Proc. of the Intl. Conf. on Advances in Computer Science and Electronics Engineering — CSEE 2013

ISBN: 978-981-07-5461-7

Hybrid Design Approach for Efficient Network Intrusion Detection using Data Mining and Network Performance Exploration

Nareshkumar D. Harale
Dr. B B Meshram

Abstract -The primary goal of an Intrusion Detection System (IDS) is to identify intruders and differentiate anomalous network activity from normal one. Intrusion detection has become a significant component of network security administration due to the enormous number of attacks persistently threaten our computer networks and systems. Traditional Network IDS are limited and do not provide a comprehensive solution for these serious problems which are causing the many types security breaches and IT service impacts. They search for potential malicious abnormal activities on the network traffics; they sometimes succeed to find true network attacks and anomalies (true positive). However, in many cases, systems fail to detect malicious network behaviors (false negative) or they fire alarms when nothing wrong in the network (false positive). In accumulation, they also require extensive and meticulous manual processing and interference. Hence applying Data Mining (DM) techniques on the network traffic data is a potential solution that helps in design and develop a better efficient intrusion detection systems. Data mining methods have been used build automatic intrusion detection systems. The central idea is to utilize auditing programs to extract set of features that describe each network connection or session, and apply data mining programs to learn that capture intrusive and non-intrusive behavior. In addition, Network Performance Analysis (NPA) is also an effective methodology to be applied for intrusion detection. In this research paper, we discuss DM and NPA Techniques for network intrusion detection and propose that an approach which will have the potential to detect intrusions in networks more effectively and help in increasing accuracy.

***Keywords-Intrusion Detection, Misuse Intrusion Detection, Anomaly Intrusion Detection, Network Intrusion Detection System, Data Mining Techniques, Network Performance Analysis*.**

Nareshkumar D. Harale
Dr. B B Meshram

I. INTRODUCTION

These days, there exists an extensive growth in Internet usage for social collaboration (e.g., instant messaging, audio/video conferences, etc.), healthcare, e-commerce, internet banking, online trading and many more other application services. These Internet applications need a satisfactory level of security and privacy. On the other hand, our computer systems and networks are vulnerable to attacks and vulnerable to many threats. There is an increasing availability of tools and tricks for attacking and intruding networks. An intrusion can be defined as any set of actions that threaten the security requirements (e.g., integrity, confidentiality, availability) of a computer/network resource (e.g., user accounts, file systems, and system kernels) [16, 17]. Intruders have promoted themselves and invented innovative tools that support various types of network attacks. Hence, effective methods for intrusion detection (ID) have become an insisting need to protect our computers from intruders. In general, there are two types of Intrusion Detection Systems (IDS); misuse detection systems and anomaly detection systems [14, 16, 17]. In such systems, known intrusions (signatures) are provided and hand-coded by human experts based on their extensive experience in identifying intrusions. Current misuse IDS are built based on: expert systems (e.g., IDES, ComputerWatch, NIDX, P-BEST, ISOA) which use a set of rules to describe attacks, signature analysis (e.g., Haystack, NetRanger, RealSecure, MuSig) where features of attacks are captured in audit trail, state-transition analysis (e.g., STAT, USTAT and NetSTAT) which uses state-transition diagrams, colored Petri nets (e.g., IDIOT), or case-based reasoning (e.g., AUTOGUARD) [16]. Anomaly detection [8, 12], in contrast to misuse detection, can identify novel intrusions. It builds models for normal network behaviour (called profiles) and uses these profiles to detect new patterns that significantly deviate from them. These suspicious patterns may represent actual intrusions or could simply be new behaviors that need to be added to profiles. Current anomaly detection systems use statistical methods such as multivariate and temporal analysis to identify anomalies; examples of these systems are IDES, NIDES, and EMERALD. Other anomaly detection systems are built based on expert systems such as ComputerWatch, Wisdom, and Sense [16].

Proc. of the Intl. Conf. on Advances in Computer Science and Electronics Engineering — CSEE 2013

ISBN: 978-981-07-5461-7

Misuse IDS suffer from a number of major drawbacks, first, known intrusions have to be hand-coded by experts. Second, signature library needs to be updated whenever a new signature is discovered, network configuration has been changed, or a new software version has been installed. Third, misuse IDS are unable to detect new (previously unknown) intrusions that do not match signatures; they can only identify cases that match signatures. Thus, the system fails to identify a new event as an intrusion when it is in fact an intrusion, this is called false negative. On the other hand, current anomaly detection systems suffer from high percentage of false positives (i.e., an event incorrectly identified by the IDS as being an intrusion when it is not) [16]. An additional drawback is that selecting the right set of system features to be measured is ad hoc and based on experience. A common shortcoming in IDS is that for a large, complex network IDS can typically generate thousands or millions of alarms per day, representing an overwhelming task for the security analysts [16, 17]. Table 1 shows a comparison between the two types of intrusion detection.

TABLE I: Network IDS Comparative Assessment

	Misuse based Intrusion Detection	**Anomaly based Intrusion Detection**
Characteristics	Make use of patterns of well-known attacks (signatures) to identify intrusions, any match with signatures is reported as a possible network attack	Make use of deviation from normal usage patterns to identify intrusions, any significant deviations from the expected behavior or defined user profile are reported as possible attacks
Drawbacks	False negatives - Unable to detect new attacks -Need signatures update - Known attacks has to be hand-coded, Overwhelming security analysts	False positives. - Selecting the right set of system features to be measured is ad hoc and based on experience however it has to study sequential interrelation between transactions , Overwhelming security analysts

From the above discussion, we conclude that traditional IDS face many limitations. This has led to an increased interest in improving current IDS. Applying Data Mining (DM) techniques such as classification, clustering, association rules, etc, on network traffic data in real time is a promising solution that helps improves IDS [15-23]. In addition, Network Performance Analysis (NPA) is also an effective technique for network intrusion detection [4, 6, 25, 26]. In this paper, we discuss DM and NPA approaches for network intrusion detection and suggest that a combination of both approaches has the potential to detect intrusions in computer networks more effectively. The rest of this paper is organized as follows: in section 2 we give background information and related work. In section 3 we discuss NPA systems. In section 4 we suggest an IDS model that integrates DM techniques and NPA feature. Finally, in section 5, we give our conclusions and future work.

A major shortcoming of the current IDSs that employ data mining methods is that they can give a Series of false alarms in case of a noticeable systems environment modification and a user can deceive the system by slowly changing behavior patterns. There can be two types of false alarms in classifying system activities in case of any deviation from normal patterns: *false positives* and *false negatives*. False positive alarms are issued when normal behaviors are incorrectly identified as abnormal and false negative alarms are issued when abnormal behaviors are incorrectly identified as normal. Though it's important to keep both types of false alarm rates as low as possible, the false negative alarms should be the minimum to ensure the security of the system. To overcome this limitation, IDS must be capable of adapting to the changing conditions typical of an intrusion detection environment. For example, in an academic environment, the behavior patterns at the beginning of a semester may be different than the behavior patterns at the middle/end of the semester If the system builds its profile based on the audit data gathered during the early semester days, then the system may give a series of false alarms at the later stages of the semester. System security administrators can tune the IDS by intervention. Again, the patterns of intrusions may be dynamic. Intruders may change their strategies over time and the normal system activities may change because of modifications to work practices. Moreover, it is not always possible to predict the level of intrusions in the future. So it is important that IDS should have automatic adaptability to new conditions.

One straightforward approach can be to regenerate the user profile with the new audit data. But this would not be a computationally feasible approach. When the current usage profile is compared with the initial profile, there can be different types of deviation as mentioned in section 2.1. Each of these deviations can represent an intrusion or a change in behavior. In case of a change in system behaviors, the base profile must be updated with the corresponding change so that it doesn't give any false positives alarms in future. So the system needs to decide whether to make a change or reject it. If the system tries to make a change to the base profile every time it sees a deviation, there is a potential danger of incorporating intrusive activities into the profile. The IDS must be able to adapt to these changes while still recognizing abnormal activities and not adapt to those. If both an intrusion and behavior change occur during a particular time interval, it becomes more complicated. Again, which rules to add, which to remove, is critical. Moreover, there are more issues that need to be addressed in case of updating. The system should adapt to rapid changes as well as gradual changes in system behavior. Selecting the time interval at which the update should take place is also an important issue. If the interval is too long, the system may miss some rapid changes or short-term attacks. If the interval is too small, the system may miss some long-term changes. So, we consider two problems as the major issues in developing a true adaptive intrusion detection system. One is to select the time when the update should be

Proc. of the Intl. Conf. on Advances in Computer Science and Electronics Engineering — CSEE 2013

ISBN: 978-981-07-5461-7

made. The other is to select a mechanism to update the profile. To tackle the first issue, we can trace the similarity pattern found by comparing each day's activities with the base profile. If the similarity goes down the threshold line and experiences a sharp shift, we would consider that as an abnormal behavior. If the similarity goes down the threshold line, but does not experience a sharp shift, rather experiences a slow downwards trend, we would consider that as a It is not computationally feasible to archive audit data for a long time. So we may employ a sliding window Technique to update the base profile. We can assume that system activities before a certain period of time are too old to characterize the current behavior, i.e., the audit records before that period are unlikely to contribute towards the rules that represent system activities. We can define a sliding window [t_1, t_2, ... , t_n] of n days. We would maintain both the large item sets and the negative border. As time goes on, a large item set may start losing its support and an item set in the negative border may start gaining support. We would discard some large item sets in the process and include some new item sets. The update technique would reject transactions outside the sliding window as they are assumed to be old and outdated. We can use different techniques to update the profile rule set [34].

II. RELATED WORK

Intrusion detection is the process of monitoring and analyzing the data and events occurring in a computer and/or network system in order to detect attacks, vulnerabilities and other security problems [16]. IDS can be classified according to data sources into: host-based detection and network-based detection. In host-based detection, data files and OS processes of the host are directly monitored to determine exactly which host resources are the targets of a particular attack. In contrast, network-based detection systems monitor network traffic data using a set of sensors attached to the network to capture any malicious activities. Networks security problems can vary widely and can affect different security requirements including authentication, integrity, authorization, and availability. Intruders can cause different types of attacks such as Denial of Services (DoS), scan, compromises, and worms and viruses [17, 18]. In this paper, we emphasize on network-based intrusion detection which is discussed in the next sub-section. The important hypothesis in intrusion detection is that user and program activities can be monitored and modeled [16,17]. A set of processes represent the framework of intrusion detection, first, data files or network traffic are monitored and analyzed by the system, next, abnormal activities are detected, finally, the system raises an alarm based on the severity of the attack [16]. Figure 1 below shows a traditional framework for ID. In order for IDS to be successful, a system is needed to satisfy a set of requirements. IDS should be able to detect a wide variety of intrusions including known and unknown attacks. This implies that the system needs to adapt to new attacks and malicious behaviors. IDS are also required to detect intrusions in timely fashion, i.e., the system may need to respond to intrusions in real-time. This may represent a challenge since analyzing intrusions is a time consuming process that may delay system response. IDS are required to be accurate in a sense that minimizes both false negative and false positive errors. Finally, IDS should present analysis in simple, easy-to understand format in order to help analysts get an insight of intrusion detection results [16].

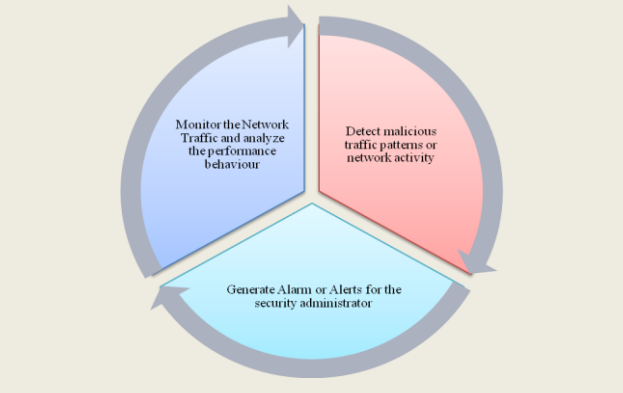


Figure 1. Traditional Network IDS Logical Function Layout

A. Network-based Intrusion Detection System

Network-based intrusion detection can be broken down into two categories: packet-based anomaly detection and flow-based anomaly detection. Flow-based anomaly detection tends to rely on existing network elements, such as routers and switches, to make a flow of information available for analysis. On the other hand, packet-based anomaly detection doesn't rely on other network components; it observes network traffic for the detection of anomalies. Flow-based anomaly detection is based on the concept of a network flow and flow records. A flow record is a summarized indicator that a certain network flow took place and that two hosts have communicated with each other previously at some point in time. Typically, the flow record contains both the source and destination IP addresses the source and destination TCP or UDP network ports or ICMP types and codes, the number of packets and number of bytes transmitted in the session, and the timestamps for both the start and end of the network flow. Routers generate these flow records as they observe network traffic. By analyzing flow records and looking for unusual amounts, directions, groupings and characteristics of the network flow, the Network Performance Analysis software can infer the presence of worms or even DoS attacks in a network. The problem is that these flow records only carry a summary of the information presented for analysis. Basically, this information is the metadata about the network traffic. The

ISBN: 978-981-07-5461-7

actual network packets are not accessible for further analysis [9]. Packet-based anomaly detection software, unlike its flow-based counterpart, does not use third party elements to generate the metadata of the network traffic. Instead, the entire packet-based analysis looks at raw packets as they traverse the network links. Observation of the network traffic can be done using either port mirroring or network taps. Port mirroring, known as SPAN (Switched Port Analyzer), is used on a network switch to send a copy of all network packets seen on one switch port to a network monitoring connection on another switch port. Network taps are used to create permanent access ports for passive monitoring. Test Access Port (TAP) can create a monitoring access port between any two network devices, including switches, routers, and firewalls. A good example to compare the two detection methodologies is that of a large-scale SYN flood denial of service attack. Typically a huge amount of connection request packets are generated by a number of compromised zombie machines. The source addresses are randomly generated [4].

A flow-based anomaly detection system only sees that there is a large number of network flows, which are established from many clients to the specific server and port that is under attack. But, no useful information beyond that is forthcoming from a flow-based solution. Therefore, the network operator has the choice to either rate-shape or blocks all traffic to that server, with disastrous impact on even the valid traffic [4]. On the other hand, a packet-based anomaly detection system can extract the signature of the offending packets. Often, large-scale attack tools initialize packet headers with certain, non-random data. For example, the TCP window size or sequence number, which is advertised in a connection request packet, could be fixed. A packet-based anomaly detection system, which has access to the raw packet data, can detect this and provide a signature of the packets that only block the offending traffic, and leaves valid traffic untouched. Since routers and switches tend to send out their network flow after there has been a period of inactivity (on average about 15 seconds), the "earliest a flow-based anomaly detection solution can begin to detect the anomaly is at least 15 seconds after its onset" [4]. After that, the detection algorithms can begin processing, which further adds to the delay in finding an anomaly or not.

In a flow-based anomaly detection system, the routers and switches are the components that produce the flow records. These flow records are the only insight into the current network traffic. The problem with this is that many anomalies and malicious activity could be either designed to affect the routers and switches or take them down as a side effect of the actual cause of the attack. In this case, during the worst possible time, the flow-based system could fail to detect anything in the middle of an attack, since the router has failed. The packet-based anomaly detection system works in real time since it doesn't depend on any third party components, such as routers or switches. Because of that there is no 15 second time delay before the statistical data on the network traffic is available to the software. As long as traffic is flowing on the network links, it is seen and analyzed. The detection algorithms are continuously at work on this data. Since flow records are generated using flow-based detection, it places a heavy burden on network infrastructure. Many routers' CPUs are heavily loaded when flow record generation occurs, which as a side effect can interfere with the other activities the router is responsible for. Looking at the previous denial of service example, each packet could represent a new flow record. Flow record generation by the routers and switches could act in effect to strengthen the attack, as the more malicious packets get generated, the more the router is loaded in processing the flow record, thus taxing the network with a plethora of flow records. On the other hand, packet-based detection does not cause any additional overhead on the routers, switches or the network since both the processing of the detection and all necessary communication are independent of the attack volume.

Detection accuracy is not as great comparatively with flow-based anomaly detection as it is with packet-based. The reason is that due to the resource intensive flow record generation, an often method of counteraction is flow sampling. Flow sampling is considering every nth packet for the generation of flow records, not every packet. This causes the number of flow records to dramatically decrease as well as decreasing the CPU load and network utilization. The price for this method is the loss of detection accuracy. In most cases, smaller flows will be seen only as one packet even if the flow does contain a packet or not. Large flows on the other hand will be over represented, since they have a higher chance of having at least one of their packets sampled, which could lead to a distorted picture of the actual state of the network. Flow sampling could cause an additional delay in detecting anomalies, as it takes more packets passing through the router to find out if there is an anomaly or not. Packet-based anomaly detection does not have to rely on sampling since it is not as resource intensive as flow-based detection. Thus, the accuracy rate of detection is higher.

B. Data Mining Techniques for Network Intrusion Detection

Many researchers have investigated the deployment of data mining algorithms and techniques for intrusion detection [13, 15-23]. Examples of these techniques include [16-18]:

Feature selection data analysis: The main idea in feature

ISBN: 978-981-07-5461-7

selection is to remove features with little or no predictive information from the original set of features of the audit data to form a subset of appropriate features [24]. Feature selection significantly reduces computational complexity resulting from using the full original feature set. Other benefits of feature selection are: improving the prediction of ID models, providing faster and cost-effective ID models and providing better understanding and virtualization of the generated intrusions. Feature selection algorithms are typically classified into two categories: subset selection and feature ranking. Subset selection algorithms use heuristic search such as genetic algorithms, simulated annealing and greedy hill climbing to generate and evaluate a subset of features as a group for suitability. On the other hand, feature ranking uses a metric to rank the features based on their scores on that metric and removes all features that do not achieve an adequate score [34].

Classification analysis: The goal of classification is to assign objects (intrusions) to classes based on the values of the object's features. Classification algorithms can be used for both misuse and anomaly detections [16]. In misuse detection, network traffic data are collected and labeled as "normal" or "intrusion". This labeled dataset is used as a training data to learn classifiers of different types (e.g., SVM, NN, NB, or ID3) which can be used to detect known intrusions. In anomaly detection, the normal behaviour model is learned from the training dataset that are known to be "normal" using learning algorithms. Classification can be applied to detect intrusions in data streams; a predefined collection of historical data with their observed nature helps in determining the nature of newly arriving data stream and hence will be useful in classification of the new data stream and detect the intrusion. Data may be non sequential or sequential in nature. Non-sequential data are those data where order of occurrence is not important, while sequential data are those data where the order of occurrence with respect to time is important to consider. Using data mining and specially classification techniques can play a very important role on two dimensions; the similarity measures and the classification schema [28]. Kumar [27] stated that any data, facts, concepts, or instructions, can be represented in a formalized manner suitable for communication, interpretation, or processing by humans or by automated means. Kumar [27] classified sequential data into temporal or non-temporal, where temporal data are those data, which have time stamp attached to it and non-temporal data are those which are ordered with respect to some other dimension other than time such as space. Temporal data can be classified into discrete temporal sequential data such as logs time or continuous temporal sequential data such as observations.

Clustering analysis: Clustering assign objects (intrusions) to groups (clusters) on the basis of distance measurements made on the objects. As opposed to classification, clustering is an unsupervised learning process since no information is available on the labels of the training data. In anomaly detection, clustering and outlier analysis can be used to drive the ID model [16]. Distance or similarity measure plays an important role in grouping observations in homogeneous clusters. It is important to formulate a metric to determine whether an event is deemed normal or anomalous using measures such as Jaccard similarity measure, Cosine similarity measure, Euclidian distance measure and longest common subsequence (LCS) measure. Jaccard similarity coefficient is a statistical measure of similarity between sample sets and can be defined as the degree of commonality between two sets [29]. Cosine similarity is a common vector based similarity measure and mostly used in text databases and it calculates the angle of difference in direction of two vectors, irrespective of their lengths [30]. Euclidean distance is a widely used distance measure for vector spaces, for two vectors X and Y in an n-dimensional Euclidean space; Euclidean distance can be defined as the square root of the sum of differences of the corresponding dimensions of the vectors [29]. Mining models for network intrusion detection view data as sequences of TCP/IP packet and K-Nearest neighborhood algorithms is commonly used in all techniques with different similarity measures. Finally, clustering and classification algorithms must be efficient scalable, and can handle network data of high volume, dimensionality, and heterogeneity [16]. Han and Kamber [16] mentioned some other DM approaches that can be used for ID and we summarize them below:

Association and correlation analysis: The main objective of association rule analysis is to discover association relationships between specific values of features in large datasets. This helps discover hidden patterns and has a wide variety of applications in business and research. Association rules can help select discriminating attributes that are useful for intrusion detection. It can be applied to find relationships between system attributes describing network data. New attributes derived from aggregated data may also be helpful, such as summary counts of traffic matching a particular pattern.

Stream data analysis: Intrusions and malicious attacks are of dynamic nature. Moreover, data streams may help detect intrusions in the sense that an event may be normal on its own, but considered malicious if viewed as part of a sequence of events [16]. Thus, it is necessary to perform intrusion detection in data stream, real-time environment. This helps identify sequences of events that are frequently encountered together, find sequential patterns, and identify outliers. Other data mining methods for finding evolving clusters and building dynamic classification models in data streams can be applied for these purposes.

Distributed data mining: Intruders can work from several different locations and attack many different destinations. Distributed data mining methods may be utilized to analyze network data from several network locations, this helps

Proc. of the Intl. Conf. on Advances in Computer Science and Electronics Engineering — CSEE 2013

ISBN: 978-981-07-5461-7

detect distributed attacks and prevent attackers in different places from harming our data and resources.

Visualization and querying tools: Visualization data mining tools that include features to view classes, associations, clusters, and outliers can be used for viewing any anomalous patterns detected. Graphical user interface associated with these tools allows security analysts to understand intrusion detection results, evaluate IDS performance and decide on future enhancements for the system.

C. Network Analysis and Performance Assessment

Within the last few years, Network Performance Analysis (NPA) has been one of these emerging technologies that have been sold as a security management tool to improve the current network security status. The main focus of NPA is to monitor inbound and outbound traffic associated with the network to ensure that nothing is getting into the servers, software, and application systems which helps enhance the overall security of the network at all levels. The author in [1] stated that approximately 25% of large enterprises systems will be using NPA by 2011. The traditional security model of network as shown in figure 2 is not clear and has too many concerns. First of all, the model have little proactive capability attitude toward preventing any security incidents because the architecture is built with technologies that discover most security events in progress while it misses opportunities to detect and resolve other small threats before it become major problems for the network. Firewalls and intrusion detection systems are typically stationed at a network gateway, which doesn't stop laptops infected with malware or subversive employees from accessing the network. A typical security tactic to overcoming this problem is to deploy firewalls and intrusion detection devices throughout the internal network [4]. This can get extremely expensive and can increase network maintenance and complexity even without addressing many of the security threats.

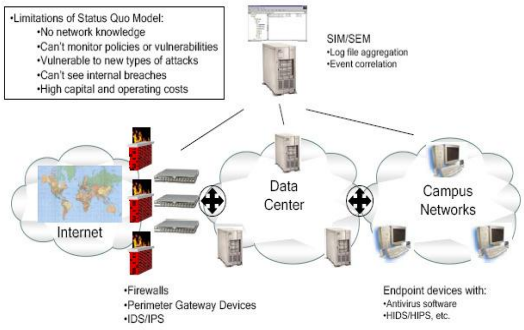


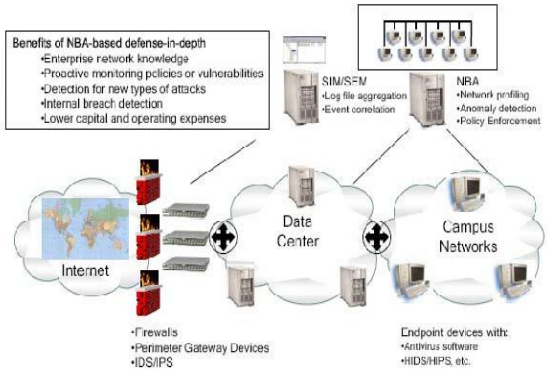
Figure 2: Traditional Network Defense Strategy Model (SANSI 2009), Source Enterprise Strategy Group

Without NPA systems added to the security model, the architecture could require three to four times more intrusion prevention system devices that if it had it. Though intrusion detection and intrusion prevention systems can spot common and signature based attacks such as port scans, denial of services, and certain viruses, but they cannot trap the security attacks that fast spreading such as zero-day worms. Other potential attacks such as reverse tunneling and island hopping look like normal traffic so there is no signature to detect the breach [4]. Since this traditional security model is event-based, log files become irrelevant as they do not provide a true picture of the internal control metrics for security administrators and auditors. This limitation forces companies into expensive manual process and lengthy audit cycles. Once these security events penetrate the internal network, the traditional model can provide little help. The security devices tend to reside at either the perimeter or at gateways of the network, so there is a possibility that they might miss internal attacks in other network segments.

The addition of a Network Performance Analysis system to anchor the traditional security architecture as in figure 3 can have several benefits. First benefit, NPA systems provide visibility into how both applications and services are being used within the network [5]. This allows for the identification of risky activities, creation of more secure network segments, fine tuning of corporate access policies, and the ability to deploy security appliances more effectively. The NPA system's network monitoring capabilities allows for monitoring historical trends to help improve security over time. For example, a security administrator may see a security attack in the sales department of a company where managers travel with their laptops and have to access unsecured networks [6].

Using this information from the historical trends, the network security team can deploy security countermeasures such as an intrusion prevention device within that network segment. NPA systems can also help detect attacks such as zero-day worms and suspicious insider activities across the network. In this respect, NPA can adapt the use of an intrusion detection system that tracks signature based security attacks faster. Finally, NPA systems can view the network in terms of the applications and users consuming services from specific servers. This helps organizations setup proper internal controls and enforces a network usage policy. This can prevent users from setting up their own servers or using inappropriate services. It also ensures that developers and development servers do not mingle with production systems [7].

ISBN: 978-981-07-5461-7



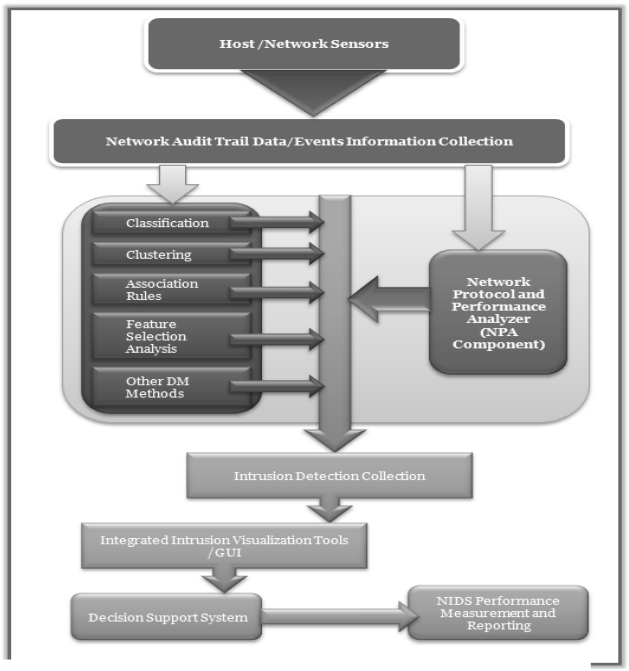
F

igure 3: Network Defense Strategy Model Anchored by Network Performance Analysis (SANSI 2010) Source Enterprise Strategy Group.NPA software examines network traffic or statistics on network traffic to identify unusual traffic flows through network monitoring activity. NPA systems can monitor the network and flag policy violations and see unwanted services, detect backdoors on host servers, and report unusual activity that may or may not be security related. Most of the mentioned services depend on the baseline profile of the network traffic that is created or modified when the NPA system is configured. Unlike traditional IDS, these sensors aren't deployed inline so they don't add latency to the network [2]. There are four major players in NPA market today. All of these vendors have marketed the tool as a network security solution. Lancope is the provider of StealthWatch [11], the most widely used NPA and response solution that claims to streamline network operations and security into one process. StealthWatch protects over 200 enterprise customers, more than all direct competitors combined [10]. Arbor Networks delivers the NPA solution Arbor Peakflow, which claims to provide real-time views of network activity enabling organizations to instantly protect against worms, DoS attacks, insider misuse, and traffic and routing instability as well as segment and harden networks from future threats. Q1 Labs' QRadar claims to be a cross between a Security Event Management (SEM) tool and a NPA system. The last major player is Mazu Networks, the leading provider of NPA systems to large enterprises, offers two applications, the Mazu Profiler, which is an internal security, application visibility and compliance solution, and the Mazu Enforcer, which allows for perin protectior

2005 and has grown its customer base to more than 100 enterprises. In conclusion, there are an underlying agenda to the majority of the sources that discussed NPA systems. Most of the sources, which had a positive analysis on NPA systems, tended to be white papers from vendors, such as Lancope, Mazu, Q1 Labs, or Arbor. Companies do seem to have benefited from using NPA systems as a security tool, rather than just a network management tool and they used flow-based anomaly detection as the major security tool.

III. PROPOSED IDS MODEL BASED ON DM AND NPA

Due to the many advantages of DM and NPA approaches in network intrusion detection, we suggest that a combination of both approaches can help develop a new generation of high performance IDS. In comparison to traditional IDS (Fig.1), IDS based on DM and NPA are generally more precise and require far less manual processing and input from human experts.



ISBN: 978-981-07-2950-9 doi:10.3850/ 978-981-07-2950-9 123

As depicted in the given figure 4, the proposed Network IDS Design Architecture based on DM and NPA. The sySystem is composed of the following essential key components given in below TABLE II:

TABLE II. Proposed NIDS Design Component and its functional Description

Proposed NIDS Design Component Name	Functional Description
Computer Network Sensors	Network Sensors collect the network audit data and network traffic events and send out these data to Intrusion Detection Blocks (IDB).

ISBN: 978-981-07-5461-7

DM- IDB	This component encompasses different modules that employ various DM Algorithms and Techniques (e.g., Classification, Clustering, etc.). Each module works independently to detect intrusions in the network traffic data in real-time.
NPA-IDB	This component deploys NPA to detect intrusions or anomalous traffic in the network audit data.
Intrusion Data Collection Block	This block is responsible for collection of detected intrusions from DM-IDB and NPA-IDB.
Integrated Intrusion Detection Visualization Block	This module helps in monitoring and visualizing the intrusion results and / trends of both ID Blocks.
Decision Making Support System	It analyzes intrusion results, evaluates system performance, takes decisions based on the detected intrusions, also checks for false positives and false negatives , controls system operation, generates a performance report periodically as well as need basis and decides if any configuration changes/updates are needed.

IV. CONCLUSIONS AND FUTURE WORK

Traditional Network IDS suffer from different problems that limit their detection effectiveness and efficiency. In contrast DM and NPA are promising approaches for network level intrusion detection in the complex enterprise environment. In this paper, we have been discussed DM and NPA approaches for network intrusion detection. We suggested that a combination of both approaches may overcome the limitations in current Network IDS and leads to high performance including the intrusion detection accuracy by reducing the false positives. NPA can help to cover the gap in traditional network systems, which considers a good move for most of industries to integrate NPA with advanced DM to achieve a better performance. NPA can significantly enhance the value of the data generated from IDS that use DM as intrusion detection technique by analyzing and correlating large amount of sequence data. We plan to put the suggested fusion system model in practice and apply it on real world intrusion detection problems.

References

[1] Schwartz, Matthew, "Beyond Firewalls and IPS: Monitoring Network Behavior." February 2006, available on http://esj.com/articles/2006/02/07/beyond-firewalls-and-ips-monitoring-networkbehavior.aspx

[2] K. Scarfone and P. Mell, "Guide to Intrusion Detection and Prevention Systems (IDPS)", NIST Special Publication 800-94, 2007, Available online: http://csrc.nist.gov/publications/nistpubs/80094/SP800-94.pdf

[3] Conry-Murray, "Anomaly Detection On the Rise", June 2005, available on http://business.highbeam.com/787/article-1G1-132920452/anomaly-detection-rise-network-behavioranomaly-detection

[4] Enterprise Strategy Group, "Network Performance Analysis Systems: The New Foundation of Defense-in-Depth", Technical White Paper, November 2005. http://www.enterprisestrategygroup.com/

[5] Mazu Networks, "What You Can't See Can Hurt You: Ensuring Application Availability through Enterprise-Wide Visibility", November 2006 , http://www.developertutorials.com/whitepapers/network-communications/

[6] Liebert, Chris, "Internal Threat Protection with Net-Based Detection, Prevention and Behavioral Systems", October 2006, http://www.mazunetworks.com/resources/analystreports/Internal_Threat_Protection_January_06.pdf.

[7] Enterprise Management Associates: Behavioral Analysis Enables a New Level of Network Security Awareness, technical White Paper, June 2004. http://security.ittoolbox.com/research/behavioralanalysis-enables-a-new-level-of-network-security-awareness-3755

[8] Tanase, Matthew, " One of These Things is not Like the Others: The State of Anomaly Detection", 2010, http://www.symantec.com/connect/articles/one-these-things-not-others-state-anomalydetection

[9] Esphion: Packet vs. flow-based anomaly detection. Technical White Paper, July 2005. http://trendmap.net/support/wp/ESP_WP_4_PACKET_V_FLOWS.pdf

[10] Network Intelligence, "Network Intelligence Integrates Network Performance Analysis Solutions to Enable Broad Internal Threat Detection", June 2006, http://www.rsa.com/press_release.aspx?id=7564.

[11] Detroit Tigers Select Lancope's StealthWatch to Protect Comerica Park Network, October 2006. http://www.lancope.com/news-events/press-releases/detroit-tigers-select-lancopes-stealthwatch-toprotect-comerica-park-networ/

[12] C. Kruegel and G. Vigna. "Anomaly detection of web-based attacks", in ACM CCS'03

[13] S. Mukkamala et al. " Intrusion detection using neural networks and support vector machines", in IEEE IJCNN May 2002.

[14] S. Axelsson, "Intrusion Detection Systems: A Survey and Taxonomy". Technical Report 99-15, Chalmers Univ., March 2000. http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.1.6603

[15] Susan M. Bridges , Rayford B. Vaughn, "Data mining and genetic algorithms applied to intrusion detection", In Proceedings of the National Information Systems Security Conference, 2000.

[16] Jiawei Han and. Micheline Kamber, Data Mining: Concepts and Techniques, Morgan Kufmann, 2nd edition 2006, 3rd edition 2011.

[17] S.J. Stolfo, W. Lee. P. Chan, W. Fan and E. Eskin, "Data Mining – based Intrusion Detector: An overview of the Columbia IDS Project" ACM SIGMOD Records vol. 30, Issue 4, 2001. th

[18] W. Lee and S.J. Stolfo, "Data Mining Approaches for Intrusion Detection" 7USENIX Security Symposium, Texas, 1998.

[19] S. Terry Brugger, "Data Mining Methods for Network Intrusion detection", University of California, Davis, 2004. http://www.mendeley.com/research/data-mining-methods-for-network-intrusion-detection/

[20] T. Lappas and K. Pelechrinis, "Data Mining Techniques for (Network) Intrusion Detection Systems" http://atl-svn.assembla.com/svn/odinIDS/Egio/artigos/datamining/dataIDS.pdf

[21] P. Dokas, L. Ertoz, V. Kumar, A. Lazaevic. J. Srivastava, and P. Tan,

ISBN: 978-981-07-5461-7

"Data Mining for Network Intrusion Detection", 2002, http://minds.cs.umn.edu/papers/nsf_ngdm_2002.pdf
[22] M. Hossain "Data Mining Approaches for Intrusion Detection: Issues and Research Directions", http://www.cse.msstate.edu/~bridges/papers/iasted.pdf
[23] A. Chauhan, G. Mishra, and G. Kumar, "Survey on Data mining Techniques in Intrusion Detection", International journal of Scientific & Engineering Research Vol.2 Issue 7, 2011.
[24] I. Guyon and A. Elisseeff, "An Introduction to Variable and Feature Selection", Journal of Machine Learning Research 3 (2003) 1157-1182
[25] Weili Han, Dianxun Shuai and Yujun Liu, "Network Performance Analysis Based on a Computer Network Model", Lecture Notes in Computer Science, 2004, Volume 3033/2004, 418-421, [26] Jack Timofte and Praktiker Romania," Securing the Organization with Network Performance Analysis", Economy Informatics, 1-4/2007.
[27] P. Kumar, P.R. Krishna, B. S Raju and T. M Padmaja, "Advances in Classification of Sequence Data", Data Mining and Knowledge Discovery Technologies. IGI Global, 2008, pp.143-174.
[28] A. Sharma, A.K. Pujari, and K.K. Paliwal, "Intrusion detection using text processing techniques with a kernel based similarity measure", presented at Computers & Security, 2007, pp.488-495.
[29] P. Kumar, M.V. Rao, P.R. Krishna, and R.S. Bapi, "Using Sub-sequence Information with kNN for Classification of Sequential Data", in Proc. ICDCIT, 2005, pp.536-546.
[30] G. Qian, S. Sural, Y. Gu, and S. Pramanik, "Similarity between Euclidean and cosine angle distance for nearest neighbor queries", in Proc. SAC, 2004, pp.1232-1237.
[31] P. Kumar, R.S. Bapi, and P.R. Krishna, "A New Similarity Metric for Sequential Data", presented at IJDWM, 2010, pp.16-32.
[32] Gudadhe, M.; Prasad, P.; Wankhade, K.; "A new data mining based network Intrusion Detection model" International Conference on Computer and Communication Technology (ICCCT), 17-19 Sept.
[33] Dartigue, C.; Hyun Ik Jang; Wenjun Zeng;" A New Data-Mining Based Approach for Network Intrusion Detection" Seventh Annual Communication Networks and Services Research Conference (CNSR), 11-13 May 2009 [34] Mohmood Husain," Data Mining Approaches for Intrusion Detection: Issues and Research Directions", Department of Computer Science, Mississippi State University, MS 39762, USA.