International Journal of Network Security & Its Applications (IJNSA), Vol.4, No.2, March 2012

A BAYESIAN CLASSIFICATION ON ASSET VULNERABILITY FOR REAL TIME REDUCTION OF FALSE POSITIVES IN IDS

G. JACOB VICTOR[1] Dr. M SREENIVASA RAO[2] Dr. V. CH. VENKAIAH[3]

[1]Director, SERP, RD Department, Hermitage Complex, Hill Fort Road, Hyderabad
jacob.victor@gmail.com
[2]Director, School of Information Technology, JNTU, Hyderabad -85
srmeda@jntuap.ac.in
[3]Professor, CRRao AIMS & CS, UoH Campus, Hyderabad – 46
venkaiah@hotmail.com

ABSTRACT

IT assets connected on internetwill encounter alien protocols and few parameters of protocol process are exposed as vulnerabilities. Intrusion Detection Systems (IDS) are installed to alerton suspicious traffic or activity. IDS issuesfalse positives alerts, if any behavior construe for partial attack pattern or the IDS lacks environment knowledge. Continuous monitoring of alerts to evolve whether, an alert is false positive or not is a major concern. In this paper we present design of an external module to IDS,to identify false positive alertsbased on anomaly based adaptive learning model. The novel feature of this design is that the system updates behavior profile of assets and environment with adaptive learning process.A mixture model is used for behavior modeling from reference data. The design of the detection and learning process are based on normal behavior and of environment. The anomaly alert identification algorithm isbuiltonSparse Markov Transducers (SMT) based probability.The total process is presented using real-time data. The Experimental results are validated and presentedwith reference to lab environment.

KEYWORDS

Anomaly, Common Vulnerability Exposure (CVE),IT policy, True positives, False Positives

1. INTRODUCTION

Computers and internet have become a part of human life. With availability of Internet connectivity everywhere, there is an increase in the number of computers and devices connected to the internet. The excess dependence on computers leads to threats and attacks on the vulnerabilities in IT setup. To address these challenging threats, security tools like Anti-viruses, Firewalls, Intrusion Detection / Prevention Systems are deployed.

In the last 30 years, after the first report on Intrusion detection publishedby [Anderson[1], 1980), Intrusion Detection Systems became an interesting and important subject of study. The concept of IDS design is based on the viewpoint that attackers pattern of actions are unusual compared to a genuine client. The difference in behavior can be detected. As per [Peng Ning[2,] 2005] Intrusion detection systems (IDSs) are a subset of preventive security mechanisms and deployed along with authentication, access control systems as a subsequent level of protection

[1] The work of this author is supported by the Department of Science and Technology, Govt. of India, New Delhi, under Project No. SR/s4/MS: 516/07 dated 21st April 2008.
DOI : 10.5121/ijnsa.2012.4205

to IT Assets. Intrusion Detection/Prevention Systemshave become mandatory security tool to monitor systems and detect possible attacks [Debar [3], 1999].

Most of IT systems and user applications were developed in their respective context without security awareness, thereby susceptible to attacks. In the rest of cases, applications and systems were developed to operate in one set of environment parameters, deployed in the different setup resulting in vulnerability.The vigilant security experts introduce more stringent rules by increasing the security thresholds, to reduce false negatives, resulting in high False Positives.

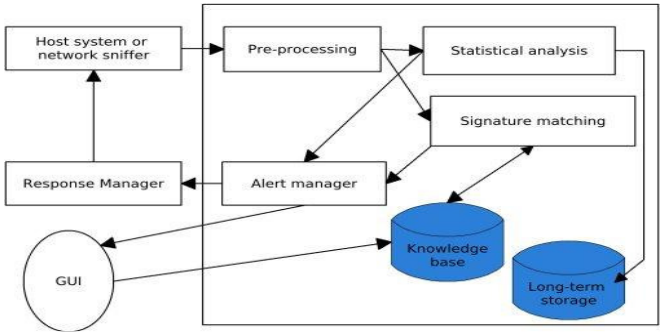


Figure 1 Intrusion Detection System – components

The components of IDS are shown in figure 1. The target system (network) is protected by the security policy, which defines legitimate actions(profile) on each entity. The Data captured by sensor will be processed by IDS, consist of commands executed by a user, attacker or by an application etc. IDS issues alerts, if an intrusion is suspected or detected.Alerts are of two categories, intrusion alerts and suspected intrusion alerts.

The main reason for large number of false positivesgenerated by IDS is that IDS classifies an event as anomalous, based on the probability threshold of event depending on current profile(asset) or rule. The other reason is anomaly-based system may not differentiate anomalous behavior of legitimate user actions with intrusive behavior (actions). The objective of this work is to address these scenarios and minimize false positives.

The contribution of this paper is in the design of an adaptive external module for real-time identification of anomaly in alerts generated by IDS. To attain a linear complexity for detection algorithm, the case of Mixture Model is used for behavior modeling and Sparse Markov Transducers based algorithm for identification of anomaly in alerts generated. The novel Feature is that, the alert is validated with vulnerability on victim system. The experimental results are showcased to validate the proposed techniques. In the remainder of this paper, we present these findingssuccessively.

2. RELATED WORK

Different approaches to build anomaly detection models have been proposed and implemented at design level by many researchers. Statistical Methods are proposed by [Johnson[4], 1998] and [Roberts[5], 1998] for Anomaly Detection, Adaptive, Model-based Monitoring for Cyber Attack Detection by [Alfonso[6], 2000]. [Stephanie[7] 3, 1996] used normal sequences for look ahead pairs and contiguous sequences. [Sobirey[8], 1996] used expert system to collect data from audit sources and used it for adaptive intrusion detection. [H S Teng[25], 1990] using inductively generated sequential patterns, performed adaptive real time anomaly detection. A statistical method to determine sequences which occur more frequently in intrusion data as opposed to normal data was implemented by [Helman[9] 5, 1997]. Many anomaly detection techniques

models were compared by [Christe[10], 1999]. A decision tree applied over the normal data was used by [Lee[11], 1999]for design of trained prediction model. Neural networks were used to model normal data by [Ghosh[12],1999]. Model to examine unlabeled data to compare during intrusion and normal use for anomaly detection was proposed by [Lane[13], 1999]

[Srivastava[14], 2006] proposed Database Intrusion Detection using Weighted Sequence Mining. [Emmanuel Hooper[15] 06] designed Intelligent IDS Using Network Quarantine Channels and Adaptive Policies and Alert Filters. [Venkatachalam[16], 2007]proposed theLAMSTAR neural network to learn patterns of normal and intrusive activities and to also classify system activities. [Thoosi[17], 2007]proposed IDS based on An Evolutionary Soft computing model using Neuro-Fuzzy classifiers.[Victor[18], 2008] proposed a model for Data Mining Approaches for Intrusion Detection in Email System. Used data mining technique to discover consistent and useful patterns of email system and recognized anomalies.[Chen[19], 2008] proposed a classification method based on Support Vector Machines (SVM) with a weighted voting schema to detect intrusions. [Neelakantan[20], 2008] proposed Approach for Obtaining Network-Specific Useful Alarms. [Ramana Murthy[21] 09] Incorporated both anomaly and misuse detection into the NIDS, to improve Performance of the NIDS. [Rung-Ching[22], 2009] "Using Rough Set and Supporting Vector Machine for Network Intrusion System to detect intrusions.[Subbulakshmi[23], 2010] et al proposed "Real Time Classification and Clustering of IDS Alerts Using Machine Learning Algorithms"**.** [Vignesh[24], 2010] et al proposed "A Cache Oblivious based Genetic Algorithm(GA) Solution for Clustering, for eliminating false positives using a parallelized version of Genetic Algorithm".

The preparation of behavior profiles to reflect behavior profile of multiple applications running on a network with multiple communication protocols is a difficult task. At times, it is difficult to identify profiles of present behavior. We modeled this base reference behavior from the manual observations of the target network, a representative set of legitimate, non-malicious entities (users, services, etc). The add-on module with reference profile was created from vulnerability of target network. The anomalies are detected to update the configuration (profile), and thereby minimize them.

Bayesian behavior classification is created manually and used for system training, using parametrical mixture model [Cheeseman[26]]. The mixture model parameters of the EM algorithm [Dampster[27], 1977], [McLachlan[28], 1999] are fitted to accomplish the unsupervised learning.The model updates the changes in behavior in last phase of process as learning. The design is such that, the model update behavior profile only after re-estimation of parameters.The Bayesian techniques used for design of learning, detection and update are adopted from the earlier works. The process additional adaptive module to IDS will updateknowledge base automatically.

3. ANOMALY IN ALERT GENERATION MODEL

The IDS functions as anintrusion detection tool by capturing data related to processes or events in target area and analyze them as per the security policy defined by the organization. Intrusion activity occurs as a sequence of events. The intrusion activity can be classified into three stages:

(i) Reconnaissance - Sweeps, automated scans and Port scans
(ii) Penetration – Ping, Denial of service (DOS) attacks
(iii) Attack and damage - Compromises the target.

IDS identifies intrusions or suspicious behavior and generate alerts. Suspicious behavior identification process depends on characterization of behavior or signature. The alerts generated by IDS can be categorized into four types:

TP 'True Positive', alarm issued on intrusive attempt;
FP 'False Positive', alarm issued on non-intrusive attempt;
TN 'True Negative', no alarm issued and no attempt;
FN 'False Negative', no alarm issued on intrusive attempt.

As per Bayesian model,detection rate D_r of IDS can be computed as

$$D_r = \frac{TP}{TP+FN} \qquad \text{...(1)}$$

The false positive rate F_r can be computed as

$$F_r = \frac{FP}{TN+FP} \qquad \text{...(2)}$$

In the event, IDS does not generate an alarm $D_r=0$ and therefore the $F_r=0$; In case, IDS generate alarms for all cases, $D_r= 1$ and $F_r= 1$, as $FN=TN=0$; in between IDS can exhibit its behavior. The detection rate Dr is proportionate to environment of target area and IDS configuration.

Total number of intrusions in a given period of time is sum of the True Positives and False Negatives. The total number of non-intrusions is the sum of True Negatives and False Positives. Therefore the Base Rate(BR), which is the probability of an attack on target system is number of intrusions over the number of events.

$$\text{BR} = \frac{TP+FN}{TP+FP+TN+FN} \qquad \text{...(3)}$$

The D_r and F_r depend on:

(i) Design criteria of IDS
(ii) Environment of target area
  a. Security Policy and configuration
  b. Vulnerability profile on assets
(iii) The intruder behavior

To build a model to minimize false positives, we propose a probability based algorithm to tolerate noises in alert data. The assumption is that non-intrusions (false positives alarms) alerts generated by IDS are more compared to intrusions (true positives) alerts. Therefore occurrence of actual intrusions is relatively small in number compared with alerts generated by IDS. A false positive identification model is devised to separate the intrusions on the target network.

4. MODELING PROBABILITY DISTRIBUTIONS TO IDENTIFY FALSE POSITIVES

A probabilistic approach is proposed to examine whether an alert generated is afalse positive or a true positive. A mixture model is proposed to model the adaptive learning. Anomaly detection techniqueusesSparse Markov Transducers probability distribution.

The assumption is that,the number of true positives (intrusions) may be very small, first it was assumed that every alert as a non-intrusion. The probability distribution is used to test each element to determine whether it is a non-intrusion or not.Therefore the model assumes two cases for each alert and λ is the probability of true positives (intrusions), while with $(1-\lambda)$ probability alerts corresponds to the non-intrusions (false positives). This motivates a mixture model for explaining the presence of false positive alerts.

The framework proposes two probability distributions, a *majority (non-intrusion)* distribution*M* and *minority* (intrusions) distribution*I*. The properties or types of the distributions *M* and *I* are independent from the framework of mixture model used for explaining the presence of false

positives in the alert data. An element is either generated in the majority distribution with probability *1-λ*or from the alternate distribution *I*with probability *λ*. The distribution for the entire alerts data is:

$$D = (1-\lambda)M + \lambda I \quad \text{.........(4)}$$

The framework is to determine which alerts generated by distribution *M* and which elements were generated by the distribution *I*. Elements generated by *I* are intrusions (true positives).The machine learning method is used to model probability distributions.

We assume that for the non-intrusions, a likelihood function $\mathcal{L}_M$which takes as a parameter, a set of non-intrusion elements M_t, and outputs a probability distribution P_{Mt} on the data. Similarly for intrusions we have likelihood function $\mathcal{L}_I$and probability distribution P_{It}.

$$P_{Mt}(X) = \mathcal{L}_M(M_t)(X) \quad \text{.........(5)}$$

$$P_{It}(X) = \mathcal{L}_I(I_t)(X) \quad \text{.........(6)}$$

To identify anomalies in alerts, we have to testelements generated by distribution *M* or *I*.For each element x_t, we have to compute and determine whether it is outlier by comparing the likelihood. If determined asintrusion, should be flagged and moved to I_{t+1}, else should remain in M_{t+1}as it is false positive alert.

To examine the likelihood of these two cases, we make this determination and the likelihood$\mathcal{L}$ of the distribution *D* at time *t* is:

$$L_t(D) = \prod_{i=1}^{N} P_D(x_i) = \left((1-\lambda)^{|M_t|} \prod_{x_i \in M_t} P_{M_t}(x_i)\right)\left((\lambda)^{|I_t|} \prod_{x_j \in I_t} P_{I_t}(x_j)\right) \text{........(7)}$$

Where, $P_{Mt}$and $P_{It}$are the probability distributions over the majority and minority (intrusions) alerts respectively. For ease of computation, log likelihood *(LL)* at a time *t* is computed.

$$LL_t(\mathrm{D}) = |M_t| \log(1-\lambda) + \sum_{x_i \in M_t} (\log P_{Mt}(x_i)) + |I_t| \log \lambda + \sum_{x_j \in I_t} (\log P_I(x_j)) \text{..........(8)}$$

We compute whether an element is more likely to be a false positive or a true positive using an external module based on the probability.In other words, we examine the change in behavior of an activity.

$$M_t = M_{t-1} \setminus \{x_t\} \quad \text{..........(9)}$$

$$I_t = I_{t-1} \cup \{x_t\} \quad \text{..........(10)}$$

If this ratio (L_t:L_{t-1}) or (L_t:L_{t-2}) or (L_t:L_{t+1}) is meeting the environment criteria, we declare the alert as false positive, $I_t = I_{t-1}$. Otherwise, the alert remains in normal distribution$M_t = M_{t-1}$.

5. METHODOLOGY

In order to determine whether alert trace subsequence corresponds to an exploit or normal trace, we build a probabilistic prediction model which predicts the last (n^th^) alert /events given the previous (n-1) events in the subsequence. In this model, this can be represented as a probability estimate of the last events conditional on the sequence of previous events. The size of the window and the placement of the wild cards correspond to the length of the conditioning sequence and the specific positions in the conditioning sequence on which the probability is conditioned. To model this type of probability distribution, we use sparse Markov transducers.Sparse Markov Transducers compute probabilistic mappings over sparse data.

A Sparse Markov Transducers is defined as a probability distribution on a finite set of inputs. A Sparse Markov Transducers of order *L* is the conditional probability distribution of the form:

$$P(Y_t|X_tX_{t-1}\,X_{t-2}\,X_{t-3}\,\ldots X_{t(L-1)})$$

Where, X_k are random variables over the input alert vectors Σ_{in} and Y_k is random variable over the output alert Σ_{out}. The distribution stochastically defines a mapping from input to output of the alerts. We use the Sparse Markov Transducers to model this distribution of order $\boldsymbol{L}$, which is a conditional probability of the form:

$$P(Y_t|\emptyset^{n1}X_{t1}\,\emptyset^{n2}\,X_{t2}\ldots\emptyset^{nk}X_{tk})$$

Where, $\emptyset$ represents a wild card symbol and $t_i = t - \sum_{j=1}^{i} n_j - (i-1)$. The Sparse Markov Transducer estimation algorithm estimates a conditional probability based on a set of inputs and their outputs. The priori data set is created manually using the tools. In variable order Markov the value of *n* changes depending on context. Few data elements may use a bigram, while others use trigram or n-gram. The Sparse Markov Transducer uses a weighted sum of n-grams for different values of n, and the weights depend on the context. In this context to have low complexity, the n is considered as 5. The specific weights of each element depends on the context or the actual values of $\boldsymbol{C_i}$,$T_{t-2,}T_{t-1,}\ T_tT_{t+1,}T_{t+2}$. Each alert in the weighted sum uses a pseudo-count predictor. This predictor computes the probability of output by number of times that that specific output was seen in a given context.

The intrusion alerts generated by IDS are independent of any specific probability distribution as they are random and sparse. The sequence of the events, activities will be tagged at time *t*. The intrusive behavior can be result of any of the activity at time $T_{t-2,}T_{t-1,}\ T_tT_{t+1,}T_{t+2}$. Using Sparse Markov Transducers Probability model, the alert will be determined to be an element of M_i or$I_{i,}$. The data packets in the window between the $\boldsymbol{T_{i-2}}$ alert to the $\boldsymbol{T_{i+2}}$alert will be tagged and analyzed. The alert vectors are tagged as:

The Common Vulnerability Exposure (CVE) of the alert is $\boldsymbol{C_i}$
The alert being examined is $\boldsymbol{T_i}$
The next alert $\boldsymbol{T_{i+1}}$
The previous alert $\boldsymbol{T_{i-1}}$
The previous alerts, previous alerts $\boldsymbol{T_{i-2}}$

The following parameters are examined from tagged alert vectors:

i. Common Vulnerability Exposure (CVE) ID (code) of the attack: Common Vulnerability Score System (CVSS) of the CVE, Security authorization controls impact base line, Source name[BUGTRAQ, OVAL(Open Vulnerability and assessment Language), CISCO, IBM, OSVDB, MS, REDHAT, SUN, SGI etc]
ii. IP address of the victim
iii. Vulnerability on the victim - Vulnerability status on the victim system (Existing | Closed)
iv. Alerts previous activity
v. Victim system environment
vi. Connection number, Timestamp, Source IP, Destination IP, Source port, Destination port, Protocol, Duration, Source bytes, Destination bytes, TCP Flags, Land packet,
vii. % wrong frag, % urgent, % Resent, % Wrong resent, % Duplicate ACK, % wrong data packet size, % data packets, # SYN flags, # RST flags, # FIN flags

The packets in the window $\boldsymbol{T_{i-2}}$ to the $\boldsymbol{T_{i+2}}$will be analyzed.To decide an alert is true positive or false positive, the environment dependent threshold is computed.The tagged data in the window is validated by computing the thresholds. This process is repeated for every element and in the

end we get a partition of alerts data into two sets, majority elements (false positives) and a set of intrusion elements.The computations are as per the Sparse Markov Transducers probability distribution.The vulnerability data created will be used for further updation by the learning process by adding a recod in that class. Further, the alert is validated with the asset vulnerability data to reconfirm respective attack is feasible and possible.

The schematic representation of the process is given below:

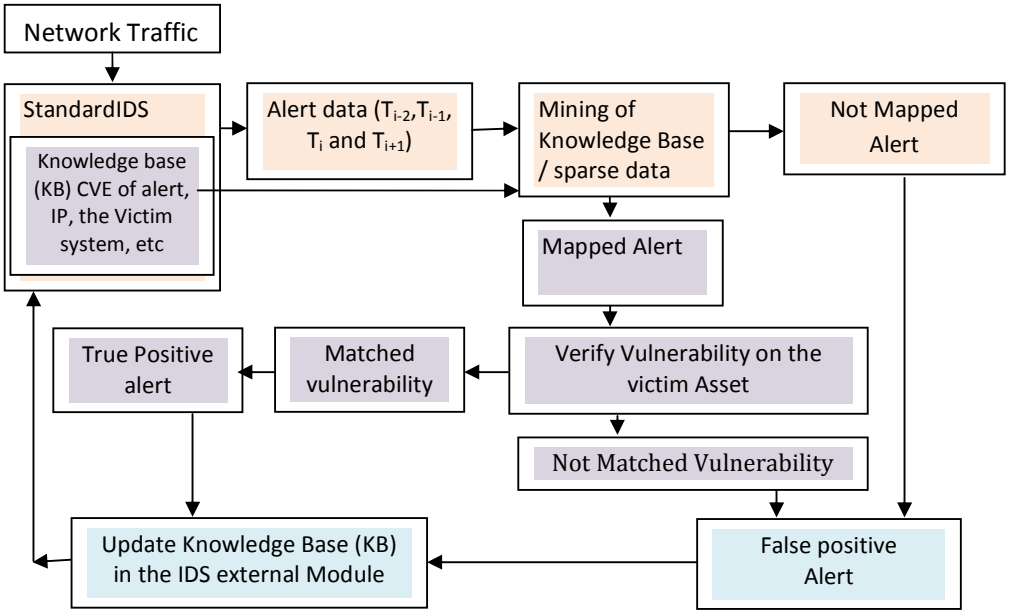


Figure 2 Model for Asset Vulnerabilities compliance to minimize the false positives

6. EXPERIMENTAL RESULTS

The work was evaluated on a gigabit network in a campus with online real-time traffic data on LAN segment. The IDS is configured on a LAN segment to observe the traffic. The environment consists of Snort(Version 2.6) intrusion detection system for obtaining network specific alarms, Winpcap(Version 4.0), software sensor to collect the packets to feed the IDS and MySql(Version 4.0.25) for storing the data. Retina free download version was used to creating the vulnerability database of the target area.

The base knowledgebase was created with major classifications. The first calcification is based on of sequences of events. The other is sequence of events linked with the asset and its vulnerability. The vulnerability data will be updated as and when the patches are released for the retina database. For every sequence of length *n,* the sequence database will be updated for every sequence validated by the system, if the same sequence not exists.

The observed results are indicative of the Knowledgebase was updated with respect to environment. 25,592 sequences were examined by the system during the experiment. 9156 records were updated in knowledgebase for cases of true positives, false positives and false negatives. The system identified 1612, 2466, 2372 and 2412 false positive alerts in four equal intervals after examining 6879, 7419, 5890 and 5404 sequences (classes) respective intervals. Further the system identified and confirmed 369, 563, 708 and 782 alerts as true positives.

The alert data could be classified in to two major categories True positive (TP) and False Positive (FP). The probability distribution for the alarms data is not clearly known, hence

Receiver Operating Characteristic (ROC) curve used for analyze and to visualize two-dimensional data at their decision threshold levels. It is considered that the real-time data used in evaluation is representative of the environment in target area. The accuracy of one set of live data varies from its accuracy, at different testing intervals. The experiment was performed in four equal intervals.

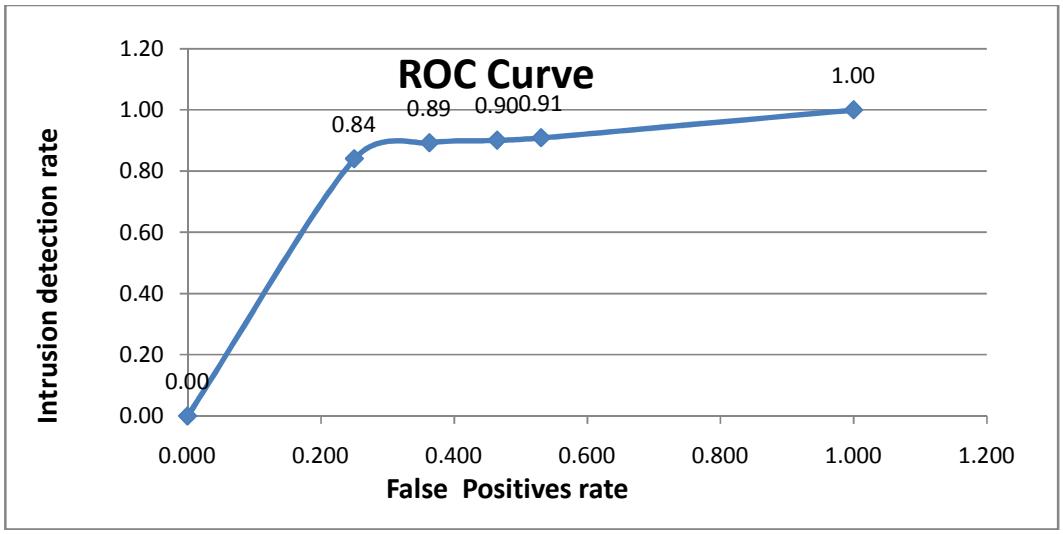
The ROC curve, illustrate the complete, i.e., two-dimensional accuracy of signal detectors on a given data set. The ROC curve gives a sense of how IDS is behaving on a particular environment. The ROC curve is starting point for more detailed analyses about expected accuracy and cost of detector.

Table1 True Positive, False Positives and base Rate

S no	True positive	False Positives	False Negatives	True Negatives	FP Rate	TP Rate	Total alerts	% of FPs
1	0	0	0	0	0.000	0.000		
2	369	1612	70	4828	0.250	0.84	6879	23.43
3	563	2466	68	4322	0.363	0.89	7419	33.24
4	708	2372	78	2732	0.465	0.90	5890	40.27
5	782	2412	78	2132	0.531	0.91	5404	44.63

We use ROC curve to indicate the accuracy of the IDS. The accuracy is revealed in the shape of curve, is two-dimensional because there are two kinds of events, and hence two kinds of accuracies possible. The first dimension is the success rate (probability of true positives) on y-axis. The second dimension falsely identifying alarms (probability of false positives) on x-axis. If, ROC curve with y-values grow at a faster rate than its x-values, resulting in a curve shape which rises swiftly upward is an ideal ROC curve. The decision threshold shall change to be more lenient, the x-values must also grow large, catching up with the success values for signals (y-values).

IDS accuracy is indicated by the rise or "bowness" of the ROC curve (a). A perfect ROC curve passes through the point (0,1) (b) each point along the ROC curve corresponds to a different operating mode, or decision threshold. Points (0,0) indicate exclusive criterion; The more accurate curves bulge outward to the upper-left, nearing the point of perfection at (0,1).



Graph 1 ROC Curve with VDB application First Trail

There is slight increase in reduction of false positives because of learning process of the system. This experiment was done on the real time data. As the alerts distribution is completely independent of the vulnerabilities and environment variables.

7. CONCLUSIONS AND FUTURE WORK

The art of detecting intrusions into IT systems is in primitive stage even after 30 years of its existence. Even though we are able to address intrusions, IDS generate a large number of false positive alarms. IDS made it mandatory for human intervention to validate alarms. Almost all work done by the researchers was on reduction of false positives at design level. The current work is done on implementation end to assist security administrator. The model with add-on module is developed and implemented. The experiments were carried out on real time data. Every alarm is examined by an add-on module whether it is a true positive or false positive. The use of mixture models implies the assumption of statistical independence between trials, which can be restrictive in some cases, hence Stochastic models are considered for identification of anomaly.

We have presented a newmodel in identifying anomalies in Alerts generated by IDS, using an add-on module to IDS, using a mixture model for behavior modeling and Sparse Markov Transducers for detection of anomaly. Continuous model update is accomplished by model parameter re-estimation. Algorithms for detection and update phases are designed for real-time operations. Our experiments show that proposed algorithms present real-time feasibility with no special or additional hardware requirement. Functional validation has been considered for intrusion detection in a real-time environment.

The objective is to minimize false positives. The alerts were analyzed and results are presented. The observed results are indicative of the environment. The system effectively analyzed and identified false positives alarms. The reduction of false was 22.46%, 32.37%, 39.36% and 43.68% in the first, second, third and fourth weeks, when compared with all alerts. The system effectively analyzed and identified 100 % true positives alerts.

The critical evaluation is done on alert using an “Anomaly in alert” and supplemented by “vulnerability complianceof the alert with asset” to reinforce verification process is presented. The work has established a “framework which enables the administrator to effectively analyze alerts”and identify the alarm is a false positive or true positives.From this framework, an adaptive module was modeled.

The model is tested with only with one type of vulnerability Bugtraq ID. The intrusions with other CVE IDs shall be considered for further stabilization of the system. The proposed system has limitation of evaluating only alerts data. The data of alerts only considered for evaluating the model, the remaining data to be considered to examine false Negatives.

8. REFERENCES

[1] Anderson, J P (1980), “*Computer Security threat Monitoring and surveillance (Technical Report)*”. Fort Washington, PA: James P Anderson Company.

[2] Peng Ning(2005), “*Intrusion Detection Systems Basics”, published in “Hand Book of Computer*”, Volume 3, edited by Hossien Bidgoli, Published by John Wiley& Sons, Inc (PP 685 to 700)

[3] H. Debar, M. Dacier and A. Wespi, “*A Revised Taxonomy for Intrusion-Detection Systems*,” IBM Research Report, 1999.

[4] R.A.Johnson, D.A. Wichern and D. W. Wichern, 1998. Applied Multivariate Statistical Analysis – 4th Edition, Prentice-Hall, pp: 198-207.

[5] S J. Roberts, R. Everson and I. Rezek, 1999. Pattern Recognition, 33:5, pp: 833-839.

[6] Alfonso Valdes, Keith Skinner, "Adaptive, Model-based Monitoring for Cyber Attack Detection", SRI International., ã Springer-Verlag Berlin Heidelberg 2000, Debar, L. Me, and F. Wu (Eds): RAID 2000, LNCS 1907, pp. 80-92, 2000.

[7] Stephanie Forrest, S. A. Hofmeyr, A. Somayaji, and T. A. Longstaff. A sense of self for unix processes. In *Proceedings of the 1996 IEEE Symposium on Security and Privacy*, pages 120–128. IEEE Computer Society, 1996.

[8] M. Sobirey, B. Richter, and M. Konig. The intrusion detection system aid. architecture, and experiences in automated audit analysis. In *Proc. of the IFIP TC6 / TC11 International Conference on Communications and Multimedia Security*, pages 278 – 290, Essen, Germany, 1996.

[9] P. Helman and J. Bhangoo. A statistically base system for prioritizing information exploration under uncertainty. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 27:449–466, 1997.

[10] Christina Warrender, Stephanie Forrest, and Barak Pearlmutter. Detecting intrusions using system calls: alternative data models. In *Proceedings of the 1999 IEEE Symposium on Security and Privacy*, pages 133–145. IEEE Computer Society, 1999.

[11] W. Lee and S. J. Stolfo. Data mining approaches for intrusion detection. In *Proceedings of the Seventh USENIX Security Symposium*, 1998.

[12] Anup Ghosh and Aaron Schwartzbard. A study in using neural networks for anomaly and misuse detection. In *Proceedings of the Eighth USENIX Security Symposium*, 1999.

[13] T. Lane and C. E. Brodley. Temporal sequence learning and data reduction for anomaly detection. *ACM Transactions on Information and System Security*, 2:295–331, 1999.

[14] Srivastava, Shamik Sural and A.K. Majumdar, " Database Intrusion Detection using WeightedSequence Mining"JOURNAL OF COMPUTERS, VOL. 1, NO. 4, JULY 2006

[15] Emmanuel Hooper (2006), "An Intelligent Intrusion Detection and Response System Using Network Quarantine Channels: Adaptive Policies and Alert Filters" , Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT 2006 Workshops)(WI-IATW'06), pp. 16-21, 0-7695-2749-3/06 $20.00 © 2006.

[16] V. Venkatachalam, S.Selvan, "Intrusion Detection an Improved Competitive Learning Lamstar Neural Networks", Computer Science and Network Security, February 2007, Vol:7 No:2, p255

[17] Adel Nadjaran Toosi and Mohsen Kahani, "A new approach to intrusion detection based on an evolutionary soft computing model using neuro-fuzzy classifiers", Computer Communications, Volume30, issue 10, 31 July 2007, Pages 2201-2212. http://portal.acm.org/citation.cfm?id=1279397

[18] Victor-Vslerriu, Liviu Rusu, Iustin Priescu, "Data Mining Approaches for Intrusion Detection Email System Internet-based", CAIM, 2008 p144

[19] R. C. Chen and S. P. Chen, Intrusion Detection using a Hybrid support Vector Machine based on Entropy and TF-IDF, International Journal of Innovative computing, Information and control(IJICIC), Vol 4 No 2 2008, P 413

[20] Neelakantan.S. & Rao, "A Threat-Aware Signature Based Intrusion Detection Approach for Obtaining Network-Specific Useful Alarms, in the proceedings of "Internet Monitoring and Protection, 2008. ICIMP '08" Publication Date: June 29 2008-July 5 2008, ISBN: 978-0-7695-3189-2, (pp 80-85)

[21] M.V. Ramana Murthy, P. Ram Kumar, E. Devender Rao, A C sharma, S.Rajender and S.Rambabu, "Performance of the Network Intrusion Detection Systems", Computer Science and Network Security, Vol:9 No:10, Oct 2009, P 198

[22] Rung-Ching Chen, Kai-Fan Cheng, Ying-Hao Chen, Chia-Fen Hsieh, Chaoyang Univ. of Technol., Wufeng, Taiwan, Using Rough Set and Supporting Vector Machine for Nwtwork Intrusion System, Conference on Intelligent Information and Database Systems (April2009, p465), http://doi.ieeecomputersociety.org/10.1109/ACIIDS.2009.59

[23] Subbulakshmi T, George Mathew,Dr. S. Mercy Shalinie, "Real Time Classification And Clustering Of Ids Alerts Using Machine Learning Algorithms", International Journal of Artificial Intelligence & Applications (IJAIA), Vol. 1, No.1, January 2010.

[24] Vignesh R, Ganesh B, Aarthi G, Iyswarya N, "A Cache Oblivious based GA Solution for Clustering, Problem in IDS", International Journal of Computer Applications (0975 – 8887), 2010, Volume 1 – No. 11

[25] H. S. Teng , K. Chen, and S. C. Lu. Adaptive real-time anomaly detection using inductively generated sequential patterns. In *Proceedings of the IEEE Symposium on Research in Security and Privacy*, pages 278–284, Oakland CA, May 1990.

[26] P. Cheeseman and J. Stutz, 1996. "*Bayesian classification (Auto Class): theory and results in Advances in Knowledge Discovery and Data Mining*", edited by U.M. Fayyad et al., California: The AAAI Press, pp: 61-83.

[27] A.P. Dempster, N. M. Laird and D. B. Rubin, 1977. Journal of the Royal Statistical Society B 39, pp: 1-38.

[28] G. J. McLachlan, D. Peel, K. E. Basford and P. Adams, 1999. Journal of Statistical Software 04.

Authors:

1. G Jacob Victor: BE(CS), from Andhra University, M.Tech (CS), from BIT, Ranchi, India; Certified Software Quality Professional (CSQP), Certified Information System Auditor (CISA), Currently, Director (IT), SERP, RD Department, Govt. of AP, India. 25 years of IT Experience in Industry & Government.

2. Dr. M Sreenivasa Rao, Director, School of Information Technology, JNT University, Hyderabad,India, Obtained his Graduation and Post graduation in Engineering from JNT University, Hyderabad and Ph D from University of Hyderabad. Over 27 Years of IT Experience in the Academia and Industry.

3. Dr. V. Ch. Venkaiah,Professor,CRRao AIMS & CS, UoH Campus, Gachibowli, Hyderabad, India, obtained PhD. from IISc, Bangalore, 24 years of research experience & academic Recipient of CSIR's both Junior and Senior Research Fellowship. Selected for both CSIR Research Associateship and Indo-USSR Post doctoral fellowships.