ISSN:0975-9646



# A Survey on Intrusion Network Detection System Using Data Mining Techniques

[1]A. S. Kapse, [1]V. P. Kshirsagar, [2]Avinash Kapse, [1]M. B. Nagori
[1]*CSE ,Government College of Engineering, Aurangabad*
[2]*CSE Dept.Anuradha Engineering College Chikhli*

**ABSTRACT As more and more data goes online, there is a pressing need to secured dissemination of a large amount of information. Because of the effort required to monitor networks and systems manually, it is not easy to detect attempts at misuse or successful attacks without the help of intelligent intrusion Detection system (IDS). IDS, much like the security industry, have grown rapidly over the past few years. These tools have become essential security components- as valuable to many organizations as a firewall However, as in any environment, things change. Networks and crackers are evolving fast, demanding that security tools keep up. Intrusion Detection Systems face several daunting, but exciting challenges in the future and are sure to remain one of our best weapons in the area of network security. The modern day Network IDS faces some very challenging problems, like switched environment, increased network traffic, and encryption. Add that, the performance consideration of IDS, such as false positives and missed attacks, and the mole hill does become a mountain! The way to go seems to be analysis and data correlation. The concept of management console dedicated to the task of correlating abnormal event notifications, with relevance measures is an emerging one. One can pictures many distributed elements performing specific jobs, each passing the results onto a higher level for correlation and analysis. In an environment where many machines have similar configurations, a complete port scan on machine may trigger alarms but slow scans across ports of different machines might go unnoticed and will result in the intruder gaining all the information about the services running on each machine, thus successfully performing a distributed port scan. We focus on detecting a distributed port scan, by sniffing packets on the network. Five types of TCP port scans, performed by nap are successfully detected, in scan sweeps of one-to-one, one-to-many, many-to-one, many-to-many hosts. Our approach also manages to detect slow scans which are typically missed by available commercial packages, because of the feature that we select to examine.**

**Keywords- Anomaly Detection, Intrusion detection, clustering.**

## 1. INTRODUCTION

- An intrusion is a deliberate, unauthorized attempt to access or manipulate information or an system o render them unreliable or unusable.
- When suspicious activity is from your internal network it can also be classified as misuse

Networks have evolved over the years to a point of blissful convenience and flexibility, but within the joyful evolution, blatant breaches of security have surfaced. With the ever-changing world, ever-changing technologies and ever changing programs coming into existence that make our technical world more exciting and productive, the security world also changes. All networks are vulnerable. New holes, bugs and exploits are found by unscrupulous and unethical individuals to exploit for their own gain. Manufacturers of security products struggle to keep up with fixes, patches, and new releases in an effort to keep up-to-date with the surrounding market.

The aforementioned access points make today's networks more vulnerable to intrusions and attacks. Cyber-crime is no longer the prerogative of the stereotypical hacker. Joining ranks with the hackers are disgruntled employees, unethical corporations, and even terrorist organizations. With the vulnerability of present-day software and protocols combined with the increasing sophistication of attacks, it comes as no surprises that network based attacks are on the rise. The 2009 annual *Computer Crime and Security survey* [4], jointly conducted by the Computer Security Institute and the FBI. Although 86 percent of the respondents used firewalls, their consensus was that firewalls by themselves are not sufficient to provide adequate protection. Moreover, according to recent studies, an average of twenty to forty new vulnerabilities in commonly used networking and computer products are discovered every month. Such wide-spread vulnerabilities in software add to today's insecure computing/networking environment. This insecure environment has given rise to the ever evolving field of intrusion detection and prevention. The cyberspace's equivalent to the burglar alarm, intrusion detection systems complement the beleaguered firewall.

An intrusion detection system gathers and analyzes information from various areas within a computer or a network to identify possible security breaches. In other words, intrusion detection is the act of detecting actions that attempt to compromise the confidentiality, integrity or availability of a system/network. Traditionally, intrusion detection systems have been classified as signature detection systems, anomaly detection systems or a hybrid/ compound detection systems. A signature detection system identifies patterns of traffic or application data presumed to be malicious while anomaly detection systems compare activities against a "normal" baseline. On the other hand, a hybrid intrusion detection system combines the techniques of the two approaches. Both signature detection and anomaly detection systems have their share of advantages and drawbacks. The primary advantage of signature detection is that known attacks can be detected fairly reliably with a low false positive rate. The major drawback of the signature detection approach is that such systems typically require a signature to be defined for all of the possible attacks that an attacker may launch against a network. The biggest advantage of anomaly detection systems is that profiles of normal activity are customized for every system, application and/or network, and therefore making it very difficult for an attacker to know with certainty what activities it can carry out without getting detected. However, the anomaly detection approach also has its share of drawbacks: the intrinsic complexity of the system and the difficulty of associating alarms with the specific events that triggered those alarms [1] and [2].

However, most intrusion detection systems have not been able to keep up with the advances in high speed networking. Intrusion detection products, currently deployed in gigabit networks, need significant improvements before they can offer

adequate protection against attacks. A majority of the products in the market today can detect less than half of the attacks directed at them, even though many of those attacks are well documented. Therefore the leitmotif of this dissertation is the development of a network based anomaly detection system that can detect intrusions/attacks in a large, high volume and high speed enterprise network.

***Intrusion Detection-***

An intrusion detection system is a software tool used to detect unauthorized access to a computer system or network. An intrusion detection system is capable of detecting all types of malicious network traffic and computer usage. This includes network attacks against vulnerable services, data driven attacks on applications, host-based attacks-such as privilege escalation, unauthorized logins and access to sensitive files-and malware. An intrusion detection system is a dynamic monitoring entity that complements the static monitoring abilities of a firewall. An intrusion detection system monitors traffic in a network in promiscuous mode, very much like a network sniffer. The network packets that are collected are analyzed for rule violations by a pattern recognition algorithm. When rule violations are detected, the intrusion detection system alerts the administrator. One of the earliest work that proposed intrusion detection by identifying abnormal behavior can be attributed to Anderson [5]. In his report, Anderson presents a threat model that classifies threats as external penetrations, internal penetrations, and misfeasance, and uses this classification to develop a security monitoring surveillance system based on detecting anomalies in user behavior. External penetrations are defined as intrusions that are carried out by unauthorized computer system users; internal penetrations are those that are carried out by authorized users who are not authorized for the data that is compromised; and misfeasance is defined as the misuse of authorized access both to the system and to its data [2].

In other words, a model of the behaviour of the entity being monitored could be constructed by an intrusion detection system, and subsequent behaviour of the entity could be verified against the entity's model. In this model, behaviour that deviates sufficiently from the norm is considered anomalous. In the paper, Denning mentioned several models that are based on statistics, Markov chains, time-series etc [3].

**Alarm:** This part of the system handles all output from the intrusion detection system. The output may be either an automated response to an intrusion or a suspicious activity alert for a system security officer. Historically, intrusion detection research on the analysis and detection stage of the architectural model. As mentioned above, algorithms for the analysis and detection of intrusions/attacks are traditionally classified into the following three broad categories:

***i. Signature or misuse detection***

***ii. An anomaly detection system***

***iii. Statistical Anomaly Detection***

**Machine Learning based Anomaly Detection-**

Machine learning can be defined as the ability of a program and/or a system to learn and improve their performance on a certain task or group of tasks over time. Machine learning aims to answer many of the same questions as statistics or data mining. However unlike statistical approaches which tend to focus on understanding the process that generated the data, machine learning techniques focus on building a system that improves its performance with experience [8].

***System Call and Sequence Analysis-***One of the widely used machine learning techniques for anomaly detection involves learning the behaviour of a program and recognizing significant deviations from the normal. In a seminal paper, Forrest et al. [4] established an analogy between the human immune system and intrusion detection. They did this by proposing a methodology that involved analyzing a program's system call sequences to build a normal profile. In their paper, they analyzed several UNIX based programs like send mail etc. and showed that correlations in fixed length sequences of system calls could be used to build a normal profile of a program. Therefore, programs that show sequences that deviated from the normal sequence profile could then be considered to be victims of an attack. The system they developed was only used off-line using previously collected data and used a quite simple table-lookup algorithm to learn the profiles of programs. Their work was extended by However et al. [5,10], where they collected a database of normal behaviour for each program of interest. Once a stable database is constructed for a given program in a particular environment, the database was then used to monitor the program's behaviour. The sequences of system calls formed the set of normal patterns for the database, and sequences not found in the database indicated anomalies.

**Self-Similarity and Network Traffic-**

In the last decade, most of the studies on network traffic argued convincingly that Internet traffic is very far from being regular, and presents large variations in its throughput at all scales [5]. These studies have shown that Internet traffic exhibits characteristics such as self-similarity [6,8], multi-factuality [1], and long-range dependence [5], which implies that in all cases network traffic can vary significantly. In addition, given the highly variable nature of Internet traffic, anomaly based intrusion detection systems are raising alarms for many disruptions that are not attacks. The high rate of false positives is one of the major shortcomings of current IDS and the current evolution of Internet traffic with larger and larger variations among time continues to limit the efficiency of anomaly based IDS.

**Properties of Network Traffic-**

Network intrusion detection must distinguish between hostile and benign traffic, and must do so quickly to keep up with a high speed network. Depending on whether the intrusion detection system uses signature or anomaly detection, it must either model attacks (of which there are thousands) or normal traffic. There are two main challenges for modelling normal traffic for anomaly detection. First, network traffic is very complex and unpredictable, and second, the model changes over time.

Wide area network traffic contains a wide range of anomalies and uncharacteristic data that cannot be easily explained for. Examples include private IP addresses, storms of packets

routed in a loop until their TTLs expire; TCP acknowledgments of packets never sent, TCP retransmissions with inconsistent payloads, SYN packets with urgent data, and so on. ICMP packets with invalid code fields, and packets addressed to nonexistent hosts and ports. Many of these were investigated and found to be not hostile. Instead, many of the aforementioned errors were caused by mis-configured routers and/or DNS servers.

## 2. PROPOSED SAMPLING ALGORITHM

Traffic measurement and monitoring serves as the basis for a wide range of IP network operations and engineering tasks such as trouble shooting, accounting and usage profiling, routing weight configuration, load balancing, capacity planning, etc. Traditionally, traffic measurement and monitoring is done by capturing every packet traversing a router interface or a link. With today's high-speed links, such an approach is no longer feasible due to the excessive overheads it incurs on line-cards or routers. As a result, packet sampling has been suggested as a scalable alternative to address this problem. We describe the *weighted least squares* predictor that is utilized for predicting the next sampling interval. This predictor has been adopted because of its capability to follow the trends in network traffic.

**Weighted Least Square Predictor-**

Let us assume that the vector Z holds the values of the *N* previous samples, such that *ZN* is the most recent sample and *Z*1 is the oldest sample. Having fixed a window size of *N*, when the next sampling occurs, the vector is right shifted such that *ZN* replaces *ZN*-1 and *Z*1 is discarded. The weighted prediction model therefore predicts the value of *ZN* given *ZN*-1;*Z*1. In general, we can express this predicted value as a function of the *N* past samples i.e.,

$ZN = \alpha^{T} Z'$ where *Z'N* is the new predicted value, Z is the vector of past

*N* -1 samples, and $\alpha^{T}$ is a vector of predictor coefficients distributed such that newer values have a greater impact on the predicted value *Z'N* . A second vector, t, records the time that each sample is taken and is shifted in the same manner as Z. The objective of the weighted prediction algorithm is to find an appropriate coefficient vector, $\alpha^{T}$ such that the following summation is minimized

$$\Sigma = \sum_{i=0}^{n-1} Wi\,(Zi - Z')^2$$

Where *i*, *Zi*, and *Z'* denote the weight, the actual sampled value, and the predicted value in the *ith* interval, respectively. The coefficient vector is given by:

$$(Z^{T} WZ')^{-1} Z^{T} W$$

where $W = W^{T} W$ is a (*N*-1)X(*N* -1) diagonal weight matrix and w is a *N X*1 weight vector with weight co-efficient *wI* that are determined according to two criteria:

1. The "freshness" of the past *N* -1 samples. A more recent sample has a greater weight.
2. The similarity between the predicted value at the beginning of the time interval and the actual value. The similarity between the two values is measured by the distance between them. The smaller the Euclidean distance is, the more similar they are to each other.

**Adaptive Weighted Sampling-**

Adaptive sampling algorithms dynamically adjust the sampling rate based on the observed sampled data. A key element in adaptive sampling is the prediction of future behaviour based on the observed samples. The weighted sampling algorithm described utilizes the weighted least squares predictor to select the next sampling interval. Inaccurate predictions by the weighted least squares predictor indicates a change in the network traffic behaviour and requires a change in the sampling rate [7, 8,10].

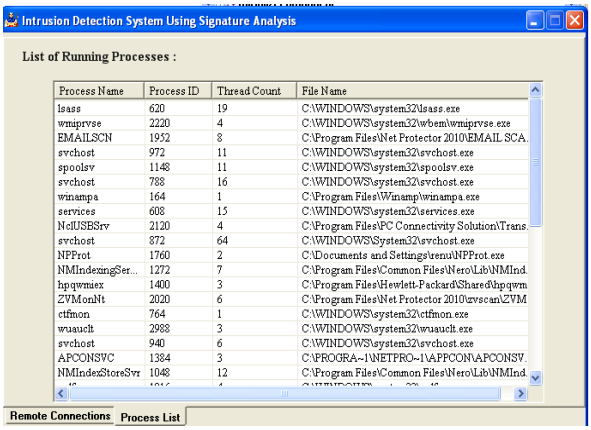


Figure: Intrusion Detection System using signature Analysis based on adaptive sampling algorithm

The proposed adaptive sampling algorithm consists of the following steps :

1. Fix the first *N* sampling intervals. (In our simulations we used $\mu = 60$ sec. and $N = 10$)
2. Apply the weighted least squares predictor to predict the anticipated value, *Z'N*, of the network parameter.
3. Calculate the network parameter value at the end of the sampling time period.
4. Compare the predicted value with the actual value.
5. Adjust sampling rate according to the predefined rule set if the predicted value differs from the actual value.

**My System**-To achieve the goal of detecting intrusions in high bandwidth environments, SCAN's architecture combines intelligent sampling and flow aggregation with data reduction and anomaly detection to achieve a high degree of accuracy in detecting intrusions with partial audit data. The design requirements for such a network based IDS were (a) stateless inspection of packets, protocols and/or packet headers at wire speed, (b) low occurrence of false alarms and high detection rate, (c) ability to track TCP states, and (d) ability to report events and/or alarms.

## 3. RESULTS

The results from the simulation based analysis that was performed on datasets. The evaluation methodology involved evaluating the components of datasets individually before combining them and testing the system as a whole. In this direction, we first evaluated the performance of the proposed sampling algorithm. By comparing its performance with the simple random sampling algorithm. We have check, through our simulation based analysis that the proposed sampling algorithm is superior to the traditionally used simple random sampling algorithm in predicting the packet mean delay while reducing the volume of traffic and maintaining the self similarity of the original network flows.

Our simulation results show that missing data, the accuracy of clustering is in the high.

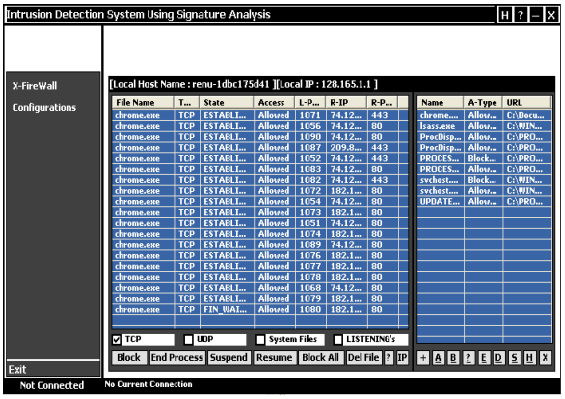


Figure: Accuracy of. percentage of missing data.

## 4. CONCLUSION

TCP port scan performed by source machine this machine passively collecting all the network traffic. The entire scans performed and captured in network data were detected. In any case, we believe that correlation is primarily a tool to support the user when examining a NIDS's output, rather than a fully automatic mechanism to find relevant alerts. Therefore, a user interface needs to integrate correlation techniques into a highly interactive interface. The user needs to be able to quickly jump back and forth between different levels of abstractions as well as different kinds of correlations. Moreover, relevant contextual information needs to be easily accessible (e.g., a summary of a host's activity). A high responsiveness is of utmost importance for such a degree of interactivity. A user should be able to play with the data to explore it many different ways.

## 5. REFERENCES

[1]. Abdul Samad bin Haji Ismail, Abdul Hanan Abdullah, Kamalrulnizam bin Abu Bak, "A Novel Method for Unsupervised Anomaly Detection using Unlabelled Data" IEEE2008.

[2]. Nam Hun Park, Won Suk Lee," Anomaly Detection over Clustering Multi-dimensional Transactional Audit Streams" IEEE2008.

[3]. C. C. Aggarwal and P. S. Yu, "Outlier detection for high dimensional data," in Proceedings of the 2001 ACM SIGMOD international conference on Management of data, T. Sellis, Ed. ACM Press, May 2001, pp. 37 – 46.

[4]. R. Agrawal, T. Imielinski, and A. Swami, "Mining association rules between sets of items in large databases," in In Proceedings of the ACM SIGMOD Conference on Management of Data. ACM Press, 1993, pp. 207–216.

[5]. D. Anderson, T. Frivold, A. Tamaru, and A. Valdes, "Next generation intrusion detection expert system (nides), software users manual," Computer Science Laboratory, SRI International, 333 Ravenswood Avenue, Menlo Park, CA 94025-3493, User Manual SRI–CSL–95–07, 1994.

[6]. D. Anderson, T. F. Lunt, H. S. Javitz, A. Tamaru, and A. Valdes, "Detecting unusual program behaviour using the statistical component of the next-generation intrusion detection expert system (nides)," Computer Science Laboratory, SRI International, 333 Ravenswood Avenue, Menlo Park, CA 94025-3493, Technical Report SRICSL-95 06, May 1995.

[7]. J. P. Anderson, "Computer security threat monitoring and surveillance," James P. Anderson Co., Fort

[8]. Washington, Pennsylvania, Technical Report, April 1980.

[9]. Shailesh Kumar Gaikwad, "A Secure Network Detection System against Noisy Unlabeled Data", IJCA, *Volume 9– No.9, November 2010*.

[10]. Lee, W., Stolfo, S.J. (2000), "A Framework for Constructing Features and Models for Intrusion Detection Systems", ACM Transactions on Information and System Security 3(4) pp. 227-26