International Journal of Engineering Science Invention
ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726
www.ijesi.org Volume 3 Issue 7ǁ July 2014 ǁ PP.46-50

Intrusion Detection System Using Pipelining Approach

M. Naveen kumar[1] , M. Pardha Saradhi[2] , G. Rajeswarappa[3]

ABSTRACT: An intrusion detection system is a device or software application that monitors network and/or system activities for malicious activities or policy violations and produces reports to a Management Station. Intrusion detection faces a number of challenges; an intrusion detection system must reliably detect malicious activities in a network and must perform efficiently to cope with the large amount of network traffic. There are many existing frameworks for intrusion detection system that finds out malicious attacks efficiently. In this paper, we propose a new approach that uses *pipelining approach* for finding malicious attacks efficiently.

KEYWORDS : Intrusion detection, network security, decision trees, naive Bayes, Pipelining

I. INTRODUCTION

INTRUSION detection as defined by the Sys Admin, Audit, Networking, and Security (SANS) Institute is the art of detecting inappropriate, inaccurate, or anomalous activity [2]. Today, intrusion detection is one of the high priority and challenging tasks for network administrators and security professionals. More sophisticated security tools mean that the attackers come up with newer and more advanced penetration methods to defeat the installed security systems [1] and [7]. Thus, there is a need to safeguard the networks from known vulnerabilities and at the same time take steps to detect new and unseen, but possible, system abuses by developing more reliable and efficient intrusion detection system. Any intrusion detection system has some inherent requirements. Its prime purpose is to detect as many attacks as possible with minimum number of false alarms, i.e., the system must be accurate in detecting attacks. We desire a system that detects most of the attacks, gives very few false alarms, copes with large amount of data, and is fast enough to make real-time decisions. Intrusion detection started in around 1980s after the influential paper from Anderson [5]. Intrusion detection systems are classified as *network based, host based*, or *application based* depending on their mode of deployment and data used for analysis [6]. Additionally, intrusion detection systems can also be classified as *signature based* or *anomaly based* depending upon the attack detection method.

II. RELATED WORK

Though the both firewalls and intrusion detection systems relate to network security, an intrusion detection system (IDS) differs from a firewall in that a firewall looks outwardly for intrusions in order to stop them from happening. Firewalls limit access between networks to prevent intrusion and do not signal an attack from inside the network. An IDS evaluates a suspected intrusion once it has taken place and signals an alarm. The field of intrusion detection and network security has been around since late 1980s. Since then, a number of methods and frameworks have been proposed and many systems have been built to detect intrusions. Various techniques such as association rules, clustering, naïve bayes classifier, support vector machines, genetic algorithms, artificial neural networks, and others have been applied to detect intrusions. In this section, we briefly discuss these techniques and frameworks. Lee et al. introduced data mining approaches for detecting intrusions in [9] and [10]. Data mining approaches for intrusion detection include association rules and frequent episodes, which are based on building classifiers by discovering relevant patterns of program and user behaviour. Association rules [4] and frequent episodes are used to learn the record patterns that describe user behaviour. These methods can deal with symbolic data, and the features can be defined in the form of packet and connection details. However, mining of features is limited to entry level of the packet and requires the number of records to be large and sparsely populated; otherwise, they tend to produce a large number of rules that increase the complexity of the system [3].

Data clustering methods such as the k-means and the fuzzy c-means have also been applied extensively for Intrusion detection. One of the main drawbacks of the clustering technique is that it is based on calculating numeric distance between the observations, and hence, the observations must be numeric. Observations with symbolic features cannot be easily used for clustering, resulting in inaccuracy. In addition, the clustering methods consider the features independently and are unable to capture the relationship between different features of a single record, which further degrades attack detection accuracy. Naive Bayes classifiers have also been used for intrusion detection [9].

However, they make strict independence assumption between the features in an observation resulting in lower attack detection accuracy when the features are correlated, which is often the case for intrusion detection. Bayesian network can also be used for intrusion detection. However, they tend to be attack specific and build a decision network based on special characteristics of individual attacks. Thus, the size of a Bayesian network increases rapidly as the number of features and the type of attacks modelled by a Bayesian network increases.Decision trees have also been used for intrusion detection [9]. The decision trees select the best features for each decision node during the construction of the tree based on some well-defined criteria. One such criterion is to use the information gain ratio, which is used in C4.5. Decision trees generally have very high speed of operation and high attack detection accuracy. Debar and Zhang discuss the use of artificial neural networks for network intrusion detection. Though the neural networks can work effectively with noisy data, they require large amount of data for training and it is often hard to select the best possible architecture for a neural network. Support vector machines have also been used for detecting intrusions [8]. Support vector machines map real valued input feature vector to a higher dimensional feature space through nonlinear mapping and can provide real-time detection capability, deal with large dimensionality of data, and can be used for binary-class as well as multiclass classification.

Other approaches for detecting intrusion include the use of genetic algorithm and autonomous and probabilistic agents for intrusion detection. These methods are generally aimed at developing a distributed intrusion detection system. To overcome the weakness of a single intrusion detection system, a number of frameworks have been proposed, which describe the collaborative use of network-based and host based systems. A data mining framework is also used for building adaptive intrusion detection models. In the paper [11], the authors classified the all network attacks into 4 categories. They are Probe attacks, DoS attacks, R2L attacks, and U2R attacks. They used layered approach for detecting the attacks. Each layer is responsible for detecting a particular type of attacks. Here each layer is independent of other. The goal of using a layered model is to reduce computation and the overall time required to detect anomalous events. The time required to detect an intrusive event is significant and can be reduced by eliminating the communication overhead among different layers. This can be achieved by making the layers autonomous and self-sufficient to block an attack without the need of a central decision-maker. Every layer in the Layered Intrusion Detection System (LIDS) framework is trained separately and then deployed sequentially. They defined four layers that correspond to the four attack groups mentioned in the data set. They are Probe layer, DoS layer, R2L layer, and U2R layer. Each layer is then separately trained with a small set of relevant features. Feature selection is significant for Layered Approach and discussed in the next section.

In order to make the layers independent, some features may be present in more than one layer. In many situations, there is a trade-off between efficiency and accuracy of the system and there can be various avenues to improve system performance. Methods such as naive Bayes assume independence among the observed data. This certainly increases system efficiency, but it may severely affect the accuracy.

III. PIPELINED LAYERED APPROACH

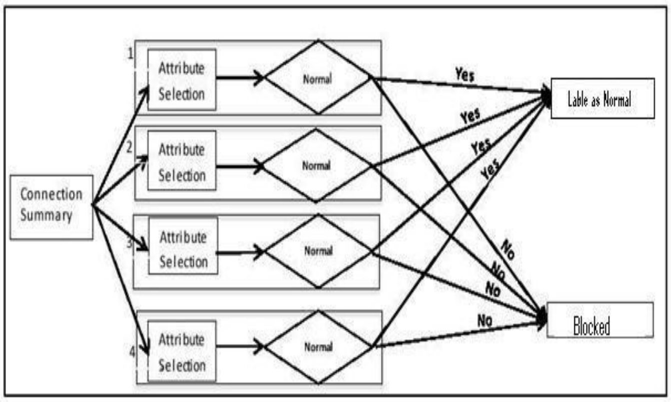
Our goal is to improve the efficiency of the intrusion detection system. This can be achieved by using pipelined layered approach. In this each layer is capable of detecting a particular type of attacks independently. Like paper[0], we also classify the all attacks into 4 types(Probe attacks, DoS attacks, R2L attacks, and U2R attacks).To detect each type of attack, a particular layer is designed and is independent of other layers. 41 features are necessary to detect all type of attacks but for detecting a particular type of attack, a small set of features are enough ,not all 41 features. Hence to improve the efficiency of the system, each layer has a particular set of features, not all 41 features.

Probe Layer : The probe attacks are aimed at acquiring information about the target network from a source that is often external to the network. Hence, basic connection level features such as the" duration of connection" and "source bytes" are significant while features like "number of files creations" and "number of files accessed" are not expected to provide information for detecting probes.

DoS Layer : The DoS attacks are meant to force the target to stop the service(s) that is (are) provided by flooding it with illegitimate requests. Hence, for the DoS layer, traffic features such as the "percentage of connections having same destination host and same service" and packet level features such as the "source bytes" and "percentage of packets with errors" are significant. To detect DoS attacks, it may not be important to know whether a user is "logged in or not."

R2L Layer : The R2L attacks are one of the most difficult to detect as they involve the network level and the host level features. We therefore selected both the network level features such as the "duration of connection" and "service requested "and the host level features such as the "number of failed login attempts" among others for detecting R2L attacks.

U2R Layer : The U2R attacks involve the semantic details that are very difficult to capture at an early stage. Such attacks are often content based and target an application. Hence, for U2Rattacks, we selected features such as "number of file creations" and "number of shell prompts invoked," while we ignored features such as "protocol" and "source bytes."



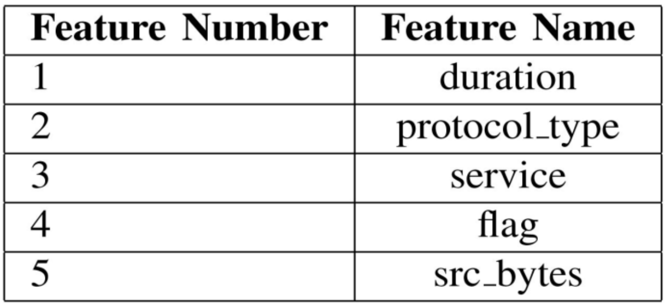
Pipelined approach for intrusion detection

In the above figure 1= probe layer, 2= dos layer, 3= R2L layer 4= U2R layer.

Each layer is separately trained with a small set of relevant features. Feature selection is significant for Layered Approach. In order to make the layers independent, some features may be present in more than one layer. The layers essentially act as filters that block any anomalous connection.

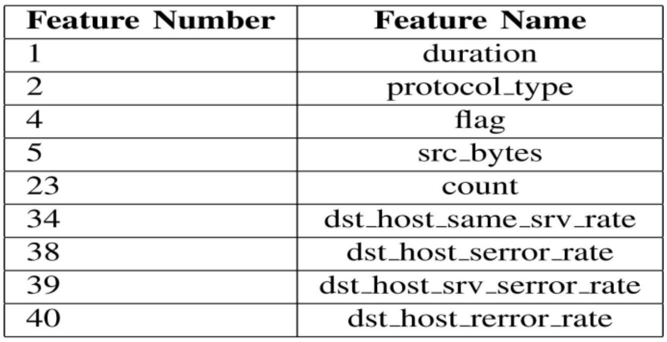
IV. FEATURE SELECTIONS

Features Selected for Probe Layer



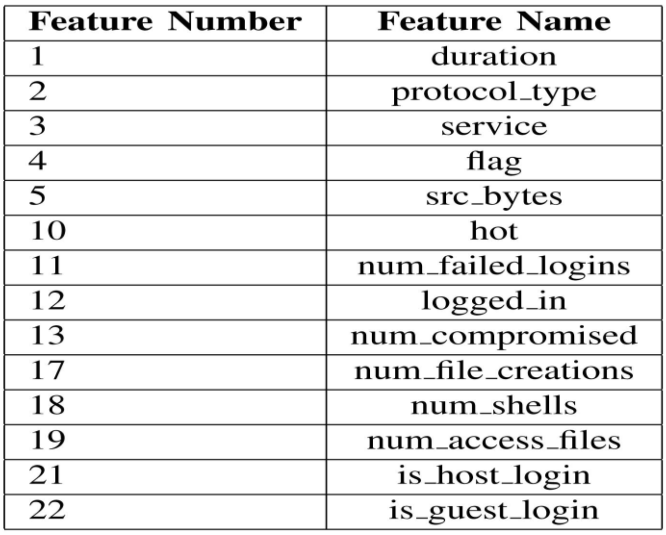
Feature Number	Feature Name
1	duration
2	protocol_type
3	service
4	flag
5	src_bytes

Features Selected for DoS Layer



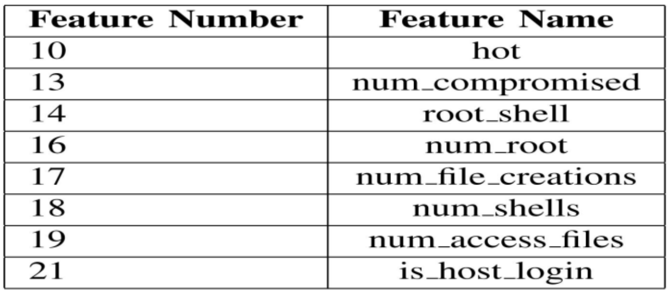
Feature Number	Feature Name
1	duration
2	protocol_type
4	flag
5	src_bytes
23	count
34	dst_host_same_srv_rate
38	dst_host_serror_rate
39	dst_host_srv_serror_rate
40	dst_host_rerror_rate

Features Selected for R2L Layer



Feature Number	Feature Name
1	duration
2	protocol_type
3	service
4	flag
5	src_bytes
10	hot
11	num_failed_logins
12	logged_in
13	num_compromised
17	num_file_creations
18	num_shells
19	num_access_files
21	is_host_login
22	is_guest_login

Features Selected for U2R Layer



Feature Number	Feature Name
10	hot
13	num_compromised
14	root_shell
16	num_root
17	num_file_creations
18	num_shells
19	num_access_files
21	is_host_login

V. CONCLUSIONS

The areas for future research include the use of our method for extracting features that can aid in the development of signatures for signature-based systems. The signature-based systems can be deployed at the periphery of a network to filter out attacks that are frequent and previously known, leaving the detection of new unknown attacks for anomaly and hybrid systems. The future research also includes that when noise occurs, then how is the system efficiency.

REFERENCES

[1] Overview of Attack Trends, http://www.cert.org/archive/pdf/attack_trends.pdf, 2002.

[2] SANS Institute—Intrusion Detection FAQ, ttp://www.sans.org/resources/idfaq/, 2010.

[3] T. Abraham, IDDM: Intrusion Detection Using Data MiningTechniques, http://www.dsto.defence./gov.au/publications/2345/DSTO-GD-0286.pdf, 2008.

[4] R. Agrawal, T. Imielinski, and A. Swami, “Mining Association Rules between Sets of Items in Large Databases,” Proc. ACMSIGMOD, vol. 22, no. 2, pp. 207-216, 1993.

[5] J.P. Anderson, Computer Security Threat Monitoring and surveillance, http://csrc.nist.gov/publications/history/ande80.pdf, 2010.

[6] R. Bace and P. Mell, Intrusion Detection Systems, Computer Security Division, Information Technology Laboratory, Nat’l Inst.of Standards and Technology, 2001.

[7] K.K. Gupta, B. Nath, R. Kotagiri, and A. Kazi, “Attacking Confidentiality: An Agent Based Approach,” Proc. IEEE Int’l Conf. Intelligence and Security Informatics (ISI ’06), vol. 3975, pp. 285- 296, 2006.

[8] D.S. Kim and J.S. Park, “Network-Based Intrusion Detection with Support Vector Machines,” Proc. Information Networking, Networking Technologies for Enhanced Internet Services Int’l Conf. (ICOIN ’03),pp. 747-756, 2003.

[9] W. Lee and S. Stolfo, “Data Mining Approaches for Intrusion Detection,” Proc. Seventh USENIX Security Symp. (Security’98),pp. 79-94, 1998.

[10] W. Lee, S. Stolfo, and K. Mok, “Mining Audit Data to Build Intrusion Detection Models,” Proc. Fourth Int’l Conf. Knowledge Discovery and Data Mining (KDD ’98), pp. 66-72, 1998.

[11] Layered Approach using CRF for Intrusion detection, IEEE Transactions on dependable and secure computing, vol. 7, no. 1, January-march 2010, Kapil Kumar Gupta, Baikunth Nath, Senior Member, IEEE, and Ramamohanarao Kotagiri, Member, IEEE.

M.Naveen kumar received M.Tech degree from Department of Computer Science, JNTU Anantapur University. Currently, he is working as an Assistant professor in SV Engineering College for women. His research interests are in computer networks, Network Security and theory of computation. (E-mail: nkumarmsc2009@gmail.com)

M. Pardha Saradhi received M.Tech degree from Department of Computer Science and Engineering, JNTU Anantapur University. Currently, he is working as an Assistant professor in SV Engineering College for women. His research interests are in Network Security, Data Mining and software engineering. (E-mail: pardhu47@gmail.com)

G. Rajeswarappa received M.Tech degree from Department of Computer Science, JNTU Anantapur University. Currently, he is working as an Assistant professor in SV Engineering College for women. His research interests are in Network Security, Data Mining and Operating Systems. (E-mail: rajeswarappag@gmail.com).