International Journal of Research Studies in Computer Science and Engineering (IJRSCSE)
Volume 2, Issue 3, March 2015, PP 25-29
ISSN 2349-4840 (Print) & ISSN 2349-4859 (Online)
www.arcjournals.org

Detection of Probe Attacks Using Machine Learning Techniques

Ch.Ambedkar
Assistant Professor, Department of C.S.E
SRKIT, Vijayawada, India
rahul59985@gmail.com

V. Kishore Babu
Assistant Professor, Department of C.S.E.
SRKIT, Vijayawada, India
kishorebabu83@yahoo.co.in

Abstract: *In recent years, the number of attacks on the computer networks and its components are getting increasing. To protect from these attacks various Intrusion detection techniques have been used. Intrusion Detection System (IDS) is a system which collects and analyzes the information from the network to identify various attacks made against the components of a network. In this paper we presented a comprehensive analysis on Probe attacks, by applying various popular machine learning techniques such as Naïve Bayes, SVM, Multilayer Perceptron, Decision Trees etc. we used KDDcup99 data set to build the model. In this paper we proposed three layer architecture for detection of probe attacks. Principal Component Analysis is used for dimensionality reduction. We also removed duplicate samples from the training data set. Finally, we compared the performance of each classifier with the help of a line chart.*

Keywords: *decision Tree, Intrusion Detection, Multilayer Perceptron, Probe Attacks, PCA.*

1. INTRODUCTION

Intrusion Detection System (IDS) is an active process or device that analyzes system and network activity for unauthorized activity [1]. An ID is hardware or software or a combination of both which is used to monitor a system or network of systems against any malicious or unauthorized activities [1]. Intrusion Detection Systems (IDSs) are used to improve network security. An ID improves the security of the network by identifying, assessing, and reporting unauthorized network activities. IDS are categorized into two classes: network-based and host-based. Network based Intrusion Detection Systems analyses network packets retrieved from the network. Host-based Intrusion Detection System analyses system calls generated by individual hosts [2].The data flows through a network is very large and it is difficult to analyze and detect the attacks using traditional methods. Today we have number of Machine learning techniques available which are very useful for analyzing the data and detecting the attacks. In this paper we have used various machine learning techniques for network intrusion detection [3].

The rest of the paper is organized as follows: section II describes the detailed analysis of the KDD cup 99 dataset. The detailed description of the Probe Attacks data set is given in section III. The proposed model and the process of building a classification model using RapidMiner is given in section IV. Experimental analysis and results are discussed in section V. Finally in section VI the conclusions and future work have been mentioned.

1.1. KDD Cup 99 Data Set Description

KDD CUP 99 data set [4] has been the mostly used data set for evaluation of anomaly detection methods. KDD training data set consists of approximately 4,900,000 samples with 41 features and each sample is labeled as either normal or attack [3][5]. KDD99 is actually composed of three datasets. The largest data set is called "Whole KDD", which consists of 4 million samples. A subset is created from the original data set by randomly selecting the samples from it.

TableI. *Categories of Attacks of KDD Cup 99 Data*

Category of attack	*Attack name*
Denial of Service Attacks	*Back, land, Neptune, pod, smurf, teardrop*
User to Root Attacks	*Buffor_overflow, loadmodule, perl, rootkit*
Remote to Local Attacks	*ftp_write, guess_passwd, imap, multihop, phf, spy, warezclient*
Probes	*Satan, ipsweep, nmap, portsweep*

This data set is called "10% KDD" data set which is used to train the IDS [3][5]. Each sample of the original KDD'99 dataset is classified as one of the following categories [3][6].

- Normal: not an attack
- DOS: denial of service
- R2L: unauthorized access from a remote to local machine.
- U2R: unauthorized access to local super user.

- Probe: a probe attack scans the network to gather the information of computers to identify the vulnerabilities.

The following Table I shows the possible types of attacks in each category.

TABLEII. *Total Number of Attributes Given in KDD Cup 99 Dataset*

Feature Index	*Feature Name*
1	*Duration*
2	*protocol_type*
3	*Sevice*
4	*Flag*
5	*Scr-bytes*
6	*dst_bytes*
7	*Land*
8	*wrong_fragment*
9	*Urgent*
10	*Hot*
11	num_failed_logins
12	logged_in
13	num_compromised
14	root_shell
15	su_attempted
16	num_root
17	num_file_creations
18	*num_shells*
19	*num_access_files*
20	*num_outbound_cmds*
21	*is_host_login*
22	*is_guest_login*
23	*Count*
24	*srv_count*
25	*serror_rate*
26	*srv_serror_rate*
27	*rerror_rate*
28	*srv_rerror_rate*
29	*same_srv_rate*
30	*diff_srv_rate*
31	*srv_diff_host_rate*
32	*dst_host_count*
33	*dst_host_srv_count*
34	*dst_host_same_srv_rate*
35	*dst_host_diff_srv_rate*
36	*dst_host_same_src_port_rate*
37	*dst_host_srv_diff_host_rate*
38	*dst_host_serror_rate*
39	*dst_host_srv_serror_rate*
40	*dst_host_rerror_rate*
41	*dst_host_srv_rerror_rate*

The KDD CUP 99 Dataset contains 41 features. All these features are listed in the Table II.

2. DATA SET DESCRIPTION OF PROBE ATTACKS

Shows the major attacks in recent years, an increasing number of programs have been used by the attackers to scan a network of computers to gather information or find known vulnerabilities [7]. In probe attacks the attacker scans the network to gather the information of computers to identify the vulnerabilities. These network probes are quite useful to an attacker who is staging a future attack. An attacker with a map of which machines and services are available on a network can use this information to look for weak points. Some of these scanning tools (satan, saint, mscan) enable even a very unskilled attacker to very quickly check hundreds or thousands of machines on a network for known vulnerabilities [7]. The following Table III probe data set. The training data set contains total 4107 records of four different kinds of probe attacks.

TableIII. *Attacks in Probe Data Set*

Attack Name	*Attacks in Data Set*
IPsweep	***1247***
Nmap	***231***
Portsweep	***1040***
Satan	***1589***
Total	***4107***

Ipsweep: An Ipsweep attack is a surveillance sweep which is used to determine which hosts are listening on a network.

Mscan: Mscan is a probing tool that uses both DNS zone transfers and/or brute force scanning of IP addresses to locate machines, and test them for vulnerabilities [8].

Nmap: Nmap is a general-purpose tool for performing network scans.

Saint: SAINT is the Security Administrator’s Integrated Network Tool [9].

Satan: SATAN is an early predecessor of the SAINT. While SAINT and SATAN are quite similar in purpose and design, the particular vulnerabilities that each tools checks for are slightly different [10].

3. DETECTION OF PROBE ATTACKS

This section describes the proposed model for the detection of Probe attacks. The architecture of the proposed model is given in Fig. 1. This architecture consists of three Layers, in each layer a set of activities have been carried out.

Layer I: Initially KDD CUP 99 data set with 4,94,020 samples is taken, from which the Probe data set is created. Probe data set contains 4107 samples with 41 attributes. The data set description of Probe data set is given in Table III.

Layer II: Principal Component Analysis (PCA) is applied for Feature Selection. Before applying PCA the input data is normalized, so that each attribute falls in the same range [11]. PCA computes c orthonormal vectors that provide a basis for the normalized input data. These vectors are referred **to** ***as the principal components. The principal*** components are sorted in order of decreasing significance or strength [11]. By using PCA a subset of 18 attributes out of 41 attributes have been selected. The set of attributes selected are listed in Table IV. After applying the PCA we have got dimensional reduced data set with 18 features with so many duplicate samples. In order to remove duplicate samples from the data set, we applied an operator called Remove Duplicates in RapidMiner. After removing the duplicates the resulting data set now consists of only 1800 samples. At the end of layer II, the modified training as well as testing data sets are created.

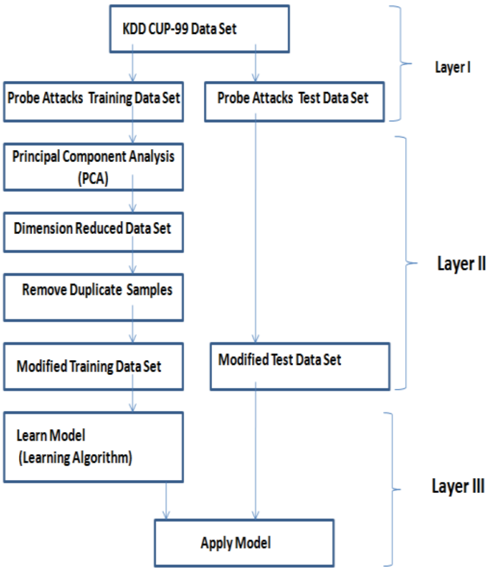
KDD CUP-99 Data Set
Layer I
Probe Attacks Training Data Set
Probe Attacks Test Data Set
Principal Component Analysis (PCA)
Dimension Reduced Data Set
Remove Duplicate Samples
Layer II
Modified Training Data Set
Modified Test Data Set
Learn Model (Learning Algorithm)
Layer III
Apply Model

Fig1. *The architecture of proposed model*

Layer III: In this layer training and testing is done. For training different classification algorithms have been used. The classification algorithms used are Naïve Bayes, SVM, k-NN, Decision Tree, Random Forest, ID3, CHAD, Neural Net, MLP, Perceptron, *etc. The performance of these algorithms is discussed in section V.*

TableIV. *Selected Attributes For Probe Attack Analysis*

Index Number	***Attribute Name***
2,3,4	***Protocol_type, Service, Flag***
23,27,28	***Count, Rerror_rate, Srv_rerror_rate***
29,30,31	***Same_srv_rate, Diff_srv_rate, Srv_diff_host_rate***
32,33,34	***Dst_hosy_count, Dst_host_srv_count, dst_host_same_srv_rate***
35,36,37	***dst_host_diff_srv_rate, dst_host_same_src_port_rate, dst_host_srv_diff_host_rate***
38,40,41	***Dst_host_serror_rate, dst_host_rerror_rate, dst_host_srv_rerror_rate***

4. Experimental Analysis

The performance of the model can be evaluated using various metrics. Some of the metrics are explained bellow [3] [13].

- True Positive (TP): The number of positive samples correctly predicted by the classification model.
- False Negative (FN): The number of positive samples wrongly predicted as negative by the classification model.
- False Positive (FP): The number of negative samples wrongly predicted as positive by the classification model.
- True Negative (TN): The number of negative samples correctly predicted by the classification model.
- True Positive Rate (TPR): The fraction of positive samples predicted correctly by the model. TPR is also called as sensitivity.

  TPR=TP / (TP+FN) (1)
- True Negative Rate (TNR): The fraction of negative samples predicted correctly by the model. TNR is also called specificity.

  TNR=TN / (TN+FP) (2)
- False Positive Rate (FPR): The fraction of negative samples predicted as a positive class.

  FPR=FP / (TN+FP) (3)
- ***False Negative Rate (FNR): The fraction of positive samples predicted as a negative class.***

  FNR=FN / (TP+FN) (4)
- Precision: The fraction of records that actually turns out to be positive in the group the classifier has declared as a positive class.

  Precision =TP / (TP+FP) (5)

  High precision indicates that the classifier has committed low number of false positive errors.
- Recall: The fraction of positive samples correctly predicted by the classifier. The value of recall is equivalent to the True Positive Rate.

  Recall=TP / (TP+FN) (6)

  Classifiers with large recall have very few positive samples misclassified as the negative class.
- Accuracy: Accuracy is the ratio between the number of correct predictions to the total number of predictions.

Accuracy=(TP+TN)/(TP+TN+FP+FN) (7)

Rule Model

We used rule induction as one of the classification algorithm. Rule Induction generates a pruned set of rules. A rule based classifier is a technique for classifying records using a collection of " if… then…" rules [13]. The following are the set of rules generated by Rule Induction for detecting probe attacks.

```
if dst_host_same_src_port_rate ≤ 0.005 then satan.
if dst_host_count ≤ 71.500 and dst_host_rerror_rate > 0.490 then ipsweep.
if rerror_rate > 0.095 then portsweep.
if dst_host_srv_diff_host_rate > 0.295 then ipsweep.
if dst_host_count > 254.500 then satan.
if service = private and dst_host_same_src_port_rate > 0.785 then nmap.
if dst_host_srv_count > 58.500 and dst_host_srv_count ≤ 201 and srv_diff_host_rate > 0.500 then nmap.
if dst_host_same_srv_rate > 0.750 and dst_host_srv_count ≤ 202.500 and dst_host_srv_count > 51 then
ipsweep.
if dst_host_srv_count > 29 and dst_host_srv_count ≤ 42.500 and srv_diff_host_rate > 0.500 then nmap.
if dst_host_same_srv_rate > 0.750 and dst_host_srv_count ≤ 47 and count ≤ 23.500 and
dst_host_srv_count ≤ 3.500 then ipsweep.
if dst_host_srv_count > 48.500 and dst_host_count ≤ 3.500 and srv_diff_host_rate > 0.500 then nmap.
if dst_host_same_srv_rate > 0.750 then ipsweep.
if flag = SF and dst_host_diff_srv_rate > 0.035 then satan.
if dst_host_rerror_rate ≤ 0.295 then nmap.
else portsweep.

correct: 1765 out of 1800 training examples.
```

The following Table V contains experimental results for various classification models. We have considered three measurements such as Precision, Recall, and Accuracy for each classification model.

TableVI. *Performance Comparison of Different Classification Algorithms*

Classifier	*Metric*	*Type of Attack*			
		Buffer_Overflow	*loadmodule*	*Perl*	*rootkit*
Naïve Bayes	***Accuracy***	***93.89***			
	Precision	***97.09***	***97.86***	***75.54***	***90.33***
	Recall	***93.91***	***93.53***	***95.86***	***93.97***
k-NN	*Accuracy*	*91.73*			
	Precision	***85.74***	***93.22***	***84.09***	***100***
	Recall	***100***	***99.89***	***51.03***	***77.87***
SVM	*Accuracy*	*98.11*			
	Precision	***98.35***	***97.56***	***96.12***	***100***
	Recall	***97.66***	***100***	***85.52***	***99.14***
Decision tree	*Accuracy*	*99.50*			
	Precision	***99.76***	***99.77***	***96***	***100***
	Recall	***98.59***	***100***	***99.31***	***99.43***
Random Forest	*Accuracy*	*85.06*			
	Precision	***80.19***	***84.29***	***100***	***100***
	Recall	***99.53***	***99.89***	***2.07***	***64.37***
ID3	*Accuracy*	*92.06*			
	Precision	***75.18***	***99.77***	***100***	***100***
	Recall	***100***	***96.82***	***22.07***	***99.43***
CHAID	*Accuracy*	*91.8*			
	Precision	***76.94***	***98.52***	***83.33***	***100***
	Recall	***100***	***98.18***	***20.69***	***95.40***
Random Tree	*Accuracy*	*89.67*			
	Precision	***81.84***	***96.47***	***41.67***	***93.37***
	Recall	***93.91***	***96.03***	***20.69***	***97.13***
Rule induction	*Accuracy*	*96.78*			
	Precision	***96.89***	***98.97***	***99.22***	***90.77***
	Recall	***9.85***	***98.41***	***87.59***	***98.85***

Neural Net	***Accuracy***	***99.44***			
	Precision	***99.53***	***99.77***	***95.92***	***100***
	Recall	***98.59***	***100***	***97.24***	***100***
Auto MLP	***Accuracy***	***99.06***			
	Precision	***99.53***	***99.21***	***94.44***	***100***
	Recall	***98.59***	***99.77***	***93.79***	***100***

Finally the comparison of performance of different classification models is carried out. Fig.2 shows the performance comparisons. X-axis represents classifier and Y-axis represents accuracy of each algorithm.

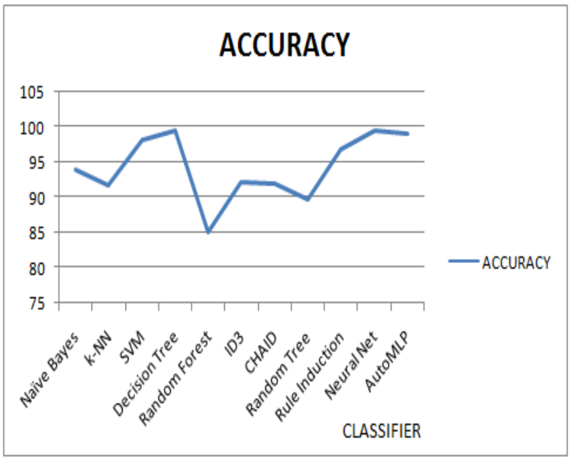

Fig2. *Comparisons of accuracy of classification algorithms*

5. Conclusion

In this paper, we have applied various data mining techniques such as Decision Tree, Naïve Bayes, SVM, Random Forest, Neural Net, AutoMLP, Random Tree, Rule Reduction, ID3, CHAID,k-NN on Probe data set. All most all the classification algorithms show high accuracy. Among these techniques Neural Net gave highest accuracy of 99.44%. AutoMLP gave second highest accuracy of 99.06%. The performances of these algorithms are clearly shown in the graph. In the future we will use ensemble methods to improve the accuracy of the system. In addition to these we will also use fuzzy techniques, genetic algorithms to improve the accuracy of the system.

References

[1] D.P. Vinchurkar, A. Reshamwala, “A Review of Intrusion Detection System using Neural Network and Machine Learning Technique”, International Journal of Engineering Science and Innovative Technology (IJESIT), Vol 1, Issue 2, Nov 2012.

[2] Li Tain, “Research on Network Intrusion Detection System Based on Improved K-means Clustering Algorithm”, Computer Science – Technology and Applications, 2009.IFCSTA ’09. International Forum.

[3] T.V.N Lakshmi, V.K Babu, “ Detection of User to Root Attacks using Machine Learning Techniques”, International Journal of Advanced Engineering and Global Technology (IJAEGT), Vol. 3, Issue 3, Mar 2015.

[4] KDD cup 99 dataset, available at http://kdd.ics.uci.edu/dataset/kddcup99/kddcup 99.html

[5] Mathod T., E.Bhagheri, Wei Lu, and A.A. Ghorbani,”A Detailed Analysis of the KDD CUP 99 Data Set”, p.2, 2009.

[6] R. Shanmugavadivu, .N.Nagarajan, “Network Intrusion Detection System using Fuzzy Logic”, Indian Journal of Computer Science and Engineering ,Vol. 2 No. 1, pp-101-111,2011.

[7] Simson Garfinkel and Gene Spafford. Practical Unix & Internet Security. O’Reilly & Associates, Inc., 101 Morris Street, Sebastopol CA, 95472, 2ndedition, April 1996.

[8] CERT Incident Note. http://www.cert.org/ incident_notes/IN-98.02.html. July 2, 1998.

[9] J. Han, M. Kamber, “Data Mining Concepts and Techniques”, Morgan Kaufmann Publishers, Elsevier.

[10] P. N. Tan, M. Steinbach, V. Kumar,” Introduction to Data Mining”, Pearson Education.