1st International Conference on Experiments/Process/System Modeling/Simulation/Optimization
1st IC-EpsMsO
Athens, 6-9 July, 2005

ADAPTIVE NETWORK ANOMALY DETECTION USING BANDWIDTH UTILISATION DATA

Vassilios C. Moussas[1*], Stylianos Sp. Pappas[2]

[1] Network Operations Centre (NOC), Technological Educational Institution (T.E.I.) of Athens, Egaleo GR-12210, Greece, e-mail: vmouss@teiath.gr

[2] Dept. of Information & Communication Systems Engineering, Univ. of the Aegean, Karlovassi GR-83200, Greece, e-mail: spappas@aegean.gr

Keywords: Adaptive Multi-Model Filtering, ARIMA, Network Security, Intrusion Detection

Abstract: *With the rapid expansion of computer networks, security has become a crucial issue. A good way to detect illegitimate use is through monitoring the network traffic for unusual user activity or for intruder activity. Methods of intrusion detection based on hand-coded rule sets or predicting commands on-line are laborious to build, not very reliable, and, require a vast amount of special traffic data (protocols, ports, connections, etc.). This paper proposes an adaptive method for network unusual activity and intrusion detection, using simple and widely found sets of data such as bandwidth utilization. Bandwidth use is the most common set of data as almost all network administrators monitor the bandwidth utilization for their servers, LAN/VLAN users, and network connections. The proposed method uses past traffic data to learn and model the normal periodic behavior of a network connection. Either ARMA or State-Space models can be used for the traffic pattern modeling. Several traffic anomalies are also modeled from the past experience. The adaptive mechanism of the proposed method applies to all available models the new traffic data collected by the monitoring program. If the traffic pattern does not match the normal behavior of the network connection an anomaly is detected, and furthermore, when the pattern matches a known case, the type of anomaly is identified. Real traffic data were used and real world cases were tested from the TEI of Athens campus network. The applied adaptive multi-model filtering algorithm identified successfully the unusual activities. The method can perform equally well in real-time using the sampling interval required by the network monitoring programs.*

1 INTRODUCTION

With the rapid expansion of computer networks, security has become a crucial issue. Intrusion Detection Systems (IDS) are being designed to protect such critical networked systems. There are two major approaches in intrusion detection: anomaly detection and misuse detection. Misuse detection first is recording and modeling specific patterns of intrusions and then is monitoring and reporting if matches are found. Anomaly detection, on the other hand, records and models the normal behavior of the network and then detects variations from the normal model in the observed data. The main advantage with anomaly intrusion is that it can detect new forms of attacks or network misuse, as they will probably deviate from the normal behavior [1].

Proposed anomaly detection systems model apply various methods to the normal behavior of the network. Some systems utilize artificial neural networks (ANN) [2] and self-organizing maps (SOM) [3]. The NN is fed initially by normal traffic to learn the normal conditions and then by the observed traffic to detect anomalies. Other systems collect statistics from certain system parameters into a profile, and construct a distance vector for the observed traffic and the specified profile [4].

Most methods of intrusion detection are based on hand-coded rule sets or predicting commands on-line, they are laborious to build, and, require a very large amount of special traffic data (detailed static logs, protocols, ports, connections, etc.) provided by hubs, routers, firewalls, hosts, and network sniffers. In addition, most of the algorithms require that the data used for training is purely normal and does not contain any attacks. The process of manually cleaning the data is quite time consuming and a large set of clean data can be very expensive, although some algorithms may tolerate mixed data [5].

In this paper we investigate the possibility to use simple and widely found datasets i.e. bandwidth utilization, in order to extract information regarding normal network utilization and model it. Subsequently, we apply an adaptive multi-model partitioning method to identify the model of the observed traffic and detect unusual events.

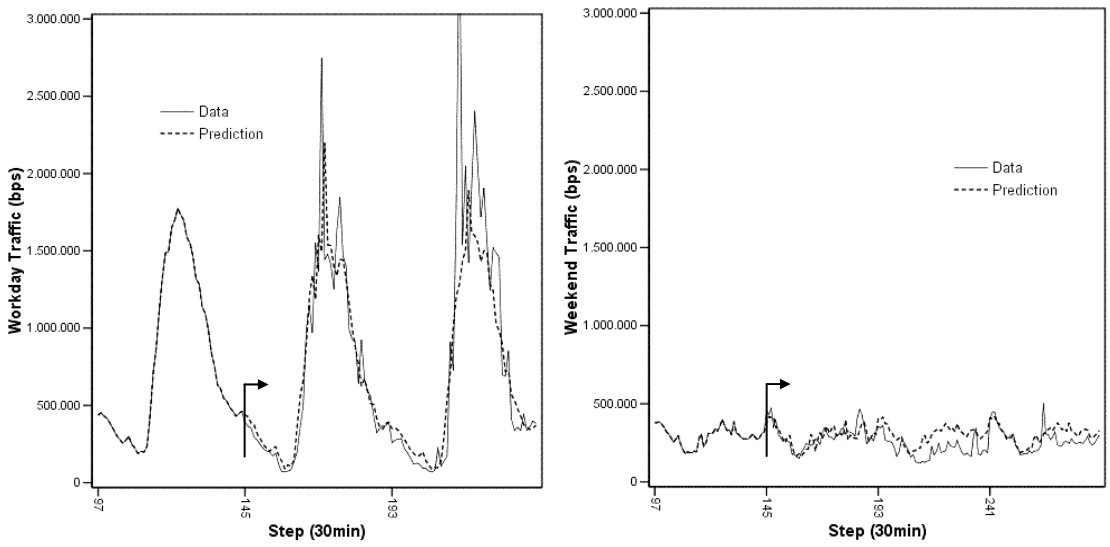
The proposed method has two advantages, it is based first, on a powerful multi model partitioning algorithm, (MMPA) proposed by Lainiotis [6, 7], known for its stability and well established in identification and modeling, and secondly, on easy to find and collect datasets. Bandwidth use is the most common set of data and almost all network administrators monitor the bandwidth utilization for their servers, LAN/VLAN users, and network connections.

In the following sections of this contribution, we first present some models of network traffic that can be used by MMPA, then we present the multi-model partitioning algorithm (MMPA), and finally, we present the detection results of the MMPA, using real datasets collected at the campus network of the Tech. Educ. Inst. of Athens.

2 ARMA AND STATE-SPACE TRAFFIC MODELS

Network traffic and utilization demonstrate daily, weekly and even yearly periodicity. One method to model such behavior is to apply the Seasonal ARIMA time series models. In a previous work [8] we successfully modeled the network utilization using SARIMA models. Here we apply the same method in order to model the daily network behavior. The daily behavior is first classified in two categories: a) the traffic during normal working days, and b) the traffic during weekends and holidays.

After several tests, we concluded to a SARIMA model that satisfies both categories. Provided that the past period data belong to the same category with the forecasting period, the seasonal ARIMA $(1,1,1)\text{x}(0,1,1)_{48}$ model predicts satisfactorily the future network traffic, as also shown in Figure 1:



(a) (b)

Figure 1. Daily traffic prediction using the Seasonal ARIMA (1,1,1) x $(0,1,1)_{48}$ model: (a) working day traffic, (b) weekend or holiday traffic. Prediction starts at step 145. The previous period (steps 97 to 144) is replaced by the average of all past periods (days) of the same type (weekends or working days).

Equation (1) represents the above model mathematically. The autoregressive (AR) and moving average (MA) parameters of the model are: $\varphi_1 = 0.413027$, $\theta_1 = 0.942437$, $\Theta_1 = 0.959323$.

$$\varphi(B)\nabla^1\nabla_{48}^1 X_k = \theta(B)\Theta\left(B^{48}\right)u_k \tag{1}$$

where, $\varphi(B) = 1-\varphi_1 B$, $\theta(B) = 1-\theta_1 B$, $\Theta\left(B^{48}\right) = 1-\Theta_1 B^{48}$, and, operators B & ∇ are defined as: $B^s X_k = X_{k-s}$ and $\nabla_s^D = (1- B^s)^D$. Therefore, the analytic expression for model (1) will be:

$$\left(1-\varphi_1 B\right)\left(1-B\right)\left(1-B^{48}\right)X_k=\left(1-\theta_1 B\right)\left(1-\Theta_1 B^{48}\right)u_k \quad \Rightarrow$$

$$X_k-\left(1+\varphi_1\right)X_{k-1}+\varphi_1 X_{k-2}-X_{k-48}+\left(1+\varphi_1\right)X_{k-49}-\varphi_1 X_{k-50}=u_k-\theta_1 u_{k-1}-\Theta_1 u_{k-48}+\theta_1\Theta_1 u_{k-49} \quad (2)$$

In order to be compatible to the notation of the MMPA and Kalman algorithms [9], model (2) can be rewritten in a state-space form. Based on the innovations representation of an ARMA process, an ARMA model of the type: $z_k+a_1 z_{k-1}+\ldots+a_n z_{k-n}=b_0 u_k+\ldots+b_m u_{k-m}$, can be written in the following state-space form [9]:

$$x_{k+1}=\begin{bmatrix} -a_1 & I & \cdots & 0 & 0 \\ -a_2 & \vdots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \cdots & I & 0 \\ -a_{n-1} & 0 & \cdots & 0 & I \\ -a_n & 0 & \cdots & 0 & 0 \end{bmatrix} x_k+\begin{bmatrix} b_1-a_1 b_0 \\ b_2-a_2 b_0 \\ \vdots \\ \vdots \\ \vdots \end{bmatrix} u_k, \qquad z_k=\begin{bmatrix} I & 0 & \cdots & 0 & 0 \end{bmatrix} x_k+b_0 u_k \quad (3)$$

Apart from normal traffic and utilization, numerous other traffic conditions also exist such as line failures or network misuse. These events are not periodic and they occur at random instances and therefore, the above seasonal models are not very helpful. Here we will model two such events using state-space models, a sudden rise (peek) in traffic (i.e. misuse) and, a constant traffic rate (i.e. failure). The corresponding state equations for these cases are:

$$z_k=x_k+v_k, \quad and, \quad a)\ x_{k+1}=10\cdot x_k, \quad b)\ x_{k+1}=x_k\ \left(=0\right) \quad (4)$$

The four candidate models described above will be used by the MMPA method to detect the type of network utilization. A sample of each traffic sequence represented by the four models is shown in Figure 2.

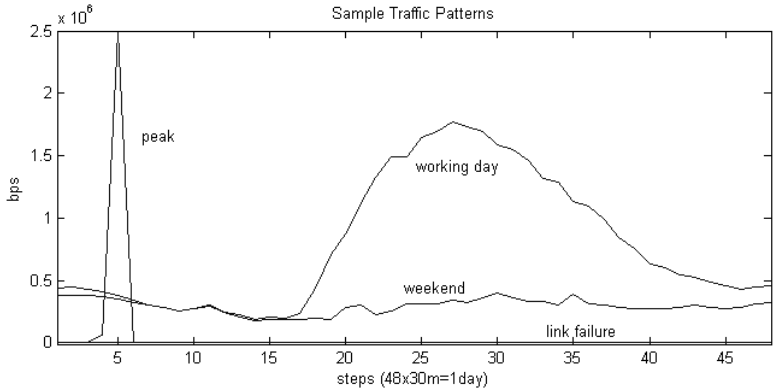


Figure 2. Sample traffic sequence with different traffic conditions.

3 THE MULTI-MODEL PARTITIONING ALGORITHM

It is clear that the correct model describing the network traffic at a certain period will be one of a family of models described by equations (3)-(4). The problem is then to select the correct model *n* among various "candidate" models. By identifying the correct model we identify the type of traffic and consequently if it is a normal behavior or a traffic anomaly.

Our approach has been to use a parallel bank of *N* Kalman filters, which operate concurrently on the same measurements (Figure 3). Each filter is based on one of the traffic models of (3)-(4). At time step *k*, each filter processes the measurement z_k and produces a state estimate $x(k / k; n)$ of x_k, conditioned on the hypothesis that the corresponding model is the correct one.

At a second level, the MMPA uses the output of all filters to select the most likely model as the one that maximizes the a posteriori probability density $p(n / k)$. This density can be calculated recursively [6], [7]:

$$p(n/k) = \frac{L(k/k;n)}{\sum_{i=1}^{N} L(k/k;i) \cdot p(i/k-1)} \cdot p(n/k-1) \tag{5}$$

where: $$L(k/k;n) = \left| P_{\tilde{z}}(k/k-1;n) \right|^{-1/2} \, e^{-1/2 \left\| \tilde{z}(k/k-1;n) \right\|^2 \cdot P_{\tilde{z}}^{-1}(k/k-1;n)} \tag{6}$$

and where $\tilde{z}(k/k-1;n)$ and $P_{\tilde{z}}(k/k-1;n)$ are the conditional innovations and corresponding covariance matrices produced by the conditional Kalman filters.

At each iteration, the MMPA algorithm selects the model that corresponds to the maximum a posteriori probability as the correct one. This probability tends (asymptotically) to one, while the remaining probabilities tend to zero. If the model changes, the algorithm senses the variation and increases the corresponding a posteriori probability, while decreasing the remaining ones. Thus the algorithm is adaptive in the sense of being able to track model changes in real time. This procedure incorporates the algorithm's intelligence.

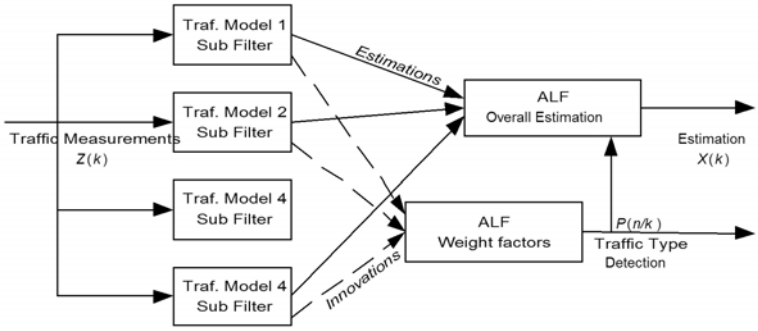


Figure 3. Structure of the Multi-Model Partitioning Algorithm for Network Anomaly Detection.

The above multi-model partitioning algorithm presented by Lainiotis [6] possess several interesting properties: A) Its structure is a natural parallel distributed processing architecture and hence it is more suitable to current computers clusters. B) By breaking a large and/or non-linear model to smaller sub-cases the algorithm has a much smaller dimensionality and hence much less architectural complexity. C) Although computationally intensive, it works faster due to parallelism and hence it is much more appropriate for real-time applications. D) It is more robust than any single filter as it is capable to isolate any diverging sub-filter. Numerous applications and simulations in the literature also show this. E) The algorithm is well structured and modular and it is easy to implement and modify on any standard programming environment (e.g. MATLAB).

4 DETECTION RESULTS USING REAL TRAFFIC DATA

In order to test the efficiency of the MMPA method, we use real data from the TEI of Athens Campus Network. The test dataset is created from real cases and, as shown in Figure 4, the dataset represents one week of traffic i.e. five working days and a weekend. In this dataset we introduced two link failures and two high traffic peaks.

The MMPA has four Kalman subfilters that correspond to the four types of traffic we are investigating. The a posteriori probability density $p(n / k)$ of each model is used to identify the type of the observed traffic. The model that maximizes this quantity is selected. If the selected model is also the correct day pattern of the current day, then we have normal traffic conditions; otherwise, an anomaly is detected.

As shown in Figure 4, the proposed method detects successfully both, the changes from weekend to working days and vice-versa. The method detects equally well, traffic peaks (misuse) and traffic zeros (i.e. link failures). In addition to the successful detection, the adaptive algorithm executes each iteration in a fraction of a second, thus, permitting us to increase the sampling rate of the data collection. The default rate is to collect measurements through the router's MIB every 5 minutes. The adjustment of the sampling rate is essential for the online detection of network anomalies.

Further work based on these results includes, the monitoring and modeling of other MIB quantities that are related to network problems or network misuse, the minimization of the sampling interval in order to obtain

better reaction times, modeling of normal user behavior, and, modeling of more unusual activities or network problems.

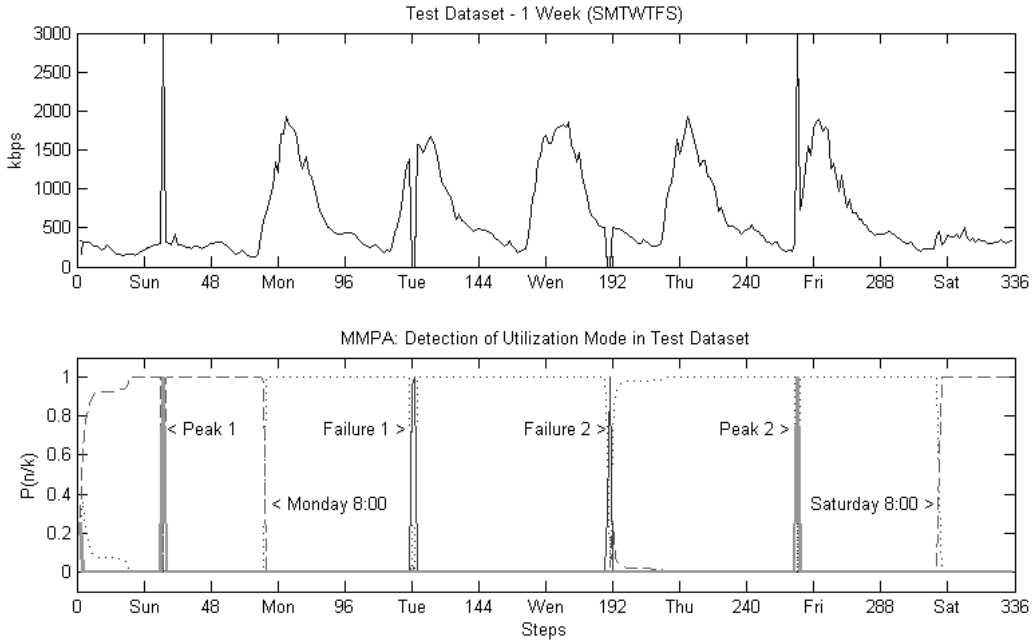


Figure 4. Test dataset for 1 week (SMTWTFS) containing peaks and failures, and, the MMPA successful detection of the changes and anomalies in the dataset.

5 CONCLUSIONS

In this paper we proposed an adaptive multi-model method for network unusual activity or misuse detection, using bandwidth utilization data. The proposed method uses past traffic data to model the normal periodic behavior of a network connection. ARMA and State-Space models were used for the traffic pattern modeling. The adaptive MMPA method processed successfully the traffic data collected in our campus network. The algorithm detected correctly all changes, failures or unusual activities included in the test datasets. The method is also very fast and it can perform equally well in real-time even with smaller than 5 min. sampling intervals.

REFERENCES

[1] Denning D. E. (1987). "An Intrusion-Detection Model". IEEE Transactions on Software Engineering, vol. SE-13, pp:222–232, Feb 1987.

[2] Debar H., Becker M., Siboni D., (1992). "A Neural Network Component for an Intrusion Detection System", *Proc of the 1992 IEEE Computer Society Symposium on Research in Security and Privacy, Oakland, CA May 1992*.

[3] Rhodes B., Mahafey J., Cannady J., (2000). "Multiple Self-Organizing Maps for Intrusion Detection", *Proc of the NISSC 2000 Conference*.

[4] Porras P., Neumann P., (1997). "Emerald: Event monitoring enabling responses to anomalous live disturbances", Proc of the 20th National Information Systems Security Conference, Baltimore, Maryland 1997.

[5] Eskin E., "Anomaly detection over noisy data using learned probability distributions" (2000). *In Proc. of ICML 2000, Menlo Park, CA, 2000*. AAAI Press.

[6] Lainiotis D.G., "Partitioning: A Unifying Framework for Adaptive Systems, I: Estimation", Proc. IEEE, Vol. 64, pp. 1126-1142, 1976.

[7] Katsikas S.K., Likothanassis S.D. and Lainiotis D.G., "AR model identification with unknown process order", IEEE Trans. Acoust., Speech and Signal Proc., Vol. ASSP-38, No. 5, pp. 872-876, 1990.

[8] Moussas V.C., Daglis M., Kolega E., (2005). "Network Traffic Modeling and Prediction using Multiplicative Seasonal ARIMA Models", *Proceedings of the 1st International Conference on Experiments/Process/System Modeling/Simulation/Optimization, Athens, 6-9 July, 2005.*

[9] Anderson B.D.O., and Moore J.B., (1979). *Optimal Filtering*, Prentice Hall, New Jersey.