International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 4, Issue 2, No 4, 2016. ISSN: 2348-6600
Reference ID: IJCS-133 PAGE NO: 783-788

A Survey on Network Anomaly Detection

D.**Priya**

Research Scholar , Bharathidasan University ,Trichy

Abstract— Networks of various kinds often experience anomalous behavior. Examples include attacks or large data transfers in IP networks, presence of intruders in distributed video surveillance systems, and an automobile accident or an untimely congestion in a road network. System administrators can attempt to prevent such attacks using intrusion detection tools and systems. There are many commercially available Intrusion Detection Systems (IDSs). However, most IDSs lack the capability to detect novel or previously unknown attacks. A special type of IDSs, called Anomaly Detection Systems, develop models based on normal system or network behavior, with the goal of detecting both known and unknown attacks. Anomaly detection systems face many problems including high rate of false alarm, ability to work in online mode, and scalability. This paper presents a selective survey of incremental approaches for detecting anomaly in normal system or network traffic.

***Keywords*:** Anomaly Detection, Incremental, Attack, Clustering.

I. Introduction

A network anomaly is a sudden and short-lived deviation from the normal operation of the network. Some anomalies are deliberately caused by intruders with malicious intent such as a denial-of-service attack in an IP network, while others may be purely an accident such as an overpass falling in a busy road network. Quick detection is needed to initiate a timely response, such as deploying an ambulance after a road accident, or raising an alarm if a surveillance network detects an intruder. Network monitoring devices collect data at high rates. Designing an effective anomaly detection system consequently involves extracting relevant information from a voluminous amount of noisy, high-dimensional data. It is also important to design distributed algorithms as networks operate under bandwidth and power constraints and communication costs must be minimized.

Different anomalies exhibit themselves in network statistics in different manners, so developing general models of normal network behavior and of anomalies is difficult. Model-based algorithms are also not portable across applications, and even subtle changes in the nature of network traffic or the monitored physical phenomena can render the model inappropriate.

There are two types of IDSs: signature-based and anomaly-based. Signature-based IDSs exploit signatures of known attacks. Such systems require frequent updates of signatures for known attacks and cannot detect unknown attacks or anomalies for which signatures are not stored in the database. In contrast, anomaly based IDSs are extremely effective in finding and preventing known as well as unknown or zero day attacks. However, an anomaly detection system has many shortcomings such as high rate of false alarm, and the failure to scale up to gigabit speeds. The main task of an incremental method is to construct or model the pattern classes and use them for correct classification of the input pattern as either belonging to a certain previously observed class or not belonging to any of referred classes. However,

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 4, Issue 2, No 4, 2016. ISSN: 2348-6600
Reference ID: IJCS-133 PAGE NO: 783-788

in network anomaly detection, the incremental approach updates normal profiles dynamically based on changes in network traffic without fresh training using all data for attack detection. Based on existing profiles, it can take the decision to raise an attack if abrupt changes happen in normal system or network traffic. Thus, it is useful to detect anomalies from normal system or network traffic from existing signatures or profiles, using an incremental approach. That is, an incremental approach updates profiles or signatures dynamically incorporating new profiles as it encounters them. It does not need to load the whole database each time into the memory and learn fresh from beginning. In the last two decades, a good number of anomaly based intrusion detection approaches have been developed, but a lot of them are general in nature, and thus quite simple. Due to the lack of papers that discuss various facets of incremental anomaly detection, and present a survey in a structured manner along with current research issues and challenges in this field.

2. Preface to Anomaly Detection

Anomaly detection refers to the important problem of finding non-conforming patterns or behaviors in live traffic data. These non-conforming patterns are often known as anomalies, outliers, discordant observations, exceptions, aberrations, surprises, peculiarities or contaminants in different application domains. In practice, it is very difficult to precisely detect anomalies in network traffic or normal data. So, anomaly is an interesting pattern due to the effect of traffic or normal data while noise consists of non-interesting patterns that hinder traffic data analysis. The central premise of anomaly detection is that intrusive activity is a subset of anomalous activity. When there is an intruder who has no idea of the legitimate user's activity patterns, the probability that the intruder's activity is detected as anomalous should be high. There are four possibilities in such a situation, each with a non-zero probability.

• Intrusive but not anomalous: An IDS may fail to detect this type of activity since the activity is not anomalous. But, if the IDS detect such an activity, it may report it as a false negative because it falsely reports the absence of an intrusion when there is one.

• Not intrusive but anomalous: If the activity is not intrusive, but it is anomalous, IDS may report it as intrusive. These are called false positives because an intrusion detection system falsely reports intrusions.

• Not intrusive and not anomalous: These are true negatives; the activity is not intrusive and should not be reported as intrusive.

• Intrusive and anomalous: These are true positives; the activity is intrusive and much be reported as such.

3. Existing Approaches for Network Anomaly Detection

Anomaly detection techniques can be of three types: supervised, semi-supervised and unsupervised.

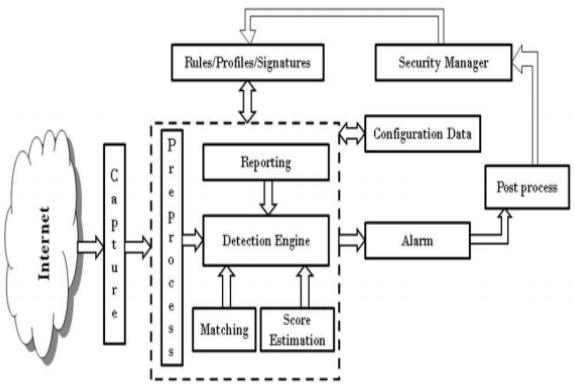

Fig 1. A generic architecture of Anomaly Detection

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 4, Issue 2, No 4, 2016. ISSN: 2348-6600
Reference ID: IJCS-133 PAGE NO: 783-788

Present a generic architecture of an incremental anomaly based network intrusion detection system in Figure 1.

3.1 Network Traffic Capture

It is useful to analyze network traffic to detect attacks or anomalies. In a capture module, real network traffic is captured by using Libpcap library, an open source C library offering an interface for capturing link-layer frames over a wide range of system architectures. It provides a high-level common Application Programming Interface to the different packet capture frameworks of various operating systems. The offered abstraction layer allows programmers to rapidly develop highly portable applications. Libpcap defines a common standard format for files in which captured frames are stored, also known as the tcpdump format, currently a de facto standard used widely in public network traffic archives. Modern kernel-level capture frameworks on UNIX operating systems are mostly based on the BSD (or Berkeley) Packet Filter (BPF).

3.2Preprocessor

In order to evaluate IDS, an unbiased intrusion dataset in a standard format is required. Generally, the live captured packet contains a lot of raw data; some of them may not be relevant in the context of IDS. Therefore, filtrations of irrelevant parameters during captures as well as extraction of relevant features from the filtered data are important preprocessing functions of IDS.

3.3 Feature Extraction

Feature extraction from raw data is an important step for anomaly based network intrusion detection. The evaluation of any intrusion detection algorithm on real time network data is difficult, mainly due to the high cost of obtaining proper labeling of network connections. The extracted features are of four types and are described below in brief.

• **Basic features:** These can be derived from packet headers without inspecting the payload. For example, the following features are basic features: protocol type, service, flag, source bytes, destination bytes;

• **Content based features:** Domain knowledge is used to assess the payload of the original TCP (Transmission Control Protocol) packets. This type includes features such as the number of failed login attempts;

• **Time-based features:** These features are designed to capture properties that mature over a 2-second temporal window. One example of such a feature is the number of connections to the same host over the 2-second interval;

• **Connection-based features:** These features are computed over a historical window estimated over the number of connections, in this case 100, instead of time. These features are designed to assess attacks, which span intervals longer than 2 seconds. It is well known that features constructed from the data content of the connections are more important when detecting R2L (Remote to Local) and U2R (User to Root) attack types in KDD99 intrusion dataset [1], while time-based and connection based features are more important for detection of DoS (Denial of Service) and probing attack types [2].

4. Supervised Approach

In this approach, a predictive model is developed based on a training dataset (i.e., data instances labeled as normal or attack class).

Yu et al. [3]: The authors propose an incremental learning method by cascading a service classifier (SC) over an incremental tree inducer (ITI) for supervised anomaly detection. A service classifier

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 4, Issue 2, No 4, 2016. ISSN: 2348-6600
Reference ID: IJCS-133 PAGE NO: 783-788

ensures that the ITI method is trained with instances with only one service value (e.g., ftp, smtp, telnet, etc.). The ITI method is trained with instances incrementally without the service attribute. The cascading method has three phases: (i) training, (ii) testing and (iii) incremental learning. During training phase, the service classifier method is first applied to partition the training dataset into m disjoint clusters according to different services. Then, the ITI method is trained with the instances in each cluster. The service classifier method ensures that each training instance is associated with only one cluster. In the testing phase, it finds the clusters to which the test instances belong. Then, the ITI method is tested with the instances. In the incremental learning phase, the service classifier and ITI cascading method are not re-trained; the authors use incremental learning to train the existing ITI binary tree. Nearest Neighbor combination rules embedded within K-Means+ITI and SOM (Self-Organizing Map)+ITI cascading methods are used in experiments. The authors also compare the performance of SC+ITI with K-Means, SOM, ITI, KMeans+ITI and SOM+ITI methods in terms of detection rate and false positive rate (FPR) on the KDD'99 dataset. Results show that the ITI method has better performance than K-Means, SOM, KMeans+ITI and SOM+ITI methods in terms of overall detection rate. However, the SC+ITI cascading method outperforms the ITI method in terms of detection rate and FPR, and obtains better detection rate compared to other methods. Like the ITI method, SC+ITI also provide additional options for handling missing values and incremental learning.

5. Semi-supervised Approach

In semi-supervised approach, the training data instances belong to the normal class only. Data instances are not labeled for the attack class. There are many approaches used to build the model for the class corresponding to normal behavior. This model is used to identify anomalies in the test data.

Burbeck et al. [4]: ADWICE (Anomaly Detection With fast Incremental Clustering) uses the first phase of the BIRCH clustering framework [5] to implement fast, scalable and adaptive anomaly detection. It extends the original clustering algorithm and applies the resulting detection mechanism for analysis of data from IP networks. The performance is demonstrated on the KDD99 intrusion dataset as well as on data from a test network at a telecom company. Their experiments show good detection quality (95%) and acceptable false positives rate (2.8 %) considering the online, real-time characteristics of the algorithm. The number of alarms is further reduced by application of the aggregation techniques implemented in the Safeguard architecture1.

6. Unsupervised Approach

Unsupervised detection approaches do not require training data, and thus are most widely applicable. These techniques make the implicit assumption that normal instances are far more frequent than anomalies in the test data. If this assumption is not true, such techniques suffer from high false alarm. Most existing unsupervised anomaly detection approaches are clustering based. Clustering is a

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 4, Issue 2, No 4, 2016. ISSN: 2348-6600
Reference ID: IJCS-133 PAGE NO: 783-788

technique to group similar objects. It deals with finding structure in a collection of unlabeled data. Representing the data by fewer clusters necessarily leads to the loss of certain finer details, but achieves simplification. In anomaly detection, clustering plays a vital role in analyzing the data by identifying various groups as either belonging to normal or to anomalous categories. There are many different clustering based anomaly detection approaches in the literature.

Hsu et al. [6]: Adaptive resonance theory network (ART) is a popular unsupervised neural network approach. Type I adaptive resonance theory network (ART1) deals with binary numerical data, whereas type II adaptive resonance theory network (ART2) deals with general numerical data. Several information systems collect mixed type attributes, which include numeric attributes and categorical attributes. However, both ART1 and ART2 do not deal with mixed type data. If the categorical attributes are transferred to binary data format, the binary data do not reflect reality with the same fidelity. It ultimately influences clustering quality. The authors present a modified adaptive resonance theory network (M-ART) and a conceptual hierarchy tree to solve the problem of clustering with mixed data. They show that the MART algorithm can process mixed type data well and has a positive effect on clustering.

7. Research Issues and Challenges

Based on the survey of published papers on incremental anomaly detectors, it is observed that most techniques have been validated using the KDD99 intrusion datasets in an offline mode. However, the effectiveness of an ANIDS based on incremental approach can only be judged in a real-time environment. Following are some of the research issues we have identified in this area.

• Most existing IDSs have been found inadequate with new networking paradigms currently used for both wired and wireless communication. Thus, adaptation to new network paradigms needs to be explored.

• Most existing methods are dependent on multiple input parameters. Improper estimation of these parameters leads to high false alarm rates.

• The clustering techniques that are used by anomaly detectors need to be faster and scalable when used on high dimensional and voluminous mixed type data.

Conclusion

In this paper, the state of modern incremental anomaly detection approaches in the context of computer network traffic has been examined. The discussion follows two well-known criteria for categorizing of anomaly detection: detection strategy and data source. Most anomaly detection approaches have been evaluated using Lincoln Lab (i.e. KDDCup99 intrusion dataset), Network Traces and LBNL datasets. Experiments have demonstrated that for different types of attacks, some anomaly detection approaches are more successful than others. Therefore, ample scope exists for working toward solutions that maintain high detection rates while lowering false alarm rates. Incremental learning approaches that combine data mining, neural network and threshold based analysis for the anomaly detection have shown great promise in this area.

References

[1] “Kdd cup 1999 data,” October 1999. [Online].Available:http://kdd.ics.uci.edu/databases/kddcup99/ kddcup99.html

[2]W. Lee and S. J. Stolfo, “Data mining approaches for intrusion detection,” in

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 4, Issue 2, No 4, 2016. ISSN: 2348-6600
Reference ID: IJCS-133 PAGE NO: 783-788

Proceedings of the 1998 USENIX Security Symposium, pp. 1-15, 1998, USENIX Association.

[3]W. Y. Yu and H.-M. Lee, "An incremental-learning method for supervised anomaly detection by cascading service classifier and ITI decision tree methods," in Proceedings of the Pacific Asia Workshop on Intelligence and Security Informatics. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 155– 160.

[4]K. Burbeck and S. Nadjm-tehrani, "ADWICE - anomaly detection with real-time incremental clustering," in In Proceedings of the 7th International Conference on Information Security and Cryptology, Seoul, Korea. Springer Verlag, pp. 4007-424, 2004.

[5]T. Zhang, R. Ramakrishnan, and M. Livny, "BIRCH: an efficient data clustering method for very large databases," SIGMOD Rec., vol. 25, no. 2, pp. 103– 114, 1996.

[6]C.C. Hsu and Y.-P. Huang, "Incremental clustering of mixed data based on distance hierarchy," Expert Syst. Appl., vol. 35, no. 3, pp. 1177–1185, 2008.