Proceeding of the IEEE 28th
Canadian Conference on Electrical and Computer Engineering
Halifax, Canada, May 3-6, 2015

A Method for Intrusion Detection in Web Services Based on Time Series

Paria Shirani[1]* Mohammad Abdollahi Azgomi[1] Saed Alrabaee[2]
[1] School of Computer Engineering, Iran University of Science and Technology, Tehran, Iran
[2] Concordia Institute for Information Systems Engineering, Concordia University, Montreal, Canada
p_shira@encs.concordia.ca azgomi@iust.ac.ir a_alraba@encs.concordia.ca

Abstract— **A prevalent issue in today's society that has attracted much attention is anomaly detection in time series. Service-oriented architecture (SOA) and web services are considered as one of the most important technologies. In this paper, we propose a model for intrusion detection in web services based on the autoregressive integrated moving average (ARIMA). First, we apply the ARIMA model to the training data. Second, we forecast their next behavior within a specific confidence interval. Third, we examine the testing data; if any instance falls out of the range of the confidence interval, it might be an anomaly, and the system will notify the administrator. We present experiments and results obtained using real world data.**

I. INTRODUCTION

Web services are becoming a more pervasive foundation technology for integrating applications and exchanging data in service-oriented architectures (SOA) [1]. The security of a system based on web services depends not only on the security of the services themselves, but also on the confidentiality and integrity of the extensible markup language (XML) based, simple object access protocol (SOAP) messages, and the HTTP requests used for communication [1]–[3]. This diversity of standards makes web services more vulnerable to attacks. Consequently, intrusion detection systems (IDS) have been proposed to secure web services. In essence, there are two types of intrusion detection systems [4]: misuse detection and anomaly detection. Misuse detection system has predefined attack signatures stored in large databases against which it compares the input traffic. If there is a match point, this traffic is categorized as an attack. In anomaly detection system, the normal behavior of the traffic is defined; the anomaly detector monitors input traffic and compares their state to the normal baseline in order to search for anomalies. Anomaly-based IDS uses one of the statistical-based techniques, knowledge-based techniques, and machine learning-based techniques for intrusion detection.

In the literature, there is a lot of existing research on network anomaly based and misused based detection systems; however, intrusion detection for web services is a new approach, and there are fewer related works on this topic. In [5], an intrusion detection system named WS-IDS is presented that uses hidden Markov models (HMM) for intrusion detection. The use of HMM in their approach is limited in order to detect overflow attacks. In [6], they represent an adaptive intrusion detection and prevention (ID/IP) framework to protect web services against SOAP/XML/SQL attacks. In [7], they provide a solution to prevent DoS attacks in web services. This solution uses a multi-agent hierarchically-distributed architecture in four layers that performs a classification mechanism in two phases. In [8], an XML-based intrusion detection system for protecting the web services is designed and implemented. Their approach detect SQL injection, oversized payloads, and recursive payloads attacks with the use of filtering policies such as XML schema validation, syntax parsing, and message size restriction.

In this paper, we introduce a statistical-based technique for intrusion detection in web services based on time series, shown as follows:

- Capture the traffic.
- Create statistics based on the captured traffic.
- Predict the normal pattern for the current traffic.
- Measure the difference between the predicted traffic and real traffic.
- Signal to the administrator any abnormality detected.

Moreover, we review the basic definitions in time series and the ARIMA model as follows:

Definition 1: **(Time Series).** A time series, $T = t_1, t_2, \ldots, t_m$ is an ordered set of m real-valued variables [9].

Definition 2: **(ARIMA Model).** The autoregressive integrated moving average (ARIMA) model is a combination of AR (autoregressive) and MA (moving average) models that is defined as follows [10]:

$Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \ldots + \phi_p Z_{t-p} + e_t - \theta_1 e_{t-1} - \theta_2 e_{t-2} - \ldots - \theta_q e_{t-q}$

Or,

$\phi_p(B) Z_t = \theta_q(B) e_t$

where Z_i is the observed time series at time t; ϕ_i is parameter of the AR aspect; θ_i is parameter of the MA aspect; d is the differentiating degree; and e_t is a measurement of the white noise at time t. B is the back-shift operator (B-operator) that transforms an observation of a time series to the previous one j times; in other words, $B^j(Z_t) = Z_{t-j}$. Typically, time series data analysis is done for investigating trends, seasonal variation, cyclical variation and for the prediction of future values.

More precisely, the ARIMA model is the combination of both AR and MA models. Thus, by plot of time series, correlograms of auto correlation (ACF) and partial autocorrelation (PACF), we can perform the following steps:

- In plot of time series, if the data is not stationary in variance, we use log transformation.
- If the data is not stationary in mean, we use differentiating and will gain d parameter.
- For estimating the order of p, q and d, we examine the ACF and PACF correlograms.
- With plot of time series and ACF and PACF correlograms, the estimated pattern is specified.

*The first author is currently a PhD student at Concrodia University, Montreal, Canada.

978-1-4799-5829-0/15/$31.00 ©2015 IEEE

Finally, we evaluate our model by using a real dataset called Amzn [11]. Considering the amzn dataset as well as denial of service (DoS) attacks, the TP rate is roughly 89%, with 97% of accuracy and 0.01 error rates.

The remainder of this paper is organized as follows. In Section II, Box and Jenkins Model, and web service security attacks will be described. In Section III, we introduce related work. In Section IV, we apply the ARIMA model to training data; the model will be predicted and the anomalies will be detected. In Section V, we evaluate our method with the ROC curve. Finally, Section VI summarizes the conclusion and future work.

II. Background

In this section, we review the Box and Jenkins Model to show how the data are forecasted within a time series, and then we introduce some web service security attacks.

A. Box and Jenkins Model

The Box and Jenkins (1976) model is a mathematical model designed to forecast data within a time series. It alters the time series to make it stationary by using the differences between data points [12]. There are three stages in building a Box-Jenkins time series model [13]: model identification, model estimation, and model validation. After fitting the model, it will be forecasted (model prediction).

1) Model Identification: The main purpose of this stage is to get an idea of the p, q, and d parameters in order to yield an effective but parsimonious model. Parsimonious refers to the fact that this model has the fewest parameters and greatest degree of freedom among all models that fit the data [14]. The major tools used in this phase are plots of the series, correlograms of ACF, and PACF [14]. The input series for ARIMA model must be stationary; that is, it should have a constant mean, variance, and autocorrelation throughout time. Typically, in order to stabilize the variance, log transformation is used; to stabilize the mean, differentiating is used. The number of times the series requires to be differenced to achieve stationarity is reflected in the d parameter. There are some criteria, such as Akaka's information criterion (AIC), the AIC corrected (AICc) criteron, and the Bayesian information criterion (BIC), that can be used to select the best fitting parameters. The AIC criterion is based on the entropy concept and illustrates the rate of lost information by means of a statistical model. On the other hand, AIC specifies the number of parameters, and according to the parsimonious principle, the selected model should have the minimum AIC. For identifying the p and q values, we examine correlograms of ACF and PACF functions:

- If ACF has an exponentially decreasing appearance or decay sine-wave shape pattern and PACF spike at lag p, we have an $AR(p)$ model.
- If PACF has an exponentially decreasing appearance or decay sine-wave shape pattern and ACF spike at lag q, we have an $MA(q)$ model.

2) Model Estimation: The purpose of model estimation is to determine the best values for parameters. For estimation, we use criteria such as the maximum likelihood estimation (MLE) and the least square, where MLE is generally the preferred technique [13]. The MLE assigns values to model parameters and results in a distribution that assigns maximum probabilities to the observations. The higher the likelihood number, the more probable the observed data. Therefore, after determining the order of ARIMA, we use a maximum likelihood estimation to estimate initial parameter values, and use AIC to select the best fitting parameters [15].

3) Model Validation: After parameter estimation, we examine the model so that the residuals have zero mean and are stationary in variance. We use the ACF and PACF correlograms for residual examination. In this stage, we should validate the adequacy of the model. These conditions are as follows [13]–[16]:

- The residuals should be white noise.
- Whether the estimated parameters are significantly different from zero; in other words, the sequence requirements mean zero.
- The fitted model is adequate.

If the conditions are not satisfied, a more appropriate model is required. That is, we must return to the model identification step and try to develop a better model.

Model Prediction

One of the most important purposes of time series is forecasting. The estimation of the parameters in the last stage of the Box and Jenkins model is used to calculate new values of the series (beyond those included in the input dataset) and confidence intervals for those predicted values [14]. The estimation process is performed on transformed (differenced) data; before the forecasts are generated, the series needs to be integrated (the inverse of differencing) so that the forecasts are expressed in values compatible with the input data. This automatic integration feature is represented by the letter *I* in the name of the methodology.

B. Web Service Security Attacks

XML messages designed for SOA-based systems can bypass firewalls over the HTTP and SMTP protocols. Bypassing firewalls is a basic security issue that may potentially lead to intrusions into systems. Some standards, such as WS-Security, WS-Policy, etc., are used for securing web services; however, these standards are not always sufficient for detecting all of the attacks. In addition, these standards are sometimes the origin of potential vulnerabilities and attacks. The main vulnerabilities in web applications and web services are DoS attacks, brute force attacks, spoofing attacks, and flooding and injection attacks which the attacks and their countermeasures are discussed in detail in [17]. One of the classic ways to perform such attacks is by sending a large-size SOAP message to the server. To prevent these attacks, we should examine the actual size of the message and reject all messages of a size greater than a predefined value. In this paper, we consider DoS attacks for amzn datasets and prevent this type of attack with the proposed method.

III. Related Work

There is a lot of existing research on network anomaly based and misused based detection systems; however, intrusion detection for web services is a new approach, and there are fewer related work on this topic. In [5], an intrusion detection system named WS-IDS is presented that uses hidden Markov models (HMM) is limited in order to detect over flow attacks, but this method can be used for other types of attacks. In [6], they represent an adaptive intrusion detection and prevention (ID/IP) framework to protect web services against SOAP/XML/SQL attacks. The framework shows that by using the agents that operate as sensors, data mining techniques such as clustering, sequential association rules accompanying fuzzy logic for further analysis, and anomaly detection, avoiding false alarms is possible. In [7], they provide a solution to prevent DoS attacks in web services. Their solution uses a multi-agent hierarchically-distributed architecture in four layers that performs a classification mechanism in two phases. In [8], an XML-based intrusion detection system for protecting the web services is designed and implemented. Their approach detect SQL injection, oversized payloads, and recursive payloads attacks with the use of filtering policies such as XML schema validation, syntax parsing, and message size restriction. In [18], kernel methods are used for WS intrusion detection. Their proposed method is capable of detecting new unknown attacks. The kernel methods map the data into a multi-dimensional feature space, where each coordinate corresponds to one feature of the data items. The data is transformed into a set of points in a Euclidean space, after which they can solve the problem. In [19], a web service firewall called CheckWay has been created. CheckWay can protect web services from DoS attacks by XML schema validation. In [20], a new approach for vulnerability detection of SQL injection and XPath injection in web services is presented. Their approach firstly learns the SQL/XPath commands, and then compares the current command structure with the previous identified structure in order to detect vulnerabilities. The ModSecurity module [21] is presented as a web application firewall with capabilities for web services; but for intrusion detection in web services, much time and progress are still needed before it reaches perfection.

In this paper, we propose a model for intrusion detection in web services based on the ARIMA.

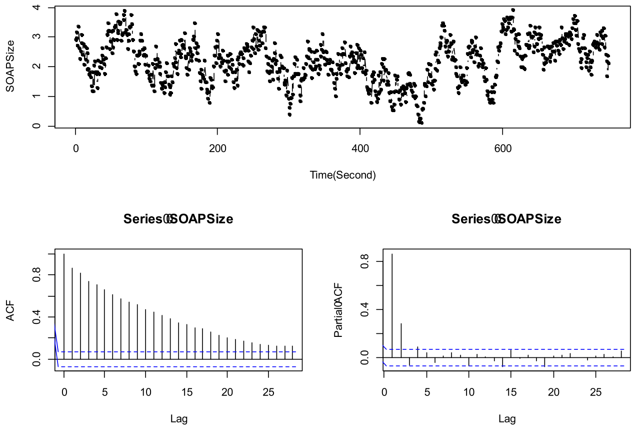


Fig. 1. Time series plot, ACF and PACF correlograms of SOAPSize

IV. Proposed Method

We consider oversized payloads attack then apply ARIMA model to the training data. In implementation phases, we recall SOAPSize field of training data with R software [22], and then plot the time series diagram and correlogram of ACF and PACF functions for these data as shown in Fig. 1. According to time series plot, the variance is stationary, but mean is not stationary and it requires to differentiating. The correlograms of ACF and PACF confirm that mean is not stationary. Fig. 2 illustrates the results of differentiating.

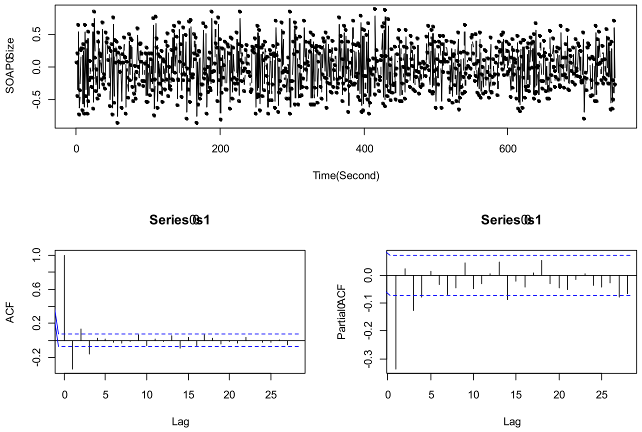


Fig. 2. Application of differentiation

Through Fig. 2, we can see that both the mean and the variance are stationary. According to correlograms of ACF and PACF, we can form an ARIMA(p, d, q) model. To estimate the p and q parameters, we examine the ACF and PACF correlograms. Fig. 2 indicates that ACF becomes zero after lag 3, therefore we can consider q = 3 or 2. PACF also becomes zero after lag 3, thus p = 3. As we have used differentiating only once, d = 1. Thus, we can apply both the ARIMA(3,1,3) and ARIMA(3,1,2) models. The results of applying the ARIMA(3,1,3) model are shown as below. As can be seen, the values of the AR model (ϕ_i) and MA model (θ_i) are estimated, and the AIC and log likelihood values are computed.

```
Series: SOAPSize ARIMA(3,1,3)
Coefficients:
    ar1      ar2     ar3     ma1     ma2     ma3
  -0.6961 0.7770 0.5647 0.3492 -0.9206 -0.3908
s.e. 0.1110 0.0546 0.0813 0.1244  0.0568  0.1116
sigma^2 estimated as 0.1176:
log likelihood = -261.8
AIC = 537.61   AICc = 537.76   BIC = 569.94
```

We apply ARIMA(3,1,2) to differentiation (for the sake of space we did not present the results); as a result AIC value is 538.09 and log likelihood equals to -263.05. One criterion for comparing the models is AIC. Models with minimum AIC are superior choices. Comparing the AIC values of the two models, the ARIMA(3,1,3) model is the one with less AIC. Another criterion is log likelihood; models with more value of log likelihood are better. In addition, as can be seen, the ARIMA(3,1,3) model suffers from lower error rates. Therefore, we choose ARIMA(3,1,3).

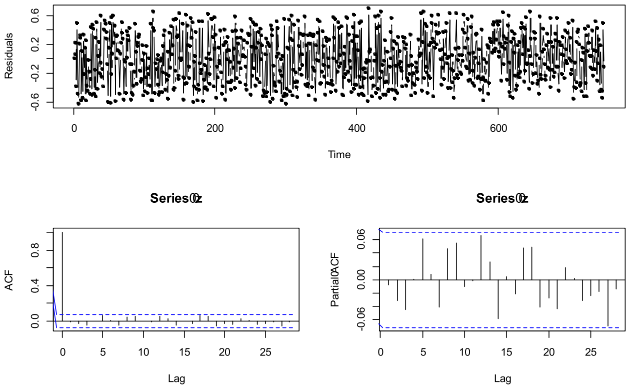


Fig. 3. Residuals of the ARIMA(3,1,3) model

In this stage, we should validate the adequacy of the model. With plotting the time series as well as the ACF and PACF correlograms of residuals in Fig. 3, we come to the realization that they are actually white noise, because ACF and PACF are zero. Likewise, the mean and variance values are stationary. It is noteworthy to mention that if the sequence of residuals are not white noise, this would mean that there is still some information in residual sequences that have not been exploited. If the residuals are not zero, or if other criteria are not adequate, we should apply a new model. There are many criteria for measuring the error, such as the Mean Squared Error (MSE), the Root Mean Squared Error (RMSE), and the Mean Absolute Error (MAE), that are defined in [8], [18]–[23]. We compute the error rate for the applied model. Subsequently shown are acceptable error values.

```
MSE      RMSE     MAE
0.117    0.342    0.296
```

After applying the model for training data, we should forecast the succeeding data in t seconds later. Fig. 4 illustrates the results of forecasting for the next 50 seconds for a confidence interval of 80% to 95%.

After forecasting, we examine the testing data. Fig. 5 illustrates the data added to the forecasted diagram. According to the confidence interval in the previous step, data which do not fall within the interval may be anomalies, and the administrator should be notified.

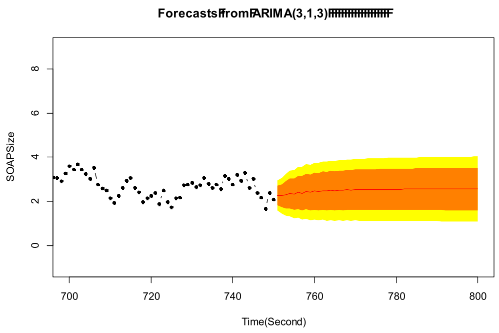


Fig. 4. Forecasting for a confidence interval of 80% to 95%

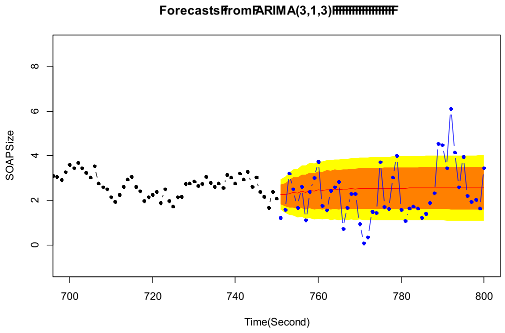


Fig. 5. Training data, testing data and detecting anomalies

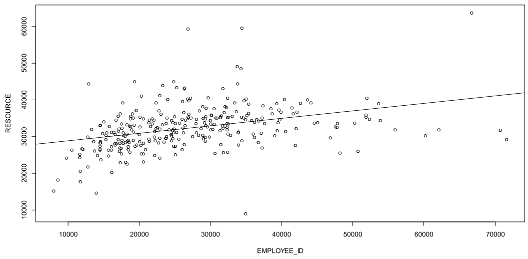


Fig. 6. Time series plot, ACF and PACF correlograms of $EMPLOYEE\_ID$ and $RESOURCE$

A. The Amzn dataset

We performed a multivariate survey on the amzn dataset [24]. We used 90% of the data as the training data, and the remainder of the data as the testing data. As it should be a correlation between two variables, we apply Pearson and Spearman tests [25] between the $EMPLOYEE\_ID$ and $RESOURCE$ variables. Fig. 6 shows that there is a correlation between these two variables.

We can now plot the time series diagram and correlogram of ACF and PACF functions for $EMPLOYEE\_ID$ and $RESOURCE$ fields separately. Fig. 7 presents these diagrams for $EMPLOYEE\_ID$. There is a trend in the time series diagram, and the variance is not stationary. Hence, we should use a *Box-Cox* [26] transformation to make it stationary. Fig. 8 illustrates the trend and *Box-Cox* transformation of $EMPLOYEE\_ID$. The best value of λ is -0.1010101. Therefore, we set $w = y^{\lambda} = y^{-}01010101$ and plot the time series diagram and correlogram of ACF and PACF functions for w (Fig. 9). The equation of w is represented as follows:

$$w_t = {y_t}^{-0.1010101} = 3.771x10^{-1} - 1.167x10^{-4}t + z_1 \qquad (1)$$

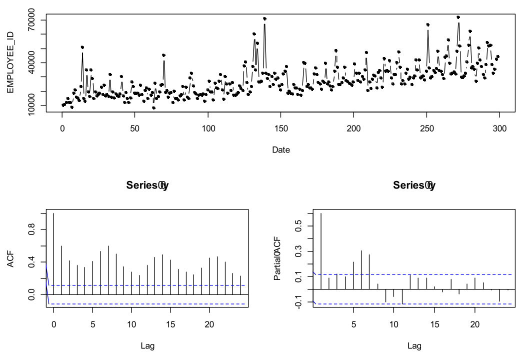


Fig. 7. Time series plot, ACF and PACF correlograms of $EMPLOYEE\_ID$

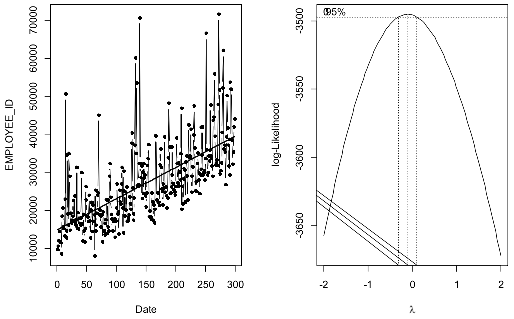


Fig. 8. Trend line and likelihood for power transformation of $EMPLOYEE\_ID$

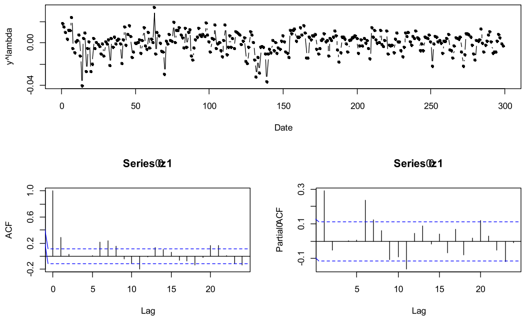


Fig. 9. Time series plot, ACF and PACF correlograms of residuals of $EMPLOYEE\_ID$

Similarly, we perform the previous steps for the $RESOURCE$ values. The equation of w is represented as follows:

$$w_t = {y_t}^{0.5858586} = 401.94521 + 0.23648t + z_2 \qquad (2)$$

We now examine the correlation between residuals. Fig. 10 confirms the correlation between them; therefore, we can apply the AR model to the residuals. The following shows the results of applying the AR model to these residuals. The order of the AR model is automatically performed by software.

```
Call: ar(x = ts.union(Z1, Z2))
$ar
, , 1
              Z1             Z2
Z1      0.2984        9.971e-06
Z2    -38.9326        1.084e-01
$var.pred
              Z1             Z2
Z1     9.147e-05     -0.09188
Z2    -9.188e-02   1899.39348
```

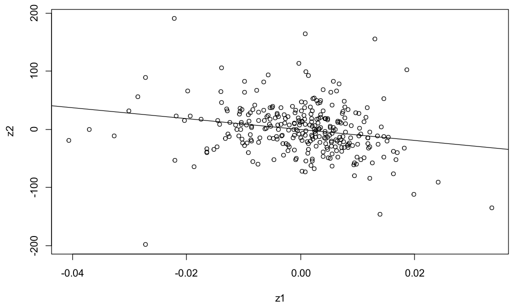
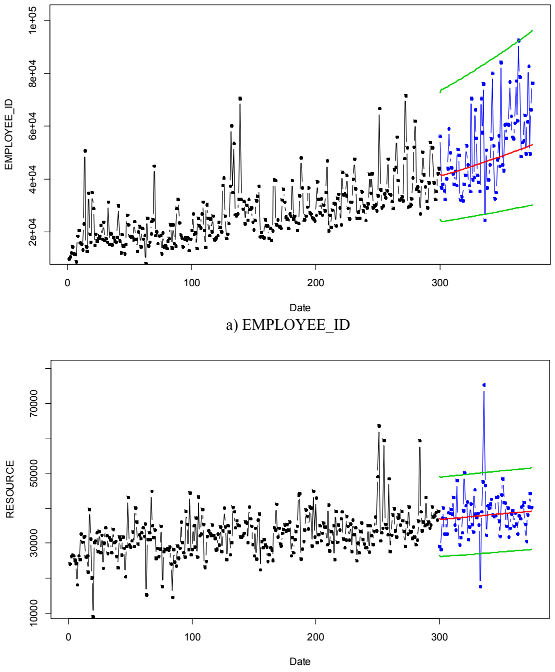


Fig. 10. The correlation between residuals of $EMPLOYEE\_ID$ and $RESOURCE$



a) EMPLOYEE_ID

b) RESOURCE

Fig. 11. Forecasting future data for a) $EMPLOYEE\_ID$ and b) $RESOURCE$

For forecasting the data, we use the VAR function in the *vars* package in R. Fig. 11 shows the results of forecasting the $EMPLOYEE\_ID$ and $RESOURCE$ values. In Fig. 12, we show the predicted values of the two variables together.

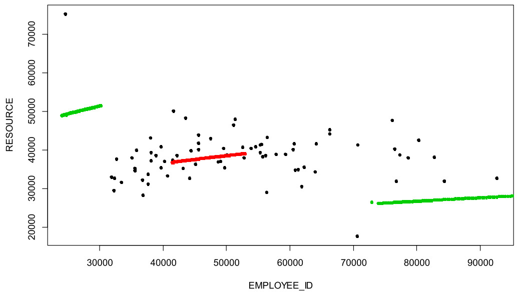


Fig. 12. Forecasting future data for $EMPLOYEE\_ID$ and $RESOURCE$ together

The methodology of the proposed forecasting is illustrated in Algorithm 1. The *mafit* variable is the value of the applied model with the *forecast* command, and the value in the *level* parameter will be predicted; the values will fit in the *prediction* variable. We start our confidence interval from 1% to 99%.

Algorithm 1: Proposed algorithm

```
Input  : $periods$
         $-Number\ of\ periods\ for\ forecasting$
         $threshold\ -Threshold\ of\ acceptable\ valuse$
Output: $Roc\ -Roc\ Curve$
begin
    $FPrate, TPrate \leftarrow \varnothing$;
    $Roc \leftarrow Matrix(\varnothing, \varnothing, 2)$;
    $rates \leftarrow Matrix(\varnothing, 100, 4)$;

    $/*t:$
    $Confidence\ interval\ for\ prediction\ intervals */$
    for $t = 1$ to 99 do
        $prediction \leftarrow$
        $forecast(mafit, periods, level = t)$;
        $lowerband \leftarrow prediction.lower$;
        $upperband \leftarrow prediction.upper$;
        $FP, FN, TP, TN \leftarrow \varnothing$;
        for $i = 1$ to $periods$ do
            if $(yTest[i] > threshold)$ && $(yTest[i] \leq up[i])$ then
                $FN = FN + 1$;
            if $(yTest[i] > threshold)$ && $(yTest[i] > up[i])$ then
                $TP = TP + 1$;
            if $(yTest[i] \leq threshold)$ && $((yTest[i] \geq low[i])$ && $(yTest[i] \leq up[i]))$ then
                $TN = TN + 1$;
            if $(yTest[i] \leq threshold)$ && $((yTest[i] > low[i])$ || $(yTest[i] < up[i]))$ then
                $TN = TN + 1$;
        $P \leftarrow TP + FN$;
        $N \leftarrow TN + FP$;
        $FPrate[t] \leftarrow FP/N$;
        $TPrate[t] \leftarrow TP/P$;

    $Roc \leftarrow cbind(FPrate, TPrate)$;
    return $Roc$;
```

V. Evaluation of Proposed Method

One of the ROC curve [27] usages is the evaluation of the quality of intrusion detection systems. This tool is appropriate for binary classification settings that classifies data into two positive and negative classes. The testing data are included of both positive (P) and negative (N) instances. The classifiers classify all data instances into one of these two classes. According to the confusion matrix, four situations exist within intrusion detection systems, corresponding to the relation between the results of the detections for an analyzed event ("normal" versus "intrusion") and its actual nature ("innocuous" versus "malicious") [28]: False Positive (`FP`), True Positive (`TP`), True Positive (`TP`), and False Negative (`FN`).

We evaluate our method by using the ROC curve. According to the oversized payload attack, we assume that the maximum size of an input SOAP message can be 4MB [29]. For drawing the ROC curve, we should compute the `TPrate` and `FPrate` values. Each test data instance is identified with a point `(FPrate, TPrate)` in ROC curve. In this method, since we have a specific confidence interval, we deal with probabilistic classifiers. Thus, we need an algorithm for generating the TPrate and FPrate values as shown in Algorithm 1. By using Algorithm1, we draw the ROC curve of our method which is plotted in Fig. 13 for SOAPSize on the amzn dataset. The true positive rate is roughly 89% and 0.01 error rates. The following shows the FP rate, the TP rate, the accuracy, and the error rate for a 95% confidence interval for amzn datset.

```
FPrate      TPrate      ErrorRates  Accuracy
0.01538462  0.88888889  0.02702703  0.97297297
```

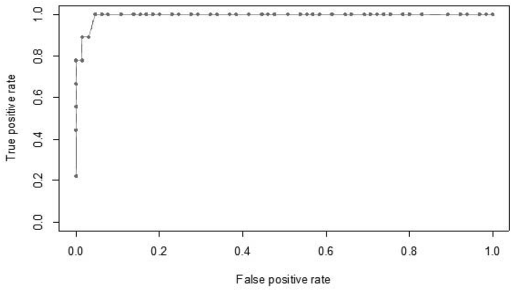


Fig. 13. ROC curve for amzn

VI. Summary

There are many techniques for performing intrusion detection. In this paper, we have used data mining techniques and time series models. We applied the ARIMA model to training data, and predicted the future values of the series for confidence intervals. Following this, we examined the testing data; if data exceed the specified threshold, an alarm is sent to administrator. In this method, we can prevent attacks with patterns embedded in data behavior. For example, we cannot defend against attacks which have a countermeasure of input validation or XML schema validation. In this paper, we have examined the oversized payloads and DoS attacks and have demonstrated a means of defense. Other attacks, such as coercive parsing, brute force, buffer over flow, XML DoS, SOAP array attack, XML document size attack, XML flooding, etc. are preventable with the proposed method. Generally, this method can be used for anomaly detection in situations where the system has learned the normal traffic and will detect abnormal input traffic. The topics and proposed subjects for future works are as follows. 1) Expanding the proposed method using multivariate time series for detecting other types of attacks. For example, in brute force attacks or session hijacking attacks, their past, present and future behavior is related; they are suitable for this proposed method. 2) False alarm correction and replacing the expected value with the predicated value in the time series. Thus, subsequent predictions will be more precise.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their valuable comments. They gratefully acknowledge the support of Professor Mourad Debbabi, the president of National Cyber Forensics and Training Alliance (NCFTA) Canada.

References

[1] A. Ghourabi, T. Abbes and A. Bouhoula, "Experimental analysis of attacks against Web services and countermeasures," iiWAS2010, pp. 195-201, (Nov. 2010).

[2] N. A. Nordbotten, "XML and Web Services Security Standards," IEEE communications surveys and tutorials, vol. 11, no. 3, pp. 4-21, Aug. (2009).

[3] A. H. Yaacob; I.K.T. Tan; S. F. Chien and H. Kh. Tan, "ARIMA Based Network Anomaly Detection," in Second International Conference on Communication Software and Networks, Cyberjaya, Malaysia, pp. 205-209, (2010).

[4] S. A. Mirheidari, S. Arshad, and R. Jalili. "Alert Correlation Algorithms: A Survey and Taxonomy." Cyberspace Safety and Security. Springer International Publishing, 183-197, (2013).

[5] M. S.A., Najjar and M. Abdollahi Azgomi,, "A Distributed Multi-Approach Intrusion Detection System for Web Services," in ACM, pp. 238-244, (2010).

[6] C. G. Yee; W. H. Shin and G.S.V.R.K. Rao, "An adaptive intrusion detection and prevention (ID/IP) framework for Web Services," in Convergence Information Technology, pp. 528-534, (2007).

[7] C. I. Pinzon, J. F. DePaz, J. Bajo, and J.M. Corchado, "An adaptive multi-agent solution to detect DOS attack in SOAP messages," in Computational Intelligence in Security for Information Systems, vol. 63, pp. 77-84, (2009).

[8] Y.S. Loh, W.C. Yau, C.T. Wong, and W.C. Ho, "Design and Implementation of an XML Firewall," in Proc. of the International Conference on Computational Intelligence and Security, Guangzhou, China, pp. 1147-1150, (2006).

[9] M. Leng, X. Chen and L. Li, "Variable Length Methods for Detecting Anomaly Patterns in Time Series," in International Symposium on Computational Intelligence and Design, vol. 2, pp. 52-56, (2008).

[10] W. W.S.Wei, "Time Series Analysis: Univariate and Multivariate Methods", Pearson, (2006).

[11] The Amzn data set, https://aws.amazon.com/datasets, accessed on Sept, 26, (2014).

[12] Investopedia. [Online]. http://www.investopedia.com/terms/b/box-jenkins-model.asp, (Jan 2012)

[13] Engineering Statistics handbook. [Online]. http://www.itl.nist.gov/div898/handbook/pmc/section4/, (Dec 2012)

[14] Electronic Statistics Textbook. [Online]. http://www.statsoft.com/textbook/time-series-analysis/(Oct 2011)

[15] H. Zare Moayedi and M.A. Masnadi-Shirazi, "Arima Model for Network Traffic Prediction and Anomaly Detection," in International Symposium on Information Technology, vol. 4, Shiraz, Iran, pp. 1-6, (Aug 2008).

[16] G. Wang, Z. Wang and X. Luo, "Research of Anomaly Detection Based on Time Series," in Software Engineering, vol. 1, Baoding, China, pp. 444-448, (2009).

[17] J. C. Estrella, M. Vieira, K. R. L. J. C Branco, "Security in Web Services," ICMC-USP and Univ. of Coimbra, (2010).

[18] R. Ghasem Esfahani and M. Abdollahi Azgomi, "Towards an Anomaly Detection Technique for Web Services Based on Kernel Methods," in Innovations in Information Technology, Tehran, Iran, pp. 345-349, (2009).

[19] N. Gruschka, N. Luttenberger, "Protecting Web Services from DoS Attacks by SOAP Message Validation," in IFIP International Federation for Information Processing, vol. 201, pp. 171-182, (2006).

[20] N. Antunes, N. Laranjeiro, M. Vieira and H. Madeira, "Effective Detection of SQL/XPath Injection Vulnerabilities in Web Services," in IEEE International Conference on Services Computing, Coimbra, Portugal, pp. 260-267, (2009).

[21] ModSecurity: Open Source Web Application Firewall. [Online]. http://www.modsecurity.org/,(Aug 2011).

[22] L. Torgo, Data Mining with R Learning with Case Studies, E. Kumar, Ed. Minnesota, U.S.A: CRC Press, (2011).

[23] R. Lang, Q. Cui and Y. Lv, "Evaluation on Forecasting Algorithms of Time Series," in Management and Service Science, pp. 1-4, (2009).

[24] 2012 IEEE International Workshop on Machine Learning For Signal Processing. [Online], http://mlsp2012.conwiz.dk/index.php?id=43,(Oct 2011).

[25] Bolboaca, Sorana-Daniela, and Lorentz Jäntschi. "Pearson versus Spearman, Kendall's tau correlation analysis on structure-activity relationships of biologic active compounds." Leonardo Journal of Sciences 5.9, 179-200, (2006).

[26] Box, G. and Cox, D.R., ŞAn analysis of TransformationŤ, Journal of Royal Statistical Society B, Vol. 26, pp. 211-243, (1964).

[27] M. Vuk and T. Curk, "ROC Curve, Lift Chart and Calibration Plot," Metodoloski zvezki, vol. 3, no. 1, pp. 89-108, (2006).

[28] P. Garc, J. Daz-Verdejo, G. Maciat'-Fernat'ndez and E. Vat'zquez, "Anomaly-based network intrusion detection:Techniques, systems and challenges," Computers and Security, vol. 28, pp. 18-28, (Aug 2009).

[29] M. Jensen, N. Gruschka, R. Herkenhoner and N. Luttenberger, "SOA and Web Services: New Technologies, New Standards - New Attacks," in Proceedings of the Fifth European Conference on Web Services, Washington, pp. 35-44, (Nov 2007).