Improved Privacy Preserving decision tree Approach for Network Intrusion Detection

S.Navya Sai, K.KishoreRaju
M.Tech Student, Assistant professor,
S.R.K.R Engineering College

Abstract

With the size of the data increases, privacy preserving plays a vital role in machine learning models. Privacy preserving becomes popular due to its privacy sensitive attributes for data analysis and decision making system.Machine learning has been used and applied in many areas including business development and internet of things. But, machine learning models occur serious problems due to its sensitive information and privacy violation. Privacy preserving data mining protects the sensitive information from disclosure without the permission of data providers. In this model, a novel privacy preserving data mining model was designed to protect the sensitive attributes in the KDD99 dataset.Experimental results show that proposed model has high true positive rate along with sensitive information compared to traditional models.

Keywords-*NID, PRIVACY RESERVING.*

I.INTRODUCTION

A typical IDS can be divided into three functional components [3]: an information source, an analysis engine and a decision maker. These three components can be applied on one single computer, or more commonly be applied on three or more different computers. Therefore the whole IDS can be a host system on one computer or be a distributed system on a local network or even across the Internet. Of the IDS, the first component, the information source is used to monitor the events occurring in a computer system or network. In the duration of monitoring, the information source provides a stream of event records for analysis. This component is working as an event generator. It senses and monitors different data sources, and generates event data that are well formatted and suitable for an analysis engine to do further analysis. The second component, the analysis engine is a key part of an IDS. An IDS relies on the analysis engine to find the signs of intrusions. All the artificial intelligence techniques can be applied to this component. The analysis engine analyzes, filters the information coming from the information source, and discards any irrelevant data in the information, thereby detecting suspicious activities. The analysis engine usually uses a detection policy database for analyzing.

Depending on different intrusion detection approaches and techniques, there could be attack signatures, normal behavior profiles and necessary parameters (for example, thresholds) in the detection policy database.
To overcome limitations of misuse detection, the second approach called anomaly detection is proposed. An anomaly detection based system analyzes the event data
of a training system, and recognizes the patterns of activities that appear to be normal [3]. If a test event lies outside of the patterns, it is reported as a possible intrusion. From this point of view, the anomaly intrusion detection can be considered as a classification technology. We first use some training data to train a model, and get a discriminant function or a classifier. Then we use this discriminant function to test and classify new coming data.
Another popular IDS is called NIDES [2]. As a comprehensive IDS, NIDES implements both misuse and anomaly detection approaches. NIDES implements anomaly detection using "Profiles". These profiles are actually patterns presenting the normal system activities. The IDS monitors and analyzes all kinds of data, such as CPU usage, command usage, and network activities, then generates profiles. The profiles in NIDES are usually updated once a day to reflect new changes. Besides the automatically updating, the system administrators can manually add extra information to the profiles, for example certain date or users information. Including human interference makes the NIDES have the ability of a misuse system. Among all the AI and software computing techniques, SVM is one of the most popular methods used for intrusion detection. SVM could be used in three different ways in the intrusion detection process. First, SVM could be used directly to find the pattern of normal activities of a computer system.

A. Deviation Detection

A task of determining the most significant changes in some key measures of data from previous or expected values.
The most commonly used techniques in

data mining are

- **Artificial neural networks**. Non – linear predictive models that learn through training and resemble biological neural networks in structure.
- **Decision Trees**. Tree-shaped structure that represent sets of decisions. These decisions generate rules for the classification of a datasets. Specific decision tree methods include Classification and Regression Trees (CART) and Chi Square Automatic Interaction Detection (CHAID).
- **Genetic Algorithms.** Optimization techniques that use processes such as genetic combination, mutation, and natural selection in a design bases on the concepts of evolution.
- **Nearest neighbor method**. A technique that classifies each record in a dataset based on a combination of the classes of the k record(s) most similar to it in a historical dataset (where k*1)). Sometimes called the k-nearest neighbor technique.
- **Rule induction**. The extraction of useful if-then rules from data based on statistical significance.

Many of these techniques have been in use for more than a decade in specialized analysis tools that work with relatively small volumes of data. These capabilities are now evolving to integrate directly with industry-standard data warehouse and OLAP platforms.

B. DATASET FILE FORMATS

1) Denials-of Service (DoS) attacks hold the goal limiting or denying services provided to the owner, computer or network. A standard tactic would be to severely overload the targeted system. (e.g. apache, smurf, Neptune, Ping of death, back, mail bomb, udpstorm, SYNflood, etc.).

2) Probing or Surveillance attacks hold the aim of gaining knowledge of this very existence or configuration regarding a computer system or network. Port Scans or sweeping regarding a given IP-address range typically fall in this category. (e.g. saint, ports weep, mscan, nmap, etc.).

3) User-to-Root (U2R) attacks provide the goal gaining root or super-user access throughout the particular computer or system on which the attacker previously had user level access. These would be attempts by way of a non-privileged user in order to increase administrative privileges (e.g. Perl, xterm, etc.).

4) Remote-to-Local(R2L) attack is undoubtedly an attack wherein an individual sends packets to your machine during the internet, which is something user lacks admittance to as a way to expose device vulnerabilities and exploit privileges which a local user is sure to have toward the computer (e.g. xclock, dictionary, guest password, phf, send mail, xsnoop, etc.).

SAMPLE KDD DATASET:

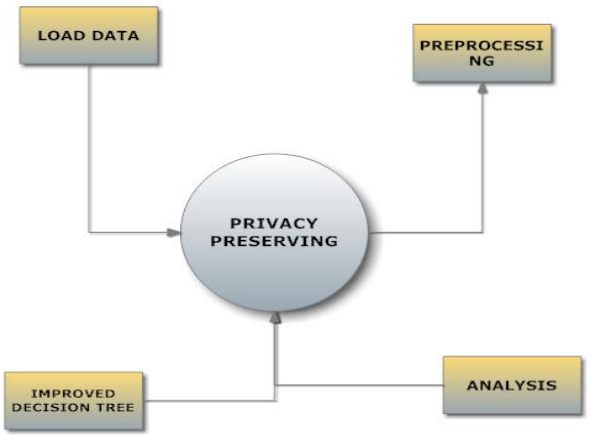
```
@relation kdddata
@attribute 0 numeric
@attribute udp {udp,tcp,icmp}
@attribute private
{private,domain_u,http,smtp,ftp_data,ftp,eco_i,oth
er,auth,ecr_i,IRC,X11,finger,time,domain,telnet}
@attribute SF {SF,RSTR,S1,REJ}
@attribute 105 numeric
@attribute 244 numeric
@attribute 0 numeric
@attribute normal.
{normal.,snmpgetattack.,named.,xlock.,smurf.}
@data
0,udp,private,SF,105,146,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,1,1,0,0,0,0,1,0,0,255,254,1,0.01,0,0,0,0,0,0,n
ormal.
0,udp,private,SF,105,146,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,1,1,0,0,0,0,1,0,0,255,254,1,0.01,0,0,0,0,0,0,n
ormal.
0,udp,private,SF,105,146,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,2,2,0,0,0,0,1,0,0,255,254,1,0.01,0,0,0,0,0,0,s
nmpgetattack.
```

PROBLEM STATEMENT

Since the same RDT code can be used for multiple data mining tasks, we focus on classification for ease of discussion. The basic problem in distributed classification is to train a classifier from the distributed data and then classify each new instance. For distributed decision tree classification, the objective is to create a decision tree classifier from the distributed data. In the privacy-preserving case, the additional constraint is that the process of building the classifier, or of classifying an instance should not leak any additional information beyond what is learned from the result (and the local input). Assuming the global data set D ðT;RÞ, where T represents the global set of transactions, and R represents the global schema.

II. PROPOSED SYSTEM

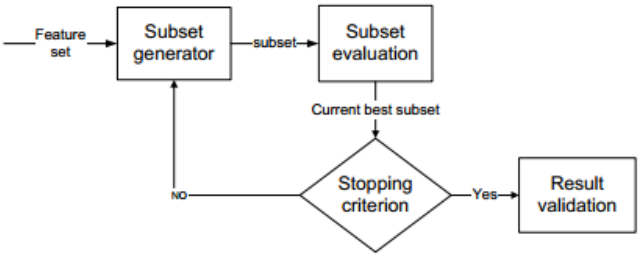
The Multiple Classifiers (MCS) approach was suggested based on pattern recognition distinct feature representation and tested with different fusion rules. The reported results proved that the MCS approach provides a better false alarm generation than that provided by an individual classifier trained on the overall feature set. Among the fusion rules, the dynamic classifier selection technique provided the best results. The fusion of multiple classifiers achieves a better trade off than that provided by individual classifiers between generalization abilities and false alarm generation. proposed a method 'bag of system calls' and experimented misuse and anomaly detection results with other machine learning techniques. With the feature representation as input, the performance has been compared with several machine learning techniques for misuse detection. The results showed that simple 'bag of system calls' combining standard machine learning and clustering techniques is effective and often performs better than other approaches.



DATALOAD: This feature is used to load the dataset for privacy preserving purpose.

PREPROCESSING:

Feature selection research has found applications in many fields where large volumes of data present challenges to effective data analysis and processing. As data evolves to be ubiquitous and abundant, new challenges arise everyday and expectations of feature selection are also elevated. Feature selection algorithms have two main components: feature search and feature subset evaluation. Feature search strategies have been widely used for searching feature space. An exhaustive search would certainly find the optimal solution; however, for a dataset of N features, a search on 2^N possible feature combinations is obviously computationally impractical. More realistic search strategies have been studied to make the problems more tractable.



Feature Selection Process

Algorithm 3.4 Extended Euclidean algorithm

Input: Integers a and b
Output: Integers x, y, and d, where $d = \gcd(a, b) = ax + by$

Set $d_0 = a$ **Set** $x_0 = 1$ **Set** $y_0 = 0$
Set $d_1 = b$ **Set** $x_1 = 0$ **Set** $y_1 = 1$
While $d_1 \neq 0$ **Do**
  Set $q = \lfloor d_0/d_1 \rfloor$
  Set $d_2 = d_1$ **Set** $x_2 = x_1$ **Set** $y_2 = y_1$
  Set $d_1 = d_0 - qd_1$
  Set $x_1 = x_0 - qx_1$
  Set $y_1 = y_0 - qy_1$
  Set $d_0 = d_2$ **Set** $x_0 = x_2$ **Set** $y_0 = y_2$
End While

Return $[d, x, y] = [d_0, x_0, y_0]$

PROPOSED ALGORITHMS:

- N, number of examples.
- Ai, continuous attributes.
- Cj, class values in training set.

. Global Threshold value
Output: Interval borders in Ai'
Procedure:
1. for each continuous attribute Ai in training dataset do
2. Do normalize the attribute within 0-1 range
3. Sorting the values of continuous attribute Ai in ascending order.
4. for each class Cj in training dataset do
5. Find the minimum (Minvalue) using StdDev attribute value of Ai for Cj•
6. Find the maximum (Max) attribute value of Ai for Cj.
7. endfor
8. Find the cut points in the continuous attributes values based on the Min and Max values of each class Cj.

Best Cutpoint range measure:
9. Find the conditional probability P(Cj/A) on each cut point and select the cut point with maximum probability value.

Stopping criteria:
10. If the cut point using the maximum probability value is exist and satisfies the global threshold value then it can be taken as an interval border else consider the next cut point, where information gain value and global threshold value satisfy the same point.
12. endfor

III. DECISION TREE ALGORITHM

The C4.5 Algorithm is the extension of ID3 algorithm .It used a mechanism of learning from large datasets .The attribute selection of the algorithm is based on an assumption the complexity of decision tree and the amount of information is represented by given attribute are closely related C4.5 expands the classify range to digital attributes. That metric standard of two-class entropy ,the most of the algorithm is based on the information entropy which is contained by produced nodal points of decision tree is least [9].The so called entropy is representative of degree of disorder of objects in the system. It is easy to understand that the smaller entropy the smaller disorder .In the other word the more sequential in the record collection, the more consistent .This is the target we seek; too .Suppose the set S is a training sample, the formula of entropy as follows:

Instance of previously-unseen class encountered. Again, C4.5 creates a decision node higher up the tree using the expected value[10].design the degree of balance coefficient of a certain attribute as

Algorithm: Generate_decision_tree.

Input: The training samples, *samples*,
set of candidate attributes, *attribute-list*.
Output: A decision tree.
Method:
(1) create a node *N*;
(2) if *samples* are all of the same class, C then
(3) return *N* as a leaf node labeled with the class C;
(4) if *attribute-list* is empty then
(5) return *N* as a leaf node labeled with the most common class in *samples*;//majority voting
(6) select *test-attribute*, the attribute among *attribute-list* with the highest information gain;
(7) label node N with *test-attribute*;
(8) for each known value *ai* of *test-attribute;*
(9) grow a branch from node N for the condition *test-attribute* = *ai;*
(10) let *si* be the set of samples in *samples* for which *test-attribute* = *ai;* // a partition
(11) if *si* is empty then
(12) attach a leaf labeled with the most common class in *samples*;
(13) else attach the node returned by Generate_decision_tree *si, attribute-listtest-attribute);*

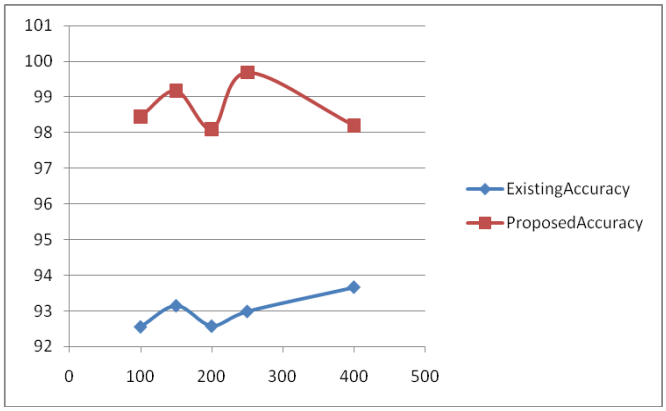
RESULTS:

dst_host_srv_rerror_rate > 0.98 &&
dst_host_srv_count <= 3 &&
dst_host_rerror_rate > 0.05 ==> anomaly
dst_host_srv_count > 155 ==> normal
flag = S0 &&
dst_host_same_src_port_rate <= 0.04 ==> anomaly
flag = REJ ==> normal
flag = RSTR ==> normal
flag = SH ==> anomaly
serror_rate <= 0.02 &&
dst_bytes <= 2638 &&
dst_bytes <= 2198 &&
dst_host_srv_diff_host_rate <= 0.05 &&
service = other &&
src_bytes > 92 ==> normal
serror_rate <= 0.02 &&
num_root <= 0 &&
dst_bytes > 2638 ==> normal
hot > 15 ==> anomaly
num_root <= 0 &&
serror_rate > 0.02 ==> normal
num_root > 0 ==> normal
service = finger &&
src_bytes > 1 ==> normal

dst_host_serror_rate <= 0.61 &&
wrong_fragment <= 0 &&

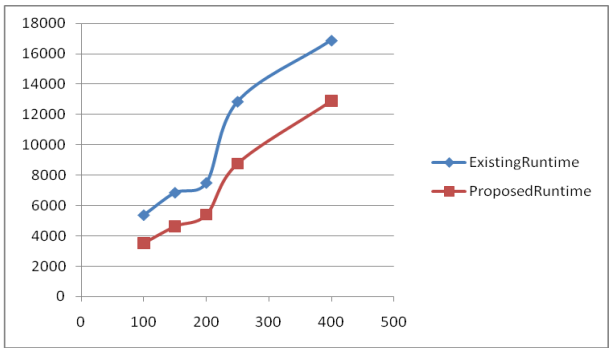
service = eco_i &&
src_bytes > 20 ==> normal
wrong_fragment <= 0 &&
dst_host_serror_rate <= 0.29 &&
service = ftp_data &&
dst_host_srv_diff_host_rate > 0.05 ==> anomaly
service = ftp_data ==> normal
service = other ==> anomaly
wrong_fragment <= 0 &&
service = domain_u ==> normal
wrong_fragment <= 0 &&
service = private ==> anomaly
wrong_fragment > 0 ==> anomaly
service = auth ==> normal
num_failed_logins <= 0 &&
dst_host_srv_serror_rate <= 0 &&
flag = SF &&
src_bytes > 20 &&
dst_host_serror_rate <= 0 ==> normal
protocol_type = tcp &&
dst_host_rerror_rate > 0.06 ==> anomaly
dst_host_srv_rerror_rate <= 0.02 ==> anomaly
: normal
Number of Rules : 32

DataSamples	ExistingAccuracy	ProposedAccuracy
100	92.56	98.45
150	93.15	99.17
200	92.58	98.1
250	92.99	99.69
400	93.67	98.19



Above table and figure indicates the proposed and existing system accuracy measures.

DataSamples	ExistingRuntime	ProposedRuntime
100	5366	3533
150	6846	4637
200	7495	5399
250	12855	8766
400	16888	12886



Above table and figure indicates the proposed and existing system Runtime measures.

IV. CONCLUSION AND FUTURE SCOPE

In this paper, we have proposed an efficient, scalable improved privacy preserving based decision tree construction algorithm which results in high processing speed and small scale. The key challenge lies in preventing the data miners from combining copies at different trust levels to jointly reconstruct the original data more accurate than what is allowed by the data owner. We address this challenge by properly correlating noise across copies at different trust levels. We prove that if we design the noise covariance matrix to have corner-wave property, then data miners will have no diversity gain in their joint reconstruction of the original data. We verify our claim and demonstrate the effectiveness of our solution through numerical evaluation. From the experimental evaluation, we have got a promising result, since our proposed algorithm outperforms the Existing decision tree based privacy preserving techniques in terms of accuracy and execution time.

V. REFERENCES

[1] D. Agrawal and C.C. Aggarwal, “On the Design and Quantification of Privacy Preserving Data Mining Algorithms,” Proc. 20th ACM SIGMOD-SIGACT-SIGART Symp. Principles of Database Systems (PODS ’01), pp. 247-255, May 2001. [2] R. Agrawal and R. Srikant, “Privacy Preserving Data Mining,” Proc. ACM SIGMOD Int’l Conf. Management of Data (SIGMOD ’00), 2000. [3] K. Chen and L. Liu, “Privacy Preserving Data Classification with Rotation Perturbation,” Proc. IEEE Fifth Int’l Conf. Data Mining, 2005. [4] Z. Huang, W. Du, and B. Chen, “Deriving Private Information From Randomized Data,” Proc. ACM SIGMOD Int’l Conf. Management of Data (SIGMOD), 2005. [5] F. Li, J. Sun, S. Papadimitriou, G. Mihaila, and I. Stanoi, “Hiding in the Crowd: Privacy Preservation on Evolving Streams Through Correlation Tracking,” Proc. IEEE 23rd Int’l Conf. Data Eng. (ICDE), 2007.

[6] G. Jagannathan, K. Pillaipakkamnatt, and R.N. Wright, “A Practical Differentially Private Random Decision Tree

Classifier," Proc. IEEE Int'l Conf. Data Mining Workshops (ICDMW '09), pp. 114-121, 2009. [7] J. Vaidya, C. Clifton, M. Kantarcioglu, and A.S. Patterson, "Privacy-Preserving Decision Trees over Vertically Partitioned Data," ACM Trans. Knowledge Discovery from Data, vol. 2, no. 3, pp. 1-27, 2008.

[8] M. Kantarcioglu and C. Clifton. Privately computing a distributed k-nn classifier. In J.-F. Boulicaut, F. Esposito, F. Giannotti, and D. Pedreschi, editors, PKDD, volume 3202 of Lecture Notes in Computer Science, pages 279–290. Springer, 2004.

[9] J. Domingo-Ferrer and V. Torra, "A Quantitative Comparison of Disclosure Control Methods for Microdata," Confidentiality, Disclosure and Data Access: Theory and Practical Applications for Statistical Agencies, P. Doyle, J. Lane, J. Theeuwes, and L. Zayatz, eds., pp. 111-134, Amsterdam: North-Holland, 2001. [10] J. Domingo-Ferrer and V. Torra, "Ordinal, Continuous and Heterogeneous k-Anonymity through Microaggregation," Data Mining and Knowledge Discovery, vol. 11, no. 2, pp. 195-212, 2005