Soft Comput (2018) 22:6621–6627
https://doi.org/10.1007/s00500-017-2679-3

FOCUS

Network anomaly detection based on probabilistic analysis

JinSoo Park[1] · **Dong Hag Choi**[1] · **You-Boo Jeon**[1] · **Yunyoung Nam**[2] · **Min Hong**[3] · **Doo-Soon Park**[3]

Published online: 15 June 2017
© Springer-Verlag GmbH Germany 2017

Abstract In this paper, we propose a method to detect network intrusions using anomaly detection technique based on probabilistic analysis. Victim's computers under attack show various symptoms such as degradation of TCP throughput, increase in CPU usage, increased round trip time, frequent disconnection to the Web sites, etc. These symptoms can be used as components to construct the k-dimensional feature space of multivariate normal distribution, in which case an anomaly detection method can be applied for the detection of the attack on the distribution. These features are generally highly correlated. Thus we choose only a few of these features for the anomaly detection in multivariate normal distribution. We use Mahalanobis distance to detect the anomalies for each data, normal, and abnormal. Anomalies are identified when their square root of Mahalanobis distance exceeds certain threshold. A detailed description of the threshold setting and the various experiments are discussed in simulation results.

Keywords Anomaly detection · Network intrusion · Traffic flood · DDoS attacks · Mahalanobis distance

Communicated by G. Yi.

✉ JinSoo Park
vtjinsoo@gmail.com

[1] Wellness Coaching Service Research Center, Soon Chun Hyang University, RM U1202, 22 Soonchunhyangro, Shinchang-myeon, Asan-si, ChoongCheongNam-do, South Korea

[2] Department of Computer Engineering, Soon Chun Hyang University, Asan-si, ChoongCheongNam-do, South Korea

[3] Department of Computer Software Engineering, Soon Chun Hyang University, Asan-si, ChoongCheongNam-do, South Korea

1 Introduction

Network intrusion refers to the unauthorized modification or use of network resources of others without the permission of the administrator or the owner. Recently, network intrusion has evolved into various forms due to the emergence of diverse network environments and the new types of application services, and the corresponding detection technologies are becoming more diverse.

Network intrusion detection technologies are classified into two different types—network intrusion detection technologies and host intrusion detection technologies depending on where the detection is taking place (Scarfone and Mell 2007). The network intrusion detection technologies monitor traffic to and from the network at any point on the network including wireless networks (Singh et al. 2014; Jingle and Rajsingh 2014; Tseng et al. 2011; Joo et al. 2015). The host intrusion detection technologies monitor traffic on a private host computer or devices on the network and informs users or administrators if there are any suspicious traffic activities (Ye et al. 2002).

The network intrusion detection technology can also be defined based on detection methods employed, and largely be classified into knowledge-based (signature-based) intrusion detection methods and behavior-based (anomaly-based) intrusion detection methods (Scarfone and Mell 2007; Keegan et al. 2016). In addition to these two classifications, much research has been conducted in the field of network intrusion detection that combines both machine learning technologies and cloud computing technologies, which have received much attention recently. We will review several related papers on these network intrusion detection technologies and discuss their technical characteristics before discussing the proposed method in this paper.

A knowledge-based (signature-based) approach is based on the detection of specific patterns (called signatures) in traffic activities during previous network intrusions, such as byte sequences in network traffic, known malicious instruction sequences used by malware. This technique is designed to prevent the same type of network intrusion. It is very accurate and efficient for known attacks experienced before, but it is very vulnerable to attacks that are new or inexperienced previously. In order to overcome these drawbacks, there is a need to continuously update related signatures, which requires a considerable amount of computing resources and overheads. The core of this technology lies in the performance of the matching algorithm that efficiently determines whether the signature information is detected or not (Tuck et al. 2004; Tan and Sherwood 2005).

On the other hand, in the case of behavior-based (anomaly-based) methods, any network activities can be considered as an intrusion in the event that the activities deviate from those under normal network environments. Most behavior-based methods adopt the machine learning technologies such as Bayesian inference based method (Valdes and Skinner 2000), PCA-based method (Shyu et al. 2003), decision tree-based method (Stein et al. 2005) and HMM-based method (Yeung and Ding 2003). For the behavior-based method, the system is trained for the normal state traffic environments, and it detects the network activities in the abnormal states based on the training results. In some cases associated with this method, system administrators sometimes define the management rules and, in this case, any violation of rules are considered as an intrusion based on these rules. There are also statistical detection approaches for the behavior-based methods (Staniford et al. 2002; Lu et al. 2007; Weon et al. 2005).

Recently, a lot of studies are being conducted to detect network intrusions by combining machine learning technologies with cloud computing technologies (Keegan et al. 2016). A common issue with these types of technologies is that the machine learning algorithms must deal with a large amount of training data and the algorithms need to be modified to run in cloud computing environments. Broadly used machine learning algorithms include k-means (Zhao et al. 2009), k-NN and SVM algorithm (Chen et al. 2014), Naïve Bayes algorithm (Bhat et al. 2013), and random forest (Singh et al. 2014), etc.

In this paper, we propose a new network intrusion detection system, especially for DDoS-type attacks, based on machine learning algorithms, which can be viewed as a kind of host intrusion detection system. To our best knowledge, applying the machine learning technologies for the host intrusion detection system have been less studied, so it is worth studying these issues.

Systems that have been attacked by the network will exhibit various symptoms. These symptoms show a lot of differences between during the normal state and the abnormal state, which can be used to determine whether or not the system is receiving network intrusions. In Kolahi et al. (2015), typical phenomena under network attacks are studied in detail. In particular, changes in TCP throughput, CPU utilization, and RTT increase are described in detail. As shown in the paper, system responses in terms of TCP throughput, CPU utilization, and RTT are significantly different during the normal and abnormal states. Sudden changes of these data can indicate a sign of network attack. From the viewpoint of machine learning, these data can be used as features of data and can be used to detect traffic anomalies. Detection of traffic anomaly is linked to the detection of data anomaly, which is one of the major technical challenges in machine learning areas. In this paper, we calculate the Mahalanobis distance (MD) for the detection of traffic anomalies. The Mahalanobis distance is mainly used for detecting outliers of data based on calculation of the distance between individual data and the center of data cluster, assuming that the data forms multivariate normal distribution in multidimensional feature space. The MD has once been used to detect the anomalous traffic interval (Bayarjargal and Cho 2014). In this paper, unlike the way the method was used in Bayarjargal and Cho (2014), the MD is applied for two dimensional data, and a method of presetting the threshold value necessary for anomaly judgement is also proposed.

This paper is organized as follows. Section 2 explains a novel anomaly detection strategy proposed in this paper including a brief summary on the multivariate normal distribution. Details on the Mahalanobis distance and the related threshold decision are discussed as well. Section 3 discusses simulation results of this research with several illustrations. Finally, we discuss conclusions of the study and future studies.

2 Proposed intrusion detection method

2.1 Anomaly detection framework

Most of the phenomena caused by DDoS attacks are highly correlated. For example, victim's computers show slow response to certain inputs, high CPU loads, frequent disconnection to Web sites, low TCP throughputs, etc. A few of these symptoms are positively or negatively correlated. TCP throughput caused by network attacks drops, but RTT and CPU utilization increase conversely. These correlated data can be represented in k-dimensional feature space, and they can be modeled by the multivariate normal distribution ($\mathbf{x} \sim N_k(\boldsymbol{\mu}, \sum)$) as follows (https://en.wikipedia.org/wiki/Multivariate_normal_distribution)

$$f_X(x_1,\ldots,x_k) = \frac{1}{\sqrt{(2\pi)^k \left|\sum\right|}} \times \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \sum^{-1} (\mathbf{x}-\boldsymbol{\mu})\right) \tag{1}$$

where $\mathbf{x} = [X_1, X_2, \ldots, X_k]$ is a real $k-$dimensional random vector, $\boldsymbol{\mu}$ is $k-$dimensional mean vector and $\sum$ indicates $k \times k$ covariance matrix.

The mean and the covariance matrix can be represented as follows, respectively:

$$\boldsymbol{\mu} = [E[X_1], E[X_2], \ldots, E[X_k]]$$
$$\sum = \left[\mathbf{Cov}\left[X_i, X_j\right]\right], i = 1, 2, \ldots, k;\ j = 1, 2, \ldots, k$$

Detection of abnormal data is based on discovery of outliers under multivariate normal distribution. The idea begins by the notion that most normal data are distributed near a mean value of data cluster, however abnormal data are located away from the mean in general. Thus if we find a proper threshold value to discriminate the normal and abnormal data under the multivariate normal distribution, we can detect the anomalies of network traffic data. The detection of anomalies are based on calculation of Mahalanobis distance of traffic data and evaluation of the distance from the center of data cluster with respect to certain criteria. Details on this are given next in Sect. 2.2.

2.2 Mahalanobis distance and threshold decision

In typical anomaly detection problems, we generally chose a threshold value heuristically to discriminate normal and abnormal data. For example, if the probability density function evaluated for certain input data is less than the threshold value, we consider the corresponding data as anomaly, otherwise we accept them as normal data (https://www.coursera.org/learn/machine-learning). In this research, we present a method to preset the threshold based on Mahalanobis distance described in Warren et al. (2011), Johnson and Wichern (2007) and (http://www.ece.vt.edu/people/profile/mili). The Mahalanobis distance of normally distributed data can be defined as follows. If the underlying distribution of the k random variables is exactly multivariate normal with a $k \times k$ covariance matrix $\sum$ and if mean vector $\boldsymbol{\mu} = [\mu_1, \mu_2, \ldots, \mu_k]^T$ formed from the population means of the k random variables, then the Mahalanobis distance MD of a particular multivariate data-point $\mathbf{x}$ from $\boldsymbol{\mu}$ is calculated as:

$$\mathrm{MD} = \sqrt{(\mathbf{x}-\boldsymbol{\mu})^T \sum^{-1} (\mathbf{x}-\boldsymbol{\mu})}. \tag{2}$$

It is known that for ($\mathbf{x_i} \sim N_k(\boldsymbol{\mu}, \sum)$), MD_i^2 follows a chi-squared distribution with n degrees of freedom $\chi^2_{n:\alpha}$ approximately. And $1-\alpha$ defines the probabilities that $\mathrm{MD}_i(X_i) \leqslant \sqrt{\chi^2_{n:\alpha}}$ (Warren et al. 2011; Johnson and Wichern 2007; http://www.ece.vt.edu/people/profile/mili).

This assumption facilitates evaluation of candidate anomalies because chi-squared distribution can be used to identify outliers based on certain value α. For given α and $\mathbf{x}_i$, we identify anomalies when

$$\mathrm{MD}_i > \sqrt{\chi^2_{n:\alpha}}. \tag{3}$$

For example, for two variables $n = 2$ and $\alpha = 0.05$, $\chi^2_{n:\alpha} = 5.99$, if only 5% of the squared Mahalanobis distances are expected to be greater than 5.99 (see Table 2). Then the threshold MD to identify the outliers can be set as $\mathrm{MD} = \sqrt{\chi^2_{n:\alpha}} = \sqrt{5.99} = 2.45$.

In our simulation results in Sect. 3.2, we choose 0.025 for α, following the classical notion in robust estimation theory to identify outliers (Warren et al. 2011; Johnson and Wichern 2007; http://www.ece.vt.edu/people/profile/mili). Therefore our chosen threshold is set as:

$$\mathrm{MD_{th}} = \sqrt{\chi^2_{n:0.025}}. \tag{4}$$

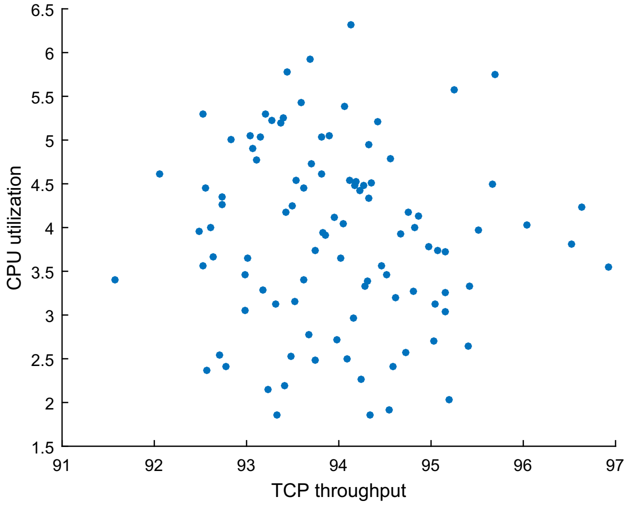
3 Estimation results

3.1 Training data generation

In Kolahi et al. (2015), the symptoms of a flood attack were analyzed in three different aspects: TCP throughput, average RTT, and CPU usage. When systems are under attack, the TCP throughput drops down significantly, and the average RTT as well as the CPU usage increases significantly. These symptoms can be used as features for the detection of network attack. Multiple types of data can be used as features that will be represented as a random vector of certain size. In this paper, we choose only two different types of data (TCP throughput and CPU usage) as features because these two features are highly correlated to each other. And also it is convenient to check the detection results in 2-D space visually. Based on the description in Kolahi et al. (2015), the TCP throughput degrades significantly during the attack, while

Table 1 Normal data generation

Features	Mean	Variance
TCP throughput	94	1
CPU utilization	4	1



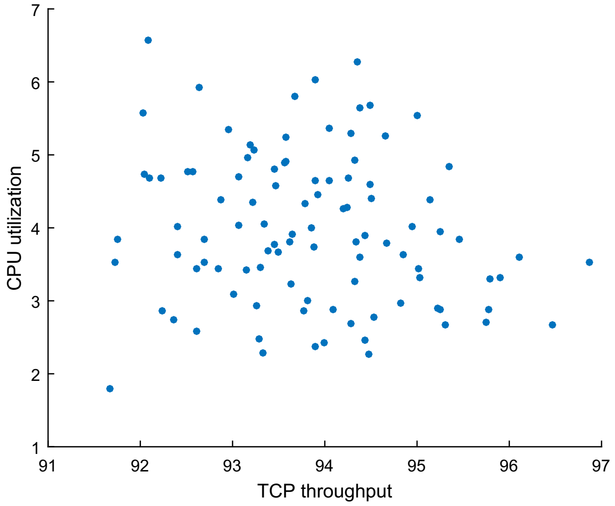


Fig. 1 Examples of training data with mean (TCP throughput = 94, CPU utilization = 4)

it is usually maintained at constant rate before the attack. Conversely, the CPU usage increases significantly during the attack, while it is maintained at nearly constant rate during a normal state. We generate two highly correlated data for TCP throughput and CPU utilization during the normal state with certain covariance between them. Also, we generate abnormal data for these two types of features with different mean and variance for various tests. As described in Kolahi et al. (2015), the TCP throughput and the CPU utilization have significant differences in their values during the normal and the attack periods. Following the data in Kolahi et al. (2015), we choose 94 (Mbps) for the mean and 1 for the variance respectively, for the normal TCP throughput data generation. And for the CPU utilization data generation, we choose 4 for the mean and 1 for the variance respectively as summarized in Table 1.

Table 2 Chi-squared distribution table for 2 degrees of freedom

df	⋯	0.9	0.1	0.05	0.025	0.01	0.005
2	⋯	0.211	4.605	5.991	7.378	9.21	10.697

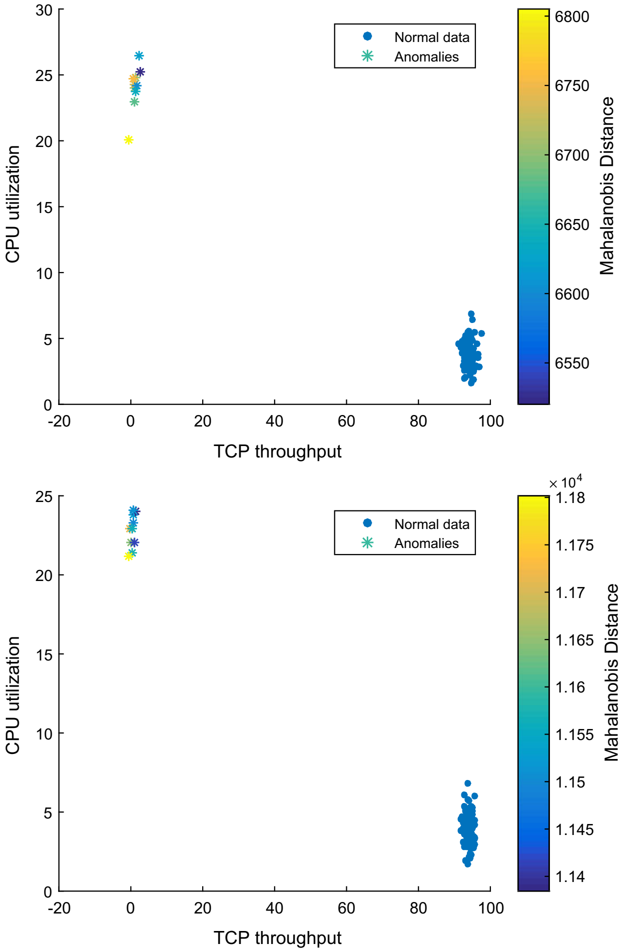


Fig. 2 Two snapshots of calculated Mahalanobis distances with respect to varying TCP throughput and CPU utilization

Figure 1 shows a few snapshots of normally distributed data in 2-D feature space. There are 100 correlated data generated, and the data is distributed near the mean with (94, 4) for TCP throughput and CPU utilization (Table 2).

3.2 Abnormal data and detection results

Figure 2 shows the calculated Mahalanobis distance for both normal and abnormal data in the 2-D feature space where most of normal data are located in lower right hand side and those of abnormal data are located in the upper left hand side. Because there are huge gap of distance between the normal and the abnormal data cloud, MD_{th} provides sufficient margin for reliable detection results.

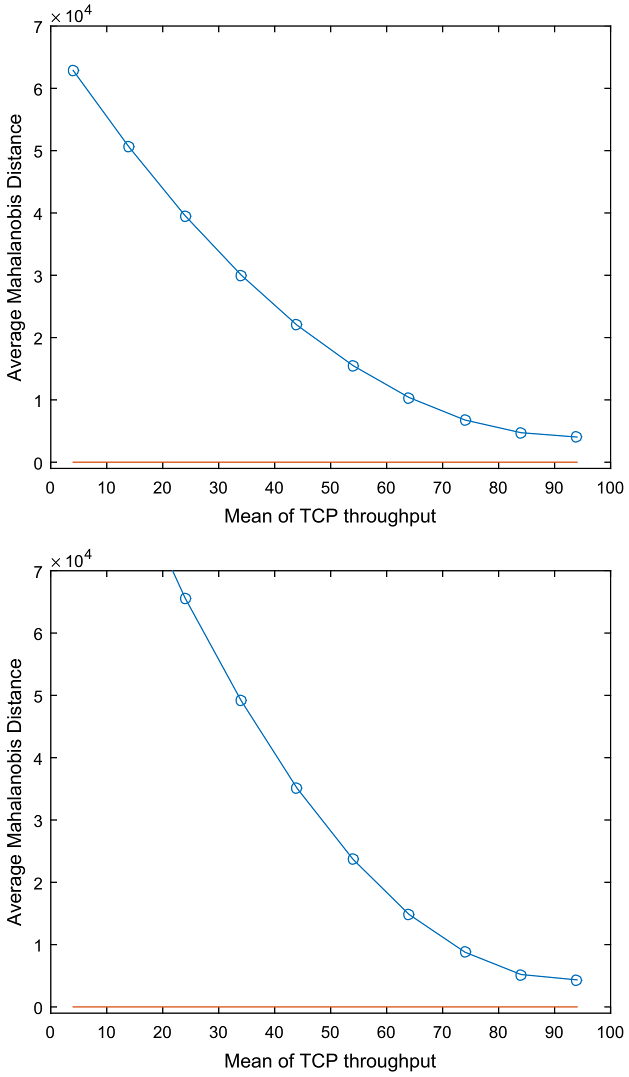


Fig. 3 Average Mahalanobis distance with respect to varying TCP throughput

Also we generate abnormal data by changing means of TCP throughput and calculate the average MDs for each mean. As shown in Fig. 3, average MDs are going to increase as the TCP throughput drops to near zero, which corresponds to a state after and during the attack. The red line indicates $\mathrm{MD}_{\mathrm{th}}$. The figure illustrates that we have enough distance margin to successfully detect the attacks. For this experiment, we generated 10 abnormal data for each average TCP throughput and used the average MDs for the plots.

In similar experimental scenario where means of CPU utilization increase from low (4) to high (25) in Fig. 4, we can see that the average MDs increase. This indicates that our chosen threshold $\mathrm{MD}_{\mathrm{th}}$ is a fairly good criterion for anomaly detection. Like in simulation for Fig. 3, we generated 10 abnormal data for each average CPU utilization and used the average MDs for the plots.

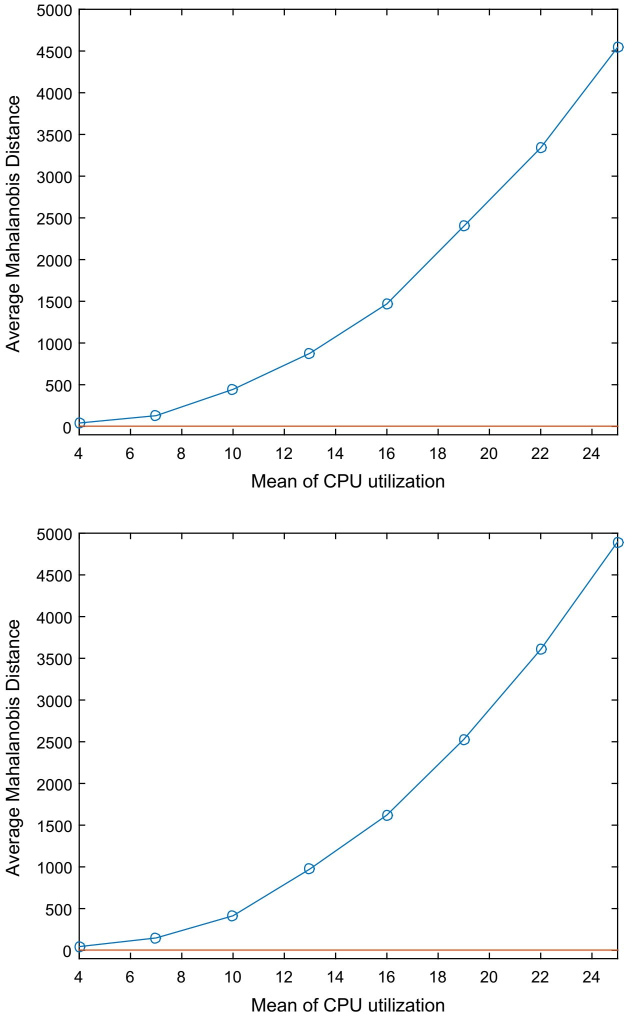


Fig. 4 Average Mahalanobis distance with respect to varying CPU utilization

Similarly, we check the performance of timely detection on the attack by plotting MDs with respect to time slots. As shown in Fig. 5, we plot MDs in log scale for each time slot because the calculated MDs are too big in scale compared to the time slot indices. It shows that even though the abnormal data (corresponding to the network attacks) is fed into the system in different time slots, it is detected whenever its corresponding MDs exceed our chosen threshold $\mathrm{MD}_{\mathrm{th}}$. And the detection process continues during the period of attack.

4 Conclusion

This paper proposes a new anomaly detection-based network intrusion detection strategy based on a probabilistic model. We use two anomaly symptoms (TCP throughput and CPU usage) as features in 2-D feature domain where we assume these features are highly correlated and can be modeled by

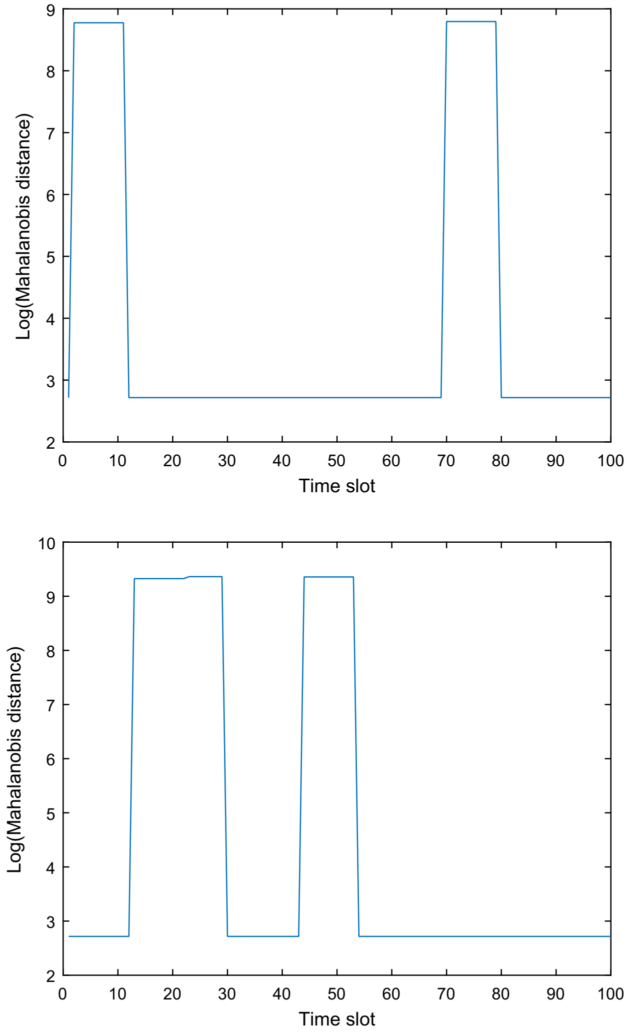


Fig. 5 Detection of attack based on Mahalanobis distance in time slot. At the *top*, two attacks occur at slots 1 and 70, and it last for 10 slot time period. At the *bottom*, two attacks occur at slots 13 and 44 wherein the first attack lasts for around 20 slot period, and the second one last for 10 slot time period

the multivariate normal distribution with certain mean and variance. Then, the detection of the attack can be achieved by calculating the Mahalanobis distance of data under the multivariate normal distribution and comparing the distance based on the proposed threshold decision method. If the MD of corresponding data exceeds $\sqrt{\chi^2_{n:0.025}}$, the data is considered as anomaly. As shown in the simulation results, the preset threshold can be used as a good criteria for stable anomaly identification results. For this study, we generated the training data for simulations. As next step, we plan to test the proposed method using real Internet trace file or real-time traffic data captured from an experimental test-bed.

Acknowledgements This research was supported by the Ministry of Science, ICT and Future Planning (MSIP), Korea, under the Information Technology Research Center (ITRC) support program (IITP-2016-H8601-16-1009) supervised by the Institute for Information and communications Technology Promotion (IITP)

Compliance with ethical standards

Conflict of interest Authors JinSoo Park, Dong Hag Choi, You-Boo Jeon, Yunyoung Nam, Min Hong and Doo-Soon Park declare that he/she has no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

Bayarjargal D, Cho G (2014) Detecting an anomalous traffic attack area based on entropy distribution and mahalanobis distance. Int J Secur Appl 8:87–94

Bhat A, Patra S, Jena D (2013) Machine learning approach for intrusion detection on cloud virtual machines. Int J Appl Innov Eng Manag 2:56–66

Chen T, Zhang X, Jin S, Kim O (2014) Efficient classification using parallel and scalable compressed model and its application on intrusion detection. Expert Syst Appl 41:5972–5983

Jingle IDJ, Rajsingh EB (2014) ColShield: an effective and collaborative protection shield for the detection and prevention of collaborative flooding of DDoS attacks in wireless mesh networks. Hum Centric Comput Inf Sci 8:1–19

Johnson RA, Wichern DW (eds) (2007) Applied multivariate statistical analysis, 2nd edn. Pearson Prentice Hall, Upper Saddle River

Joo J, Lee J, Park J (2015) Security considerations for a connected car. J Converg 6:1–9

Keegan N, Ji S, Chaudhary A, Concolato C, Yu B, Jeong DH (2016) A survey of cloud-based network intrusion detection analysis. Hum Centric Comput Inf Sci 6:1–16

Kolahi SS, Treseangrat K, Sassafpour B (2015) Analysis of UDP DDoS flood cyber attack and defense mechanisms on web server with Linux Ubuntu 13. In: 2015 international conference on communications, signal processing, and their applications (ICCSPA), vol 17–19

Lecture notes.http://www.ece.vt.edu/people/profile/mili

Lecture notes.https://www.coursera.org/learn/machine-learning

Lu K, Wu D, Fan J, Todorovic S, Nucci A (2007) Robust and efficient detection of DDoS attacks for large-scale internet. Comput Netw 51:5036–5056

Scarfone K, Mell P (2007) Guide to intrusion detection and prevention systems (IDPS). NIST special publication 800–94, Gaithersburg, MD, USA

Shyu M-L, Chen S-C, Sarinnapakorn K, Chang L (2003) A novel anomaly detection scheme based on principal component classifier. In: Proceedings of the IEEE foundations and new directions of data mining workshop, Melbourne, FL, USA, pp 172–179

Singh R, Singh P, Duhan M (2014) An effective implementation of security based algorithmic approach in mobile adhoc networks. Hum Centric Comput Inf Sci 4:1–14

Singh K, Guntuku SC, Thakur A, Hota C (2014) Big data analytics framework for peer-to-peer botnet detection using random forests. Inf Sci 278:488–497

Staniford S, Hoagland JA, McAlerney JM (2002) Practical automated detection of stealthy portscans. J Comput Secur 10:105–136

Stein G, Chen B, Wu A, Hua KA (2005) Decision tree classifier for network intrusion detection with GA-based feature selection. In: Proceedings of the 43rd annual Southeast regional conference, vol 2, pp 136–141

Tan L, Sherwood T (2005) A high throughput string matching architecture for intrusion detection and prevention. In: 32nd international symposium on computer architecture, pp 112–122

Tseng F-H, Chou L-D, Chao H-C (2011) A survey of black hole attacks in wireless mobile ad hoc networks. Hum Centric Comput Inf Sci 1:1–16

Tuck N, Sherwood T, Calder B, Varghese G (2004) Deterministic memory-efficient string matching algorithms for intrusion detection. In: IEEE Infocom, pp 333–340

Valdes A, Skinner K (2000) Adaptive model-based monitoring for cyber attack detection. In: Recent advances in intrusion detection, Toulouse, France, pp 80–92

Warren R, Smith R, Cybenko A (2011) Use of Mahalanobis distance for detecting outliers and outlier clusters in markedly non-normal data: a vehicular traffic example. Interim Report (United States Air Force), pp. 9–11

Weon I, Song D, Ko S, Lee C (2005) A multiple instance learning problem approach model to anomaly network intrusion detection. Int J Inf Process Syst 1:14–21

Ye N, Emran SM, Chen Q, Vilbert S (2002) Multivariate statistical analysis of audit trails for host-based intrusion detection. IEEE Trans Comput 51:810–820

Yeung D-Y, Ding Y (2003) Host-based intrusion detection using dynamic and static behavioral models. Pattern Recognit 36:229–243

Zhao W, Ma H, He Q (2009) Parallel k-means clustering based on mapreduce, (Cloud Computing 2009). Lect Notes Comput Sci 5931:674–679