ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 3, Issue 2, August 2013

Real Time Peer-Peer Intrusion Detection Using Rough Set Dynamic Reduct Approach

R. Ravinder Reddy, Dr. Y.Ramadevi, Dr. K V N Sunitha, P Karteek

Abstract: ***There's a humongous rise in the usage of internet over the past decade with the introduction of new services and constantly progressing technology. Some of these services imbibe in them sensitive information thereby resulting in high exposure for compromising the data. There comes a deep necessity to inculcate the features-confidentiality, integrity and accuracy and offer reliable services. In this paper, we proposed a peer-peer real-time intrusion detection approach using a supervised machine learning technique to classify online network data as normal or anomalous. Our approach is elegant and efficient. We used some essential features which offer the potential to identify the attack using the Rough set theory for finding the reducts of the dataset for improve the performance of the system using the relevance of the feature. Our RT-IDS system identifies Dos and Probe attacks with an accuracy capping at 99.7%.***

Index terms—**Intrusion detection, Data mining, Rough sets.**

I. INTRUSION DETECTION

Network intrusions protects a computer network from unauthorized users, including perhaps insiders Intrusion detection is the process of identifying and responding to wary activities aimed at computing and communication resources, and it has become the mainstream of information assertion as the progressive increase in the number of attacks. Intrusion detection system (IDS) monitors and collects data from a target system that should be secured, processes and jibes the gathered information, and initiates responses when evidence of an intrusion is found. Internet services have become essential to business commerce as well as to individuals. With the increasing dependency on network services, the availability, confidentiality, and integrity of critical information have become increasingly compromised by remote intrusions. Enterprises are impelled to beef up their networks against malicious activities and network threats. Therefore, a network system must use one or more security tools such as a firewall, antivirus software or an intrusion detection system to protect critical data/services from hackers or intruders. The network IDS is contains so many features from these the KDD [22] dataset contains 42 features; it is maintained and stored for intrusive purpose. The KDD99 dataset is a statistically pre-processed dataset which has been available from DARPA since 1999.

A. On-line (real-time) network intrusion detection

A system that can detect network intrusion while an attack is occurring is called a real-time detection system. A real-time IDS captures the present network traffic data which is on-line data. In this paper, the terms ‘‘on-line detection’’ and ‘‘real-time detection’’ are used interchangeably. Puttini [11] used a Bayesian classification model for anomaly detection to classify normal network activity and attack using a 3-month training dataset and a 1-month test dataset. They evaluated their approach by adjusting a penalty value to see how it affected the classification results. They also needed human expert to visualize the normal and abnormal network behaviours. No detection rate was reported. Su [13] created a real-time network IDS using fuzzy association rules and conducted their experiments by using four computers with 30 DoS attack types in WIN32. They could separate the normal network activity from network attacks but they did not identify the attack type. They pre-processed the network data to have 16 features. After testing, the results showed that the 30 DoS attack types have a similarity ratio of less than 0.4 while normal network activity gave a similarity ratio more than 0.75. The similarity ratio represents how close or similar the data is to normal data, i.e. 1.0 means that they are perfectly matched.

B. TYPES OF IDS

There are two types of Intrusion Detection Systems based on the operation where it is monitored they are Network Intrusion Detection Systems and Host Based Intrusion Detection Systems.

1 Network intrusion detection system (NIDS)

A network-based IDPS (NIDPS)[7] resides on a computer or appliance connected to a segment of an organization’s network and monitors network traffic on that network segment[2], looking for indications of ongoing or successful attacks. When the NIDPS identifies activity that it is programmed to recognize as an attack, it responds by sending notifications to administrators. When examining incoming packets, an NIDPS looks for patterns within network traffic such as large collections of related items of a certain type—which could indicate that a denial-of-service attack is under-way—or the exchange of a series of related packets in a certain pattern—which could indicate that a port scan is in progress. An NIDPS can detect many more types of attacks than a host based IDPS, but it requires a much more complex configuration and maintenance program. A NIDPS is installed at a specific place in the network (such as on the inside of an edge router) from where it is possible to monitor the traffic going into and out of a particular network segment. The

NIDPS can be deployed to monitor a specific grouping of host computers on a specific network segment, or it may be installed to monitor all traffic between the systems that make up an entire network. When placed next to a hub, switch, or other key networking device, the NIDPS may use that device's monitoring port. The monitoring port also known as a switched port analysis (SPAN) port or mirror port, is a specially configured connection on a network device that is capable of viewing all of the traffic that moves through the entire device.

2 Host-based intrusion detection system (HIDS)

A host-based IDPS (HIDPS) resides on a particular computer or server, known as the host, and monitors activity only on that system. HIDPSs are also known as system integrity verifiers11 because they benchmark and monitor the status of key system files and detect when an intruder creates, modifies, or deletes monitored files. An HIDPS has an advantage over an NIDPS in that it can access encrypted information traveling over the network and use it to make decisions about potential or actual attacks. Also, since the HIDPS works on only one computer system, all the traffic it examines traverses that system. The packet delivery mode, whether switched or in a shared-collision domain, is not a factor.

II. BASIC CONCEPTS OF ROUGH SET

Rough set theory (RST)[4] is a useful mathematical tool to deal with imprecise and insufficient knowledge, find hidden patterns in data, and reduce dataset size (Pawlak, 1982; Komorowski, *et al*, 1998). Also, it is used for evaluation of significance of data and easy interpretation of results. RST contributes immensely to the concept of reducts. Reducts is the minimal subsets of attributes with the most predictive outcome. Rough Set is a machine learning method which generates rules based on examples contained within an information table. Rough set theory has become well established as a mechanism for solving the problem of how to understand and manipulate imprecise and insufficient knowledge in a wide variety of applications related to artificial intelligence.

Let K = (U,C) be an approximation space, where U is a non-empty, finite set called the universe; A subset of attributes $R \subseteq C$ defines an equivalent on U. Let $[x]_R$ ($x \in U$) denote the equivalence class containing x.

Given $R \subseteq C$ *and* $X \subseteq U$. X can be approximated using only the information contained within R by constructing the R-lower and R-upper approximations of set X defined as:

$\underline{R}X = \{ x \in X \| [x]_R \subseteq X \}$ _

$RX = \{ x \in X \| [x]_R \cap X \neq 0 \}$where

$\underline{R}X$ is the set of objects that belong to X with certainty, belong to X. The R-positive region of X is $POS_R(X) = \underline{R}X$

A. Dynamic Reducts

Knowledge reduct is an important step in knowledge discovery, and also a favourable method to extract the more generalized rules. Dynamic reducts can put up better performance in very large dataset, and also enhances effectively the ability to accommodate noise data. Dynamic reducts[5] are in some sense the most stable reducts of a given decision table, i.e. they are the most frequently appearing reducts in sub tables created by random samples of a given decision table. The set of decision rules can be computed from dynamic reducts in two different ways. One can choose the best dynamic reducts and compute the decision rules using only attributes from the dynamic core i.e. from the union of these dynamic reducts. Another possibility is to compute the set of rules separately for any of the chosen dynamic reducts and to create the union of the constructed decision rule sets

III. COMMON DETECTION METHODOLOGIES

IDS technologies use many methodologies to detect incidents. We discuss the primary classes of detection methodologies: signature-based, anomaly-based, and stateful protocol analysis, respectively. Most IDS technologies use multiple detection methodologies, either separately or integrated, to provide more broad and accurate detection.

A. Signature-Based Detection

A signature is a pattern that corresponds to a known threat. Signature-based detection is the process of comparing signatures against observed events to identify possible incidents. Examples of signatures are as follows A telnet attempt with a user name of "root", which is a violation of an organization's security policy. An e-mail with a subject of "Free pictures!" and an attachment file name of freepics.exe, which are characteristics of a known form of malware. An operating system log entry with a status code value of 645, which indicates that the host's auditing has been disabled.

B. Anomaly-Based Detection

Anomaly-based [8] detection is the process of comparing definitions of what activity is considered normal against observed events to identify significant deviations. An IDS using anomaly-based detection has profiles that represent the normal behaviour of such things as users, hosts, network connections, or applications. The profiles are developed by monitoring the characteristics of typical activity over a period of time. For example, a profile for a network might show that Web activity comprises an average of 13% of network bandwidth at the Internet border during typical workday hours. The IDS then uses statistical methods to compare the characteristics of current activity to thresholds related to the profile, such as detecting when Web activity comprises significantly more bandwidth than expected and alerting an administrator of the anomaly. Profiles can be developed for many behavioural attributes, such as the number of e-mails sent by a user, the number of failed login attempts for a host, and the level of processor usage for a host in a given period of time

IV. PROBLEM SPECIFICATION

An attack is an act that takes advantage of a vulnerability to compromise a controlled system. It is accomplished by a threat agent that damages or steals an organization's information or physical asset. Threats are always present, attacks only exists when a specific act may cause a loss. For example, the threat of damage from a thunderstorm is present throughout the summer in many places and its associated risk of loss only exists for the duration of an actual thunder storm. In the real world peer-peer system environment, there's a constant flow of data between two hosts. We need to monitor the traffic constantly to identify if there's any kind of intrusion. Attacker performs eavesdropping to view the information in the data that is flowing between these systems. When the infrastructure is set up we need to set up the environment to identify the attacks that are possible. We need to construct a data set which must consist of reliable data, datum of probe & Dos attack. Once we have this training set constructed we now put the system under functioning. We now analyse the traffic with the training set constructed to identify if there are anomalies or if the data is valid.

A. COMMON TYPES OF ATTACKS

In the network environment mainly the Attacks fall into four main categories. To find the IDS basically most of the threat will be in these formats only.

1 Denial of Service Attack (DoS)

A denial-of-service attack (DoS attack) or distributed denial-of-service attack (DDoS attack) is an attempt to make a machine or network resource unavailable to its intended users. Although the means to carry out, motives for, and targets of a DoS attack may vary, it generally consists of the efforts of one or more people to temporarily or indefinitely interrupt or suspend services of a host connected to the internet. Perpetrators of DoS attacks typically target sites or services hosted on high-profile web servers such as banks, credit card payment gateways, and even root name servers. The term is generally used relating to computer networks, but is not limited to this field; for example, it is also used in reference to CPU resource management. One common method of attack involves saturating the target machine with external communications requests, so much so that it cannot respond to legitimate traffic, or responds so slowly as to be rendered essentially unavailable. Such attacks usually lead to a server overload In general terms, DoS attacks are implemented by either forcing the targeted computer(s) to reset, or consuming its resources so that it can no longer provide its intended service or obstructing the communication media between the intended users and the victim so that they can no longer communicate adequately

2 PROBE ATTACK

If an attacker launches an attack on a given site, the attacker typically probes the victim's network or host by searching these networks and hosts for open ports. This is done using a sweeping process across the different hosts on a network and within a single host for services that are up by probing the open ports. This is referred to as Probe attacks.

3 REMOTE TO LOCAL ATTACK

Unauthorized access from a remote machine, e.g. guessing password a common way to classify attacks is whether they are done remotely by a hacker from across the Internet, or whether they are done locally by a user who already has privileges on the system. The important difference is that a "remote" attack can be launched by any of the hundreds of the of millions of people on the Internet at any time without first logging on. Point: A hacker may need to use a combination of remote and local exploits in order to gain control over a system. More and more services are running within sandboxes in order to limit the "spread of the infection". A local exploit may be needed in order to break out of the sandbox. Key point: The most common remote exploits are buffer overflow and other unchecked input attacks. They are either done against public services (such as HTTP and FTP) or during the logon of protected services (such as POP and IMAP). From Hacking-Lexicon

4 USER TO REMOTE ATTACK

Unauthorized access to local super user (root) privileges, e.g., various ``buffer overflow" attacks. These are critical for processing speed and performance of the system in calculating IDS.

V. IMPLEMENTATION

The implementation of the system is in different phases for avoiding the complexity of the system. in the pre processing stage we collect the packets and finding the features for the IDS in the faster manner we used rough set approach for calculating the reducts from the data set.

A. Pre-processing phase

In the pre-processing phase we use a packet sniffer, which is built with WinPcap [19] library, to extract network packet information including IP header, TCP header, UDP header, and ICMP header from each packet. After that, the packet information is partitioned by considering connections between every pair of IP addresses (source IP and destination IP) and formed into a record by aggregating information every 2s. Each record consists of data features considered as the key signature features representing the main characteristics of network data and activities. We performed extensive experiments to find key features that define the signatures of normal vs. attack network traffic. We used information gain to select 12 essential features for our IDS approach. The information gain value of each feature represents the relevance of the feature to the output class. Parameters of information gain are X and Y, where X defines individual features such as number of TCP packets, number of TCP source ports, and Y defines class groups which are Normal data, Probe attack and DoS attack. The results from information gain indicate that we have to consider all 12 features of the network data for the intrusion detection and classification.

B. PROBLEM SPECIFICATION

Here we try to implement the IDS in between the peer to peer systems to achieve the more reliability and accuracy of the system we observer the flow in between the two peers and prepared the log file by using network tool to capture the packets and extracting the desirable features from the log file. Finding the intrusive behaviour as important as in the given time is very important. The user behaviour is mainly dependent on some of the features in the given dataset for finding the minimal features we used the rough set based approach.

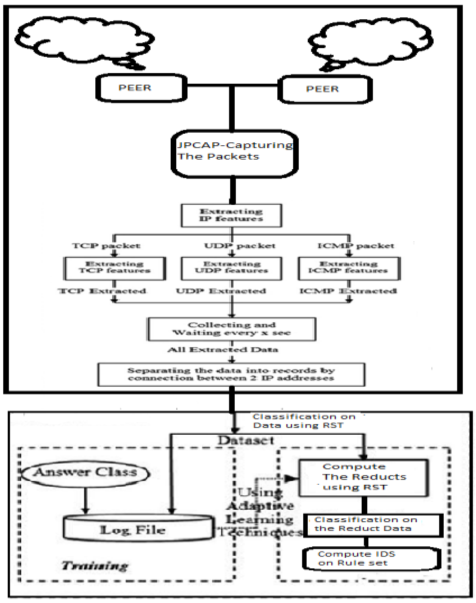

PEER
PEER
JPCAP-Capturing The Packets
Extracting IP features
TCP packet
UDP packet
ICMP packet
Extracting TCP features
Extracting UDP features
Extracting ICMP features
TCP Extracted
UDP Extracted
ICMP Extracted
Collecting and Waiting every x sec
All Extracted Data
Separating the data into records by connection between 2 IP addresses
Classification on Data using RST
Dataset
Answer Class
Log File
Training
Using Adaptive Learning Techniques
Compute The Reducts using RST
Classification on the Reduct Data
Compute IDS on Rule set

Fig 1 shows the control flow of the RT-IDS

C. ROUGH SET DYNAMIC REDUCTS

Rough sets are used to find the reducts, based on these reducts for finding faster IDS behaviour. To minimize the data set here we used the rough set approach in this we used the genetic algorithm for finding the reducts and remove the reducts from the dataset and generate the rules for classifying the IDS. Using these rules we calculated the IDS in faster manner for improving the reliability and accuracy of the real time IDS.

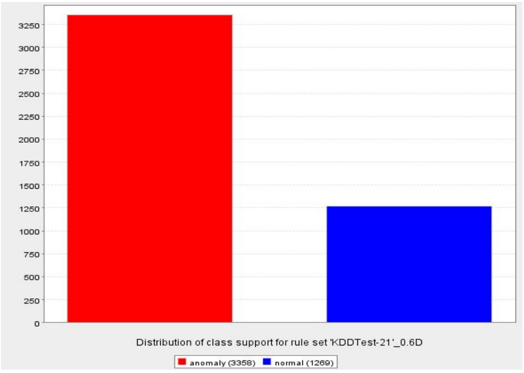
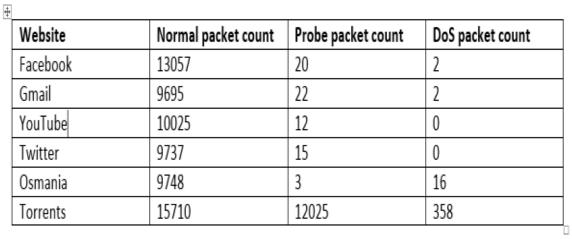


FIg:5.2 THe rule set for the KDD dataset

We have observed in between the peer to peer systems results for the different web sites and try to find out the attacks and evaluated the results shown in the below table. We have cleanly find out the mainly DOS and probe attacks.



Website	Normal packet count	Probe packet count	DoS packet count
Facebook	13057	20	2
Gmail	9695	22	2
YouTube	10025	12	0
Twitter	9737	15	0
Osmania	9748	3	16
Torrents	15710	12025	358

VI. CONCLUSION AND FUTURE ENHANCEMENTS

Conclusion: - The Intrusion Detection System developed by us is used for the peer-peer system environment. Here the monitoring is done between two systems which maintain constant IP Address. This system can be deployed into banking systems where there's a constant flow of data between two systems. We apply rules to the incoming traffic to check whether they are reliable or anomaly packets. So when an intruder attempts to break into the network we can trigger an alarm. This will help the network administrator to identify the problem and check if there's a loss of information or if the data if modified.

Future Enhancements

The entire system must be automated. This must be done to construct the rules automatically, auto merging of flooded packets with the reliable packets to form the training dataset. Auto conversion of the file obtained after matching patterns to comma separated value (.csv). Setting a timer to evaluate the time consumed for processing. After the post processing phase a push notification to system administrator must be delivered. The notification message must consist of details like the number of packets captured, amount of time the system is up monitoring, alert level messages for the anomaly packets. Auto temporary system shut-down when the intrusion level is above a threshold limit.

REFERENCES

[1] C. Jirapummin, N. Wattanapongsakorn, J. Kanthamanon, Hybrid neural networks for intrusion detection system, in: The International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC), Thailand, 2002, pp. 928-931.

[2] Wenke Lee , Salvatore J. Stolfo , Kui W. Mok, Mining in a data-flow environment: experience in network intrusion detection, Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining, p.114-124, August 15-18, 1999, San Diego, California, United States [doi>10.1145/312129.312212].

[3] Z. Pan, S. Chen, G. Hu, D. Zhang, Hybrid neural network and C4.5 for misuse detection, in: The 2nd International Conference on Machine Learning and Cybernetics, China, 2003, pp.2463-2467.

[4] Pawalak Z, “Rough sets[J],” International Journal of Computer and Information Sciences,vol.11,no. 5,PP.341-356,1982.

[5] Bazan J, Skowron A,Synak P, “Dynamic Reducts as a Tool for Extracting Laws from Decision Tables in: Methodologies for Intelligent System,”Proc. 8^{Th} International Symposium ISMIS’94, Charlotte, NG, October 1994, LNAI vol. 869, Springer Verlag 1994,346-355.

[6] N. Ngamwitthayanon, N. Wattanapongsakorn, C. Charnsripinyo, D.W. Coit, Multi-stage network-based intrusion detection system using back propagation neural networks, in: Asian International Workshop on Advanced Reliability Modeling (AIWARM), Taiwan, 2008, pp. 609-619.

[7] Abraham, R. Jain, Soft computing models for network intrusion detection systems, in: Knowledge Discovery, Computational Intelligence, vol. 4, Heidelberg, 2005, pp. 191-207.

[8] Chi-Ho Tsang , Sam Kwong , Hanli Wang, Genetic-fuzzy rule mining approach and evaluation of feature selection techniques for anomaly intrusion detection, Pattern Recognition, v.40 n.9, p.2373-2391, September, 2007 [doi>10.1016/j.patcog.2006.12.009]

[9] Chia-Mei Chen , Ya-Lin Chen , Hsiao-Chung Lin, An efficient network intrusion detection, Computer Communications, v.33 n.4, p.477-484, March, 2010 [doi>10.1016/j.comcom.2009.10.010]

[10] Labib, K. and Vemuri, R., NSOM: a real-time network-based intrusion detection system using self-organizing maps. Networks and Security

[11] R.S. Puttini, Z. Marrakchi, L. Me, A Bayesian classification model for real-time intrusion detection, in: The 22nd International Workshop on Bayesian Inference and Maximum Entropy Methods in Science and Engineering. AIP Conference Proceedings, vol. 659, 2003, pp. 150-162.

[12] Amini, M., Jalili, A. and Reza Shahriari, H., RT-UNNID: a practical solution to real-time network-based intrusion detection using unsupervised neural networks. Computer & Security. v25. 459-468.

[13] Su, M-Y., Yu, G-J. and Lin, C-Y., A real-time network intrusion detection system for large-scale attacks based on an incremental mining approach. Computers and Security. v28. 301-309.

[14] Zhichun Li , Yan Gao , Yan Chen, HiFIND: A high-speed flow-level intrusion detection approach with DoS resiliency, Computer Networks: The International Journal of Computer and Telecommunications Networking, v.54 n.8, p.1282-1299, June, 201010.1016/j.comnet.2009.10.016]

[15] S. Chakrabarti , M. Chakraborty , I. Mukhopadhyay, Study of snort-based IDS, Proceedings of the International Conference and Workshop on Emerging Trends in Technology, February 26-27, 2010, Mumbai, Maharashtra, India [doi>10.1145/1741906.1741914]

[16] J.H. Lee, J.H. Lee, S.G. Sohn, J.H. Ryu, T.H. Chung, Effective value of decision tree with KDD 99 intrusion detection datasets for intrusion detection system, in: International Conference on Advanced Communication Technology (ICACT), Korea, 2008.

[17] Pang-Ning Tan , Michael Steinbach , Vipin Kumar, Introduction to Data Mining, (First Edition), Addison-Wesley Longman Publishing Co., Inc., Boston, MA, 2005

[18] P. Sangkatsanee, N. Wattanapongsakorn, C. Charnsripinyo, Network intrusion detection with artificial neural network, decision tree and rule based approaches, in: The International Joint Conference on Computer Science and Software Engineering, Thailand, 2009.

[19] Jpcap library {Online}. .

[20] Weka 3.6.0 tools {Online}.

[21] http:// nilslab.no

[22] http://kdd.ics.uci.edu/