ISSN:0975-9646



# Rule Based Classifier Models For Intrusion Detection System

Vivek kshirsagar[1],Madhuri S.Joshi[2]

*Computer Science and Engineering Department,Government Engineering College Railway Station Road,Aurangabad,Maharashtra,India*

*Jawaharlal Nehru Engineering College Cidco,Aurangabad,Maharashtra,India*

***Abstract*— One key feature of intrusion detection systems is their ability to provide a view of unusual activity and issue alerts notifying administrators and/or block a suspected connection. Intrusion detection is a process of identifying and responding to malicious activity targeted at computing and networking resources. Over the past decade, the field of IDS has been driven into overdrive by the explosive proliferation of personal and server-based computers.There is need of a systematic and automated IDS development process rather than the pure knowledge based and engineering approaches which rely only on intuition and experience.This encourages studying some Data Mining based frameworks for Intrusion Detection. These frameworks use data mining algorithms to compute activity patterns from system audit data and extract predictive features from the patterns. Machine learning algorithms are then applied to the audit records that are processed according to the feature definitions to generate intrusion detection rules**

***Keywords*- intrusion detection,association rules,JRip,attacks**

## I. INTRODUCTION

An Intrusion Detection System (abbreviated as IDS) is a defense system, which detects hostile activities in a network. The key is then to detect and possibly prevent activities that may compromise system security, or a hacking attempt in progress including reconnaissance/data collection phases that involve for example, port scans. The ultimate desire of IDS functionality is the identification of all intrusive behavior within an environment, and the reporting of that behavior in a timely manner. These frameworks use data mining algorithms to compute activity patterns from system audit data and extract predictive features from the patterns. Machine learning algorithms are then applied to the audit records that are processed according to the feature definitions to generate intrusion detection rules.Raw (binary) audit data is first processed into ASCII network packet information (or host event data), which is in turn summarized into connection records (or host session records) containing a number of basic features, such as service, duration, source IP address, destination IP address *etc.* Data mining programs are then applied to the connection records to compute the frequent patterns (i.e., association rules and frequent episodes), which are in turn analyzed to construct additional features for the connection records [1]. Classification programs are then used to inductively learn the detection models.

## II. RELATED WORK

Primarily, An ID is concerned with the detection of hostile actions. This network security tool uses either of two main techniques. The first one, anomaly detection, explores issues in intrusion detection associated with deviations from normal system or user behavior. The second employs signature detection to discriminate between anomaly or attack patterns (signatures) and known intrusion detection signatures. Both methods have their distinct advantages and disadvantages as well as suitable application areas of intrusion detection.

When considering the area being the source of data used for intrusion detection, another classification of intrusion detection systems can be used in terms of the type of the protected system. There is a family of IDS tools that use information derived from a single host (system) — host based IDS (HIDS*)* and those IDSs that exploit information obtained from a whole segment of a local network (network based IDS, i.e.NIDS).

Two primary types of HIDS can be distinguished:

Systems that monitor incoming connection attempts (RealSecure Agent, PortSentry). These examine host-based incoming and outgoing network connections. These are particularly related to the unauthorized connection attempts to TCP or UDP ports and can also detect incoming portscans.

Systems that examine network traffic (packets) that attempts to access the host. These systems protect the host by intercepting suspicious packets and looking for aberrant payloads (packet inspection). Systems that monitor login activity onto the networking layer of their protected host (HostSentry). Their role is to monitor log-in and log-out attempts, looking for unusual activity on a system occurring at unexpected times, particular network locations or detecting multiple login attempts..Systems that monitor actions of a super-user (root) who has the highest privileges (LogCheck). IDS scans for unusual activity, increased super-user activity or actions performed at particular times, etc.

Systems that monitor file system integrity (Tripwire, AIDE). Tools that have this ability (integrity checker) allow the detection of any changes to the files that are critical for the operating system.

Systems that monitor the system register state (Windows platform only). They are designed to detect any illegal changes in the system register and alert the system administrator to this fact.[2]

Kernel based intrusion detection systems. These are especially prevalent within Linux (LIDS, OpenWall). These systems examine the state of key operating system files and streams, preventing buffer overflow, blocking unusual interprocess communications, preventing an intruder from attacking the system. In addition, they can block a part of the actions undertaken by the super-user (restricting privileges). The HIDS reside on a particular computer and provide protection for a specific computer system. They are not only equipped with system monitoring facilities but also include other modules of a typical IDS, for example the response module. HIDS products such as Snort, Dragon Squire, Emerald eXpert-BSM, NFR HID, Intruder Alert all perform this type of monitoring. The network-based type of IDS (NIDS) produces data about local network usage. The NIDS reassemble and analyze all network packets that reach the network interface card operating in promiscuous mode. They do not only deal with packets going to a specific host – since all the machines in a network segment benefit from the protection of the NIDS. Network-based IDS can also be installed on active network elements, for example on routers. Since intrusion detection (for example flood-type attack) employs statistical data on the network load, a certain type of dedicated NIDS can be separately distinguished, for example, those that monitor the traffic (Novell Analyzer, Microsoft Network Monitor). These capture all packets that they see on the network segment without analyzing them and just focusing on creating network traffic statistics. Typical network-based intrusion systems are: Cisco Secure IDS (formerly NetRanger), Hogwash, Dragon, E-Trust IDS.

## III. PROPOSED MODELS

Steps in proposed model 1 is as follows :
Step 1: Input KDD Train dataset.
Step 2: Pre processing of the dataset.
Step 3: Generate rules using JRip algorithm.
Step 4: Evaluate Model using KDD Test Data Set.
Step5:Generate Alert message for attack otherwise store packet information in database.

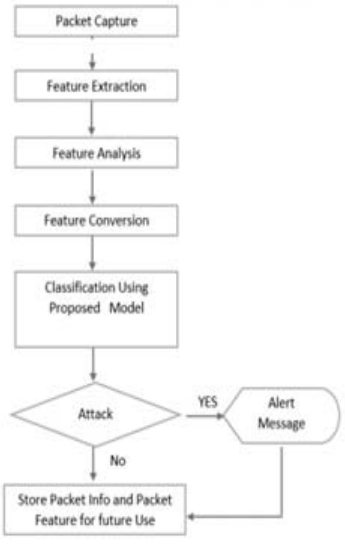


Fig.1 Flowchart of Proposed Model 1

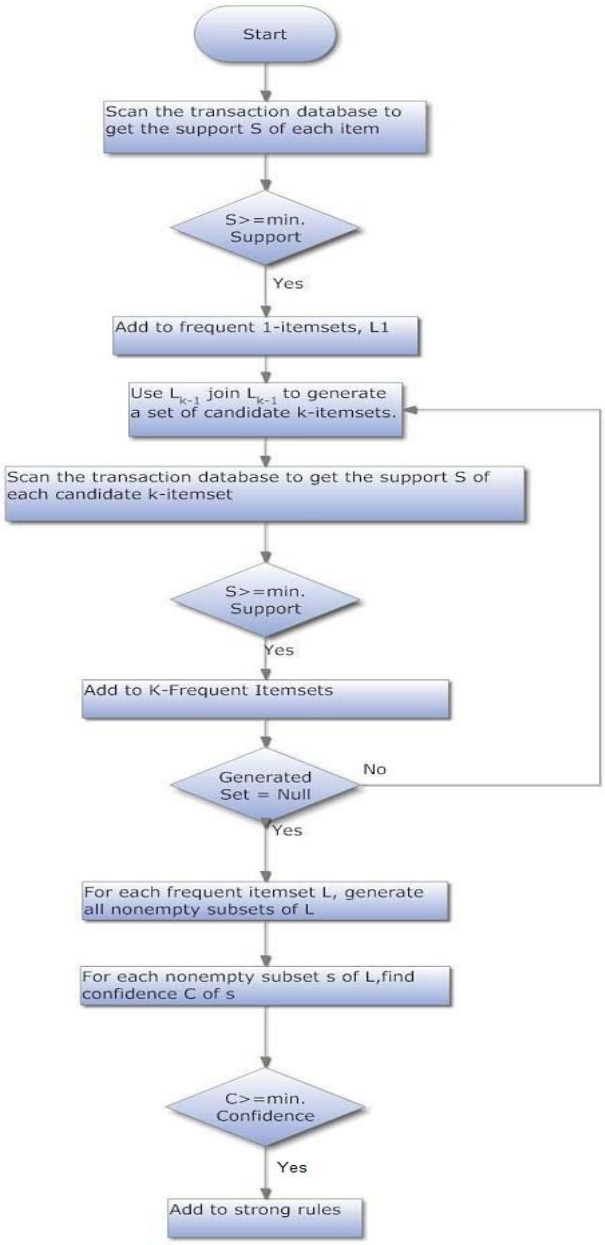


Fig.2 Flowchart of Proposed Model 2

The proposed model 2 describes an intrusion detection system for effectively identifying the intrusion activities within a network [3]. The system will be able to detect an intrusion behavior of the networks since the rule base contains a better set of rules. Here, we have used Improved Apriori algorithm for generation of association rules, which are obtained using frequent itemsets. data The efficiency of association rule mining can be improved by sampling, reducing the number of passes, hash-based itemset counting, transaction reduction and partitioning.[4]The experiments and evaluations of the proposed intrusion detection system are performed with the KDD Cup 99 intrusion detection dataset. [5]The objectives of this method is that to detect worm by using rules that are established through the analysis of behaviour of a specific application and worm activities (signature)

## IV. DATA SET

The KDD Cup 1999 dataset used for benchmarking intrusion detection problems is used in our experiment.The known attack types are those present in the training dataset while the novel attacks are the additional attacks in the test datasets not available in the training datasets. The attacks types are grouped into four categories:
(1). DOS: Denial of service – e.g. syn flooding
(2). Probing: Surveillance and other probing, e.g. port scanning
(3). U2R: unauthorized access to local super user (root) privileges, e.g. buffer overflow
attacks.
(4). R2L: unauthorized access from a remote machine, e.g. password guessing

## V. DATAMINING TECHNIQUES

JRIP:

JRip popularly known as Repeated Incremental Pruning to Produce Error Reduction (RIPPER) is one of the basic and most popular algorithms [6]. In this algorithm the five attack Classes are examined in increasing size and an initial set of rules for each class is generated using incremental reduced error i.e growing of one rule by adding combination of attributes in the antecedents to the rule.Here all possible values of each attributes gets tested and then the rule is finalized.Similarly pruning step also results in dropping attributes from antecedents until the minimum possible attributes are remaining to generate the rule.The rules are selected based on information gain.[7] The algorithm terminates on generation of rules for the five attack classes.The stategy of replacing and revising the rules hence improves the accuracy of the generated rules.[8]

## VI. EXPERIMENTAL RESULTS

Table 1. Sample Of Rules For R2L Attack Type

| |
|---|
| (service = ftp_data) and (dst_host_same_srv_rate>= 1) and (dst_host_srv_count<= 12) => attack=R2L (97.0/4.0) |
| (hot >= 1) and (dst_host_serror_rate>= 0.04) and (dst_host_diff_srv_rate<= 0.02) and (dst_host_srv_count<= 52) => attack=R2L (80.0/1.0) |
| (duration >= 4) and (hot >= 1) and (duration >= 156) and (dst_bytes<= 2551) and (dst_host_diff_srv_rate<= 0.29) => attack=R2L (12.0/0.0) |
| (service = imap4) and (count <= 4) => attack=R2L (10.0/0.0)<br>(num_access_files>= 1) and (src_bytes<= 116) => attack=R2L (7.0/0.0) |

Table 2. Sample Of Rules For U2R Attack Type

| |
|---|
| (service = telnet) and (num_file_creations>= 1) and (dst_bytes<= 8356) and (dst_host_srv_count<= 6) => attack=U2R (20.0/1.0) |
| (root_shell>= 1) and (dst_host_count<= 10) => attack=U2R (21.0/5.0) |
| (dst_host_count<= 4) and (dst_host_srv_count<= 4) and (src_bytes<= 4) and (logged_in>= 1) => attack=U2R (6.0/1.0).(duration >= 7) and (num_file_creations>= 1) and (src_bytes<= 230) and (duration <= 21) =>attack=U2R (3.0/0.0) |

Table 3. Sample Of Rules For PROBE Attack Type

| |
|---|
| (dst_host_diff_srv_rate>= 0.11) and (src_bytes<= 1) and (dst_host_same_src_port_rate>= 0.15) => attack=PROBE (631.0/1.0) |
| (srv_count<= 2) and (flag = RSTR) and (src_bytes<= 0) => attack=PROBE (14.0/1.0) |
| (protocol_type = udp) and (src_bytes<= 1) => attack=PROBE (43.0/1.0) |
| (srv_count<= 3) and (protocol_type = icmp) and (src_bytes<= 18) => attack=PROBE (6.0/0.0) |

Table 4. Sample Of Rules For NORMAL Attack Type

| |
|---|
| (count <= 56) and (dst_host_srv_diff_host_rate>= 0.01) and (logged_in>= 1) => attack=NORMAL (53037.0/2.0) |
| (count <= 76) and (dst_host_serror_rate<= 0) and (src_bytes<= 1031) and (src_bytes>= 29) and (dst_bytes>= 1) => attack=NORMAL (42433.0/2.0) |
| (count <= 24) and (src_bytes<= 35195) and (dst_host_serror_rate<= 0.02) and (wrong_fragment<= 0) and (dst_host_srv_count<= 254) and (dst_host_rerror_rate<= 0) and (dst_host_srv_count>= 8) => attack=NORMAL (5919.0/1.0) |
| (count <= 2) and (src_bytes<= 1339) and (dst_host_serror_rate<= 0.87) and (dst_host_srv_count>= 3) and (service = http) => attack=NORMAL (135.0/0.0) |
| (count <= 19) and (dst_host_same_src_port_rate>= 0.02) and (src_bytes<= 1010) and (wrong_fragment<= 0) and (dst_host_srv_serror_rate<= 0.02) and (dst_host_same_src_port_rate<= 0.44) => attack=NORMAL (422.0/2.0) |

Table 5. Sample Of Rules Generated by Association Rule Classifier

| Intrusion Rules | Support | Confidence | Normal Rules | Support | Confidence |
|---|---|---|---|---|---|
| (phf=1) | a(1)/N | a(1)/a | (phf=1) | a(2)/N | a(2)/a |
| (phf=1)Λ(pro=http) | b(1)/N | b(1)/b | (phf=1)Λ(pro=http) | b(2)/N | b(2)/b |
| (phf=1) Λ(pro=http)Λ (count=low) | c(1)/N | c(1)/c | (phf=1)Λ(pro=http)Λ (count=low) | c(2)/N | c(2)/c |
| (phf=1)Λ(pro=http)Λ (count=medium) | c(1)/N | c(1)/c | (phf=1)Λ(pro=http)Λ (count=medium) | c(2)/N | c(2)/c |
| (phf=1)Λ(pro=http)Λ (count=high) | c(1)/N | c(1)/c | (phf=1)Λ(pro=http)Λ (count=high) | c(2)/N | c(2)/c |

## VII. CONCLUSIONS

The aim of the research was to test different rule based classifiers for intrusion detection system.For choosing appropriate classifiers the accuracy of classification was taken into consideration. Overall the proposed models produces high accuracy of classification. The entire network intrusion detection framework is developed using WEKA environment with java packages. The KDD dataset was used to train and test the classifiers for the 5- classes (normal, dos, probe, u2r and r21). Once the algorithms were trained they were used to detect attacks form live traffic.

## REFERENCES


[1] Zhan Jiuhua,Leshan Teachers Coll., Leshan,".Intrusion Detection System Based on Data Mining",.In proc. 1st International Workshop on Knowledge Discovery and Data Mining, January, 2008,pp. 402-405.

[2] Yusufovna,S.F.,".Integrating Intrusion Detection System and Data Mining",Symp.,October,2008,pp. 256 – 259.

[3] Rasha G. Mohammed Helali, "Data Mining Based Network Intrusion Detection System:ASurvey.Novel Algorithms and Techniques in Telecommunications and Networking", 2010,pp. 501-505

[4] E. Frank, I. H. Witten,".Generating Accurate Rule Sets Without Global Optimization",InternationalConference on Machine Learning,1998,pp.144-151

[5] MahbodTavallaee, EbrahimBagheri, Wei Lu, Ali A. Ghorbani.,"A Detailed Analysis of the KDD CUP 99 Data Set",Computational intelligence in security and defense applications.Symp. , 2009.

[6] Thanvaratkomviriyavat, PhirivitSnagatanease,"Network intrusion detection and classification using decision tree and rule based approach", International conference on machine learning models,technologiesandapplications,Jan2009

[7] MohammadrezaEktefa, SaraMemar, Fatimah Sidi,LillySurianiAffendey, " Intrusion Detection Using Data Mining Techniques" , International Conference on Information Retrievaland KnowledgeManagement(CAMP) 2010

[8] Vivek Kshirsagar,Madhuri Joshi,"Comparative Analysis of Various Classifiers for Performance Improvement in Intrusion Detection Systems by reducing false positives",International journal of Computer Science and Information Technologies,Vol 6(5),2015,pp 4825-4828