Finding Frequent Itemsets using Apriori Algorihm to Detect Intrusions in Large Dataset

Kamini Nalavade1, B.B. Meshram2,

Abstract

With the growth of hacking and exploiting tools and invention of new ways of intrusion, Intrusion detection and prevention is becoming the major challenge in the world of network security. The increasing network traffic and data on Internet is making this task more demanding. There are various approaches being utilized in intrusion detections, but unfortunately any of the systems so far is not completely flawless. The flase positive rates makes it extremely hard for to analyze and react to attacks. Intrusion detection systems using data mining approaches make it possible to search patterns and rules in large amount of audit data. In this paper, we represent an model to integrate association rules to intrusion detection to design and implement an network intrusion detection system. Our technnique is used to generate attack rules that will detect the attacks in network audit data using anomaly detection. This shows that the association rules mining algorithm is capable of detecting network intrusions. The KDD dataset which is freely available online is used for our experimentation and results are discussed. Our aim is to experiment with dfiiferent parameters of apriori algorithm to build a string intrusion detection system using association rule mining.

Keywords

Intrusion, Security, Association rule mining, Network, Data mining, Apriori

1. Introduction

Communication on Internet, in spite of implementation of advanced security measures, is always under innovative and inventive attacks. Given the different type of attacks like Denial of Service, Probing, Remote to Local, User to Root and others, it is a challenge for any intrusion prevention system to detect a wide variety of attacks. The goal of intrusion detection systems is to automatically detect attack from the continous stream of network data traffic. Once an attack is detected, alarm is generated for administrator. Advanced version Intrusion prevention systems provide actions against the attacks instead of just producing alarms. Two approaches are used for intrusion detection: Misuse detection and Anomaly detection. Generally signature based systems are used to detect known attacks. These methods are pre-coded with signatures of attacks and perform rule matching to detect intrusions. But these kinds of systems are not sufficient to detect new or unknown attacks. Examples of misuse detection include expert systems, keystroke monitoring, and state transition analysis. Anomaly detection systems assume that an intrusion should deviate the system behaviour from its normal pattern. This approach can be implemented using statistical methods, neural networks, predictive pattern generation. In both areas, some form of user intervention is required, such as coding the attacks into the system or checking if the deviation from normal usage is a true attack. As audit logs may contain useful and rich information that can be used to build a better detection model, data mining techniques, which can discover meaningful knowledge in large volumes of data, can be used to analyze the network audit logs. By analyzing the audit logs, meaningful data can be extracted to generate better detection models. When data mining is to be applied to large volumes of network traffic data to search for patterns, it can provide valuable insights to attack patterns, thus allowing us to build a more effective detection model. These insights can identify new signatures as and when they appear, reduce the need of administrator's experience and intuition in detecting a new intrusion, and protect against constantly changing future threats.

The major objective of this paper is to build an inrusion detection system using association rule mining. Generating interesting rules from audit data to detect unknown intrusions. We investigate the performance of our intrusion detection system in terms of performance criteria as detection rate and false positive rate. The paper is organised as follows: Section 2 depicts the related work. In section 3 we explain the association rule mining and apriori algorithm for intrusion detection. In section 4 we describe our proposed system for network intrusion and detection. Section 5 illustates the experimentation and results followed by conclusion.

2. Related Work

Here a newly developed technique named, "The Research on the Application of Association Rules Mining Algorithm in Network Intrusion Detection" [6] is discussed. In this paper author have describe about Network Intrusion Detection System (IDS), as the main security defending technique, is second guard for a network after firewall. Since it can discern and respond to the hostile behavior of the computer and network resource, it is a hot area for research network security nowadays.

Furthermore author descried about Data mining technology which can be applied to the network intrusion detection, and Precision of the detection will be improved by the superiority of data mining. Basically In this paper, author have presents the study of an example running to contract two algorithms. Presented results have shown that the fuzzy rule mining algorithm is more convenient than Apriori algorithm to mine mass network log database.

In [24] paper, authors have integrated two technique data mining and fuzzy technique. Where fuzzy association rules have applied to design and implement an abnormal network intrusion detection system. Here author presents that when the association rules used in traditional information detection cannot effectively deal with changes in network behavior, it will better meet the actual needs of abnormal detection to introduce the concept of fuzzy association rules to strengthen the adaptability. Basically in This paper author mainly focused on the study of Denial of Service (DOS). According to the author's experimental results, they have found that their system can correctly identify all DOS attacks on test after appropriate adjustment of system parameters. Moreover, they have also proved, in the experiment, that their system would not result in false positives under such circumstances as a large amount of instantaneous FTP normal packet flow. In addition, if source of an attacker can be determined, the system will also be able to promptly inform the firewall to alter its rules and cut off the connection. According to another research network security is becoming an increasingly important issue, since the rapid development of the Internet. Network Intrusion Detection System (IDS), as the main security defending technique, is widely used against such malicious attacks.

Lee and Stolfo (1998) utilized the basic association rules algorithms to mine rules from system audit data into an aggregate rule set to form the user's normal profile. Two rules are merged if heir right and left hand sides are exactly the same or their RHSs can be combined and LHSs can also be combined, and the support and confidence values are close. For example, service = http→src_bytes=20 and service= http→src_bytes=30 can be combined into service =http→20 ≤src_bytes ≤ 30. Any subsequent system activities are analyzed to mine frequent patterns and the new pattern set is compared with the normal profile. Similarity functions are used to evaluate deviations involving missing or new rules, violation of the rules (same antecedent but different consequent), and significant changes in support of the rules. M.Sulaiman khan, Maybin Muyeba and Frans Coenen[1] described weighted association rule mining from fuzzy data in their paper. Then some other proposed association rule mining for weighted value not necessarily binary value. The value should be continuous or discrete value to be presented in the database. In 2009 Flora S. Tsai [9] described network intrusion detection system using association rule mining in his paper. This helped to generated interesting rules from the KDD data set.

3. Association Rule Mining for Intrusion Detection

The main aim of intrusion detection or protection systems is to protect their systems from the threats that come with increasing network connectivity and reliance on information systems. Given the level and nature of modern network security threats, the question for security professionals should not be whether to use intrusion detection, but which intrusion detection features and capabilities to use. IDSs have gained acceptance as a necessary addition to every organization's security infrastructure. Despite the documented contributions intrusion detection technologies make to system security, in many organizations one must still justify the acquisition of IDSs. There are several compelling reasons to acquire and use IDSs: 1. To prevent problem behaviors by increasing the perceived risk of discovery and punishment for those who would attack or otherwise abuse the system.

2. To detect attacks and other security violations that is not prevented by other security measures

3. To detect and deal with the preambles to attacks

The main aim is to develop a network based intrusion detection system based on modified Apriori approach for attack detection and test the results. In this section we describe the desin and mehodology of our proposed system

Data mining generally refers to the process of extracting or mining knowledge from a large amount of data. This process first understands the existing data and then predicts the new data. It is the core of Knowledge Discovery and Data mining (KDD). Kind of Patterns found in Data Mining Task are specified by Data Mining Functionalities. In general, data mining tasks are categorized into two categories: predictive and descriptive. The general properties of the data in the database are characterized by Descriptive mining. Inference on the current data in order to make predictions is performed by Predictive mining. Association rule mining. Specifically, two data mining approaches have been proposed and used for anomaly detection : association rules and frequency episodes. Association rule algorithms find correlations between features or attributes used to describe a data set.Association rules mining started as a technique for finding interesting rules from transactional databases [1].

Association rule problem:
Given a set of items I={I1,I2,…,Im} and a database of transactions D={t1,t2, …, tn} where ti={Ii1,Ii2, …, Iik} and Iij ∈ I, the Association Rule Problem is to identify all association rules X ➔ Y with a ***minimum support and confidence***.

The support of the rule is the percentage of transactions that contains both X and Y in all transactions and is calculated as |X Y| / |D|. The support of the rule measures the significance of the correlation between itemsets. The confidence is the percentage of transactions that contain Y in the transactions that contain X. The confidence of a rule measures the degree of correlation between the itemsets and is calculated as |X Y | / |X|. The support is a measure of the frequency of a rule and the confidence is a measure of the strength of the relation between the sets of items.
Two Step Process of association rule mining :
i) Frequent itemset identification (Support as the criterion): Discover the large (frequent) itemsets that have transaction support above a predefined minimum threshold. Given d items, there are 2^d possible candidate itemsets

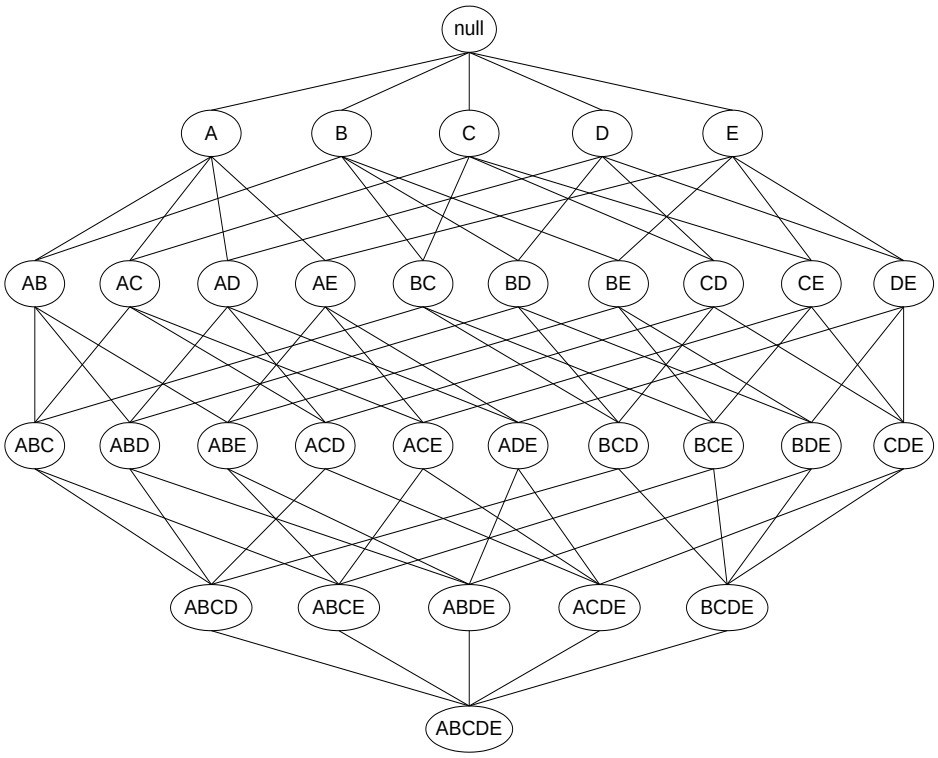


Figure 1. Frequent itemset generation

ii) Rule generation from frequent itemset (Confidence as the criterion): Use the obtained large itemsets to generate the association rules that have confidence above a predefined minimum threshold.

To discover the association rules from the DARPA dataset, various approaches should be applied for different kind of

dataset as follows.

A. Association Rule Mining for Binary Value

For binary weighted value it is easy to find out

the frequent item set. generates The frequent item set is generated by the Apriori algorithm from a large number of data set. In association rules mining weights are considered as the highest priority. In those rules Apriori algorithm can be imagined as two steps. Firstly it generates candidate sets. Secondly, it prunes the entire non-frequent item set after each step using the minimum support and the weight of the item from the data base. Many item sets could be eliminated by the pruning process which are not frequent.

B. Association Rule Mining for Continuous Value

If a particular attribute takes a value in the range [0…1] it is taken as a continuous attribute in the Tanagra tool. This could be taken as Fuzzy data and hence fuzzy weighted Association rule mining as described in fuzzy mining paper could be used here. The weight of fuzzy data can be defined as Fuzzy Item Weight (FIW). Now Fuzzy Item set Transaction Weight (FITW) is the aggregate weights of all the fuzzy sets associated with the items in the item set present in a single transaction. From this FITW support and Confidence value can be calculated as per generalizing the notion of support.

C. Association Rule Mining for Discrete value

The normalization of the data becomes very difficult If the range of values that an attribute in the data set can take is very large . The traditional approach to deal with this type of data in association analysis is to convert each value into a set of binary values. The discrete attributes are normalized i.e. we find a set of thresholds that can be used to convert the attributes into a categorical variable. This kind of normalization affects the accuracy of the rule generation technique which may lead to higher misclassification rate.

Apriori Algorithm

Here I am using apriori algorithm for association rule

mining technique to produce association rule There are different types of algorithms used to mine frequent item sets. One of the most popular data mining approaches is to find frequent item sets from a transaction dataset and derive association rules. A finding frequent item set (item sets with frequency larger than or equal to a user specified minimum support) is not trivial because of its combinatorial explosion [14]. Once frequent item-sets are obtained, it is straightforward to generate association rules with confidence larger than or equal to a user specified minimum confidence.

Apriori is a seminal algorithm for finding frequent item-sets using candidate generation [1]. It is characterized as a level-wise complete search algorithm using anti-monotonicity of

item-sets, "if an item-set is not frequent, any of its superset is never frequent". By convention, Apriori assumes that items within a transaction or item-set are sorted in lexicographic order. Let the set of frequent item-sets of size *k* be Fk and their candidates be Ck. Apriori first scans the database and searches for frequent item-sets of size 1 by accumulating the count for each item and collecting those that satisfy the minimum support requirement. It then iterates on the following three steps and extracts all the frequent item-sets.

Apriori Property

It is an **anti-monotone** property: if a set cannot pass a test, all of its supersets will fail the same test as well. It is called **anti-monotone** because the property is monotonic in the context of failing a test. All nonempty subsets of a frequent itemset must also be frequent. An itemset I is not frequent if it does not satisfy the minimum support threshold: P(I) < min_sup If an item A is added to the itemset I, then the resulting itemset I □ A cannot occur more frequently than I.

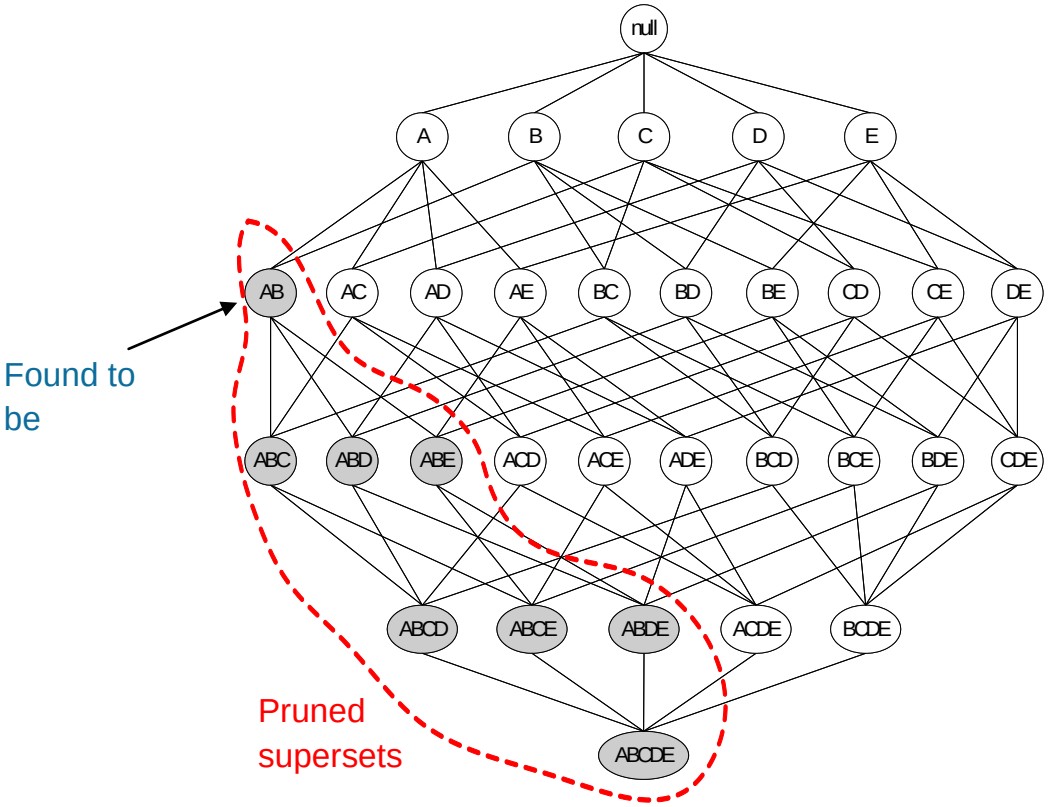


Figure 2: Apriori frequent itemset generation[10]

Pseudo code

Ck: Candidate itemset of size k Lk : frequent itemset of size k

L1 = {frequent items};

for (k = 1; Lk !=Æ; k++) **do**

- Ck+1 = candidates generated from Lk;

- **for each** transaction t in database **do**

increment the count of all candidates in Ck+1 that are contained in t;

endfor;

- Lk+1 = candidates in Ck+1 with min_support

endfor; return ∪k Lk;

Our approach is to implement the Apriori algorithm with minimum resources including hardware and less computational heads. Because apriori algorithm suffers from data complexity problems, i.e. For every step of candidate generation the algorithm has to scan the entire database, and as we are aware that the larger the database the difficult it is to scan completely, therefore candidate generation and candidate pruning are considered to be tedious as it involves bringing new data, after random unexpected intervals of time. We have to also take care that less memory should be utilized during the scanning process.

4. System Evaluation and Results

In this section we demonstrate how the system parameters are decided and fixed. Also the different performance parameters used to evaluate the system. Then we discuss the results achieved in intrusion detection using k-means clustering algorithm on KDD cup dataset. Next we evaluated the performance of k-means clustering algorithm with different values of initial cluster.

4.1 Dataset and Normalization of data

The 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. 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

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).
In order to know how to read the data from the audit data, we need to analyze how the audit da ta is being recorded. The audit data is processed for data mining purpose and is split into two files, the training set which contains around five million rows and the test set with 10% of the training set.

4.2 Performance Parameters

There are many measures available for evaluating system performance. For evaluating intrusion detection results following measure are generally used.

1. True positive (TP) means number connections that were correctly classified as intrusion.
2. True Negative (TN) means number of connections that were incorrectly classified as intrusion.
3. False positive (FP) means number of intrusion connections that were incorrectly classified as normal.
4. False negative FN) means number of normal connections that were incorrectly classified as intrusion.

To determine how many misclassification are found we use term Recall. Precision is how many records are correctly classified by the system.

Precision=TP/TP+FP……..(1)
Recall=TP/TP+FN…… .(2)

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad \text{.....(3)}$$

Confusion Matrix for Intrusion Protection System

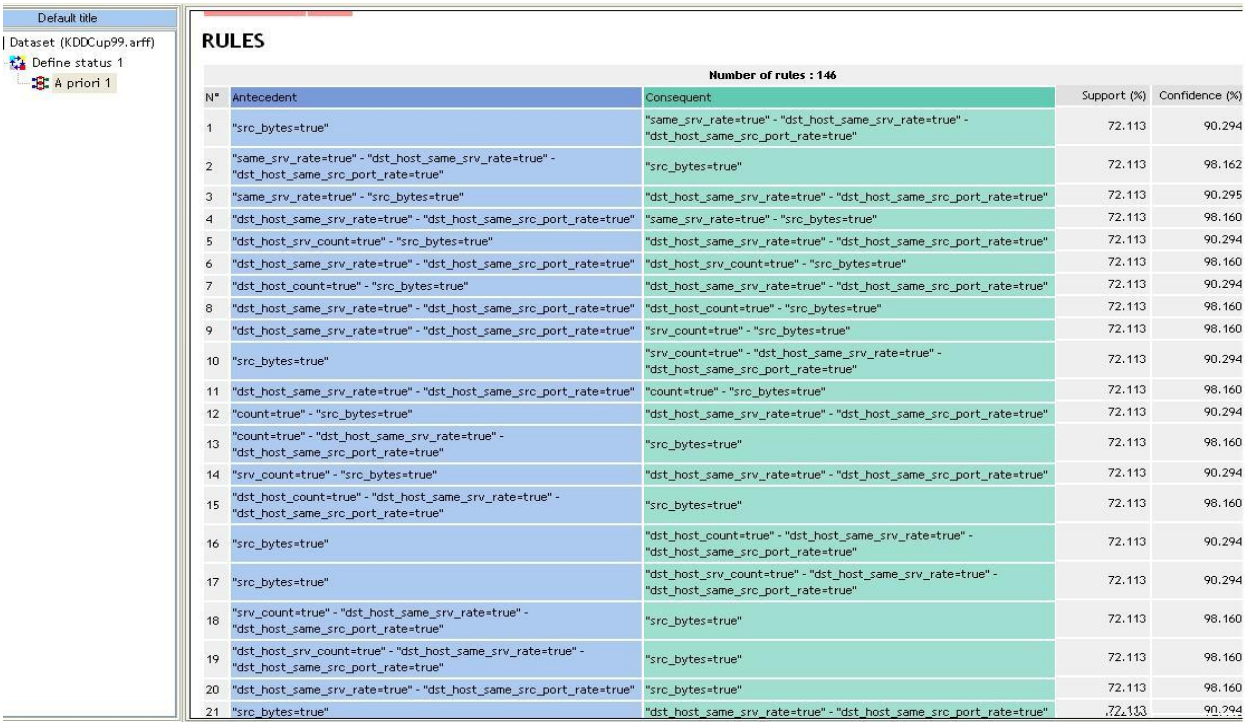
Table 2: Confusion Matrix

	Predicted Class		
Actual Class		Normal	Attack
	Normal	TP	FN
	Attack	FP	TN

We mainly concentrate on false positive rate (fpr), recall, precision and overall accuracy. In the next two sections, we present two sets of experiments, each designed to demonstrate a different point. The first set is used to intrusion detection results using unsupervised anomaly detection method. We also show that clustering methods can be employed to help the performance of classification-based intrusion detection techniques. The second set of results shows that how the different cluster values for k-means algorithm affects the performance of system.

4.3 Experimental results

The software framework of this work has been developed with Tanagra tool. Tanagra is a data mining suite build around graphical user interface. Tanagra is particularly strong in statistics, offering a wide range of uni and multivariate parametric and nonparametric tests. Equally impressive is its list of feature selection techniques. Together with a compilation of standard machine learning techniques, it also includes correspondence analysis, principal component analysis, and the partial least squares methods. Tanagra is more powerful, it contains some supervised learning but also other paradigms such as clustering, supervised learning, meta supervised learning, feature selection, data visualization supervised learning assessment, statistics, feature selection and construction algorithms.



N°	Antecedent	Consequent	Support (%)	Confidence (%)
1	"src_bytes=true"	"same_srv_rate=true" - "dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	72.113	90.294
2	"same_srv_rate=true" - "dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	"src_bytes=true"	72.113	98.162
3	"same_srv_rate=true" - "src_bytes=true"	"dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	72.113	90.295
4	"dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	"same_srv_rate=true" - "src_bytes=true"	72.113	98.160
5	"dst_host_srv_count=true" - "src_bytes=true"	"dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	72.113	90.294
6	"dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	"dst_host_srv_count=true" - "src_bytes=true"	72.113	98.160
7	"dst_host_count=true" - "src_bytes=true"	"dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	72.113	90.294
8	"dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	"dst_host_count=true" - "src_bytes=true"	72.113	98.160
9	"dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	"srv_count=true" - "src_bytes=true"	72.113	98.160
10	"src_bytes=true"	"srv_count=true" - "dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	72.113	90.294
11	"dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	"count=true" - "src_bytes=true"	72.113	98.160
12	"count=true" - "src_bytes=true"	"dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	72.113	90.294
13	"count=true" - "dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	"src_bytes=true"	72.113	98.160
14	"srv_count=true" - "src_bytes=true"	"dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	72.113	90.294
15	"dst_host_count=true" - "dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	"src_bytes=true"	72.113	98.160
16	"src_bytes=true"	"dst_host_count=true" - "dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	72.113	90.294
17	"src_bytes=true"	"dst_host_srv_count=true" - "dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	72.113	90.294
18	"srv_count=true" - "dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	"src_bytes=true"	72.113	98.160
19	"dst_host_srv_count=true" - "dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	"src_bytes=true"	72.113	98.160
20	"dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	"src_bytes=true"	72.113	98.160
21	"src_bytes=true"	"dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	72.113	90.294

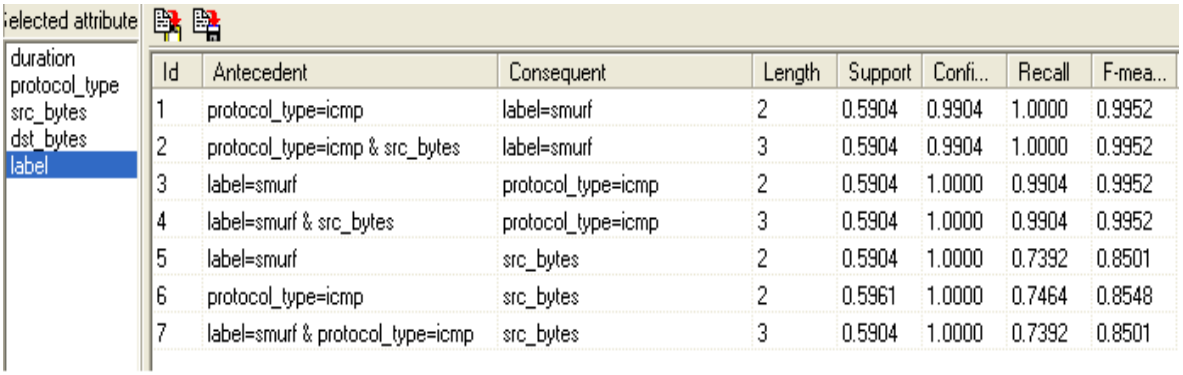
Figure 3 Apriori Result on KDDcup dataset

The rules generated are

"src_bytes=true"	"same_srv_rate=true" - "dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"
"same_srv_rate=true" - "dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	"src_bytes=true"
"same_srv_rate=true" - "src_bytes=true"	"dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"
"dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	"same_srv_rate=true" - "src_bytes=true"
"dst_host_srv_count=true" - "src_bytes=true"	"dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"
"dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	"dst_host_srv_count=true" - "src_bytes=true"
"dst_host_count=true" - "src_bytes=true"	"dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"
"dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	"dst_host_count=true" - "src_bytes=true"
"dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	"srv_count=true" - "src_bytes=true"
"src_bytes=true"	"srv_count=true" - "dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"
"dst_host_same_srv_rate=true" - "dst_host_same_src_port_rate=true"	"count=true" - "src_bytes=true"

Figure 4 Rules generated from Apriori on continous atributes

Tanagra can be considered as a pedagogical tool for learning programming techniques.



Id	Antecedent	Consequent	Length	Support	Confi...	Recall	F-mea...
1	protocol_type=icmp	label=smurf	2	0.5904	0.9904	1.0000	0.9952
2	protocol_type=icmp & src_bytes	label=smurf	3	0.5904	0.9904	1.0000	0.9952
3	label=smurf	protocol_type=icmp	2	0.5904	1.0000	0.9904	0.9952
4	label=smurf & src_bytes	protocol_type=icmp	3	0.5904	1.0000	0.9904	0.9952
5	label=smurf	src_bytes	2	0.5904	1.0000	0.7392	0.8501
6	protocol_type=icmp	src_bytes	2	0.5961	1.0000	0.7464	0.8548
7	label=smurf & protocol_type=icmp	src_bytes	3	0.5904	1.0000	0.7392	0.8501

Figure 5 Rules generated from Association rule mining on selected atributes

The figure above shows the rules generated when duration, protocol_type, src_bytes,dst_bytes and label these attributes are selected. We used KDD cup99 dataset to generate rules which contains 42 attributes having continuous and discrete values. Some more interesting results were found when all the attributes of the KDDdataset were selected. We experimented with continous as well as discrete attributes. When all 41 attributes selected we could generate 5932 rules and 1925 frequent itemsets. We are now exploring the techniques for removing redundancy from the generated ruleset.

5. Conclusion and Future Work

Currently Network-based intrusion detection detects intrusions based on signatures. Evasion technique is used for detecting new attacks based on the information of known attacks. The aim of our framework is not to break the detection of the NIDS. Research is to implmement association rule mining effectively to build a strong and expert intrusion protection system. In this paper we experimented we discussed the experimental results which we carried out during our research using the KDD cup 1999 dataset and applying the signature Apriori algorithm which is well known and widely used for intrusion detection. This framework used to detect the unknown attacks with high accuracy rate and high efficiency.

References

[1] R. Aggrawal, T. Imielinski, A. Swami, “Mining association rules between sets of items in very large databases,” Proceedings of the ACM SIGMOD Conference, 1993.

[2 L.P. Cordella and C. Sansone, “A multi-stage classification system for detecting intrusions in computer networks,” Pattern Anal Applic, 10: 83 – 100, 2007.

[3] L. Portnoy, S. Stolfo, “A geometric framework for unsupe rvised anomaly detection: detecting intrusions in unlabeled da ta,” In: D. Barbara, S. Jajodia (Eds.), Applications of Data Mining in Computer Security, Kluwer, 2002.

[4] D. Newman, “KDD Cup 1999 Data”, The UCI KDD Archive, Information and CS, University Of California, Irvine. Source: http://kdd.ics.uci.edu//databases/kddcup99/kddcup99.html

[5] Lincoln Laboratory, “APRA Intrusion Detection Evaluation”, Massachusetts Institute of Technology. Source: http://www.ll.mit.edu/IST/ideval/index.html

[6] L. Portnoy, E. Eskin, S. Stolfo., “Intrusion detection with unlabeled data using clustering,” Proceedings of ACM CSS Workshop on Data Mining Applied to Security, DMSA-2001.

[7] F.S. Tsai and C.K. Chan (eds), Cyber Security, Pearson
Education, Singapore, 2006.

[8] F. S. Tsai, K. L. Chan, Detecting cyber security threats in weblogs using probabilistic models , in: Intelligence and Security Informatics, vol. 4430, 2007, pp. 46–57.

[9] F. S. Tsai, K. L. Chan, Blog data mining for cyber security threats, in: Data Mining for Business Applications, 2009, pp. 169–182.

[10] Y. Wang, Inyoung Kim, G. Mbaten g, S.-Y. Ho, “A latent class modeling approach to detect network intrusion,” Computer

.Kamini C. Nalavade received the B.E. degree in computer science and engineering from the SGGS, College of engineering and technology, Nanded in 2001 and M.Tech degree in computer engineering from Veermata Jijabai Technological Institute (VJTI), Mumbai in 2007. She is currently PhD student in the department of computer

engineering, VJTI, Mumbai. Her research interest includes intrusion detection, network security, data mining and data privacy. She have published more than 20 papers in International journals and Conferences.

Dr. B. B. Meshram is currently professor and Head of Computer Technology Department of Veermata Jijabai Technological Institute (VJTI), Matunga, Mumbai (INDIA). His areas of interest include Object oriented database management systems, Computer network security and multimedia systems. He has published more than 200 papers in National & International Conferences & refereed Journals. He has submitted more than five patents in his research interest area.