Modeling the Inter-arrival Time of Packets in Network Traffic and Anomaly Detection Using the Zipf’s Law

Ali Naghash Asadi
Trustworthy Computing Laboratory, School of Computer Engineering, Iran University of Science and Technology, Tehran, Iran
aliasadi@comp.iust.ac.ir
Mohammad Abdollahi Azgomi*
Trustworthy Computing Laboratory, School of Computer Engineering, Iran University of Science and Technology, Tehran, Iran
azgomi@iust.ac.ir

Received: 15/Jan/2018 Revised: 10/Aug/2018 Accepted: 06/Oct/2018

Abstract

In this paper, a new method based on the Zipf’s law for modeling the features of the network traffic is proposed. The Zipf's law is an empirical law that provides the relationship between the frequency and rank of each category in the data set. Some data sets may follow from the Zipf’s law, but we show that each data set can be converted to the data set following from the Zipf’s law by changing the definition of categories. We use this law to model the inter-arrival time of packets in the normal network traffic and then we show that this model can be used to simulate the inter-arrival time of packets. The advantage of this law is that it can provide high similarity using less information. Furthermore, the Zipf’s law can model different features of the network traffic that may not follow from the mathematical distributions. The simple approach of this law can provide accuracy and lower limitations in comparison to existing methods. The Zipf's law can be also used as a criterion for anomaly detection. For this purpose, the TCP_Flood and UDP_Flood attacks are added to the inter-arrival time of packets and they are detected with high detection rate. We show that the Zipf’s law can create an accurate model of the feature to classify the feature values and obtain the rank of its categories, and this model can be used to simulate the feature values and detect anomalies. The evaluation results of the proposed method on MAWI and NUST traffic collections are presented in this paper.

Keywords: Network Traffic Modeling; Inter-arrival Time; Anomaly Detection; DoS Attack; The Zipf’s Law.

1. Introduction

Today, network traffic analysis and examination of its different aspects have great importance. Researchers need the artificial traffic to evaluate the efficiency and security of networks. In other words, they require a traffic generator to simulate the actual traffic. For this purpose, researchers must study the actual traffic without anomaly and extract patterns from it; then they can generate traffics that follow from the extracted patterns, and import them into the network. By doing this, they can monitor the networks based on efficiency and security. Furthermore, researchers can create abnormal traffic by identifying the normal traffic behavior and examine the network security with it. The models can also be used to provide security in the network. One way to detect anomalies is that we identify the normal behavior of network traffic and define any deviation from it as an anomaly. In other words, any deviation from the model obtained from normal traffic is considered as an anomaly. Thus the modeling of different features of the network traffic is important in anomaly detection.

In this paper, the Zipf’s law is proposed to model the features of the network traffic and detect anomalies. This law can model the network traffic using less information from it, and provide accurate simulation using the resulting model. Furthermore, we show that the model obtained from the Zipf’s law can detect anomalies. To do this, firstly we create a normal traffic collection and model its features using the Zipf’s law. The resulting normal model can be used to simulate the normal network traffic, and to detect anomalies by examining the deviations from the normal traffic. The main contribution of this paper can be summarized in four major categories:

1) Modeling the features of the network traffic using less information.

2) The Zipf’s law can model different features of the network traffic that may not follow from mathematical distributions.

3) Using the model obtained from the Zipf’s law, we can simulate the features of the network traffic and detect anomalies which affect these features.

4) The Zipf’s law is very suitable for detecting attacks that have frequent sequential patterns, for example, Denial-of-Service (DoS) attack.

The rest of this paper is organized as follows. In section 2, the Zipf’s law, the reasons for network traffic analysis, and anomaly detection approaches will be introduced. In section 3, the related works will be presented in the field of network traffic modeling and anomaly detection. In section 4, firstly we prove mathematically that all data sets can be converted to the data sets following from the Zipf’s law. Then the proposed method based on the Zipf’s law, for modeling network traffic and using it for simulation and

* Corresponding Author

anomaly detection, is presented. In this section, the traffic collections and experiment method will also be defined. Furthermore, we will show how the number of categories must be determined in the Zipf's law. In section 5, the experiments related to the traffic modeling and anomaly detection are evaluated more accurately. Furthermore, the Zipf's law is compared with Benford's law and the entropy theory. In the final section, the results are evaluated.

2. Background

2.1 The Zipf's law

George Kingsley Zipf, professor of linguistics at Harvard University, in 1949 reached conclusions about the words and their frequencies in the text by studying the words in books. His initial result showed that if you count all words in a book and sort them in descending order, you will see that the rank of each word is inversely proportional to its frequency. In other words, the number of each word appeared in the text is inversely proportional to its rank. This relationship is known as the Zipf's law. According to this law, a word with the rank of 1 appears twice more than a word with the rank of 2 in the text. It also appears three times more than a word with the rank of 3 [1].

This law shows the relationship between a frequency *F* and a rank *R* (Eq. (1)) [1][2]. Based on this relation, the frequency of each word multiplied by its rank will be almost a constant value in the text. In this relation, *N* is the total number of the words and *A* is the constant value close to 0.1 *(0<A<1)*. Figure 1 shows the frequency of each rank in the Zipf's law [1][2].

$$\frac{(\mathrm{R_i} \times \mathrm{F_i})}{\mathrm{N}} = \mathrm{A} \quad (1)$$

It can also be defined as a logarithmic relation (Eq. (2)) [1][2]. The logarithmic form of the Eq. (1) and its graph are important and they will be used in this paper.

$$\log \mathrm{R_i} + \log \frac{\mathrm{F_i}}{\mathrm{N}} = \log \mathrm{A} \quad (2)$$

This law can also be used in other topics. It is very strange how and why a simple relation occurs in many complex topics. For example, the relationship between the Zipf's law and the coherence property of the urban system (e.g. the city size distribution) has been studied [3][4]. Furthermore, the relationship between this law and the distribution of user-generated passwords has been investigated [5]. Researchers, with the concrete knowledge of password distributions, suggest a new metric based on the Zipf's law for measuring the strength of password datasets. In [6], the capacity scaling law of a device-to-device (D2D) caching network according to the Zipf popularity distribution has been studied. Also, in [7], researcher investigate the use of Benford's law and the Zipf's law to distinguish between humans using keystroke biometric systems and non-humans for auditing application. The law is also used for the simulation of a number of hits on the World Wide Web, page rank prediction, and viral email detection [8][9].

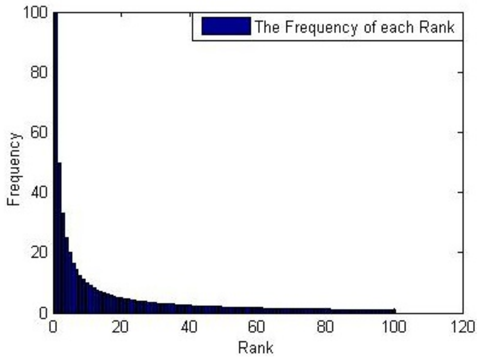


Fig. 1. Zipf's law diagram

2.2 Network Traffic Analysis

Researchers are interested to display and understand the network traffic. Network traffic analysis includes the processes of capturing and evaluating the network traffic [10]. This concept is also known as network analysis, protocol analysis, and packet sniffing.

Network traffic analysis is done for different objectives and it can provide important information on the user behavior patterns. Network managers can get a better understanding of the networks with these patterns. An important purpose of the network traffic analysis is the modeling of different features of the network traffic. The models created from different features of the network traffic can be used in various fields, including development of the normal and attack traffic simulators, detection of the anomalies and attacks, study of the service performance, examination of the network security policies, projection of the future network requirements, and prevention of the network traffic monitoring.

Development of the normal and attack traffic simulators is an important issue in network traffic modeling. Due to the lack of proper traffic collection for experiments, researchers are trying to create traffic generators with the most similarity to the actual traffic. These traffics must be generated based on the needs of the network managers; therefore, the global traffics that are available cannot be suitable. Traffic generators can have an important role in the evaluation of the service performance of a newly established network, and the monitoring of the security policies before implementing them. Researchers have created many network traffic generators with different characteristics, including Harpoon [11], D-ITG [12], TCPReplay [13], and Avalanche [14]. Network traffic analysis and modeling can provide adequate information about network requirements in the future, and network administrators can define proper policies and programs for these requirements.

The attackers are interested to access network traffic. They receive critical information in this way. To prevent it, some systems add artificial traffic to their actual traffic [15]. For this purpose, the artificial traffic should be added to the actual traffic to conceal important information and the actual behavior of the network. Furthermore, the artificial traffic should not allow that attackers distinguish it from the actual traffic. Therefore, network traffic modeling is important for generating the artificial traffics that are similar to the actual traffic. In [16], the problems of those who want to simulate internet traffic have been described; however, the solutions have also been proposed to address the problems [17].

The network traffic modeling can also be used to detect anomalies and attacks. For example, by modeling the different features of the normal network traffic and comparing it with actual traffic, any deviation from the normal model can be considered as an anomaly [18]. Therefore, accuracy in modeling for accurately identifying anomalies and attacks and reducing false alarm is very important. Traffic analysis is used to detect errors in the network and also can help to understand the main reason of the error and its effect on communication between users.

2.3 Anomaly and Attack Detection

Intrusion detection systems (IDS) can be classified into three approaches: signature-based, behavior-based, and a hybrid approach [19]. A signature detection system identifies patterns of traffic to detect a malicious activity, while an anomaly detection system compares the activities occurring with normal activities [19]. The hybrid approach uses a combination of approaches. The main advantage of a signature detection system is that the known attacks can be detected with a low false alarm rate. These systems require a signature for every attack. The main advantage of an anomaly detection system is that the unknown attacks can be detected, but these systems have the high percentage of false alarms.

Network traffic anomalies can be classified into outages, flash crowds, attacks, and measurement failures [20]. Network outages include any network failure event or temporary misconfigurations. Flash crowds are mainly caused by the sudden increase in the users of a special service. Network attacks can typically be any kind of intentional failures, including flood-based denial-of-service events. Measurement failures can be caused by problems with the data collection infrastructure itself.

A general approach for anomaly detection is to define a normal behavior and observe the actual behavior of the system and compare them. Any deviations from normal behavior should be considered as an anomaly [18]; however, there are many challenges to this simple approach. The key point is the difficulty of defining a normal behavior that should include any possible normal behaviors. Also, anomalies and methods of attacks change rapidly. Another major challenge is the massive amounts of data. Anomaly detection techniques require efficient computing to handle the large incoming data. In addition, data are usually online, therefore they need online analysis. Another important issue that arises because of the high volume input is that even a low percentage of false alarms can make analysis overwhelming.

3. Related Work

3.1 Network Traffic Modeling

Researchers have introduced many models on network traffic. According to their goals, they have investigated various features of the network traffic and modeled some of them. For example, important features of the Telnet, SMTP, HTTP, and FTP protocols are modeled in [21]. Before it, researchers had described most features of the network traffic with the Poisson distribution, but the others showed that except for user-initiated TCP session arrivals, other TCP connection arrivals don’t follow from the Poisson distribution [21].

In [22], the inter-arrival time of the TCP and UDP packets are studied by the Kolmogorov-Smirnov method to match with the mathematical distributions. In this paper, the Pareto and Weibull distributions are introduced as the most appropriate distributions for representing the inter-arrival time of the TCP and UDP packets.

In a network, packets can be transferred with a specific maximum size which is determined by the MTU parameter, and thus large packets should be divided into small ones. It leads to the inaccurate simulation of the actual packet size. In [23], researchers have been studied the packet size to estimate the probability density function that has a minimum difference with the packet size distribution graph.

Before considering the concept of self-similarity, the Poisson was the most important distribution for modeling the network traffic. The concept means that the statistical graphs of features will never change at different scales. Researchers show that the features of the network traffic have the self-similarity feature, and so they cannot be modeled by the Poisson distribution [24][25]. In [26], methods have been developed for modeling of self-similar traffic and loading process of telecommunication networks. After discovering this, the Weibull became the most important distribution to describe the different features of the network traffic [27]. In addition, the majority of features of the network traffic are long-tail, and it leads to more importance of the Weibull distribution because it can show the long-tail behavior for some shape and scale parameters. In [28][29], the role of the Weibull distribution in internet traffic modeling has been explained. In [29], researchers empirically show that despite the variety of data networks in size, number of users, applications, and load, the inter-arrival times of normal flows correspond to the Weibull distribution.

In [30], the network jitter has been modeled by mathematical distributions. Jitter can affect the quality of service. So jitter can be modeled to identify the factors that cause it. In these papers, the packet jitter has been studied with considering its path nodes. For this purpose, type and number of nodes are used for modeling the packet jitter.

3.2 Anomaly and Attack Detection in Network Traffic

Fernandes et al. have been proposed a survey to review the most important aspects related to anomaly detection [31]. They show that the anomaly detection methods can be divided into six categories, including statistical methods, clustering methods, finite state machine methods, classification-based methods, information theory, and evolutionary computation.

As mentioned, intrusion detection systems (IDS) are divided into three categories: signature-based, behavior-based, and hybrid approach [19]. Snort and Bro are signature-based intrusion detection systems which have rules to detect attacks [32]. SPADE, NIDES, PHAD, and ALAD are examples of behavior-based intrusion detection systems which are based on the statistical models and produce the anomaly detection alarms when a large deviation from normal behavior occurs [32].

Behavior-based intrusion detection systems, to evaluate the deviation from normal behavior, need to define the fast and accurate criteria that be able to work with a large volume of data. For this purpose, many methods have been proposed, including Benford's law and entropy theory. For example, in [33], Benford's law is used as a criterion to detect anomalies in the inter-arrival-time of SYN packets (the TCP flow initiator) which follows from a Weibull distribution with the shape parameter less than one. The difference between the inter-arrival-time of the actual flows and the Weibull distribution can be used to detect anomalies affecting the flow of SYN packets. This paper showed that the random variable of the Weibull distribution follows the Benford's law, so easily and without loss of generality, the Weibull conformance test can be replaced with the first-digit test which is less complicated. Also in [34], this law has been used to detect anomalies in the UDP packets and flows.

The entropy theory is more used in the anomaly detection process [35][36][37][38][39][40]. For example, in [35], the entropy theory is used to detect anomalies created by the SYN and Port_Scan attacks. In other words, the actual network traffic is compared with the basic distribution defined for normal traffic with the help of the entropy theory. In [36], important features of packets are selected for defining rules that prevent from the DoS attacks with the help of the entropy theory. In [37], the performance of IDS based on data mining and K-means algorithm is modified by using entropy theory. In our paper, the behavior-based method based on the Zipf's law will be used to detect anomalies.

4. The Zipf's Law in Modeling and Anomaly Detection

In section 3, different models of the features of the network traffic have been introduced. In these models, the researchers compare the features of the network traffic with mathematical models and distributions, and if they match, those models will be used as the best representation of the normal behavior of features. However, this approach has limitations. For example, the features that don't follow from a specific mathematical distribution can't be modeled. Furthermore, the model obtained from a traffic collection may not be usable in other traffic collections. In this paper, the Zipf's law is used to solve the above problems. This law can provide a normal behavior model from feature values with ranking the different categories.

In this section, firstly we prove that all data sets can be converted to the data sets following from the Zipf's law. Then we use this law for modeling the network traffic and using it for simulation and anomaly detection. Furthermore, we will show how the number of categories must be determined in the Zipf's law.

4.1 Following all Data Sets from the Zipf's Law

According to the Zipf's law (Eq. (1)), the rank of each category is inversely proportional to its frequency. In other words, the frequency of each category (f_i) can be obtained by multiplying the sum of all frequencies (N) in the constant value (A), and then divided it by its rank (r_i). According to this law, the frequency of a category with the rank of 1 ((f_1), Maximum frequency) is equal to multiplication the sum of all frequencies in the constant value $(f_1 = N \times A)$. Furthermore, this value (f_1) can be calculated by multiplying the rank in the frequency other categories $(f_1 = r_i \times f_i)$.

In this subsection, we prove that all data sets (even those that don't follow from the Zipf's law) can be converted to the data sets following from the Zipf's law by changing the definition of categories. For this purpose, we assume that there is a finite data set DS, and the categories $C = \{c_1, c_2, \dots, c_i, \dots, c_n\}$ has been defined for it. We show the frequency of each category with $f(c_i)$. Furthermore, according to Eq. (3), the categories with the rank of k_{th} and their frequency are shown with r_k and $f(r_k)$, respectively. If the C can be converted to the $C' = \{c'_1, c'_2, \dots, c'_j, \dots, c'_m\}$ that $f(r_1) = k \times f(c'_j)$, it can be proven that the DS follows from the Zipf's law. The r_k includes the categories (c'_j) which have the rank of k_{th}, and their frequency is equal to the maximum frequency ($f(r_1)$), the rank of 1) division by the k. We assume that the r_k can be null except $k = 1$.

$$\mathrm{r_k} = \{\mathrm{c'_j} | \mathrm{f(r_k)} = \mathrm{f}(\mathrm{c'_j}) = {}^{\mathrm{f(r_1)}}/_{\mathrm{k}}\} \quad (3)$$

Now, with the above assumptions, we prove our hypothesis. We assume that the categories defined from the DS have the same frequency $(f(c_i) = K, K$ is constant). In this case, our hypothesis can be proved by aggregating categories. The $C = \{c_1, c_2, \dots, c_n\}$ can be converted to the $C' = \{c_1, c_{2,3}, c_{4,5,6,7}, \dots\}$that the $r_k(k = 1,2,4,8, \dots)$ may have only one member, the others are null, and $f(r_1) = k \times f(r_k)$. The r_1 is equal to the aggregation of the maximum number of the $c_i s$. It is important that we can always define the $C = \{c_1, c_2, \dots, c_n\}$ with $f(c_i) = 1 (K = 1)$. So all data sets can follow from the Zipf's law. Figure 2 shows an example for this purpose.

We show that the rank of each category is the best description of that category, and by having it and the total number of samples (N) (in this paper, packets) can calculate the frequency of each category according to the Zipf's law (Eq. (1)). In this paper, the rank of each category is our model, and it can be used in the simulation and anomaly detection.

4.2 The Proposed Method

In this subsection, the proposed methods for modeling, simulation, and anomaly detection are defined. The summary of our methods is shown in Figure 3. To perform the experiments, firstly we create a data set of the selected feature values. Then we model the data set with the Zipf's law. This model can be used in the simulation and anomaly detection. As mentioned in subsection 4.1, all data sets can follow from Zipf's law with changing the definition of their categories. In the Zipf's law, the number and range of categories must be specified. This part is the most important step in the Zipf's law. If the number and range of categories were properly selected, an accurate model would obtain that can simulate features and detect anomalies with high efficiency. In subsection 4-3, we explain how to determine the number of categories. In the next step, according to Eq. (4), the maximum value of the data set is subtracted from the minimum value and then it is divided by the number of categories. By doing so, the range of categories is determined.

$$Range = \frac{Max(Feature\_Values) - min(Freature\_Values)}{The\ Number\ of\ Categories} \qquad (4)$$

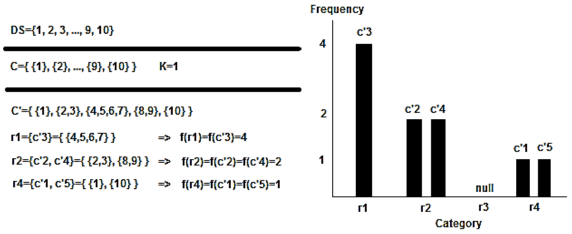


Fig. 2. An example of changing the definition of categories for following a data set from the Zipf's law

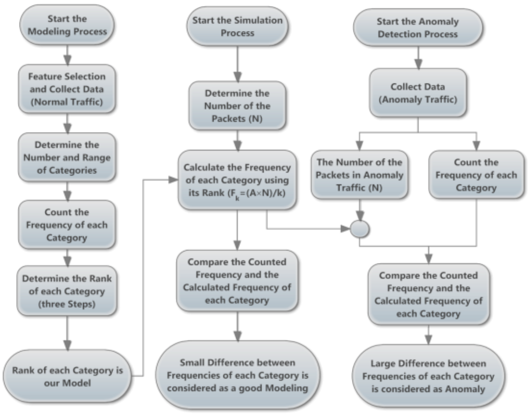
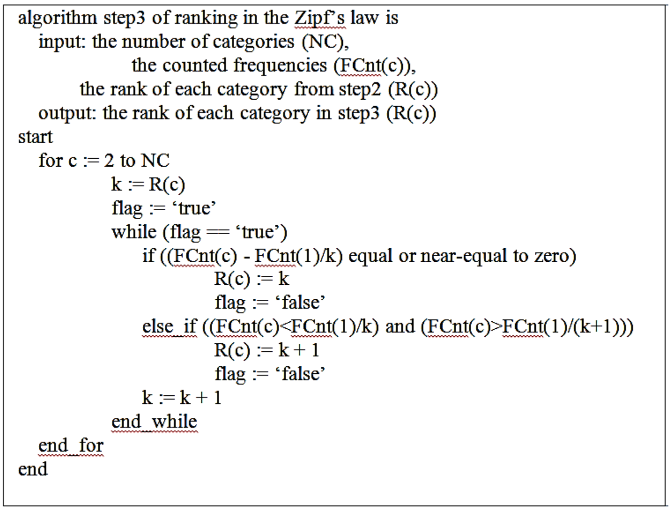

Start the Modeling Process
Feature Selection and Collect Data (Normal Traffic)
Determine the Number and Range of Categories
Count the Frequency of each Category
Determine the Rank of each Category (three Steps)
Rank of each Category is our Model
Start the Simulation Process
Determine the Number of the Packets (N)
Calculate the Frequency of each Category using its Rank (F_k=(A×N)/k)
Compare the Counted Frequency and the Calculated Frequency of each Category
Small Difference between Frequencies of each Category is considered as a good Modeling
Start the Anomaly Detection Process
Collect Data (Anomaly Traffic)
The Number of the Packets in Anomaly Traffic (N)
Count the Frequency of each Category
Compare the Counted Frequency and the Calculated Frequency of each Category
Large Difference between Frequencies of each Category is considered as Anomaly

Fig. 3. our proposed methods for modeling, simulation and anomaly detection

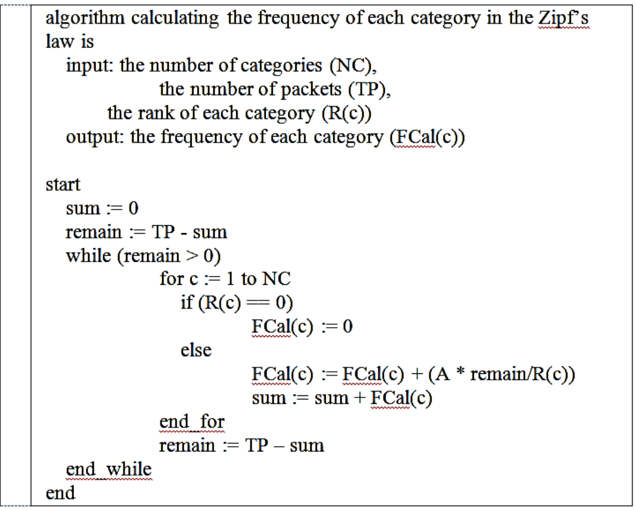
Then the frequency of each category must be counted by viewing the data set. After obtaining the frequency of each category, the rank of each category should be determined. As mentioned, the rank of categories is used as a model in the Zipf's law. In order to determine the rank of categories, three steps have been defined:

- Step 1: The frequency of categories is sorted in ascending order, then the first category is assigned the rank of 1, and the last category is assigned a rank equal to the number of categories. Categories that have a frequency of zero or one will have the rank of zero.
- Step 2: The rank of categories that have the same frequency must be set to the lower rank. For example, if the fourth and fifth categories, which have the rank of 4 and 5 respectively, have the same frequency, their rank will be 5.
- Step 3: As mentioned in the definition of the Zipf's law, the frequency of a category with the rank of 1 will be twice more than the frequency of a category with the rank of 2, and three times more than the frequency of a category with the rank of 3. To comply with this law, the pseudocode of Algorithm 1 is used. In this pseudocode, the rank of each category, obtained from step 2, will be modified. For this purpose, if the frequency of the category with the rank of k is smaller than the frequency of the first category divided by k and is greater than the frequency of the first category divided by $k+1$, the rank of the category is suitable for it. Otherwise, the k value will increase, and step 3 will be done again. In this pseudocode, the NC is the number of categories. The $R(c)$ is the rank of the c_{th} category. The $FCnt(c)$ is the counted frequency of the c_{th} category.



```
algorithm step3 of ranking in the Zipf's law is
   input: the number of categories (NC),
          the counted frequencies (FCnt(c)),
          the rank of each category from step2 (R(c))
   output: the rank of each category in step3 (R(c))
start
   for c := 2 to NC
           k := R(c)
           flag := 'true'
           while (flag == 'true')
               if ((FCnt(c) - FCnt(1)/k) equal or near-equal to zero)
                       R(c) := k
                       flag := 'false'
               else_if ((FCnt(c)<FCnt(1)/k) and (FCnt(c)>FCnt(1)/(k+1)))
                       R(c) := k + 1
                       flag := 'false'
               k := k + 1
           end_while
   end_for
end
```

Algorithm. 1 Step 3 of the ranking



```
algorithm calculating the frequency of each category in the Zipf's
law is
   input: the number of categories (NC),
          the number of packets (TP),
          the rank of each category (R(c))
   output: the frequency of each category (FCal(c))

start
   sum := 0
   remain := TP - sum
   while (remain > 0)
           for c := 1 to NC
              if (R(c) == 0)
                     FCal(c) := 0
              else
                     FCal(c) := FCal(c) + (A * remain/R(c))
                     sum := sum + FCal(c)
           end_for
           remain := TP – sum
   end_while
end
```

Algorithm. 2 Calculating the frequency of each category

By doing the above steps, a model obtains from the normal behavior of the network traffic feature. We show that this model is accurate. To do this, the frequency of each category is calculated by using the Zipf's law and the obtained ranks, and their results are compared with actual normal counted frequencies. In other words, the calculated frequencies are compared with the counted frequencies. As mentioned, the counted frequencies obtain from actual normal traffic, and the calculated frequencies obtain using the pseudocode of Algorithm 2. To perform the comparison, the frequency graph is used in both simple and logarithmic types. The SSE difference criterion is also used to show the difference between the counted and calculated frequencies. This criterion is calculated by using Eq. (5).

$$SSE = \sum_{i=1}^{NC}(log_{10} FCal(r_i) - log_{10} FCnt(r_i))^2 \qquad (5)$$

In this relationship, the *NC* is the number of categories. Also, *FCal* and *FCnt* are the calculated and counted frequencies respectively. The r_i is the rank of the i_{th} category. To prove that the Zipf's law can provide an accurate model, the *SSE* value of each category must be small.

The pseudocode of Algorithm 2 is used to calculate the frequency of each category. This pseudocode will be executed until the total of the calculated frequencies is equal to the number of the required packets. In order to evaluate the model, the total of the calculated frequencies is considered equal to the total number of packets of the actual normal network traffic, but this value in anomaly detection is considered equal to the total number of packets of the abnormal network traffic. Furthermore, the total of the calculated frequencies in the simulation is considered equal to the number of the required packets. As can be seen in the pseudocode of Algorithm 2, only the rank of each category is used for calculating the frequency of each category. In this pseudocode, the frequency of each category is calculated by dividing the number of the required packets on its rank. The constant value of *A* can be set to *0.1.* however, for greater precision in the *A* value, according to Eq. (6), the counted frequency of each category can be multiplied by its rank and divided by the total number of packets, and then their sum can be divided by the number of the non-zero categories. This value can also be considered as the A value that will always be a number close to *0.1*. In this relation, the *NCNZ* is the number of the non-zero categories.

$$A = \frac{1}{N\times NCNZ} \times \sum_{i=1}^{NC}(FCnt(i) \times r_i) \qquad (6)$$

We use from the pseudocode of Algorithm 2 to show the accuracy of the model, the network traffic simulation, and anomaly detection. We can use the calculated frequencies for network traffic simulation. In other words, we specify the number of required packets for simulating and then use from the resulting model and the pseudocode of Algorithm 2 for calculating the frequency of each category and generate packets according to it. In this paper, packet generation isn't done actually. We just calculate the frequency of each category using the model and show that these frequencies are similar to frequencies of the actual normal traffic. In other words, we know how many packets must be in each category in the total number of packets in the normal network traffic.

For detecting anomalies, the same above activities are done. For example, suppose the feature and number of categories have been selected and the rank of each category has been extracted from normal traffic. In other words, we have the normal behavior model. Now we will examine abnormal traffic with this model. For this purpose, the frequency of each category must be counted in abnormal traffic. Then according to the total number of abnormal traffic packets, the expected frequency of each category in normal traffic is calculated by the pseudocode of Algorithm 2 and the normal model. Then the calculated frequency of each category from the normal model is compared with the counted frequency of each category from abnormal traffic. Categories which their frequency shows a large difference according to the SSE value are considered as the anomaly. If the counted frequency of each category is not proportional to its rank, the SSE value will be increased.

We add the TCP_Flood and UDP_Flood attacks to the normal traffic of MAWI and NUST to create two abnormal traffic collections. There are these attacks in the NUST traffic collection separately. Since we know the location of attacks in the traffic collections, the deviation in the areas proves that the proposed method works properly. The SSE criterion is used to detect anomalies. For this purpose in Eq. (5), the calculated frequency from model normal is compared with the counted frequency from abnormal traffic.

4.3 Determining the Number of Categories

Determining the exact number of categories is the most important step in the Zipf's law. If it is properly selected, an exact model will be achieved that leads to high efficiency in the anomaly detection. The number of categories influences the speed of processes and the accuracy of results. To determine the number of categories, a balance must be established between the increase in the number of categories and the decrease in the number of categories with zero frequency (meaningless categories).

The increase in the number of categories leads to the lower speed of the modeling, simulation and anomaly detection processes. It also increases categories with zero frequency. For example, the range of 0.5 byte is meaningless for packet size because it doesn't get a decimal value. Moreover, if the maximum and minimum values of the data set are too far away, and the values aren't uniformly distributed over the entire range, many categories with zero frequency will be produced which may not have a role in the modeling, simulation and anomaly detection. Nevertheless, the increase in the number of categories will increase the accuracy of results. For example, assume the frequency of packets that their packet size is between 80 and 100 bytes is 1000. If we divide this range into two parts, the frequency of packets that their packet size is between 80 and 90 bytes may be 800. This example clearly shows the effect of increasing the number of categories in modeling. Moreover, if an anomaly event occurs in packets that their packet size is 95 bytes, anomaly detection in the range of between 90 and 100 bytes will be done much easier than the range of between 80 and 100 bytes. Nevertheless, the increase in the number of categories or the decrease in the range of categories is preferred because the modeling process is done only once.

The minimum range of categories can vary in different features. For example, the minimum range of categories in packet size feature is one byte because it cannot be a decimal value. However, the minimum range of categories in inter-arrival time feature is different and it can be selected based on the accuracy and unit of time (e.g., seconds or milliseconds). We can also use several ranges for each experiment. Figure 1 shows that the frequency of early categories is much more than the frequency of other categories. We can choose a smaller range for these categories and a larger range for others; however, we use the same range for all categories of each experiment in this paper. This range is calculated by Eq. (4).

4.4 Experimental Results

4.4.1 Traffic Collections

NUST Traffic collection [41] is used in our paper. It has been collected at the National University of Sciences and Technology in Pakistan, and its benign and attack packets have been marked. There is header and payload information of more than 6 million TCP packets and 1 million UDP packets in this traffic collection. Also, there are three subsets (home, isp and soho) in it that show traffic capture sources. Furthermore, there are attack packets include TCP_Flood, TCP_Port_Scan, and UDP_Flood in this traffic collection. These attack packets can be injected into normal packets for evaluation of intrusion detection systems.

The other traffic collection that is used in our paper is MAWI [42]. MAWI, the measurement and analysis on the WIDE internet, captures the packets from a trans-pacific backbone connecting a combination of general and academic hosts and servers to the internet. The MAWI traffic collection is a public traffic extracted from the MAWI working group traffic archive. Each file of the traffic collection, collected over a 150 Mbps trans-pacific backbone link, consists of the first 96 bytes of all IP packets sent over the link from 14:00 to 14:15 every day. We just used one minute of the collection collected in 2014/11/01. There is header and payload information of more than 3 million TCP packets and 3 thousand UDP packets in it. We have removed attacks from this traffic collection using the Snort software. Therefore, it can be used as a normal traffic collection.

4.4.2 The Zipf's Law in Modeling and Simulation

In this part, we show two graphs for each experiment (Figure 4). The number of categories is 10000 in all experiments. The first graph compares the counted frequency and the calculated frequency of each category. The vertical axis is the frequency of each category, and the horizontal axis is the number of each category. The second graph compares the logarithm of the counted frequency and the calculated frequency of each category. The vertical axis is the logarithm of the frequency of each category, and the horizontal axis is the logarithm of the rank of each category. In the second graph, the logarithm of the frequency of each category has been shown in order of its rank.

Experiments have been conducted on the inter-arrival time of packets. Details of these experiments have been shown in Table 1. The inter-arrival time of packets obtains through the subtraction of arrival time of each packet from the arrival time of the previous packet. Because these values are very small, all values are multiplied by the large constant value. The results of experiments have been shown in Figure 4.

As can be seen in Table 1, the SSE value for all experiments is very small. Also in Figure 4's graphs, the frequency of each category, counted from the normal traffic, conforms to the frequency of the same category, calculated from the normal model. Therefore, this model can be used in network traffic simulation. In other words, the number of packets of each category in a normal artificial traffic can be specified by this model. Also, we know how many packets with a certain feature value must be generated in network traffic simulation. This simulation provides the highest accuracy with minimum information.

4.4.3 The Zipf’s Law in Anomaly Detection

In this part, we show two graphs for each experiment (Figure 5). The number of categories is 10000 in all experiments. The first graph compares the calculated SSE of each packet in normal traffic. The second graph compares the calculated SSE of each packet in abnormal traffic. The calculated SSE value of each category is assigned to all packets of it. So if a category is abnormal, all packets of it will be abnormal. In both graphs, the vertical axis is the SSE value of each packet, and the horizontal axis is the number of each packet.

We create abnormal traffic collections by adding 3000 packets of the TCP_Flood and UDP_Flood attacks to normal traffic collections. These packets are added from the $200{,}000_{th}$ packet onwards of the TCP and UDP normal traffic collections. The packets of these attacks are sent at certain time intervals, 10 packets per second for 300 seconds, to achieve their objectives (disrupting the target system). For this reason, a specific inter-arrival time in the traffic collection will significantly increase. Thus, according to the Zipf's law, one or more categories will have better rank, and the changing rank against the normal rank is easily detected by the Zipf's law. In other words, the SSE value of these categories increases significantly. As can be seen in the second graph, the SSE value from the $200{,}000_{th}$ packet onwards increased significantly. However, as mentioned, some of the normal packets are known as an anomaly.

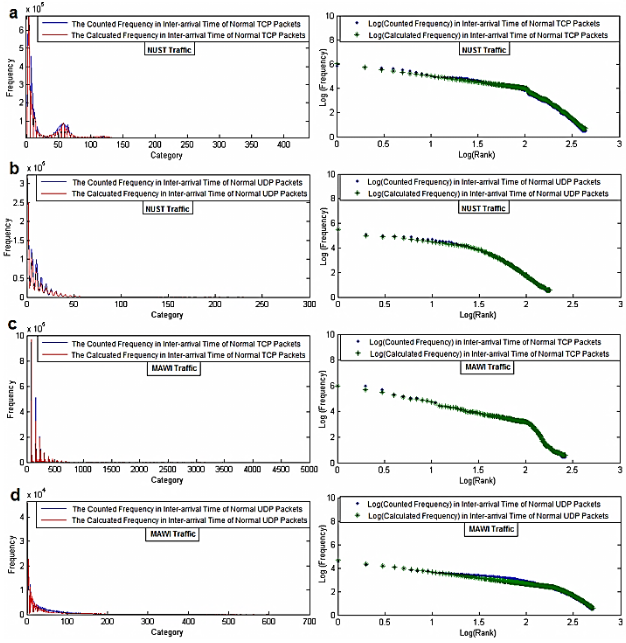


Fig. 4. The counted and calculated frequencies graphs in both simple and logarithmic in different traffic collections: (a) TCP packets of NUST collection; (b) UDP packets of NUST collection; (c) TCP packets of MAWI collection; (d) UDP packets of MAWI collection;

Table 1. Details of the modeling experiments

Traffic Collection	Inter-arrival time	Number of packets	Minimum value	Maximum value	Number of categories with a non-zero frequency	Range of categories	A	SSE
NUST	TCP packets	6276243	zero	7935	438	0.7935	0.1293	0.0026
	UDP packets	1472908	0.001	98.141	177	0.0098	0.2131	0.0024
MAWI	TCP packets	3768306	zero	416	524	0.0416	0.2569	0.0028
	UDP packets	327740	zero	3543	831	0.3543	0.1387	0.0233

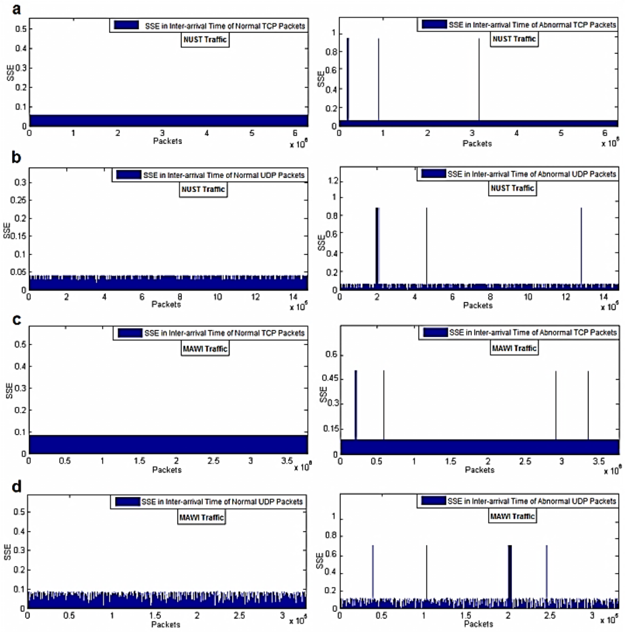


Fig. 5. The calculated SSE graphs in both normal and abnormal traffic collections in different traffic collections: (a) TCP packets of NUST collection; (b) UDP packets of NUST collection; (c) TCP packets of MAWI collection; (d) UDP packets of MAWI collection;

5. Evaluation

5.1 Evaluation of Experiments in the Different Number of Categories

In this paper, the Zipf's law was used for modeling, simulation and anomaly detection in network traffic. At the end of this paper, the proposed method is evaluated more accurately. The evaluations were done on a system with a quad-core CPU and 4 GB RAM.

5.1.1 Evaluation of the Modeling and Simulation

The modeling time and SSE Criteria are used to evaluate the modeling and simulation. In other words, the elapsed time for modeling and the SSE value which shows the difference between the calculated frequency from the model and the counted frequency from actual traffic, are used to evaluate them. According to the algorithms 1 and 2, the modeling time depends on the number of packets and categories. It is obvious that the lower values for both criteria indicate the better efficiency of the experiment. Summary of results has been shown in Table 2. As mentioned, the number and range of categories have an important role in modeling and simulation in the Zipf's law. We show the importance of this issue by repeating the experiments in the different number of categories, including 1000, 5000, and 10,000 categories.

As can be seen in Table 2, by increasing the number of categories, the modeling time increases; however, the SSE value doesn't change much with this increase. The modeling time and the SSE value are appropriate in a fewer number of categories, but increasing the number of categories can provide more information from events occurring in the network traffic. For example, assume the frequency of packets that their packet size is between 80 and 100 bytes is 1000. If we divide this range into two parts, the frequency of packets that their packet size is between 80 and 90 bytes may be 800. This example clearly shows the effect of increasing the number of categories in modeling. However, increasing the number of categories may lead to produce the categories with zero frequency which do not have a role in the modeling, simulation and anomaly detection. Therefore, the number of categories must be determined according to the needs. Table 3 shows the number of categories with a non-zero frequency in different number of categories.

As can be seen in Table 2, the SSE value is very low in all results; this means that the Zipf's law is effective in network traffic modeling and simulation. So by increasing the number of categories, the modeling and simulation are done accurately.

Table 2. The modeling and simulation experiments in the different number of categories

Traffic Collection	Inter-arrival time	Number of categories	Range of categories	Modeling time (s)	SSE
NUST	TCP packets	1000	7.9350	100.462	0.0052
		5000	1.5870	400.259	0.0026
		10000	0.7935	784.905	0.0026
	UDP packets	1000	0.0981	20.330	0.0042
		5000	0.0196	92.727	0.0024
		10000	0.0098	183.580	0.0024
MAWI	TCP packets	1000	0.4160	59.476	0.0029
		5000	0.0832	245.049	0.0029
		10000	0.0416	476.790	0.0028
	UDP packets	1000	3.5340	5.125	0.0748
		5000	0.7086	21.698	0.0748
		10000	0.3543	42.267	0.0233

Table 3. The number of categories with a non-zero frequency in the different number of categories

Traffic Collection	Inter-arrival time	1000	5000	10000
NUST	TCP packets	287	375	438
	UDP packets	53	92	177
MAWI	TCP packets	273	395	524
	UDP packets	532	667	831

5.1.2 Evaluation of the Anomaly Detection

The efficiency of anomaly and attack detection methods is often measured by two criteria: the detection rate and false alarm rate. To define the two criteria, two types of possible error which are their variables must be introduced. These two types of error are:

- False positive or false alarm error: A false positive error occurs when a normal event is detected as an attack event.
- False negative error: A false negative error occurs when an attack event is identified as a normal event.

$$False\ Positive\ Rate = \frac{Number\ of\ False\ Positive}{Total\ Number\ of\ Non\_Attacks} \quad (7)$$

$$False\ Negative\ Rate = \frac{Number\ of\ False\ Negative}{Total\ Number\ of\ Attacks} \quad (8)$$

$$Detection\ Rate = 1 - \frac{Number\ of\ False\ Negative}{Total\ Number\ of\ Attacks} \quad (9)$$

In this part, the two measures, the detection rate and false positive rate, are used to assess the proposed anomaly detection method. The elapsed time (detection time) to calculate the deviation is another criterion to assess this method. According to the algorithms 1 and 2, the detection time depends on the number of packets and categories. If the detection rate gets high value and the detection time and the false positive rate get low value, anomaly detection method will have better efficiency. Table 4 shows the results of the anomaly detection in the different number of categories. It is observed that by increasing the number of categories, the elapsed time to calculate the deviation increases, and also the detection rate and false positive rate are improved. For this reason, the number of categories must be determined according to the needs.

Table 4. The anomaly detection experiments in the different number of categories

Traffic Collection	Inter-arrival time	Number of categories	Detection time (s)	FPR (%)	DR (%)
NUST	TCP packets	1000	198.115	0.014	85.4
		5000	525.944	0.006	95.27
		10000	923.977	0.003	98.5
	UDP packets	1000	46.655	0.061	91.83
		5000	121.930	0.040	96
		10000	217.352	0.014	98.87
MAWI	TCP packets	1000	130.305	0.040	71.03
		5000	315.321	0.013	92.97
		10000	574.729	0.007	96.77
	UDP packets	1000	10.577	0.44	63.23
		5000	27.821	0.19	93.23
		10000	49.131	0.082	97.03

5.2 The Theoretical Comparison of the Zipf's Law with the Benford's Law and Entropy Theory

In section 3, we showed examples that used from Benford's law and entropy theory in anomaly detection. In this subsection, we propose more details of these laws and show advantages of the Zipf's law against them. Benford's law is an empirical law used in various fields. According to this law (Eq. (10)), the occurrence probability of the d digit as the first digit of any value in a data set isn't unexpectedly uniform. It has a logarithmic relationship [43].

$$P_d = \log_{10} \frac{d+1}{d} \quad (10)$$

We can say that Benford's law is a special case of the Zipf's law. Although Benford's law is a powerful law in anomaly detection, it has some limitations in this field. In this subsection, some limitations of Benford's law are presented and the Zipf's law is recommended due to fewer limitations and more benefits:

- Benford's law can only be done on numerical values because it needs to extract their first digit, but Zipf's law can be done on non-numeric values.
- Categories in Benford's law are the first digit of values, but they could be anything in the Zipf's law.
- The frequency graph of the first digit of feature values in Benford's law must be logarithmic, and this causes some data sets don't follow from Benford's law. The main advantage of the Zipf's law is that the frequency graph should not be necessarily logarithmic.
- Benford's law doesn't provide a model for feature values, and only when the first digit of feature values of network traffic is logarithmic, Benford's law can be used to detect anomalies. However, the Zipf's law can extract a model from feature values and use it in the simulation and anomaly detection.

In spite of all the advantages of the Zipf's law, Benford's law has advantages over the Zipf's law. The most important advantage of Benford's law is that it just counts the frequency of the first digit from the traffic collection and compares it with the logarithmic distribution, then it can detect anomalies. But for anomaly detection in the Zipf's law, the frequency of the normal model of each category is calculated and then compared with the counted frequency of the same category in the abnormal traffic. For this purpose, the categories must be defined and their normal rank extracted. So Zipf's law needs to extract the normal model.

Entropy is used in different science to study disorder and uncertainty. It emphasizes that normal systems have few disorders, and anomalies increase them; thus anomalies can be detected by this theory. In Eq. (11), $P(x_i)$ is the probability of the independent variable xi [44].

$$Entropy(X) = -\sum_i P(x_i) \times log_2 P(x_i) \qquad (11)$$

The Zipf's law and the entropy theory are very similar to each other. Zipf's law extracts an order from a feature in the normal traffic collection and thereby can detect anomalies affecting it. The entropy theory has essentially been defined by order and disorder and can identify anomalies affecting a system.

In spite of the high similarity between the Zipf's law and the entropy theory, there are differences in the two approaches. For example, the entropy theory is more used in the anomaly detection process [35][36][37][38][39][40]. In other words, the entropy theory cannot use in the modeling and simulation processes. However, the Zipf's law can extract a model from normal traffic and use it in the simulation and anomaly detection. Thus the Zipf's law provides more capabilities than the entropy theory.

5.3 The Practical Comparison of the Zipf's Law with the Benford's Law

In the related works, we presented two papers that use the Benford's law to detect anomalies in the inter-arrival time of packets [33][34]. We select them for comparison because they use the same MAWI and NUST traffic collections, albeit with different law and assumptions. Table 5 shows the best results of the three papers for comparison. As can be seen, according to the descriptions of the subsection 5-2, the results of the three papers are almost the same. However, the detection time of our method is more than the other ones. This is because the Zipf's law needs to calculate the normal frequencies and to count the anomaly frequencies. But the Benford's law can only be used to detect anomalies, while the Zipf's law can be used as a perfect method for the modeling, simulation, and anomaly detection.

Table 5. Results obtained for comparing our proposed method based on the Zipf's law with the presented methods based on the Benford's law

Traffic Collection	Inter-arrival time	Paper	Detection time (s)	FPR (%)	DR (%)
NUST	TCP packets	Our	923.977	0.003	98.5
		[33]	45.23	0.010	98.62
		[34]	-	-	-
	UDP packets	Our	217.352	0.014	98.87
		[33]	-	-	-
		[34]	-	-	-
MAWI	TCP packets	Our	574.729	0.007	96.77
		[33]	32.52	0.012	97.24
		[34]	-	-	-
	UDP packets	Our	49.131	0.082	97.03
		[33]	-	-	-
		[34]	2.188	1.53	98.93

6. Conclusions

In this paper, the Zipf's law was used to model and simulate the normal behavior of network traffic and detect anomalies. The Zipf's law is an empirical law that has been used in various research topics. Some data sets may follow from the Zipf's law, but we proved that each data set can be converted to the data set following from the Zipf's law by changing the definition of categories. We used this law to model the inter-arrival time of TCP and UDP packets in the normal network traffic and then we proposed a method to detect anomalies by using the resulting model. For this purpose, the TCP_Flood and UDP_Flood attacks were added to the normal traffic collections and they were detected with high detection rate with the help of this law. We also showed that this model can be used to simulate the inter-arrival time of packets. Then we compared the Zipf's law with Benford's law and entropy theory in anomaly detection and showed that it can be used as a perfect method for the modeling, simulation, and anomaly detection.

For future works, we can examine other features of the network traffic which that may not follow from a particular mathematical distribution. The results of this work can be effective in detecting other attacks and anomalies. Furthermore, models created from other features can be used to develop a network traffic simulator.

References

[1] G. Zipf, "Human behavior and the principle of least effort," The Economical Journal, vol. 60, no. 3, pp. 808-810, 1950.

[2] A. I. Saichev, Y. Malevergne and D. Sornette, Theory of Zipf's Law and Beyond, Springer-Verlag Berlin Heidelberg, 2010.

[3] S. Arshad, S. Hu and B. N. Ashraf, "Zipf's law and city size distribution: A survey of the literature and future research agenda," Statistical Mechanics and its Applications, vol. 492, no. 15, pp. 75-92, 2018.

[4] S. Arshad, S. Hu and B. N. Ashraf, "Zipf's law, the coherence of the urban system and city size distribution: Evidence from Pakistan," Physica A (2018), https://doi.org/10.1016/j.physa.2018.08.065.

[5] D. Wang, H. Cheng, P. Wang and G. Jian, "Zipf's Law in Passwords," IEEE Transactions on Information Forensics and Security, vol. 12, no. 11, pp. 2776-2791, 2017.

[6] A. Liu, V. Lau and G. Caire, "Capacity scaling of wireless device-to-device caching networks under the physical model," in IEEE International Symposium on Information Theory, Germany, 2017.

[7] A. Iorliam, A. T. Ho, N. Poh, S. Tirunagari and P. Bours, "Data forensic techniques using Benford's law and Zipf's law for keystroke dynamics," in International Workshop on Biometrics and Forensics, Norway, 2015.

[8] M. Jauhari, A. Saxena and J. Gautom, "Zipf's Law and Number of hits on the World Wide Web," Annals of Library and Information Studies, vol. 54, no. 2, pp. 81-84, 2007.

[9] L. Adamic and B. Huberman, "Zipf's Law and the Internet," in Glottometrics, 2007.

[10] B. R. Chang and H. F. Tsai, "Improving network traffic analysis by foreseeing data packet- flow with hybrid fuzzy-based model prediction," Expert Systems with Applications, vol. 36, no. 3, pp. 6960-6965, 2009.

[11] J. Sommers and P. Barford, "Self-Configuring Network Traffic Generation," in the 4th ACM SIGCOMM conference on Internet measurement, Italy, 2004.

[12] A. Botta, A. Dainotti and A. Pescape, "A tool for the generation of realistic network workload for emerging networking scenarios," Computer Networks, vol. 56, no. 1, pp. 3531-3547, 2012.

[13] "TCPReplay," [Online]. Available: http://tcpreplay.synfin.net/wiki. [Accessed 23 08 2018].

[14] "Network, devices & services testing-Spirent," [Online]. Available: http://www.spirent.com/. [Accessed 23 08 2018].

[15] W. M. Shbair, A. R. Bashandy and S. I. Shaheen, "A New Security Mechanism to Perform Traffic," in International Conference on Computational Science and Engineering, 2004.

[16] F. Sally and P. Vern, "Difficulties in simulating the internet," IEEE/ACM Transactions on Networking, vol. 9, no. 4, pp. 392-403, 2001.

[17] V. Paxon, "Strategies for sound internet measurement," in the 4th ACM SIGCOMM conference on Internet measurement, Italy, 2004.

[18] V. Chandola, A. Banerjee and V. Kumar, "Anomaly detection: a survey," ACM Computing Surveys, vol. 41, no. 3, pp. 1-58, 2009.

[19] A. Patcha and J. M. Park, "An overview of anomaly detection techniques: existing solutions and latest technological trends," Computer Networks, vol. 51, no. 12, pp. 3448-3470, 2009.

[20] P. Barford, J. Kline, D. Plonka and A. Ron, "A signal analysis of network traffic anomalies," in the 2nd ACM SIGCOMM Workshop on Internet measurement, France, 2002.

[21] S. Luo and G. A. Marin, "Generating Realistic Network Traffic for Security Experiments," in IEEE SoutheastCon, USA, 2004.

[22] E. Garsva, N. Paulauskas, G. Grazulevicius and L. Gulbinovic, "Packet Inter-arrival Time Distribution in Academic Computer Network," ELEKTRONIKA IR ELEKTROTECHNIKA, vol. 20, no. 3, pp. 87-90, 2014.

[23] M. Fras, J. Mohorko and Z. Cucej, "Packet Size Process Modeling of Measured Self-similar Network Traffic with Defragmentation Method," in 15th International Conference on Systems, Signals and Image Processing, Slovakia, 2008.

[24] W. E. Leland, M. S. Taqqu, W. Willinger and D. V. Wilson, "the self-similar nature of Ethernet traffic (extended version)," IEEE/ACM Transactions on Networking, vol. 2, no. 1, pp. 1-15, 1994.

[25] X. An and L. Qu, "A Study Based on Self-Similar Network Traffic Model," in Sixth International Conference on Intelligent Systems Design and Engineering Applications, China, 2015.

[26] A. Pashko and V. Tretynyk, "Methods of the statistical simulation of the self-similar traffic," Advances in Intelligent Systems and Computing, vol. 754, no. 1, pp. 54-64, 2018.

[27] V. I. Strelkovskaya, T. I. Grygoryeva and I. N. Solovskaya, "Self-similar traffic in G/M/1 queue defined by the Weibull distribution," Radioelectronics and Communications Systems, vol. 61, no. 3, pp. 128-134, 2018.

[28] M. A. Arfeen, K. Pawlikowski, D. McNickle and A. Willig, "The Role of the Weibull Distribution in Internet Traffic Modeling," in 25th International Conference on Teletraffic Congress, China, 2013.

[29] L. Arshadi and A. H. Jahangir, "An empirical study on TCP flow interarrival time distribution for normal and anomalous traffic," International Journal of Communication Systems, vol. 30, no. 1, pp. 1-19, 2017.

[30] T. K. Bandhopadhya, M. Saxena and A. Tiwari, "Jitter's Alpha-Stable Distribution Behavior," Computer Technology and Electronics Engineering, vol. 3, no. 1, pp. 13-16, 2013.

[31] G. J. Fernandes, J. P. Rodrigues, L. F. Carvalho, J. F. Al-Muhtadi and M. J. Proença, "A comprehensive survey on network anomaly detection," Telecommunication Systems (2018), https://doi.org/10.1007/s11235-018-0475-8.

[32] "IDS Distribution," [Online]. Available: http://cs.fit.edu/~mmahoney/dist/. [Accessed 23 08 2018].

[33] L. Arshadi and A. H. Jahangir, "Benford's law behavior of Internet traffic," Journal of Network and Computer Applications, vol. 40, no. 1, pp. 194-205, 2014.

[34] A. N. Asadi, "An approach for detecting anomalies by assessing the inter-arrival time of UDP packets and flows using Benford's law," in 2nd International Conference on Knowledge-Based Engineering and Innovation, Tehran, 2015.

[35] Y. Gu, A. McCallum and D. Towsley, "Detecting Anomalies in Network Traffic Using Maximum Entropy Estimation," in the 5th ACM SIGCOMM conference on Internet measurement, USA, 2005.

[36] S. Honda, T. Nakashima and S. Oshima, "Entropy Based Analysis of Anomaly Access of IP Packets," in 3rd International Conference on Innovative Computing Information and Control, China, 2008.

[37] L. I. Han, "Research of K-MEANS Algorithm based on Information Entropy in Anomaly Detection," in Fourth International Conference on Multimedia Information Networking and Security, China, 2012.

[38] S. K. Gautam and H. Om, "Anomaly detection system using entropy based technique," in First International Conference on Next Generation Computing Technologies, India, 2015.

[39] A. A. Waskita, H. Suhartanto and L. T. Handoko, "A performance study of anomaly detection using entropy method," in International Conference on Computer, Control, Informatics and its Applications, Indonesia, 2016.

[40] D. Hong, D. Zhao and Y. Zhang, "The Entropy and PCA Based Anomaly Prediction in Data Streams," Procedia Computer Science, vol. 96, no. 1, pp. 139-146, 2016.

[41] "NUST," [Online]. Available: http://wisnet.seecs.nust.edu.pk/downloads.php. [Accessed 13 06 2013].

[42] "MAWI Working Group Traffic Archive," [Online]. Available: http://mawi.wide.ad.jp/mawi/. [Accessed 13 10 2016].

[43] A. E. Kossovsky, Benford's Law: Theory, the General Law of Relative Quantities, and Forensic Fraud Detection Applications, NewYork: WorldScientific, 2014.

[44] C. E. Shannon, "A mathematical theory of communication," Bell System Technical Journal, vol. 27, no. 4, pp. 623-656, 1948.

Ali Naghash Asadi is a Ph.D. candidate of computer engineering at Iran University of Science and Technology, Tehran, Iran. He obtained his B.Sc. degree in computer engineering from Guilan University in 2013, and his M.Sc. degree in computer engineering from Iran University of Science and Technology in 2015. His research include network traffic analysis, Petri nets, power and performance modeling, and stochastic and analytical modeling.

Mohammad Abdollahi Azgomi received the B.S., M.S. and Ph.D. degrees in computer engineering (software) (1991, 1996 and 2005, respectively) from Department of Computer Engineering, Sharif University of Technology, Tehran, Iran. His research interests include Petri nets, hybrid systems, stochastic modeling, quantitative evaluation, and trustworthy computing. Dr. Abdollahi Azgomi is currently an associate professor at School of Computer Engineering, Iran University of Science and Technology, Tehran, Iran.