International Journal of Computer Applications Technology and Research
Volume 3– Issue 6, 364 - 369, 2014, ISSN: 2319–8656

Layered Approach for Preprocessing of Data in Intrusion Prevention Systems

Kamini Nalavade
Department of Computer Engineering,
VJTI, Matunga, Mumbai,
India

Dr. B. B. Meshram
Department ofComputer Engineering
VJTI, Matunga, Mumbai,
India

Abstract: Due to extensive growth of the Internet and increasing availability of tools and methods for intruding and attacking networks, intrusion detection has become a critical component of network security parameters. TCP/IP protocol suite is the defacto standard for communication on the Internet. The underlying vulnerabilities in the protocols is the root cause of intrusions. Therefor Intrusion detection system becomes an important element in network security that controls real time data and leads to huge dimensional problem. Processing large number of packets and data in real time is very difficult and costly. Therefor data pre-processing is necessary to remove redundant and unwanted information from packets and clean network data. Here, we are focusing on two important aspects of intrusion detection; one is accuracy and other is performance. The layered approach of TCP/IP model can be applied to packet pre-processing to achieve early and faster intrusion detection. Motivation for the paper comes from the large impact data preprocessing has on the accuracy and capability of anomaly-based NIPS. In this paper it is demonstrated that high attack detection accuracy can be achieved by using layered approach for data preprocessing in Internet. To reduce false positive rate and to increase efficiency of detection, the paper proposed framework for preprocessing in intrusion prevention system. We experimented with real time network traffic as well as he KDDcup99 dataset for our research.

Keywords: Intrusion, Security, Network, Layered approach

1. INTRODUCTION

The continuous improvements in technology have made the use of computers easy for gathering and sharing information using the Internet. The Transmission Control Protocol and Internet protocol suite (TCP/IP) is the de-facto standard for using the internet. Due to a number of reported attacks on networks originating from the Internet, security has become a primary concern for organizations connecting to the Internet. The Information ow on Internet is constantly under various attacks because of vulnerabilities lying in the structure of networks. Therefore it is essential to provide security to the information in transit. The secure connection itself must be established and maintained securely. The Transmission Control Protocol and Internet protocol (TCP/IP), which is the protocol suite that Internet was first developed in 1979. The primary focus was to ensure reliable communications between groups of networks connected by computers. At that time, security was not a primary concern as the users of the Internet were less. The information flow on Internet is constantly under various attacks. The root cause of these exploits is weaknesses in the protocols of underlying TCP/IP protocol suite.

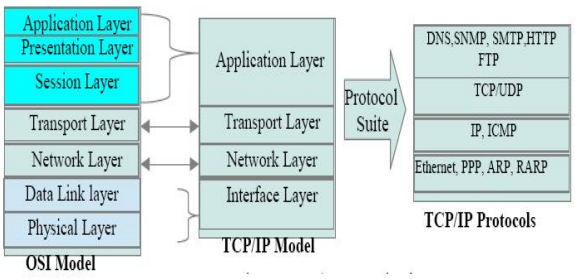

Figure 1 TCP/IP model

The TCP/IP protocol suite suffers from a number of vulnerabilities and security flaws inherent in the protocols. Those vulnerabilities are often exploited by attackers for session hijacking, sniffing, spoofing, Denial of Service (DOS) attacks and other attacks. The key vulnerability in most of the protocols of TCP/IP is lack of authentication mechanisms. This is the severe flaw which enables attacker to access the confidential information. The IP layer believes that the source address on any IP packet it receives is the same IP address as the system that actually sent the packet. The other vulnerability is connectionless communication between peers. IP layer does not ensure that a packet will reach its final destination. Also it does not guarantee that packets forwarded on network will arrive in the order. The following are the major TCP security problems. A malicious host can exhaust the server's buffer by sending several SYN requests to a host, but never replying to the SYN & ACK the other host sends back. By doing so server will stop accepting new connections, until a partially opened connection in its queue is completed or times out. This ability to effectively remove a server from the network can be used as a denial-of-service attack. It can be used to implement other attacks, like IP Spoofing, reconnaissance.

RIP, OSPF and BGP are the widely used de facto standard of routing protocols on the Internet. These protocols suffer from major vulnerabilities which causes attacks on network such as denial of service, invalid route information. Routing attacks takes advantage of Routing Information Protocol (RIP), which is an essential component in a TCP/IP network. RIP is used to distribute routing information within networks and advertising routes out from the local network. RIP has no inbuilt authentication, and the information provided in a RIP packet is often used without verifying it. RIP's update messages are sent over UDP and can be modified by attackers. Attacks on RIP change the destination where data goes to, not where it came from. For example, an invader could forge a RIP packet, claiming his host "B" has the fastest path out of the network. All packets sent out from that network would then be routed

through B, where they could be modified or scanned. An invader could also use RIP to effectively impersonate any host, by causing all traffic sent to that host to be sent to the attacker's machine instead. RIP, OSPF and BGP were studied with respect to their architecture, functionality and message types. OSPF suffers from implementation and configuration problems. BGP have vulnerabilities related confidentiality, integrity and authentication. This study provides immense help in describing security architecture for routing protocols.

Security protocols are the addition to the basic protocol set of TCP/IP suite to overcome the vulnerabilities lying in the design of these protocols. Security Protocols such IPSec, DNSSec, SSL, SSH, TLS are also prone to attacks such as DOS, spoofing, flooding etc. Attack detection in security protocols is crucial task. DNSSEC does not guard against poor configuration or bad information in the authoritative name server, and does not protect against buffer overruns or DDoS attacks. Small queries can generate larger UDP packets in response. DNSSEC has a hierarchical trust model. To securely resolve a name in DNSSEC, a root public key must be available at the resolver. The IPSEC protocols rely on a number of underlying technologies to achieve encryption and authentication. Specific SSH versions and implementations have been vulnerable to brute force attack.

In our research work we aim to develop an Intrusion Protection Systems which detects broad range of attacks along with reducing false alarms and increasing attack detection accuracy. During our research work we explored many of the vulnerabilities of these protocols and defense mechanisms for this. Although many defense techniques are the configuration based. The paper is organized as below. In section II we provide a brief overview of Intrusion Prevention Sytems. In section III Layered approach for intrusion detection is discussed. In Section IV Experimentation and results generated for our system is discussed followed by conclusion.

2. INTRUSION PREVENTION SYSTEM

Intrusion detection as defined by the Sysadmin, Audit, Networking, and Security (SANS) institute is the act of detecting activities that attempt to negotiate the confidentiality, integrity or availability of a resource [2]. Current network systems provide critical services for businesses to perform optimally and are target of attacks which aim to bring down the services provided by the network.

An Intrusion detection system (IDS) is software designed to detect unwanted attempts at accessing, manipulating, or disabling of computer systems, especially through a network. It is a specialized tool that knows how to parse and interpret network traffic and host activities. IDS technologies are not really effective against prediction a new attacks. There are several limitations, such as performance, flexibility, and scalability. The inadequacies inherent in current defenses have driven the development of a new breed of security products known as Intrusion Prevention Systems (IPS). Intrusion Prevention System (IPS) is a new approach system to defense networking systems, which combine the technique firewall with that of the Intrusion Detection properly, which is proactive technique, prevent the attacks from entering the network by examining various data record and detection demeanor of pattern recognition sensor, when an attack is identified, intrusion prevention block and log the offending data IPS make access control decisions based on application content, rather than IP address or ports as traditional firewalls had done. These systems are proactive defenses mechanisms designed to detect malicious packets within normal network traffic and stop intrusions dead, blocking the offending traffic automatically before it does any damage rather than simply raising an alert as, or after, the malicious payload has been delivered IPS use several response techniques. The comparison of IDS and IPS is shown in figure 2.[16]

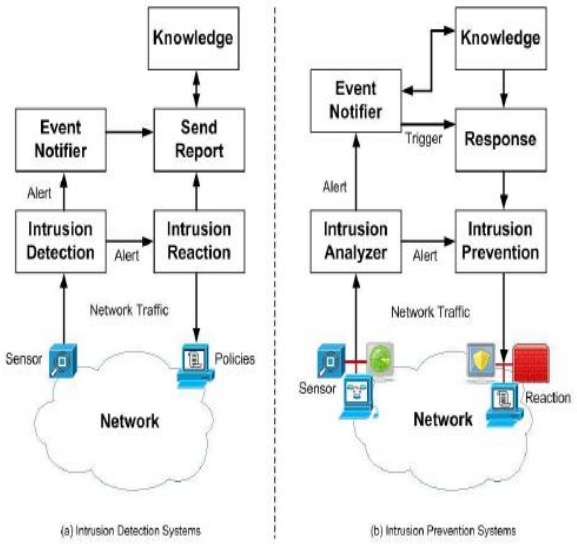

Figure 2 Comparison of IDS and IPS

Approaches to Intrusion Prevention Systems: There are different types of approaches is used in the IPS to secure the network.[14]

1. Signature-Based IPS: - It is commonly used by many IPS solutions. Signatures are added to the devices that identify a pattern that the most common attacks present. That's why it is also known as pattern matching. These signatures can be added, tuned, and updated to deal with the new attacks.

2. Anomaly-Based IPS: - It is also called as profile-based. It attempts to discover activity that deviates from what an engineer defines as normal activity. Anomaly-based approach can be statistical anomaly detection and non-statistical anomaly detection.

3. Policy-Based IPS: - It is more concerned with enforcing the security policy of the organization. Alarms are triggered if activities are detected that violate the security policy coded by the organization. With this type approaches security policy is written into the IPS device.

4. Protocol-Analysis-Based IPS - It is similar to signature based approach. Most signatures examine common settings, but the protocol-analysis-based approach can do much deeper packet inspection and is more flexible in finding some types of attacks.

IPS technologies: Basically IPS Host based and network-based.

1) Host-based IPS: Host-based IPSs [13] monitors the characteristics of a single host and the events occurring within

that host for suspicious activity. Examples of the types of characteristics a host-based IPS might monitor are wired and wireless network traffic, system logs, running processes, file access and modification, and system and application configuration changes. Most host-based IPSs have detection software known as agents installed on the hosts of interest. Each agent monitors activity on a single host and also performs prevention actions. The agents transmit data to management servers. Each agent is typically designed to protect a server, a desktop or laptop, or an application service. The agents are deployed to existing hosts on the networks, the components usually communicate over those networks instead of using a management network. Host-based IPSs run sensors on the hosts being monitored, they can impact host performance because of the resources the sensors consume.

2) Network-based IPS: A network-based IPS [13] monitors network traffic for particular network segments or devices and analyzes network, transport, and application protocols to identify suspicious activity. Network-based IPS components are similar to HIPS technologies, except for the sensors. A network-based IPS sensor monitors and analyzes network activity on one or more network segments. Sensors are available in two formats: appliance-based sensors, which are comprised of specialized hardware and software optimized for IPS sensor use, and software-only sensors, which can be installed onto hosts that meet certain specifications.

3. LAYERED APPROACH FOR INTRUSION DETECTION AND PREVENTION

Preprocessing is the organization of collected data from sensors in a particular pattern. This data is then placed in a structured database format by means of parsing and reconstructing. The cleansing process is protocol specific as we need different attributes of packets for intrusion analysis. If packet is from blacklisted source then system should discard packet without verifying it. When the packets are transformed and stored in the respective data stores it triggers intrusion detection.

Layered-based intrusion detection system gets its motivation from TCP/IP model, where a number of protocols are assigned different task at different level. Similar to this model, the layered intrusion detection system represents a sequential layered approach. 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 layered intrusion detection system framework is trained separately and then deployed sequentially. We define four layers that correspond to the four attack groups mentioned in the dataset. They are interface layer, network layer, transport layer and application layer. Each layer is then separately trained with a small set of relevant features. Feature selection or reduction is important for layered approach and discussed in next section. 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, thereby eliminating the need of further processing at subsequent layers enabling quick response to intrusion. The effect of such a sequence of layers is that the anomalous events are identified and blocked as soon as they are detected [2].

Data preprocessor is responsible for collecting and providing the audit data (in a specified form) that will be used by the next module to make a decision. Data preprocessor is, thus, concerned with collecting the data from the desired source and converting it into a format that is understandable by the intrusion detector. Data used for detecting intrusions range from user access patterns to network packet level features such as the source and destination IP addresses, type of packets . We refer to this data as the audit patterns.

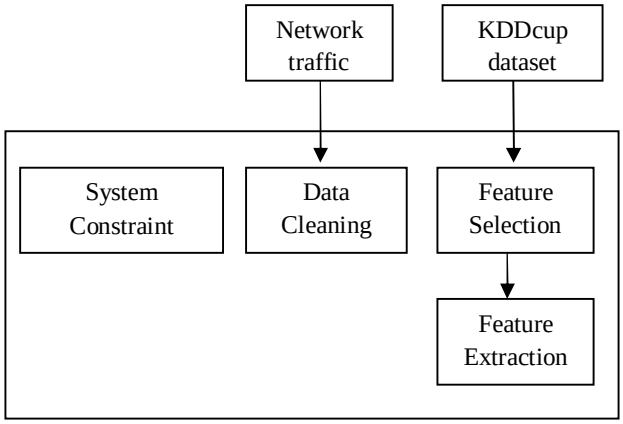

Figure 3 Preprocessing of Data

In the proposed model we have used four major functionalities in preprocessing module as shown in figure 2. Two different datasets are used for our experiments. Some experiments are carried out on real time network audit trails collected over high speed network. Often Intrusion Detection Systems are loaded with huge amount of data to be processed. Processing this enormous amount of data in real-time is major challenge faced in this area. Reduction in input data rate will provide additional time to detection engine for thoroughly process data and give more detection accuracy with less false positive. In the first round, input data cleaning by removing unwanted parameters is performed. Removal of noise and incomplete data makes the task of intrusion detection faster. But it also increases overlapping behavior of normal and intrusion data. Most modern data mining and soft computing based Intrusion Detection Systems uses data cleaning techniques to provide quality data to detection engine and in turn results in improved intrusion detection rate.

Our proposed system uses feature selection and extraction on KDD cup dataset which is freely available intrusion dataset. This dataset contains 41 features for intrusion specification. Not all the features available in raw input dataset are useful for intrusion detection. For detecting

particular category of intrusion, we require only subset of these features. Removal of forged and duplicate data will help in reducing false positive rate.

Another reason for false positive is lack of knowledge about network topology, hosts and services running on the hosts. In proposed model third functionality is system constraint check or configuration based processing. Configuration data about existing network, hosts, and services are stored in a file. Configuration parameters help in differentiating normal and intrusion data by providing additional information. Some portion of overlapping behavior is the challenge for Intrusion Detection Systems. The data for which Intrusion Detection System is not sure results in false detection, either false negative or false positive. Such ambiguity can be reduced by collecting information from various sources. This again helps in reducing false positive rate in proposed system. In our approach, we perform preprocessing based on type of packet. For proliferation of performance and reducing time factor in detection, we separate the packets into TCP/IP protocols, routing protocols and security protocols. Algorithm for preprocessing is given below

Algorithm: PreprocessPacket(p)
Input: Packet p, System Configuration Constraints List L
Begin
2. Read packet header ψ.
3. Detect Type of Protocol Δ= ψ ->T
4. If (ψ ->T=TCP/UDP/IP/ICMP/ARP/RARP) Δ = 1. // To separate the TCP/IP , routing and security protocols.
else if (ψ ->T = RIP/ BGP/EGP) Δ=2.
else Δ =3.
5. CleanPacket(Packet, Type) //This method will remove unnecessary header fields
6. If incomplete/duplicate Packet then discard packet;
7. End

We successful created data records for TCP/IP Packets and separate log files for the routing and security protocols for our experimentation. To collect the attack data, both, the web requests and the data accesses were logged. For the first data set, we generate 45 different attack sessions with 275 web requests resulting in 54,390 data requests. Combining the two together, the unified log has 45 unique attack sessions with 275 event vectors.

For the second dataset we used KDD dataset. Every record in the KDD 1999 data set symbolizes 41 features representing a variety of attacks such as the Probe, DoS, R2L and U2R. However, using all the 41 features for detecting attacks belonging to all these classes severely affects the performance of the system and also generates superfluous rules, resulting in fitting irregularities in the data which can misguide classification. Hence, we performed feature selection to effectively detect different classes of attacks. We now describe our approach for selecting features for every attack and why some features were chosen over others.

Algorithm: FeatureSelection
Input: Set of 41 features from KDD cup Data Set
Output: Reduced set of features R.

Step 1. Calculate the information gain for each attribute AiεD using (3).
Step 2. Choose an attribute Ai from D with the maximum information gain value.
Step 3. Split the data set D into subdatasets {D1,D2, . . . Dn} depending on the attribute values of Ai where Cj stands for jth attribute of class C.
Step 4. Find all the attributes whose information gain ratio > threshold.
Step 5. Store the selected attributes in the set R and output it.
Step6: End

We tested our algorithm for each category of attack. For every category, we applied all relevant attributes for that category, calculated gain for them and generated small subset which contains most relevant attributes for that category.

4. EXPERIMENTATION & RESULTS

Data preprocessing is major component of our proposed architecture. We have considered two datasets for our experimentation as mentioned in previous sections. The first data is collected over real time network using packet generators. We have developed a Java program for data formatting and implementing a layered approach. The program works as given in algorithm 1. The results achieved are logged and stored in the database. Three separate tables for TCP/IP protocols, routing protocols and security protocols are created. This helps in further analysis of packets. Before storing the packet info in the database, signatures for the attack on a specific protocol are searched. This reduces the time complexity rapidly as there is no need to check with signatures which are for other protocols.

The other dataset used is KDDcup1999 intrusion dataset which contains wide variety of intrusions simulated in network environment to acquire nine weeks of raw TCP dump data for a local-area network. A connection is a sequence of TCP packets starting and ending at some well-defined times, between which data flows to and from a source IP address to a target IP address. Each connection is labelled as either normal, or as an attack, with exactly one specific attack type. It is important to note that the testing data is not from the same probability distribution as the training data. This makes the task more realistic. The datasets contains a total of 22 training attack types. There are 41 features for each connection record that are divided into discrete sets and continuous sets according to the feature values. It consists of number of total records 494021. The 22 different types of network attacks in the KDD99 dataset fall into four main categories: DOS (Denial of Service), Probe, R2L(Remote to

Local), U2R(user to remote). The attacks in each class are as shown below:

Table 1: Classes of Attacks

S.N	Class	Attack Types
1	DOS	Back, Land, Neptune,pod, smurf, Teardrop,
2	U2R	Buffer_overflow, loadmodule, perl, rootkit
3	R2L	ftp_write, guess_passwd, imap, multihop, phf, spy,warezlient, warezmaster
4	Probe	IPsweep,nmap, satan,portsweep

For intrusion analysis all the 41 features are not required. Some specific features are only contributing for a specific attack. This reduces the amount of work for intrusion detection and increases accuracy. The feature selecion algorithm is given above in section III. The results we achieved after applying the algorithm is given below.

Feature Selection from KDD dataset

1. Feature Selection for Probe Layer

Probe attacks are aimed at acquiring information about the target network from a source that is often external to the network. For detecting Probe attacks, basic connection level features such as the 'duration of connection' and 'source bytes' are significant. We selected only four features for Probe layer. The features selected for detecting Probe attacks are presented in Table B.1.

Table B.1: Features for Probe Detection

S.N.	Name of Feature	Feature_No
1	src_bytes	5
2	duration	1
3	protocol_type	2
4	flag	4

2. Feature Selection for DoS Attacks

DoS attacks are meant to prevent the target from providing service(s) to its users by flooding the network with illegitimate requests. Hence, to detect attacks at the DoS layer, network traffic features such as the 'percentage of connections having same destination host and same service'and packet level features such as the 'duration' of a connection, 'protocol type', 'source bytes', 'percentage of packets with errors' and others are significant. To detect DoS attacks, it may not be important to know whether a user is 'logged in or not', or whether or not the shell' is invoked or 'number of files accessed' and, hence, such features are not considered in the DoS layer. From all the 41 features, we selected only nine features for the DoS layer.

Table B.2: DoS Layer Features

S.N.	Name of Feature	Feature_No
1	src_bytes	5
2	duration	1
3	protocol_type	2
4	flag	4
5	count	23
6	dst host same srv rate	34
7	dst host serror rate	38
8	dst host srv serror rate	39
9	dst host rerror rate	40

The features selected for detecting DoS attacks are presented in Table B.2.

3. Feature Selection for U2R attacks

U2R attacks involve the semantic details which are very difficult to capture at an early stage at the network level. Such attacks are often content based and target an application. Hence, for detecting U2R attacks, we selected features such as 'number of file creations', 'number of shell prompts invoked', while we ignored features such as 'protocol' and 'source bytes'. From all the 41 features, we selected only eight features for the U2R layer. Features selected for detecting U2R attacks are presented in Table B.3.

Table B.3: U2R Layer Features

S.N.	Name of Feature	Feature_ No
1	num_compromised	13
2.	root_shell	14
3	num_root	16
4.	num_file_creations	17
5	num_shells	18
6	num_access_files	19
7	is_host_logins	21

4. Feature Selection for R2L Attacks

R2L attacks are one of the most difficult attacks to detect and most of the present systems cannot detect them reliably. However, our experimental results presented earlier show that careful feature selection can significantly improve their detection. We observed that effective detection of the R2L attacks involve both, the network level and the host level features. Hence, to detect R2L attacks, we selected both, the network level features such as the 'duration of connection', 'service requested' and the host level features such as the 'number of failed login attempts' among others. Detecting R2L attacks, require a large number of features and we selected 14 features. The features selected for detecting R2L attacks are presented in Table B.4

Table B.4: R2L Layer Features

S.N.	Name of Feature	Feature_No
1	src_bytes	5
2	duration	1
3	protocol_type	2
4	flag	4
5	num_failed_logins	11
6	num_file_creations	17
7	num_shells	18
8	num_access_files	19
9	is_host_login	21
10	is_guest_login	22

Feature selection is an important task of Network Intrusion application. Large amount of attacks are threats to network and information security. Using Feature selection approach kdd attacks are detected with less error rate and high accuracy.

5. CONCLUSION

Data preprocessing is widely recognized as an important stage in anomaly detection. Data preprocessing is found to predominantly rely on expert domain knowledge for identifying the most relevant parts of network traffic and for constructing the initial candidate set of traffic features. Motivation for the paper comes from the large impact data preprocessing has on the accuracy and capability of anomaly-based NIPS. The review finds that many NIPS limit their view of network traffic to the TCP/IP packet headers. Time-based statistics can be derived from these headers to detect network behavior, and denial of service attacks. A number of other NIPS perform deeper inspection of request packets to detect attacks against network services and network applications. On the other hand, automated methods have been widely used for feature extraction to reduce data dimensionality, and feature selection to find the most relevant subset of features from this candidate set. These context sensitive features are required to detect current attacks. In our proposed system, we try to evaluate attack at every level of TCP/IP Model by combining network Intrusion detection and layered approach. Our preprocessing module has packet capture, feature selection and storing it in databases. But along with these basic features it also evaluates known network attacks by protocol layer wise inbuilt detection algorithm.

6. REFERENCES

[1] Shun-ichi Amari and Si Wu, “Improving support vector machine classifiers by modifying kernel function”, RIKEN Brain Science Institute Japan.

[2] Kapil Kumar Gupta, Baikunth Nath and Ramamohanarookotagiri, “A layered approach using conditional random fields for intrusion detection”, IEEE Tranc. on Dependence and secure computing, Vol.7, 2010

[3] G.MeeraGandhi, Kumaravel Appavoo and S.K Srivasta, “Effective network intrusion detection using classifiers decision trees and decision rules”,Int. J. Advanced network and application, Vol2, 2010

[4] Bernhard scholkopf, Kah kay Sung, Chris Burges, Federico and other, IEEE Transactions on signal processing, Vol. 45 , 1997

[5] Richard Machlin and David Opitz, “An empirical Evaluation of bagging and boosting”, National conference on A.I, providence Rhode Island 1997.

[6] Sandy Peddabachigari, Ajit Abraham and Johmson Thomas, “Intrusion detection system using decision trees and SVM”, Oklahoma state university USA.

[7] Huy Anh Nguye and Deokjai choi, “Application of data mining to network intrusion detection”, Korea.

[8] Weiming Hu, Wei Hu and Steve Maybank, “Adaboost based algorithm for network intrusion detection”, Tranc. On system man and cybernetics, 2008.

[9] Shilpa Lakhina, Sini Joseph and Bhupendra Verma, “Feature reduction using PCA for effective Anomaly- based intrusion detection on NSL-KDD”, Int. J. of engineering science and technology, 2010

[10] Snehal A.Mulay, P.R Devale and G.V Garje, “Intrusion detection using SVM and decision tree”, Int. J. of computer application, 2010

[11] J.Vishumathi and K.L Shunmuganathan, “A computational intelligence for evaluation of intrusion detection system ”, Indian J. of science and technology, Jan 2011

[12] Ritu Ranjani Singh, Neetesh Gupta and Shiv Kumar, “To reduce the false alarm in intrusion detection system”, Int. J. of soft computing and engineering, May 2011

[13] Defending yourself: IEEE software September/October 2000 tutorial

[14] Xunyi Ren, Ruchuan Wang and Hejunzhou, “intrusion detection system method using protocol classification and Rough set based SVM”, www.ccsenet.org/journal.html,2009

[15] Peyman Kabiri and Ali A. Ghorbani, “Research on ID and Response:A survey ”, Int. J. of network security, 2005 M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.

[16] Deris Stiawan, Abdul Hanan Abdull ,”Characterizing Network Intrusion Prevention System ”*International Journal of Computer Applications (0975 – 8887) Volume 14– No.1, January 2011*

[17] Davis, Jonathan Jeremy & Clark, Andrew J. (2011) Data preprocessing for anomaly based network intrusion detection : a review. *Computers & Security*, *30*(6-7), pp. 353-375.