Network Intrusion Detection System With Data Mining Approach

*Nayyar Ahmed Khan
**Varsha Sharma

Abstract

Network intrusion detection systems have become a standard component in security infrastructures. Unfortunately, current systems are poor at detecting novel attacks without an unacceptable level of false alarms. We propose that the solution to this problem is the application of an ensemble of data mining techniques which can be applied to network connection data in an offline environment, augmenting existing real-time sensors. In this paper, we expand on our motivation, particularly with regard to running in an offline environment, and our interest in multisensor and multimethod correlation. We then review existing systems, from commercial systems, to research based intrusion detection systems. Next we survey the state of the art in the area. Standard datasets and feature extraction turned out to be more important than we had initially anticipated, so each can be found under its own heading. Next, we review the actual data mining methods that have been proposed or implemented. We conclude by summarizing the open problems in this area and proposing a new research project to answer some of these open problems.

Keywords: NIDS, data mining, intrusion, attach, firewall, IDS, detection, network mining, intrusion mining

1. Introduction: Network Intrusion Detection Systems (NIDS) have become a standard component in security infrastructures as they allow network administrators to detect policy violations. These policy violations range the gamut from external attackers trying to gain unauthorized access (which can usually be protected against through the rest of the security infrastructure) to insiders abusing their access (which often times is not easy to protect against). Detecting such violations is a necessary step in taking corrective action, such as blocking the offender (by blocking their machine at the parameter, or freezing their account), by reporting them (to their ISP or supervisor), or taking legal action against them. Alternatively, detecting policy violations allows administrators to identify areas where their defenses need improvement, such as by identifying a previously unknown vulnerability, a system that wasn't properly patched, or a user that needs further education against social engineering attacks. The problem is that current NIDS are tuned pecifically to detect known service level network attacks. Attempts to expand beyond this limited realm typically results in an unacceptable level of false positives. At the same time, enough data exists or could be collected to allow network administrators to detect these policy violations. Unfortunately, the data is so volumous, and the analysis process so time consuming, that the administrators don't have the resources to

*Senior Lecturer, Chameli Devi Institute of Technology & Management
**Department of Computer Science, Chameli Devi Institute of Technology & Management

go through it all and find the relevant knowledge, save for the most exceptional situations, such as after the organization has taken a large loss and the analysis is done as part of a legal investigation. In other words, network administrators don't have the resources to proactively analyze the data for policy violations, especially in the presence of a high number of false positives that cause them to waste their limited resources.

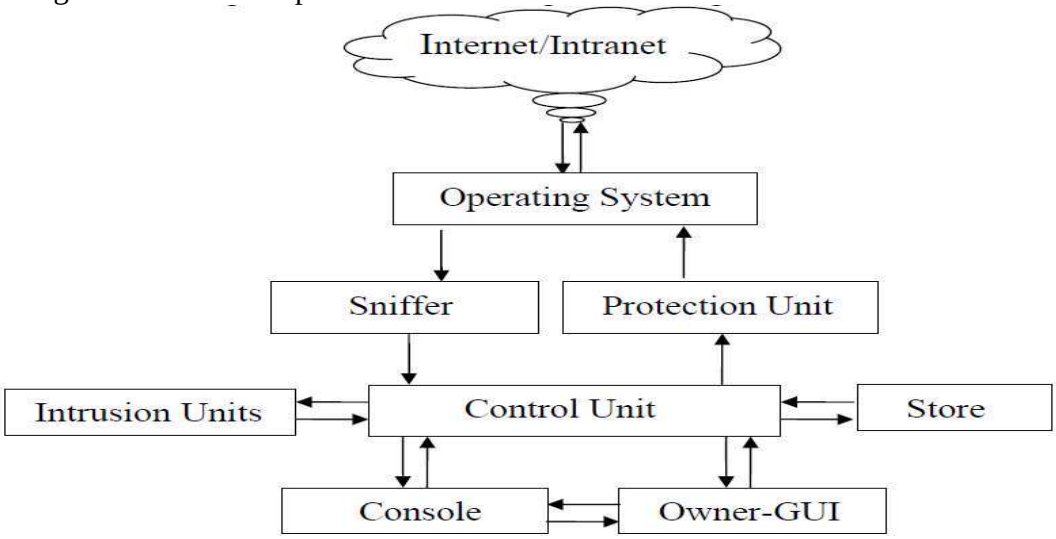
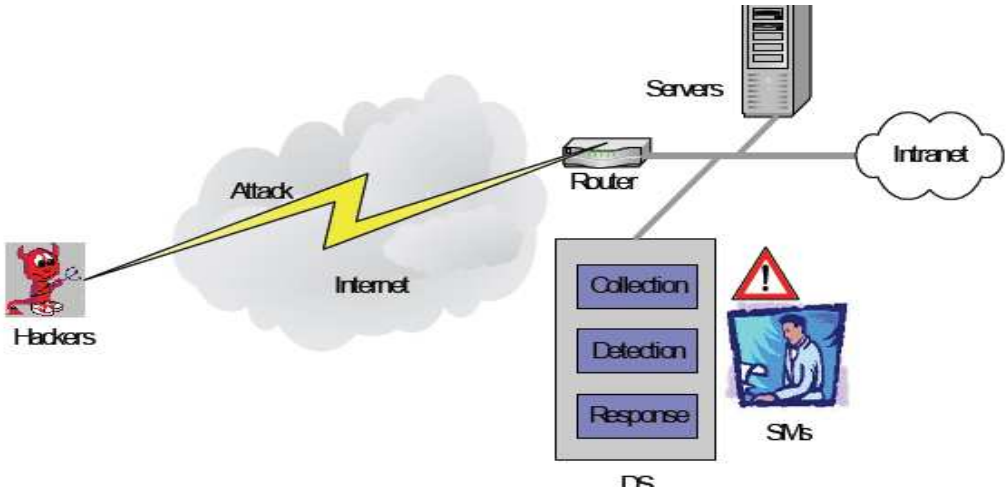


Fig.1 Network Intrusion Detection System Modal

2. Overview For Intrusion System: Given the nature of this problem, the natural solution is data mining in an offline environment. Such an approach would add additional depth to the administrators defenses, and allows them to more accurately determine what the threats against their network are though the use of multiple methods on data from multiple sources. Hence, activity that it is not efficient to detect in near real-time in an online NID, either due to the amount of state that would need to be retained, or the amount of computational resources that would need to be expended in a limited time window, can be more easily identified. [1].



Some examples of what such a system could detect, that online NIDS can not detect effectively, include certain types of malicious activity, such as low and slow scans, a slowly propagating worm, unusual activity of a user based on some new pattern of activity (rather than a single connection or small number of connections, which are bound to produce a number of false positives), or even new forms of attacks that online sensors are not tuned for. Additionally, such a system could more easily allow for the introduction of new metrics, that can use the historical data as a baseline for comparison with current activity. It also serves to aid network administrators, security officers, and analysts in the performance of their duties by allowing them to ask questions that would not have occurred to them a priori. Ideally, such a system should be able to derive a threat level for the network activity that it analyzes, and predict future attacks based on past activity. In this paper, we concentrate on the mining of network connection data as a first step.

3. Data Coagulation: Network connection data is easy to collect from most firewalls and online network intrusion sensors, or it can be constructed based on packet logs. It presents less legal hassle
than other forms of data that could be collected in many environments since it does not identify users (only machines), it does not contain details of what was done (just what service was contacted, and perhaps the duration and number of bytes transferred), and it is easily anonymous able. While we concentrate on mining connection information, the methods presented here should be applicable to other data sources, for
instance the alert logs from online NIDS, which would not be replaced, but augmented by this approach. Certainly, we expect that the incorporation of numerous forms of data will serve to increase the accuracy of such a system.[6] We begin by looking at the general motivation for doing data mining in an offline environment, with an emphasis on the advantages of an offline system versus online systems, using an ensemble of classifiers, and multisensor correlation. We'll then look at existing systems, from IDSs to services that incorporate some aspects of the methods discussed. Next, we'll focus on current research in this area, particularly datasets, feature selection, and methods, with a brief look at visualization and the potential for predictive analysis. The second part of this work presents the open problems in this area and presents a proposal for a project to answer some of those questions.

4. Existing System: ISOA conglomerated the audit information for numerous hosts whereas DIDS conglomerated the audit information from numerous host and network based IDSs. Both used a rules based expert system to perform the centralized analysis. The primary difference between the two was that ISOA was more focused on anomaly detection and DIDS on misuse detection. [2] Additional features of note were that ISOA provided a suite of statistical analysis tools that could be employed either by the expert system or a human analyst, and the DIDS expert system featured a limited learning capability. EMERALD extended some of the seminal IDS work at SRI (Denning 1987;NIDES 2002) with a hierarchical analysis system: the various levels (host, network, enterprise, etc) would each perform some level of analysis and pass any interesting results up the chain for correlation (Porras and Neumann 1996; Neumann and Porras 1999; Porras and Valdes 1998).[3] It provided a feedback system such that the higher

levels could request. more information for a given activity. The data fusion and correlation capabilities of commercial intrusion detection systems spans over a wide range. A few products are specifically designed to do centralized alarm collection and correlation. For example Real Secure SiteProtector, which claims to do "advanced data correlation and analysis" by interoperating with the other products in ISS's Real Secure line (Internet Security Systems 2003b). Some products, such as Symantec ManHunt and nSecure nPatrol, integrate the means to collect alarms and the ability to apply multiple statistical measures to the data that they collect directly into the IDS itself (Symantec 2003b; nSecure Software 2002).Most IDSs, such as the Cisco IDS, or Network Flight Recorder (NFR) provide the means to do centralized sensor configuration and alarm collection (Cisco 2003; NFR Security 2003).

5. Datamining Technique To Be Used Datasets: We'll use multiple datasets to provide a While we will adhere to current best practices, we retain the belief that these currently available data available for research purposes does not sufficiently model either real-world normal nor malicious traffic, and we encourage further work in this area. Should better datasets (or at least new datasets that correct some of the obvious flaws with existing ones) become available, we will incorporate them into this research. **Connection mining** In order to apply our data mining methods to network connection logs, we need to derive connection records from our datasets. Most intrusion detection techniques beyond basic pattern matching require sets of data to train on. When work on advanced network intrusion detection systems began in earnest in the late 1990's, researchers quickly recognized the need for standardized datasets to perform this training. [5] Such datasets allow different systems to be quantitatively compared. Further, they provide a welcome alternative to the prior method of dataset creation, which involved every researcher collecting data from a live network and using human analysts to thoroughly analyze and label the data. The most popular data format to do analysis on is the connection log. Besides being readily available and a much more reasonable size than other log formats (such as packet logs), the connection record format affords more power in the data analysis step, as it provides multiple fields that correlation can be done on (unlike a format such as command histories). Additionally, not examining data stream contents saves significant amounts of processing time and storage, and avoids privacy issues (Hofmeyr and Forrest 1999).[4] Essential Attributes to be handled and controlled in the data mining system is as:

1. 1.Timestamp
2. Source IP
3. Destination IP
4. Source port
5. Destination port
6. Protocol
7. Duration
8. Source bytes
9. Destination bytes
10. TCP Flags
11. Land packet
12. Wrong Fragment
13. Resent rate
14. Wrong resent rate
15. Duplicate ACK rate
16. Hole rate
17. Wrong data packet size rate
18. Data packets Loss

Classification techniques: A classification based IDS attempts to classify all traffic as either normal or malicious in some manner. The primary difficulty in this approach is

how accurately the system can learn what these patterns are. This ultimately affects the accuracy of the system both in terms of whether nonhostile activity is flagged (false positive) and whether malicious activity will be missed (false negative). Some classification techniques are binary (they classify data into one of two classes), while others are n-ary (they classify data into one of an arbitrary number of classes). We do not differentiate, as one can use multiple binary classifiers to emulate an n-ary classifier

Clustering techniques: Clustering is a data mining technique where data points are clustered together based on their feature values and a similarity metric. Frank (1994) breaks clustering techniques into five areas: hierarchical,statistical, exemplar, distance, and conceptual clustering, each of which has different ways of determining cluster membership and representation. Berkhin presents an excellent survey of specific methods for techniques in most of these areas in (2002) [6.] Frank (1994) notes that clustering is an effective way to find hidden patterns in data that humans might otherwise miss. Clustering is useful in an intrusion detection as malicious activity should cluster together, separate from non-malicious activity Another approach that has been successfully applied for intrusion detection is the use of graphs. This approach was pioneered by GrIDS, the Graph based Intrusion Detection System. [4] GrIDS creates graphs of network activity which reveal the causal structure of the network traffic, hence allowing coordinated attacks to be easily detected (Staniford- Chen et al. 1996). This concept was expanded on by T¨olle and Niggermann (2000) who note, “We believe that graph clustering delivers patterns that make it possible to visualize and automatically detect anomalies in the network traffic.” In their system, the traffic is used to construct a graph, where nodes representing similar traffic are clustered together, and a mapping function is used to classify the type of traffic the cluster contains, based on the properties of the cluster and the nodes in it. Interestingly, they noticed that, “the learning method relies mainly on average values of cluster properties when deciding whether an intrusion happens.” The primary disadvantage that they found in their approach was “that this method is only able to detect intrusions producing a considerable amount of network traffic.”

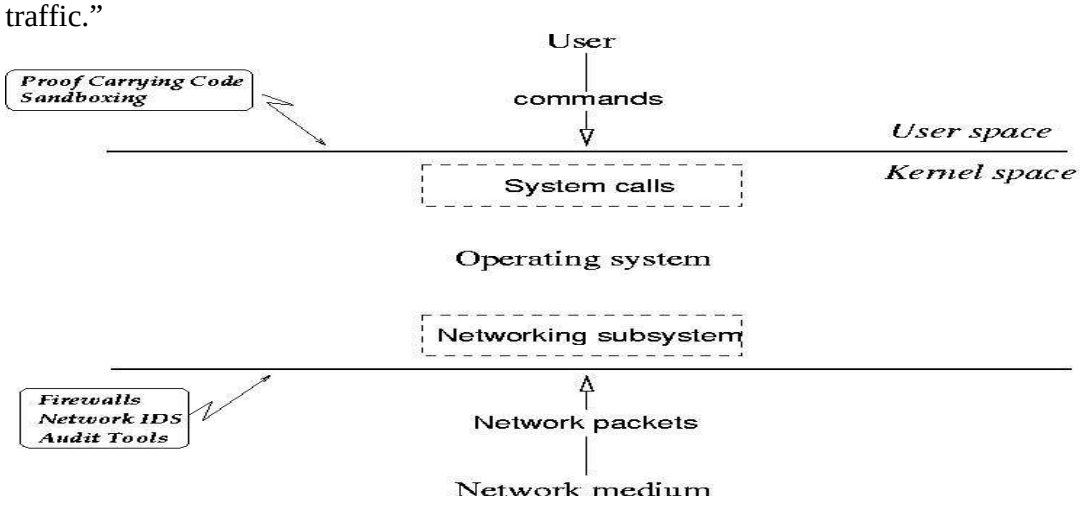


Fig 3. IDS Working at Operating System Level

The relative work to be done on the IDS system with the datamining approach as proposed is based on the number of mined user information based on the above parameters in a sense that tell that the system will decide how and when a particular mechanism will be generated depending on the system user interaction and the intruder will be decided based on the clustering of the system performance and requests generation. A part of the data mining system will make use of the system management done in an artificial intelligent manner in which the mining will yield some results and it is going to show the basic of the work done by the intruder in order to enter the network and then trying to handle the system parameters in the context of the system policies. The implementation logic is under revision and will be soon provided with a data mining approach.

6. Conclusions: In this paper we have proposed a schema based on certain factors like data sets and clustering for the Intrusion detection mechanism in various network related system. At an enterprise level the system is so well expanded and managed that the clustering of data is done on the basis of mining of information from some factors stated above like Timestamp Source IP, Destination IP, Source port, Destination port, Protocol, Duration, Source bytes, Destination bytes, TCP Flags, Land packet, Wrong Fragment , Resent rate, Wrong resent rate, Duplicate ACK rate, Hole rate, Wrong data packet size rate, Data packets Loss etc. Based on the data obtained from the above factors a data mining concept is applied on the system and the result set is obtained to detect an intruder or a malicious person in the network.

7. References

1. Warrender, C., S. Forrest, and B. A. Pearlmutter (1999). Detecting intrusionsusing system calls: Alternative data models. In Proc. of the 1999 IEEE Symp. on Security and Privacy, Oakland, CA, pp. 133–145. IEEEComputer Society Press.
2. Winkler, J. R. and W. J. Page (1990). Intrusion and anomaly detection in trusted systems. In Fifth Annual Computer Security Applications Conf.,1989, Tucson, AZ, pp. 39–45. IEEE.
3. Yeung, D.-Y. and C. Chow (2002, 11–15 August). Parzenwindow network intrusion detectors. In Proc. of the Sixteenth International Conference on Pattern Recognition, Volume 4, Quebec City, Canada, pp. 385–388. IEEE Computer Society.
4. Dickerson, J. E. and J. A. Dickerson (2000, July). Fuzzy network profiling for intrusion detection. In Proc. of NAFIPS 19th International Conference of the North American Fuzzy Information Processing Society, Atlanta, pp. 301–306. North American Fuzzy Information Processing Society(NAFIPS).
5. Lippmann, R. P., J. W. Haines, D. J. Fried, J. Korba, and K. J. Das (2000,October). The 1999 DARPA off-line intrusion detection evaluation. Computer Networks 34,579–??
6. Ning, P., X. S. Wang, and S. Jajodia (2000). Modeling requests among cooperating intrusion detection systems. Computer Communications 23 (17), 1702–1716.nSecure Software (2002). nSecure nPatrol.
7. http://www.nsecure.net/features.htm.

8. Paxson, V. (2004, 10 March). Bro: A system for detecting network intruders in real-time.
9. http://www-nrg.ee.lbl.gov/bro.html