Paper: ASAT-14-236-CM

14th International Conference on
AEROSPACE SCIENCES & AVIATION TECHNOLOGY,
ASAT - 14 – May 24 - 26, 2011, Email: asat@mtc.edu.eg
Military Technical College, Kobry Elkobbah, Cairo, Egypt
Tel: +(202) 24025292 –24036138, Fax: +(202) 22621908

Detecting Abnormal Network Traffic in the Secure Event Management Systems

A. Abd Elmomen[*], A. Bahaa El Din[†], A. Wahdan [‡]

Abstract: State-of-the-art intrusion detection and monitoring systems produce hundreds or even thousands of events every day. Unfortunately, most of these events are false positives, or irrelevant and can be considered as background noise, which makes their correlation, analysis and investigation very complicated and resource consuming. This paper attempts to simulate the modeling of background noise using the non-stationary time series analysis with lag smoothing Kalman filter. Then introduce and compare a second technique applying a multi-layered perceptron neural network with back propagation network; an approach that is used for the first time in modeling and correlating the background noise. DARPA Dataset is used to analyze and compare both techniques and finally a verification experiment is conducted using a gathered dataset from real network environment.

Keywords: Intrusion Detection, Alert Correlation, Time Series Modeling, Kalman Filtering, Neural Network

1. Introduction

The internet has become now the most commonly used means of communication among companies, business partners, and end users. Therefore, most organizations put their critical resources online, which increase cyber crime, attacks, and malicious activities. Despite, the importance of firewalls and antivirus applications, these tools are not sufficient to protect data from network attacks. Since most attacks evolve from inside the network, event management and intrusion detection systems (IDSs) are necessary as a complementary solution. IDSs and monitoring systems produce hundreds or even thousands of events every day [1], most of those alerts are false positive or irrelevant [2, 3]. So correlation of such events is required in order to reduce the huge amount of alerts as well as predict the high-level-structured network threats. Alert correlation process has three major phases: alert collection, alert aggregation and verification then finally the high-level alert structures [4]. Each phase has a specific role in alert reduction. We notice that the alert verification component in the second phase reduces the highest number of alerts. Due to the existence of false and irrelevant positives, those types of alerts may contain interesting traffic that needs more investigation. This research focuses on the alert verification component by modeling and analyzing the background noise and therefore detecting any irregular traffic. Those anomalies may be a new type of attack, an

[*] Senior Security Engineer, France Telecom – Orange Business Services, Cairo, Egypt, eng.ahmedsaeed@hotmail.com .

[†] Ph. D. Computer and System Engineering Department – Faculty of Engineering – Ain Shams University – Cairo, Egypt, ayman.bahaa@eng.asu.edu.eg .

[‡] Professor, Computer and System Engineering Department – Faculty of Engineering – Ain Shams University – Cairo, Egypt, wahdan47@hotmail.com .

attempt of an existing attack, or just an abnormal behavior. In this paper, we model regularities of the background noise events flow, so we can find interesting phenomena or anomalies that need more attention. First, we use non-stationary time series model and lag smoothing Kalman filter to model regularities of irrelevant traffic alerts inspired from research [5]. Second, we introduce a multi-layered perceptron neural network approach (MLP) with back propagation network (BPN) to model normal flow behavior and predict future values. Finally, both methods are implemented and compared using Matlab toolbox. We utilize DARPA Data set to experiment and analyze both methods then we made a verification experiment using data set collected from real network setting.

2. Related Work

The present approach focuses on alert aggregation and reduction rather than activity tracking and content improvement. Moreover, the background noise is of more interest in this approach than the high impact alerts. This is how such approach is different from other methods applied in this field, except for few researches that focus on the same interest. In the following lines, some related works focusing on the alert correlation process, yet from a different point of view will be presented.

The first research is concerned with situation and projection axes method. It was introduced by H.Debar and A.Wespi in 2001 [6] through inserting an implicit aggregation component. According to this method, alerts are projected along three axes: source address, destination address, and alert class. If the alerts match, hence they are aggregated together in a Meta alert.

The second approach is based on probabilistic and expected similarity, which was introduced by Valdes and Skinner [7] as the first probabilistic approach in alert correlation. They define a similarity value between 0 and 1, 1 meaning perfect match for each comparable alert attribute (source IP, destination IP, port numbers, attack class, sensor, time, etc.). Then similar alerts are grouped together to form Meta alerts, the attributes of which are compared with the new alert to see whether matches or not.

Many researches were introduced in the statistical causality analysis, Qin and Lee's is the most famous [8,9]. It groups alerts in a time series model and then try to find causality relationship between them. If they are related then they are grouped as Meta alerts.

Another useful technique in event management is alert correlation using data mining. It was first introduced by S. Manganaris, M. Christensen, D. Zerkle, and K. Hermiz [3]. They mine association rules and frequent item sets using real data from IBM’s Emergency Response Services. They report hundreds or even thousands of innocent alerts per day and sensor in real environment. The goal is to model the innocent behavior and filter out those alerts by taking into account the alert context in terms of other alerts. For more details about the association rules refer to [10, 11].

On the other hand, Julisch uses data mining in root cause analysis in [2, 12, and 13]. The root cause of an alert is the phenomenon causing the alert to occur. Empirical results show that only few root causes create large proportion, up to 90%, of alerts. Examples of root causes are misconfigurations, flawed TCP/IP stacks, and proxies, the behavior of which resembles scanning activity.

Finally yet importantly, J. Viinikka and H. Debar [14] introduce stationary time series modeling to analyze background traffic and detect anomalies, which is the core of the study in this paper, where two other approaches are built. The stationary model is based on modeling regularities in alert flows with classical time series methods like Trend modeling, Estimated Weighted Moving Average (EWMA) and Stationary auto regressive model (AR). The proposed methodology builds on three basic ideas:

1. Instead of individual alerts, alert flows are processed.
2. The component of a flow that can be attributed to the normal system behavior is modeled using autoregressive time series model.
3. Only deviations from the normal flow behavior are reported to the operator as meta-alerts.

3. Event Flow Data Source

The accuracy of our model depends greatly on the source of the input data. Standard laboratory data set is being used -1999 DARPA intrusion detection evaluation Data set - so that the obtained results can be evaluated and compared. This data set is extracted from DARPA for twelve continuous days. The data set flow is composed of two types of alerts that are most of the time false positive or irrelevant. The portsweep and the ipsweep alerts, which are alerts of category probes, represent surveillance sweep to determine which hosts are listening on a network. This information is useful to an attacker in staging attacks and searching for vulnerable machines. It depends on sending ICMP Ping packets to every possible address within a subnet and wait to see which machines respond. ICMP messages are part of system functioning but can be used as an information gathering or a part of a multi-step attack. This type of alerts is chosen to prove that some abnormal behavior can be observed from the irrelevant alerts flows, which can be an interesting structure attack. For more details about the DARPA lincolin laboratory dataset, reference may be made to [15].

Figure (1) below shows the data set flow composed of two types of alerts from DARPA data set for twelve continuous days.

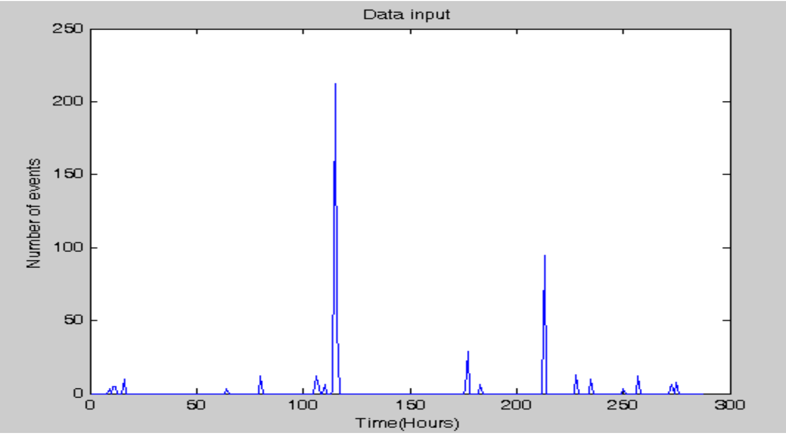

Figure (1): Graph of data set flow collected for twelve successive days

From the figure above, it can be noticed that most of alert intensity varies between 0 and 50 alerts per hour, except for 2 peaks that exceed this number. Those peaks do not guarantee that no other attacks exist during the twelve days. However, from the background noise traffic perspective, they indicate that abnormal behaviors may happen during those peaks. Even though, one cannot assure that such peaks represent abnormal behavior or a structured attack. In next sections, data flow normalization models will evaluate whether the two peaks constitute abnormal behavior or an attack.

4. Non Stationary Time Series Model

4.1 Overview

There are three types of time series models: trend modeling using Exponentially Weighted Moving Average (EWMA), stationary Auto Regressive (AR) time series and non-stationary AR time series. Here non-stationary AR time series modeling is chosen for many reasons. First, because it helps avoid limitations of the stationary AR model. Second, the high accuracy of this model is impressive. Third, it gives a stable performance regarding alert filtering and anomaly detection. For these reasons, the results of non-stationary AR model are easier to interpret than with the stationary model, and less dependent on the data set than with EWMA model. The non-stationary time series model depends on modeling the normal flow behavior then analyzing the difference between the observed flow behavior and the modeled output to detect the deviation from the normal profile. Unlike the stationary algorithm, this one models the alert flow directly without removing any component. The parameters of the model are time-dependent and can be estimated with recursive algorithms like Bayesian filtering, Kalman filter and Kalman lag smother. Matlab is used to simulate the experiment and analyze the results.

The non-stationary AR (p) model of degree p is defined as:

$$y_t = \sum_{k=1}^{p} a_t^k y_{t-k} + e_t \tag{1}$$

where y represents the observations, e represents the white noise and a_t^k represents the time-varying model parameters.

For more details about the algorithm used in non-stationary time series and smoothing Kalman filter, refer to [5].

4.2 Specifying the Model Degree

Two important variables should be specified in order to run the experiment; the sampling interval and the model degree. The sampling interval only affects real time detection of anomalies. Sampling interval t_s is chosen equal one hour to stabilize this parameter and focus on the model degree. Since the background traffic is to be analyzed, real time is not the focus in such research, but may be considered in the future. Model degree (p) indicates the number of previous samples $y(t)$ used in algorithms in order to predict the next sample $y(t+1)$.

4.3 Anomaly Detection

In the neural network model, anomaly detection can be easily specified. The abnormal component z_t is represented by the difference between the actual and the predicted data. Through analyzing the abnormal component and comparing it with a threshold, anomalies can be detected. Threshold can be determined from the formula utilized in the non-stationary algorithm.

$$Threshold = n\sigma_Z \tag{2}$$

where the standard deviation can be calculated from equations

$$\sigma_Z = \sqrt{\frac{\sum(Z-\bar{Z})^2}{N}} \qquad (3)$$

$$\bar{Z} = \frac{\sum Z}{N} \qquad (4)$$

where n represents the control limit width so it indicates how large a deviation from trend is acceptable, and N represents the number of values that exist.

If the abnormal value Z_t exceeds the threshold, it is considered as an interesting behavior for investigation $|Z_t| > n\sigma_Z$.

4.4 Experiment Results

Two important parameters are needed to tune in order to obtain perfect results, sampling interval and model degree. A fixed sampling interval equal 1 hour for all experiments are chosen. For model degree, several experiments are needed to get the best model degree since it depends on the data set itself and the number of anomalies detected. Putting into consideration that model degree should not be high in order to reduce the complexity of computation and since sampling interval was chosen to be equal one hour.

Several model degrees were tried *p* (from *p* = 5 till *p* = 100) in order to detect the one that is suitable for the input data. The model degree is chosen according to the least root mean square error (RMSE) value.

Based on the statistics in Table (1), the model degree, which gives the lowest RMSE value, is 5.

Table (1): RMSE for different model degrees

Model Degree (p)	5	8	15	25	45	60	80	100
RMSE	0.2816	2.1789	5.1388	4.4953	7.9728	9.9881	11.355	6.8512

Figure (2) illustrates a graph of RMSE values for different model degrees, where the one with the lowest RMSE is chosen.

The graph illustrates that the RMSE values do not have a linear relationship with the model degree. It could be observed that for higher model degrees, the RMSE value is too large, and when the RMSE value exceeds 6, it does not give same experiment results. This proves that accuracy of results is absolutely related to RMSE values. The lowest model degree should avoid computational complexity but sometimes more importance is to be given to the least RMSE value to get results that are more accurate.

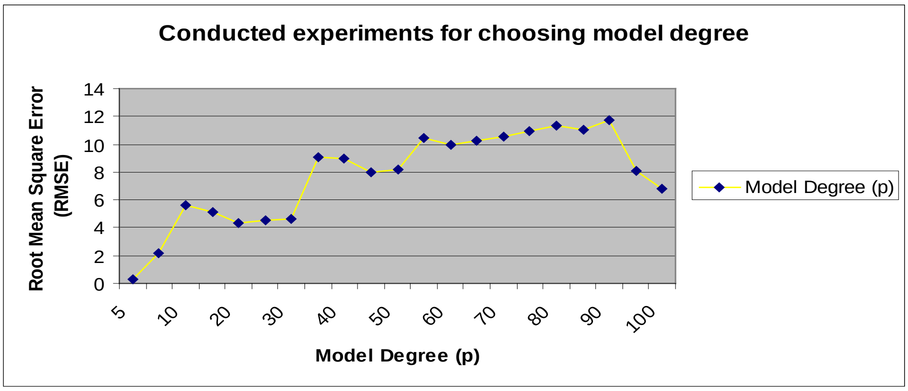

Figure (2): Conducted experiments for choosing the model degree

With the model degree (p) selected equal to 5, Figure (3) represents the error (z_t) and the anomalies (if any) that should be investigated.

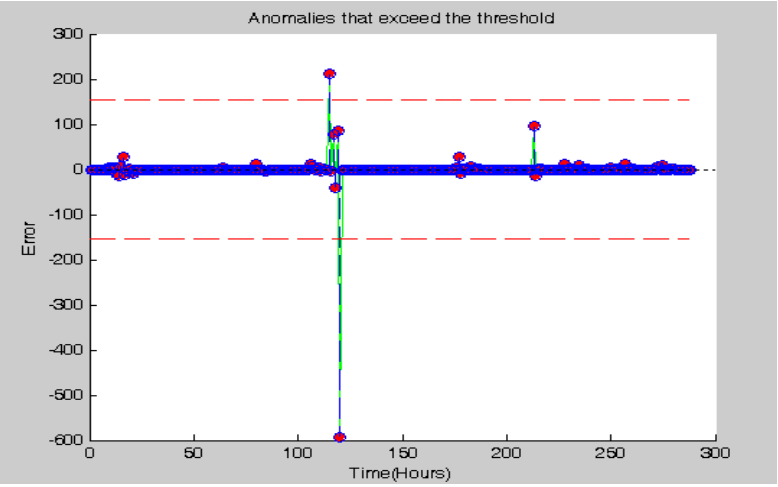

Figure (3): Error model for port/Ipsweep alert flow

As shown in Figure (3), there are two anomalies exceeding the threshold in two consecutive hours. This indicates that an interesting behavior exists, which requires attention and analysis. The non-stationary model detects only one peak from the dataset flow as anomaly; however the second peak has a big value but does not exceed the threshold.

After checking the DARPA dataset during the two consecutive hours where the abnormal behavior exists, a critical alert called “sechole” and another one called “secret” are found. The first indicates a privilege elevation, while the latter, “secret” indicates a file transfer from a trusted zone to another untrusted zone.

Therefore, after analyzing the two consecutive anomalies detected within the system, a structured attack is captured.

5. Artificial Neural Network Model

5.1 Overview

After applying the non-stationary auto regressive model in the experiment, another model is introduced using neural network that is thought to be more accurate and would give better results. The concept of a neural network was used in various different applications including the intrusion detection researches and the evaluation of intrusion detection, where it gives tremendous results. However, this is the first time to apply neural network in analyzing the background traffic.

A neural network is chosen because, similar to the non-stationary method, it depends on one-step prediction. In addition, it can produce non-linear model that is more convenient to the behavior of background traffic. On the other hand, the back propagation algorithm is chosen because it is the best tool to use when dealing with new data. Back propagation network is a layered, feed forward network that is fully interconnected by layers. For more details about Back propagation algorithm and Artificial Neural Network (ANN), it is advised to check references [16, 17].

5.2 Proposed Neural Network Model

In the present scenario, it is decided to utilize a multi layered perceptron (MLP) feed forward network (with one hidden layer) that is fully connected, besides the back propagation as a learning algorithm. Use is made of twenty-four input neurons the values of which are the same that those previously processed by the hidden layer and the transfer functions to predict one future value as an output. The target error is 0.015 or 500 iteration. Figure (4) shows the proposed neural network architecture.

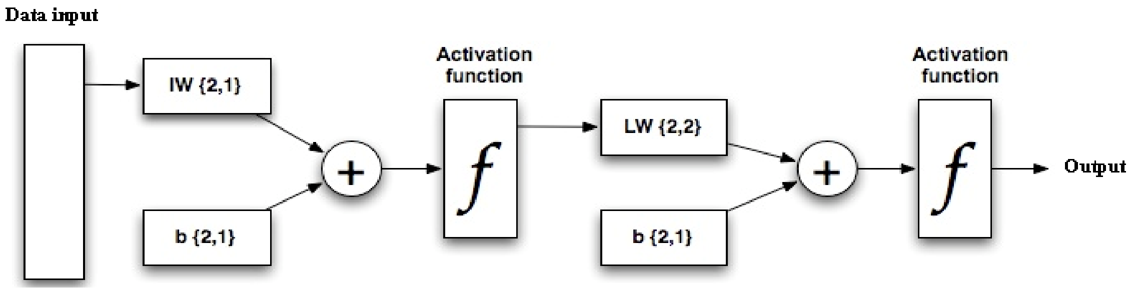

Figure (4): Proposed MLP ANN

The data are divided into two segments: the first one to train the network, the second one to obtain the forecasting data and finally calculate error difference between actual and forecasted data. Then the performance of the predicted data is measured.

5.3 Anomaly Detection

In the neural network model, anomaly detection can be easily specified. The abnormal component (z_t) is represented by the difference between the actual and the predicted data. Through analyzing the abnormal component and comparing it with a threshold, anomalies can be detected. Threshold can be determined from the formula utilized in the non-stationary algorithm, equation (2), where the standard deviation can be calculated from equations (3,4). If the abnormal component (z_t) exceeds the threshold, it is considered as an interesting behavior for investigation.

5.4 Experiment Results

Neural network is one of the most accurate algorithms. That is why it is believed to be better than the non-stationary algorithm. First, training the network model is performed using the first 24 values (the number of hours in each day) as the input data. Then iterations are processed until the network reaches its target performance, which is equal to 0.015.

Figure (5) illustrates that the performance achieves its 0.015 goal after 70 iterations. After training the network, the model becomes ready to predict and analyze new data from the real data set under consideration.

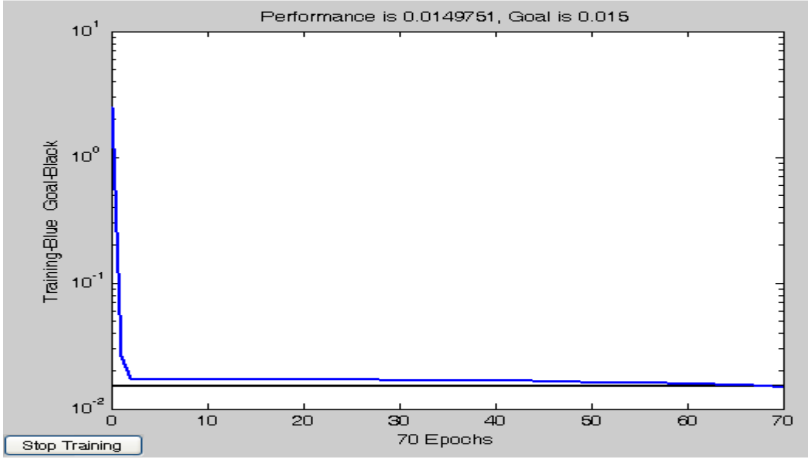

Figure (5): Neural network training phase

Figure (6) shows both the actual and the predicted data over the 290 hours of the data set. The first 24 values (hours) are zeros because the first day is chosen as an input to the model to train it so no values are assigned for them. After that, the type of alerts port/ipsweep does not exist until the hour number 60.

From the difference between actual and predicted, the error model (z_t) can be displayed to discover any abnormal behavior that exceeds a certain threshold.

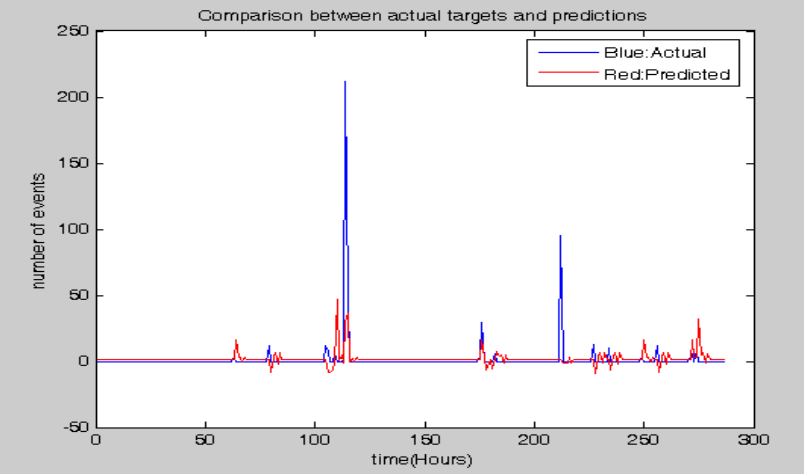

Figure (6): Actual vs. predicted data results

As noticed from Figure (7), the threshold red line value is not the same as previous experiment; this is valid because the threshold calculation uses standard deviation equation, which depends on the input data set, so it varies each time a different data set is used. It is observed that the two peaks existing in the data set flow are detected as anomaly; however, the non-stationary model detects only one peak of them.

Beside the first attack detected in the previous experiment, on checking - from DARPA dataset - the period during which the second anomaly exists, another type of alert called "ncftp" is discovered. This attack affects and intercepts the FTP applications. In conclusion, neural network model successfully discovered another attack existing on DARPA data set that was not detected by the non-stationary model.

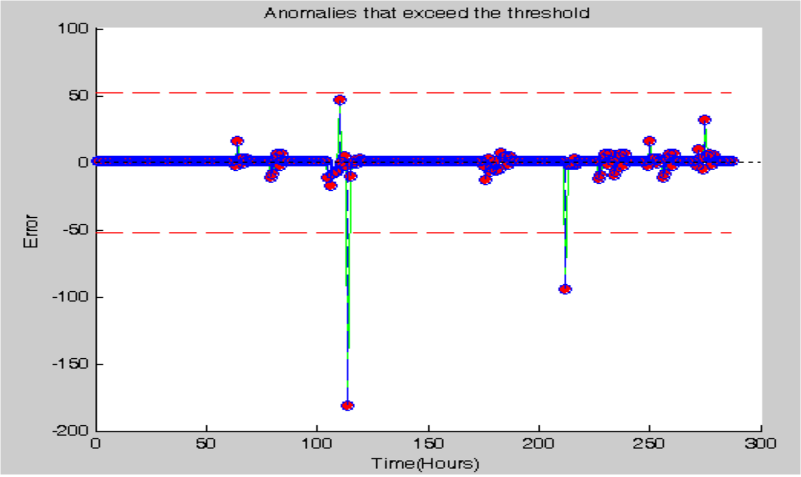

Figure (7): Error model for port/Ipsweep alert flow

6. Comparison

Based on the nature of both experiments, it can be noted that the neural network approach has many advantages over the non-stationary model.

Actually, the non-stationary model depends on the model degree, which needs modifications and tuning in each new data set. On the other side, the neural network is more flexible and can be used with various types of data without changing their parameters. Sometimes the non-stationary technique needs a high model degree to guarantee accurate results, which increases the computational complexity and the time processing. On the contrary, neural network consumes moderate processing time depending on the number of iterations used.

Moreover, the neural network model detects more anomalies that are not detected by non-stationary model, and this does not mean that non-stationary model fails to detect anomalies, since it is normal that critical alerts are already detected by the intrusion detection systems. However, it indicates the efficiency and the accuracy provided by the neural network model in analyzing the background noise.

Finally, the error measurement using the Root Mean Square Error (RMSE) equation is considered the most accurate way to evaluate and compare between models as used before in determining the best model degree of the non-stationary model. When experimenting data set with non-stationary and neural network method, both models successfully detect the inserted anomaly. However, by calculating the RMSE of both experiments, it is found that the neural network model has less RMSE than the non-stationary AR model with the best model degree equal 5, and hence it is better.

From Table (2), it is perceived that neural network model was benchmarked against non-stationary model.

Table (2): RMSE for both models

Model	Non stationary AR model (p=44)	Neural network model
RMSE	0.2816	0.084

7. Verification Experiment

7.1 New Event Flow Data Source

Verification tests with a new data set from a real network environment are needed in order to be sure that the model works with the same behavior as expected.

Events are gathered from two Juniper IDP devices placed in the main site of the Egyptian universities network, which is connected to the internet and to other IPVPN networks. The first IDP device was placed between the IPVPN networks and the headquarter gateway router, the other device was placed inside the Dematerialized zone (DMZ). Figure (8) shows the network architecture of the main site of the Egyptian universities network from which the data set are collected.

The number of events occurring each hour is recorded for seven continuous days. Figure (9) is a graph of events of the collected data set.

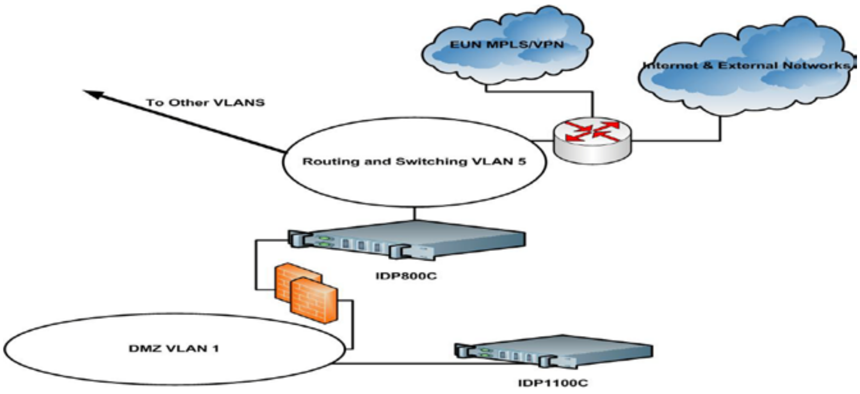

Figure (8): Architecture of part of the Egyptian universities network used for collecting data set

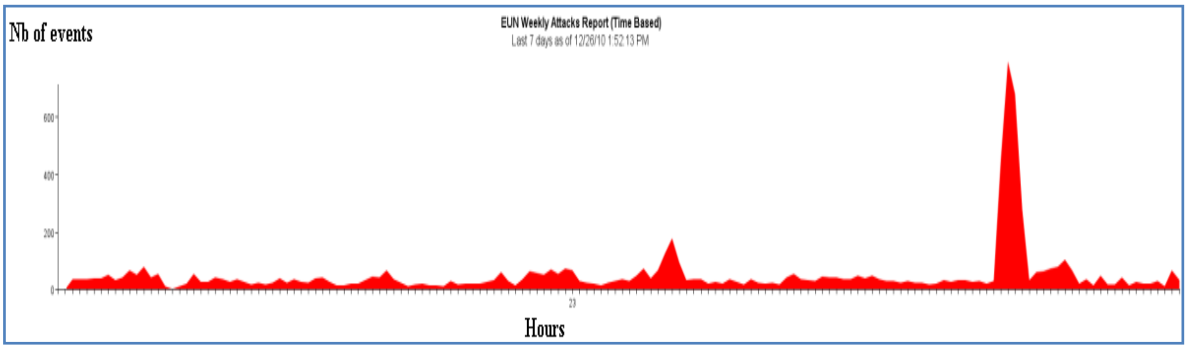

Figure (9): Graph of events collected for seven days

It can be noticed from the graph above that number of events per hour varies between zero and twenty events except for some extreme values (peaks) that exceed this limit. It will be seen in verification experiments below how the noticed peaks could be handled by both methods. The verification experiment will also check the accuracy of both systems and confirm whether the neural network is still benchmarked or not.

7.2 Time Series Experiment

Despite the fact that the model degree of the verification experiment can be the same as the previous experiment, another model degree will be chosen to better detect anomalies and to obtain the least root mean square error. From the new data set, anomalies based on the remarked triggers can be easily identified. This time the model degree will be chosen based on the detected anomalies, putting into consideration the lowest possible model degree to reduce complexity.

Several experiments are conducted with different model degrees (from p=2 to p=95) to choose the best one. It is noticed that higher model degrees (from p=50 and higher) do not detect peaks in data flow as anomalies. Also, the model degrees (between p=20 and p=50) do not detect the anomalies correctly. In fact, the only model degrees that detect the peaks as anomalies correctly are equal to 15 and lower. This does not provide the least root mean square error. However, it leads to less computational complexity and less time processing.

As noticed in figure (10), few anomalies are actually detected but delayed by four samples, which is normal due to some parameters initialization. It is also noticed that the error model normally reflect the peaks existing in the data input flow.

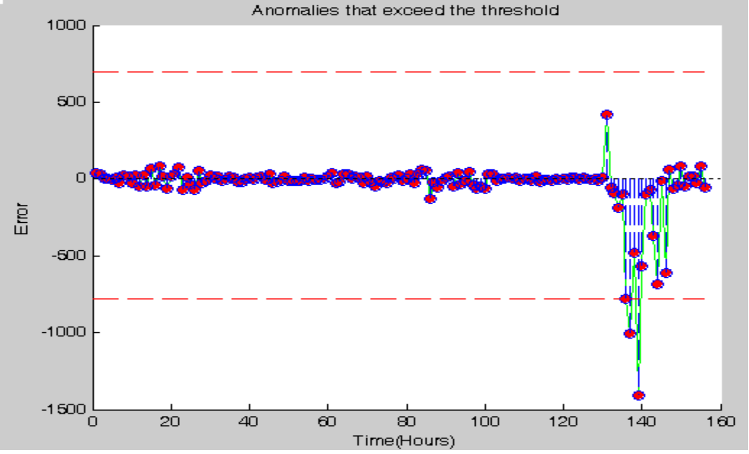

Figure (10): Anomalies detection with P=15

7.3 Neural Network Experiment

Figure (11) below illustrates that only two iterations are needed in order to reach 0.015 which is too small compared to the previous experiment with 70 iterations.

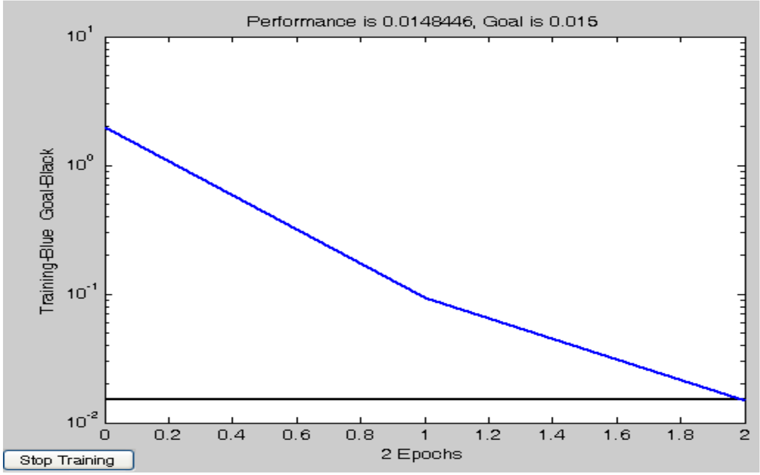

Figure (11): Neural network training phase

Figure (12) shows actual-versus-predicted data and it is noticeable that there is a small difference between the actual and the predicted compared to the first experiment.

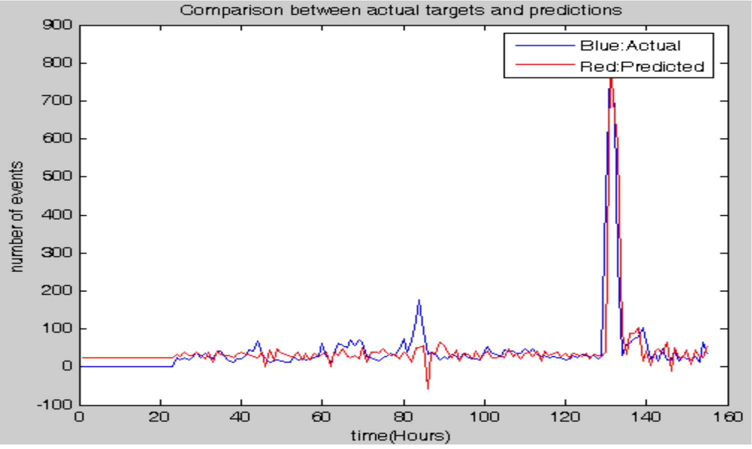

Figure (12): Actual vs. predicted verification experiment

After the new data set was verified by the neural model, it could be concluded that it detects anomalies with excellent results. Figure (13) shows the two detected anomalies that need attention. The two anomalies were represented by two peaks in the data set that exceeds the variation limit. In addition, some peaks in the middle of the data flow were not detected as anomalies by the model. This proves that the presented model does not detect the variations between values. It nevertheless detects the anomalies only, which is an indication of its high accuracy.

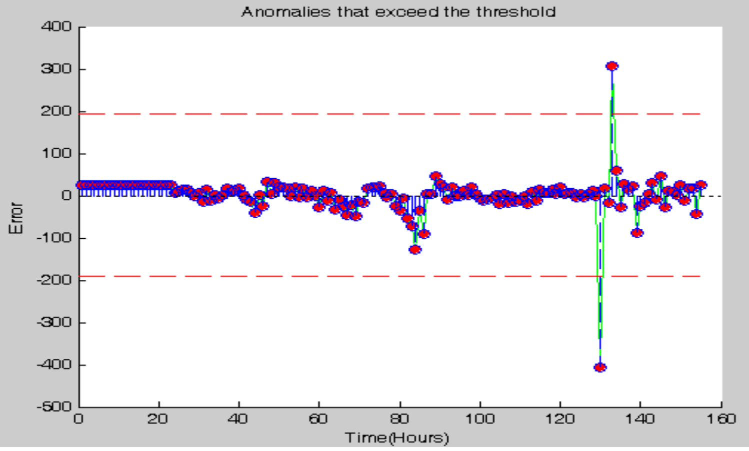

Figure (13): Anomaly detection from the new data set

7.4 Verification Experiment Results

The evaluation criteria based on the root square mean error will be applied. It is considered the most accurate measurement based on the first experiment results.

When calculating the root mean square error of both models, it is found that both their RMSE's are higher than the first experiment results. However, they are still relatively small and acceptable. Table (3) shows a comparison between RMSE values of all experiments. As noticed from Table (3) the RMSE value of non-stationary model is increased in the verification experiment with a much higher value compared to the neural network model, hence the RMSE value of neural network is better than non-stationary.

In fact, the neural network model has lower RMSE values than the non stationary model in both the first and the verification experiment, which indicates that the neural network model is more accurate and generating less error; therefore it can benchmarked against non stationary model in term of background traffic analysis.

Table (3): RMSE values of all experiments

RMSE\Model	Non stationary AR model	Neural network model
first experiment	0.2816	0.084
verification experiment	40.3	1.4279

8. Conclusion and Future Work

In this research, the background traffic flow is analyzed for twelve consecutive days taken from DARPA dataset. Non-stationary time series model with lag smoothing Kalman filter is implemented using Matlab toolbox, which is the most recent technique for analyzing background traffic. Then a new technique is introduced, namely, the multi layer neural network with back propagation algorithm with the use of Matlab. By conducting a comparison between both techniques, it is found that the neural network approach gives better results in terms of accuracy regarding detecting anomalies and was benchmarked against the non-stationary time series model with a lower RMSE value even after conducting a verification experiment with a different data set of seven continuous days from a real network environment.

In fact, modeling background traffic using neural network is a promising topic that needs further research, experimentation as well as analysis and evaluation. Endless opportunities in this field are open for discussion and research. Future work shall involve testing the neural network model in different environments and observing its accuracy to confirm this model as a general model for detecting anomalies in background traffic. Moreover, focusing on the alert flow is also significant; so more researches are to be carried out in order to get more detailed alert flow. This can be done by collecting events from different types of devices such as HIDS, Syslog servers, and SNMP traps, in order to reduce the sampling interval from 1 hour to 1 min, leading to more accurate detection of the real time anomalies in the background traffic. Another suggestion open for future investigation is to push the research in the sensor level instead of the flow level, as the sensor level is believed to give better results and reduce the effort of traffic analysis.

9. References

[1] S. Axelsson. The Base-Rate Fallacy and Its Implications for the Difficulty of Intrusion Detection. In Proc. of the ACM CCS'99, Nov. 1999.

[2] K. Julisch. Mining Alarm Clusters to Improve Alarm Handling Efficiency, Proceedings of the 17th Annual Computer Security Applications Conference (ACSAC2001), Dec. 2001.

[3] S. Manganaris, M. Christensen, D. Zerkle, and K. Hermiz. A Data Mining Analysis of RTID Alarms. 2nd International Symposium on Recent Advances in Intrusion Detection (RAID 1999), 1999. Available online:
http://www.raid-symposium.org/raid99/PAPERS/Manganaris.pdf

[4] Christopher Kruegel, Fredrik Valeur and Giovanni Vigna. Intrusion Detection and Correlation Challenges and Solutions, 1st.ed, Vol. 14, Springer Verlag, ISBN 0-387-23398-9. January 2005.

[5] J. Viinikka, H. Debar, Ludovic, A. Lehikoinen, M. Tarvainen. Processing intrusion detection alert aggregates with time series Modeling. 2009, Journal of Information Fusion Volume 10, Issue 4, October 2009, Pages 312-324

[6] H. Debar and A. Wespi. Aggregation and Correlation of Intrusion-Detection Alerts. In W. Lee, L. M´e, and A. Wespi, editors, Proceedings of the 4th International Symposium on Recent Advances in Intrusion Detection (RAID 2001), volume 2212 of Lecture Notes in Computer Science, pages 85–103, Heidelberg, Germany, 2001. Springer–Verlag.

[7] A. Valdes and K. Skinner. Probabilistic Alert Correlation. In W. Lee, L. M´e, and A. Wespi, editors, Proceedings of the 4th International Symposium on Recent Advances in Intrusion Detection (RAID 2001), volume 2212 of Lecture Notes in Computer Science, Heidelberg, Germany, 2001. Springer–Verlag.

[8] X. Qin and W. Lee. Statistical Causality Analysis of INFOSEC Alert Data. In G. Vigna, E. Jonsson, and C. Kruegel, editors, Proceedings of the 6th International Symposium on Recent Advances in Intrusion Detection (RAID 2003), volume 2820 of Lecture Notes in Computer Science, pages 73–93, Heidelberg, Germany, 2003. Springer–Verlag.

[9] X. Qin and W. Lee. Discovering novel attack strategies from infosec alerts, Proceedings of The 9th European Symposium on Research in Computer Security (ESORICS 2004), Sophia Antipolis, France, Sept. 2004.
URL: http://www.cc.gatech.edu/~wenke/papers/esorics paper 2004.pdf.

[10] H. Mannila. Data mining: machine learning, statistics, and databases, Proceedings of the Eight International Conference on Scientific and Statistical Database Management, pages 1–8, 1996. Available online: http://www.cs.helsinki.fi/mannila/postscripts/ssdbm.ps.

[11] H. Mannila. Methods and Problems in Data Mining. In F. Afrati and P. Kolaitis, editors, Proceedings of International Conference on Database Theory (ICDT'97), pages 41–55. pub:springer, 1997. Available online: http://www.cs.helsinki.fi/mannila/postscripts/icdt-tutorial.ps.gz

[12] K. Julisch and M. Dacier. Mining Intrusion Detection Alarms for Actionable Knowledge, Proceedings of Knowledge Discovery in Data and Data Mining (SIGKDD), 2002.

[13] K. Julisch. Clustering intrusion detection alarms to support root cause analysis. ACM Transactions on Information and System Security, 6(4), nov 2003. URL: http://www.zurich.ibm.com/kju/tissec.pdf.

[14] J. Viinikka and H. Debar. Time Series Modeling for IDS Alert Management, 2006.

[15] Lincoln Laboratory, Massachusetts Institute of Technology, "DARPA Intrusion Detection Evaluation", 1999. http://www.ll.mit.edu/mission/communications/ist/CST/index.html

[16] Tom Baker and Dan Hammerstrom, "Characterization of Artificial Neural Network Algorithms", Dept. of Computer Science and Engineering, Oregon Graduate Center, IEEE, 1989.

[17] Chun Lu, Bingxue Shi and Lu Chen, "An expandable on-chip Back Propagation learning neural network chip", Int. J.Electronics, Vol 90, No. 5, 331-340, 2003