Karaelmas Science and Engineering Journal
Journal home page: http://fbd.beun.edu.tr
DOI: 10.7212%2Fzkufbd.v8i1.998

Research Article
Received / Geliş tarihi : 17.05.2017
Accepted / Kabul tarihi : 29.06.2017

Identifying the General Pattern of the Academic Computer Networks Based on Users Daily Behaviors

Kullanıcıların Günlük Davranışına Bağlı Olarak Akademik Ağların Genel Örüntüsünün Belirlenmesi

Fidan Kaya Gülağız*, Onur Gök, Suhap Şahin

[1]Kocaeli University, Computer Engineering Department, Kocaeli, Turkey

Abstract

The use of the internet has become wide spread with the developments in technology as a result of this data has been removed to electronic environment. With the increase of data stored in the electronic environment, the security of the data has become much important. For this reason, network anomalies and attacks should be detected early. There are many different data mining methods used to detect network anomalies. In this study general behavior of academic networks determined to detect network anomalies. For this purpose, a network state analysis method using Iterative K-Means and Hidden Markov Model methods is proposed.

Keywords: Academic networks, Intrusion detection, Hidden markov model

Öz

Teknolojideki gelişmeler ile birlikte internet kullanımı da yaygınlaşmıştır bu durumun bir sonucu olarak veriler de elektronik ortama taşınmıştır. Elektronik ortamda saklanan verinin boyutunun artması ile birlikte verilerin güvenliğinin sağlanması daha da önemli hale gelmiştir. Bu nedenle ağda meydana gelen anormalliklerin ve saldırıların erken teşhisi önemlidir. Ağdaki anormallikleri tespit etmek amacıyla kullanılan pek çok farklı veri madenciliği yöntemi mevcuttur. Bu çalışmada akademik ağlardaki anormallikleri tespit etmek amacıyla ağın genel davranışı tanımlanmıştır. Bu amaçla Iterative K-Means ve Hidden Markov Model (HMM) yöntemlerini kullanan bir ağ durum analizi yöntemi önerilmiştir.

Anahtar Kelimeler: Akademik ağlar, Atak tespiti, Hidden markov model

1. Introduction

The Internet has been one of the most important mediums preferred for knowledge sharing in recent years. The number of websites has increased rapidly, data is transferred to the electronic environment and web-based services have become an important part of our lives. In addition to this, using the internet environment is risky because of the intrusions carried out against the internet-based systems (Foltz 2004).

There are many methods/tools that can be used for detection of the web intrusions and one of them is the intrusion detection systems. Intrusion Detection Systems (IDS) are security systems that detect the intrusions aiming your computer system and network resources, determine the abnormal circumstances while monitoring the systems and target to take the necessary pre-caution. Attack detection mechanisms used by IDS are basically divided into two. These are signature-based methods and anomaly based methods. Signature-based methods are used more for known intrusion types prevalently. Anomaly based methods model the normal behavior of the network and determines abnormal circumstances through the obtained model of normal state. Anomaly based methods are preferred more as they are more successful than the signature-based model in terms of detecting the new intrusion types. Anomaly detection systems determine behaviors that show deviation from the expected normal usage profiles as an anomaly. In the detection of the anomaly the normal behavior of the system is generally obtained using the statistical methods. One of the statistical methods used generally for this purpose is the Hidden Markov Model (HMM).

Markov model is used to represent the tiered observations happen in time. HMM is a special version of the Markov

*Corresponding author: fidan.kaya@kocaeli.edu.tr

Fidan Kaya Gülağız orcid.org/0000-0003-3519-9278
Onur Gök orcid.org/0000-0002-9833-9031
Suhap Şahin orcid.org/0000-0003-1340-8972

Model. In HMM a sequence of states cannot be observed. Instead, a sequence of states must be obtained from the sequence of the observations. States being obtained from observations is what makes the model hidden. HMM is a method that is used generally in recent years in the detection of the network anomalies.

In the study conducted by Dorj and Altangerel (Dorj and Altangerel 2013), using the Discrete Hidden Markov Model, anomaly detection is made for discrete sequences. As a result of the tests conducted it is seen that the method works on the discrete data with 85.7% of correctness. In another study conducted by Jain and Abouzakhar (Jain and Abouzakhar 2012), HMM is used for understanding whether there is an intrusion in TCP services. The normal behavior of the TCP data is modeled using HMM. It is determined that the success of the method is in between the range of 76%-99% according to the TCP service type. Again by Sultana et. al. (Sultana et al. 2012) a host-based intrusion detection system is developed using HMM. In the study, it is mentioned that HMM is successful in detection of host-based intrusions but the processing time must be shortened. With the study conducted, the processing time of the HMM-based methods is decreased at the rate of 48%. Another technology where the network based applications are run is cloud computing technology. Intrusion detection is important in the cloud computing-based application too. In the study conducted by Hong et. al. (Hong et el. 2015), HMM is used to do accessibility analysis of the cloud architecture. The normal state of the cloud substructure is modeled and by detecting the cases where there is overcrowding the accessibility of the system is increased. Again, for detecting the intrusions in the network, a hybrid model is constituted by Karthick et. al. (Karthick et al. 2012). In the model constituted different HMM case diagrams are obtained for different intrusion types. Thus, the system is provided to recognize all of the potential intrusions' general behavior. By Fan (Fan 2012), whether the demands from the real network traffic data are intrusions or a normal access demand is determined. When results belonging to different data set are examined it is seen that HMM could detect the abnormal cases.

When the studies in the literature are examined, it is seen that HMM method gives out successful result on different intrusion types in terms of detection of the intrusions on the network. Within the context of this study, the normal behavior of the network is tried to be modeled at two step using Iterative K-Means and HMM method.

2. Materials and Method

In Figure 1, main steps to create the proposed architecture are given. These are (i) establishment of an academic campus network, (ii) obtaining network normal behavior parameters during a day, (iii) implementation of Iterative K-means algorithm on recorded parameters for clustering, (iv) processing HMM to obtain network normal behavior. Each stage is detailed in the following subsections.

2.1. Network Traffic Statistical Analysis

The method suggested within the context of the study is a method that is applicable to the networks whose daily behavior has a specific pattern. University (campus) networks, school networks, institutions that work within specific working hours can be given examples to this type of networks. In order these networks to be modeled the behaviors of the users in the network must be modeled. The most general method used for this purpose is the modeling of packet's inter arrival time distribution within some specific periods of time. The most common probability distributions used for the modeling can be listed as Pareto, Weibull, Gamma and Lognormal.

In Figure 2 the traffic density constituted by the packets in different types belonging an academic network is shown diagrammatically. Here, a daily traffic being divided into different time periods is seen. When the graphic is examined it is determined that the network traffic is intensified in some specific hours within the working hours. As it can be understood from the figure it be can divided into four different parts according to the network's usage density. These are the 07:00-10:00, 10:00-16:30, 16:30-22:00 and 22:00-07:00 time intervals. It is seen that, from these time

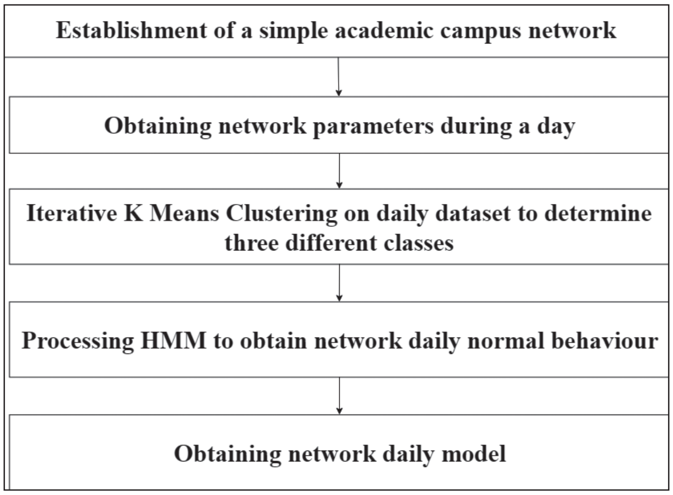

Establishment of a simple academic campus network
Obtaining network parameters during a day
Iterative K Means Clustering on daily dataset to determine three different classes
Processing HMM to obtain network daily normal behaviour
Obtaining network daily model

Figure 1. Main steps to create proposed system.

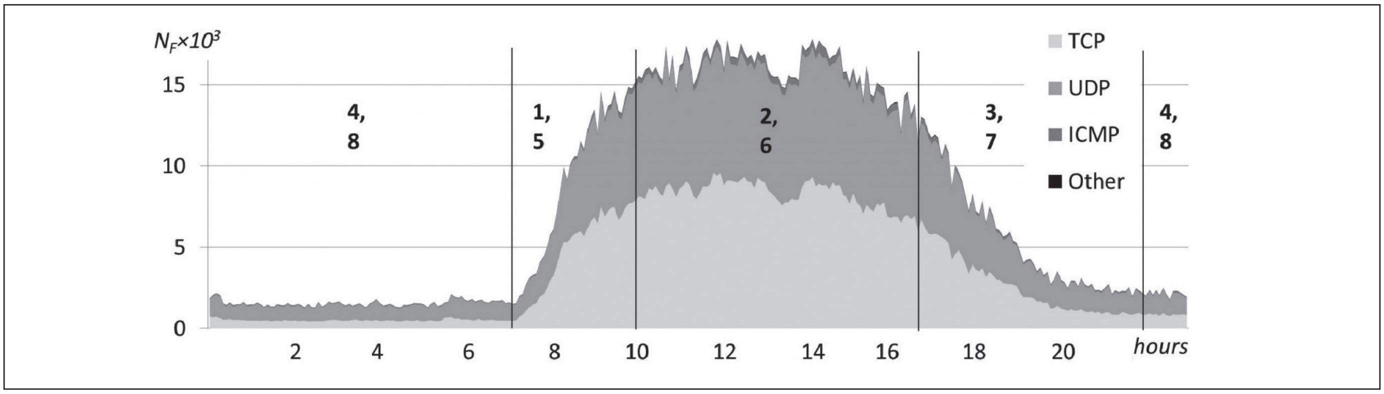


Figure 2. The daily network traffic density for an academic network (Garsva et al. 2014).

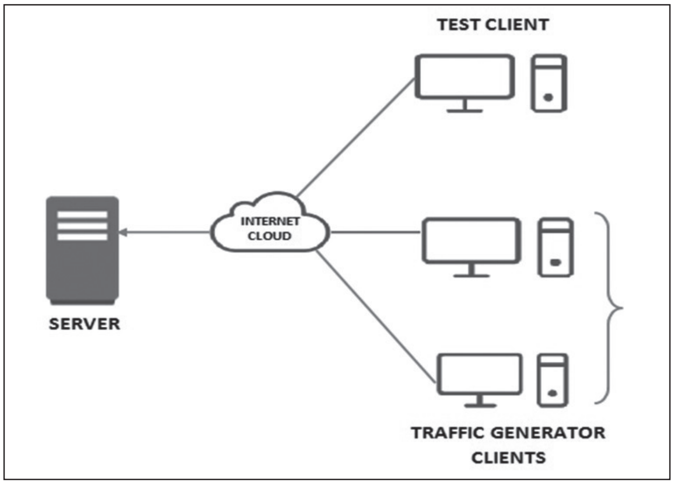


Figure 3. The network architecture from which the data set is obtained.

intervals, there is an increase in the network traffic density within the 07:00-10:00 interval and the traffic reaches the densest levels within the time interval 10:00-16.30 and this density continues for a while and it gradually decreases within the time interval 16:30-22:00 and at night, the traffic density drops down to the lowest levels.

Distributions that will model the network traffic behavior having different patterns in different times of the day in accordance with the time intervals shown in Figure 2 , is shown in Table 1. In this study, the general behavior of the network is tried to be modeled for a half hour example presenting the time interval numbered 3 and 7 in Figure 2. Using the probability distributions specified in the table, data regarding the state of the network through the network architecture shown in Figure 3 is obtained. This architecture is a simple client-server model aimed at obtaining data. The architecture here is formed using the OPNET (OPNET Technologies 1986) simulation software devoted to a single main server.

The server in Figure 3 demonstrates the main server that the requests are made to. Packet transmission and traffic density in the network is provided with two separate network traffic generator clients. These clients are accepted to be in different LANs. Via a test client in a different LAN, some data belonging to the main server daily density is recorded. Through this data, the analysis of the network's density within a half-hour time interval is made. The aim of the study is to show that the density state of the network can be modeled through example data. Therefore, for the sampling, a half-hour time sampling interval is decided to be enough. And for the sampling period, two minute interval is used.

There are many parameters that can be used to determine different density states of the network. These parameters can be listed basically as response time, connection time, throughput, jitter for video data, retransmission count and delay (Dorj and Altangerel 2013). For the analysis of the network structure given in the article the parameters, throughput, delay, retransmission count and response time are used. These parameters are standard parameters used aiming detection of deadlock in the networks and prevention of the deadlock. After the network architecture was established, modeling of the overall behavior of the network was performed. At this step Iterative K-Means algorithm is used to cluster daily network data and HMM is used to obtain network daily behavior using clustered data. In the next sub section of the study, Iterative K-Means, HMM and modeling phase for detection of the general behaviour is explained in detail.

2.2. Detecting General Behavior of Network

After the network traffic model was created and the parameters were saved for one day, daily data set should be clustered. For this purpose Iterative K-Means algorithm is used.

The Iterative K-Means algorithm is based on the K-Means algorithm. K-Means algorithm processes with the thought that new clusters should be formed according to distance between points and center of clusters. Distance between elements of data set and center of clusters also give error rate of clustering. K-Means algorithm consists of four basic steps (Kaya and Köymen 2008). These are; determination of centers, assigning points to clusters which are outside of the centers according to distance between centers and points, calculation of new centers and repeating these steps until obtaining decided clusters. Pseudo code of K-Means algorithm is given in Figure 4.

Iterative K-Means algorithm is an iterative version of classic K-Means algorithm. This algorithm processes according to two parameters. These are minimum number of clusters and maximum number of clusters. Algorithm takes these parameters as input to increase accuracy. Method works between minimum and maximum number range of K-Means algorithm and calculates a score value for each number of cluster. Number of clusters which have the highest score values, returns as a result (Kaya Gülağız and Şahin 2017). After network daily data clustered, HMM is used to obtain daily network model.

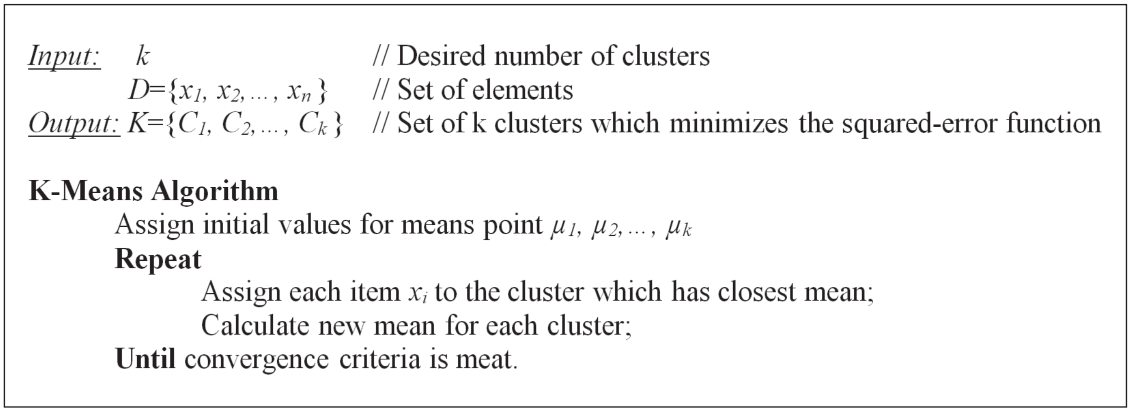
In data mining in order to represent the tiered observations occur in time, the Markov Model is used (Ramage 2007). Markov models accept that examples taking place in the learning cluster are not independent of each other and the input sequence is constituted by the probabilistic process (Alpaydın 2013). There are many different application fields of HMM including the computer networks. In this study with the aim of detection of the density of a network, general HMM, is used.

HMM is a special case of the Discrete Markov Model. Here, there is not any state sequence that can be observed as in the Discrete Markov model. Instead of this, the state sequence must be obtained from the observation sequence. That the situations are obtained from observations is the state which makes the model hidden. There may be many different situation sequences that have produced an observation sequence. However, the probability of each of these sequences is different. The goal of the method is to obtain the state sequence that have the highest probability to produce a specific observation sequence.

The most general representation of the Hidden Markov model is made by using the λ symbol. λ = (A, B, π) is the

Table 1. Packet inter arrival time distributions according to day sections (Garsva et al. 2014).

Protocol	Section	Distribution	α	B	KS	A
TCP	1	Pareto2	2.7278	0.0034	0.0478	359.63
	2	Weibull	0.8978	0.0011	0.0623	1890.70
	3	Pareto2	1.9981	0.0022	0.0558	698.54
	4	Pareto2	3.1931	0.0361	0.0246	23.18
	5	Weibull	0.7750	0.0027	0.0441	230.12
	6	Gamma	0.7810	0.0023	0.0545	1201.46
	7	Weibull	0.7578	0.0028	0.0467	437.86
	8	Weibull	0.7897	0.014	0.0211	24.93



```
Input:    k                          // Desired number of clusters
          D={$x_1, x_2, \dots, x_n$}          // Set of elements
Output: K={$C_1, C_2, \dots, C_k$}         // Set of k clusters which minimizes the squared-error function

K-Means Algorithm
    Assign initial values for means point $\mu_1, \mu_2, \dots, \mu_k$
    Repeat
        Assign each item $x_i$ to the cluster which has closest mean;
        Calculate new mean for each cluster;
    Until convergence criteria is meat.
```

Figure 4. The pseudo code of K-Means algorithm (Kaya Gülağız and Şahin 2017).

parameter set of a Hidden Markov model. These parameters must be obtained from a given training set in order to create the Hidden Markov model. When we obtain these parameters, we can get the observation sequences in length we desire which are suitable for the model, the parameters of which we know. The diagram of the general HMM model can be represented by using the trellis diagram as shown in Figure 5 (Alpaydın 2013).

In Hidden Markov model, there are three problems, the answers of which are required to be found. These are; finding the actual probability for an observation sequence, finding the state sequence having highest derivation probability for this observation sequence when an observation sequence is given, and obtaining the model parameters through data set when a training (learning) data set is given. Forward algorithm is used for solving the problem given in the first case, Viterbi algorithm is used for solving the problem in the second case, and Forward-Backward algorithm is used for solving the problem in the last state along with Expectation Maximization or BaumWelch algorithm. Within the scope of the study, emphasis shall be laid on the problem of how to create the model when the training sequence is given. The most commonly used method for this progressing is the forward-backward algorithm. The goal here is to obtain the parameter values that optimize the system. It is possible to perform both forward and backward operations on a path given with its forward and backward variables used in forward algorithm. The partial probability formula used in the Backward algorithm is similar to the partial probability formula used in the Forward algorithm. This calculation is shown below with (1) (Alpaydın 2013).

$$\beta_t(i) = P(O_{t+1}...O_T \mid q_t = S_i, \lambda) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j) \tag{1}$$

During the stage of learning the model parameters, the probability formulas for both the Backward and Forward algorithms are used. With the help of these formulas, both forward and backward calculations will be done in a iterative way on the sequences in the training data set. We have defined the parameters of the Hidden Markov model with the variant λ. Let's assume that the variant λ* is defined to be able to get the most suitable model parameters. The probability of transition to S_j state at t+1 moment from S_i state at t moment through the observation sequence with $\varepsilon_t(i,j)$ is shown (Alpaydın 2013). In this case;

$$\varepsilon_t(i,j) = P(q_t = S_i, q_{t+1} = S_j \mid O, \lambda)$$
$$\varepsilon_t(i,j) = \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{\sum_k \sum_l \alpha_t(k) a_{kl} b_l(O_{t+1}) \beta_{t+1}(l)} \tag{2}$$

This probability can be calculated with (2) (Alpaydın 2013). The partial probability variant $\alpha_t(i)$ in the formula

Table 2. State list of HMM.

N: Number of States	$S = \{S_1, S_2, S_3, ..., S_N\}$
M: Number of Observation Variables	$V = \{v_1, v_2, v_3, ..., v_M\}$
Transition Probabilities	$A = [a_{ij}], a_{ij} \equiv P(q_{t+1} = S_j \mid q_t = S_i)$
Observation Probabilities	$B = [b_j(m)], b_j(m) \equiv P(O_t = v_m \mid q_t = S_j)$
Initial Probabilities	$\pi = [\pi_i],\ \pi_i \equiv P(q_1 = S_i)$

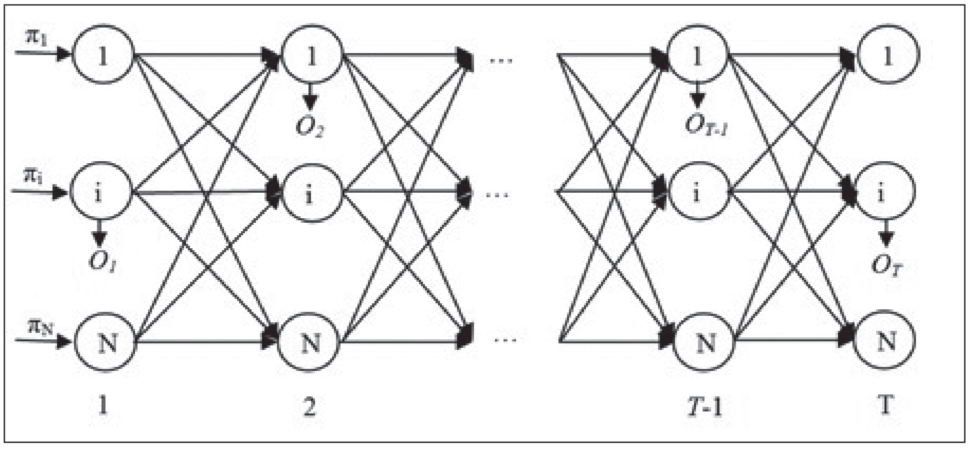


Figure 5. General representation of HMM (Alpaydın 2013).

calculates the probability for the first t observation forward. Then, the probability of transition to S_j state at t+1 moment from S_i state at t moment through the observation sequence with $a_{ij}b_j(O_{t+1})$ is calculated. Then, the observation values from t + 1 moment to T moment are also calculated with the backward variant $\beta_{t+1}(j)$. On the other hand, in the denominator, the values of all possible states between t and t+1 times are calculated. Thus and so, the value of the transition probability is calculated.

The probability of being at S_i state at the t moment can be calculated as the sum of all the possible states in t + 1. We can show this probability value with (3) (Alpaydın 2013).

$$\gamma_t(i) = \sum_{j=1}^{N} \varepsilon_t(i,j) \tag{3}$$

Considering that there are K pcs of observation lines and that these sequences are independent from each other. The estimated model parameters can be calculated with (4), (5) and (6) (Alpaydın 2013).

$$\hat{a}_{ij} = \frac{\sum_{k=1}^{K} \frac{1}{P(O^k | \lambda)} \sum_{t=1}^{T_{k-1}} \varepsilon_t^k(i,j)}{\sum_{k=1}^{K} \frac{1}{P(O^k | \lambda)} \sum_{t=1}^{T_{k-1}} \gamma_t^k(i)} \tag{4}$$

$$\hat{b}_j(m) = \frac{\sum_{k=1}^{K} \frac{1}{P(O^k | \lambda)} \sum_{t=1}^{T_k} \gamma_t^k(j) 1(O_t^k = v_m)}{\sum_{k=1}^{K} \frac{1}{P(O^k | \lambda)} \sum_{t=1}^{T_k} \gamma_t^k(j)} \tag{5}$$

$$\hat{\pi}_i = \frac{\sum_{k=1}^{K} \frac{1}{P(O^k | \lambda)} \gamma_1^k(i)}{\sum_{k=1}^{K} P(O^k | \lambda)} \tag{6}$$

After the model is established, the probability of an output belonging to the model can also be calculated. For the calculation operation, the Forward algorithm can be used again.

3. Results and Discussion

In this study, a HMM that models the specific half hour of academic network is proposed to show that the normal behavior of the network can be modeled. For this purpose firstly a daily dataset is obtained and clustered. This dataset contains four network parameters. These are load, response time, delay and number of retransmissions. Parameters are recorded with two minute sampling over a period of one day and daily dataset is obtained. Clustering results are shown in Figure 6. Busy cluster is shown with blue, normal cluster is shown with red and idle cluster is shown with green color. As seen as Figure 6, Iterative K-Means algorithm is clustered network data successfully. High load, delay and response time values are located at busy cluster and low values are located at idle cluster.

Clusters are labeled as idle, normal and busy. Idle cluster is symbolized as B, normal cluster symbolized as N and busy cluster is symbolized as Y. After clustering step, for modelling network behavior a small time interval is sampled

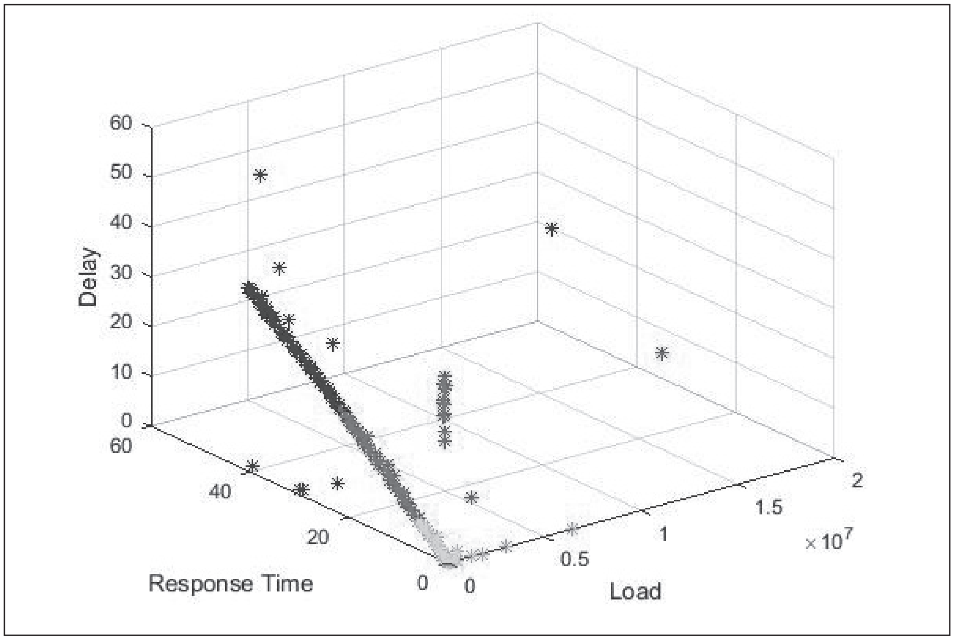


Figure 6. Clustering results of daily network data.

Table 3. Sample data set obtained in accordance with an academic network.

Load (Bytes / sec.)	Response Time (sec.)	Delay (sec.)	# of Retransmission
993028.01	0.43	0.279	28
266123.76	1.41	1.134	13
2136413.58	2.34	1.888	2
1410108.82	3.26	2.656	94
531934.43	4.20	3.452	120
1154402.34	0.25	0.214	6
73.34	0.0078	0.004	1
73.6	0.0078	0.003	1
72.84	0.0078	0.003	1
73.81	0.0078	0.032	1
5725763.11	0.0081	0.004	1
173.50	0.0083	0.004	1
191.091	0.00831	0.018	2
185.24	0.00831	0.033	1
135.75	0.008313	0.0042	3

Table 4. Obtained daily state patterns for HMM.

D1	*Y*	*N*	*Y*	*Y*	*Y*	*B*	*B*	*B*	*B*	*B*	*B*	*B*	*B*	*B*	*B*
D2	*N*	*N*	*N*	*Y*	*Y*	*B*	*B*	*B*	*B*	*B*	*B*	*B*	*B*	*B*	*B*
D3	*N*	*N*	*Y*	*Y*	*Y*	*B*	*B*	*B*	*B*	*B*	*B*	*B*	*B*	*B*	*B*
D4	*Y*	*N*	*B*	*Y*	*Y*	*N*	*B*	*B*	*B*	*B*	*B*	*B*	*B*	*B*	*B*
D5	*N*	*N*	*B*	*Y*	*Y*	*N*	*B*	*B*	*B*	*B*	*B*	*B*	*B*	*B*	*B*
D6	*N*	*N*	*N*	*Y*	*Y*	*B*	*B*	*B*	*B*	*B*	*B*	*B*	*B*	*B*	*B*
D7	*N*	*N*	*Y*	*Y*	*Y*	*B*	*B*	*B*	*B*	*B*	*B*	*B*	*B*	*B*	*B*
D8	*N*	*N*	*N*	*Y*	*Y*	*N*	*B*	*B*	*B*	*N*	*B*	*B*	*B*	*B*	*B*
D9	*N*	*N*	*Y*	*Y*	*Y*	*N*	*B*	*B*	*B*	*B*	*B*	*N*	*B*	*B*	*B*
D10	*N*	*B*	*N*	*Y*	*Y*	*B*	*B*	*B*	*B*	*B*	*B*	*B*	*B*	*B*	*B*

because a HMM modeling all day would be too long. The data set obtained with the trial packets sent using the test server as a result of two minute sampling of the half-hour time interval is shown in Table 3. This time interval shows the 18:30-19:00 time intervals in Figure 2.

According to Figure 2, the time intervals having different patterns that can be used in the daily analysis of the network are 07:00-10:00, 10:00-16:30, 16:30-22:00 and 22:00-07:00. Four different HMMs that belong to these different time intervals can be constituted and by using the relative model for in each time interval the normal state of the network can be obtained using four different HMMs. In this case while constituting the first model, data that belongs to the time interval 07:00-10:00 and that is obtained in different days; while constituting the second model, data that belongs to the time interval 10:00-16:30 and that is obtained in different days; while constituting the third model, data that belongs to the time interval 16:30-22:00 and that is obtained in different days; and while constituting the last model, data that belongs to the time interval 22:00-07:00 will be used. Thus, the daily behavior of the network will be obtained completely with more than one models coming together. In the modeling stage of the network, four parameters mentioned above will be recorded periodically within a time interval specified.

After clustering process, half-hour data is obtained in a way that this data belong to the same time intervals of different ten days. This data is different from the dataset used at clustering step. Data which belongs the same time interval of ten days were classified according to k-nn algorithm logic and can be used to generate HMM. The classified version of ten days data according to obtained clusters is shown in Table 4. The first column of the table states which day that the data belongs to. D1 statement corresponds to the first day.

A HMM having two states can be obtained by using the log directories given in Table 4. The HMM to be obtained will be in the form of the trellis diagram shown in Figure 5. It will consist of two states to be expressed as YS and NS. Outputs related to NS state will be represented with the variants of Y and N, and outputs related to YS state will be represented with the variant B. The size of the Trellis diagram will be the same as the length of the given state sequence. The transition probabilities of the states and outputs constituted in each state are calculated using (4), (5) and (6). As the half-hour data samples are taken at two minutes, intervals a sample of daily data contains 15 samples. Within the context of the study, samples are taken at two minutes, but appropriate sample intervals can be required to be specified for different networks. Similarly, the model can also be established for longer time intervals. Thus HMMs can be obtained after modeling all the time intervals having different patterns can be used in detecting the general behavior of academic network. Thus during a day, if we realize a dangerous data density at any time we can record four parameters of main server at this time point and can check the probability of this pattern on related time interval's HMM using Forward algorithm. If probability of pattern is low, attacks can be detected.

4. Conclusion

In this study, a method for modeling the network behaviors having a specific pattern during the day is suggested. Proposed method is appropriate for many networks such as school networks, university networks, institution networks and organizations that work within a specific working hour.

Obtained model can be used effectively in different areas such as density determination of the servers, load distribution between servers. In the coming period, in the stage of clustering, the different data clustering methods will be used. Also proposed method will be used to balance distribution of the load between the servers.

5. References

Alpaydın, E. 2013. Yapay Öğrenme. Boğaziçi Üniversitesi Yayınevi, İstanbul.

Dorj, E., Altangerel, E. 2013. Anomaly detection approach using hidden markov model. *International Forum on Strategic Technology, s.*1-4, Ulaanbaatar, Mongolia.

Fan, W. K. G. 2012. An adaptive anomaly detection of WEB-based attacks. *7th International Conference on Computer Science & Education*, s. 690-694, Melbourne, Australia.

Foltz, C. B. 2004. Cyberterrorism, computer crime and reality. *Info. Mngmt. & Comp. Sec.*12: 154-166.

Garsva, E., Paulauskas, N., Grazulevicius, G., Gulbinovic, L. 2014. Packet inter-arrival time distribution in academic computer network. *Elektron. Electrotech.* 20: 87-90.

Hong, B., Hu, Y., Peng, F., Deng, B. 2015. Distributed state monitoring for IaaS Cloud with continuous observation sequence. *IEEE 15th Conference on Scalable Computing and Communications and Its Associated Workshops*, s. 1037-1042, Beijing, China.

Jain, R., Abouzakhar, N. S. 2012. Hidden markov model based anomaly intrusion detection. *International Conference for Internet Technolog and Secured Transactions,* s. 528-533, London, United Kingdom.

Karthick, R. R., Hattiwale,V. P., Ravindran, B. 2012. Adaptive network intrusion detection system using a hybrid approach. *Fourth International Conference on Communication Systems and Networks*, s. 1-7, Bangalore, India.

Kaya Gülağız, F. , Şahin, S. 2017. Comparison of hierarchical and non-hierarchical clustering algorithms. *Int. J. of Comp. Eng. and Info.Tech.,* 9: 6-14.

Kaya, H., Köymen, K. 2008. Veri madenciliği kavrami ve uygulama alanlari. *Doğu Anadolu Bölgesi Araştırmları,* s. 159-164.

OPNET Technologies 1986, Optimum network simulation and engineering tool. https:// www.riverbed.com/ gb/products/ steelcentral/ opnet.html?redirect=opnet

Ramage, D. 2007. *http://cs229.stanford.edu/section/cs229-hmm.pdf*

Sultana, A., Hamou-Lhadj, A., Couture, A. 2012. An improved hidden markov model for anomaly detection using frequent common patterns. *IEEE International Conference on Communications (ICC),*s. 1113-1117, Ottawa, ON, Canada.