

INTRUSION DETECTION SYSTEM USING K MEANS AND NAIVE BAYES CLASSIFICATION

POONAM RANI, ANJU GANDHI

Department of Computer Engineering, India

Abstract

The rapid development of business and other transaction systems over the Internet makes computer security a critical issue. In this paper, we present an overview of our research in intrusion detection systems (IDSs) using K-means & Naïve bayes. We focus on issues related to deploying a data mining-based IDS in a real time environment selecting important features from input data lead to a simplification of the problem, faster and more accurate detection rates. We describe our approaches into three steps: First step will define Intrusion detection system architecture. Second step will define the KDD Cup 1999 dataset used for Train data or extract data, test data. Third step will introduce K-Mean algorithm one of the most important clustering algorithms and Naïve Bayesian classifiers which is highly dependent on the assumptions about the behavior, the accuracy, efficiency, and usability of the target system. To improve accuracy, data mining programs are used to analyze audit data and extract features that can distinguish normal activities from intrusions. The investigation revealed many interesting results about the protocols and attack types preferred by the hackers for intruding the networks.

Keywords: Attack, KDD, Intrusion detection, NIDS, K-means, Protocols.

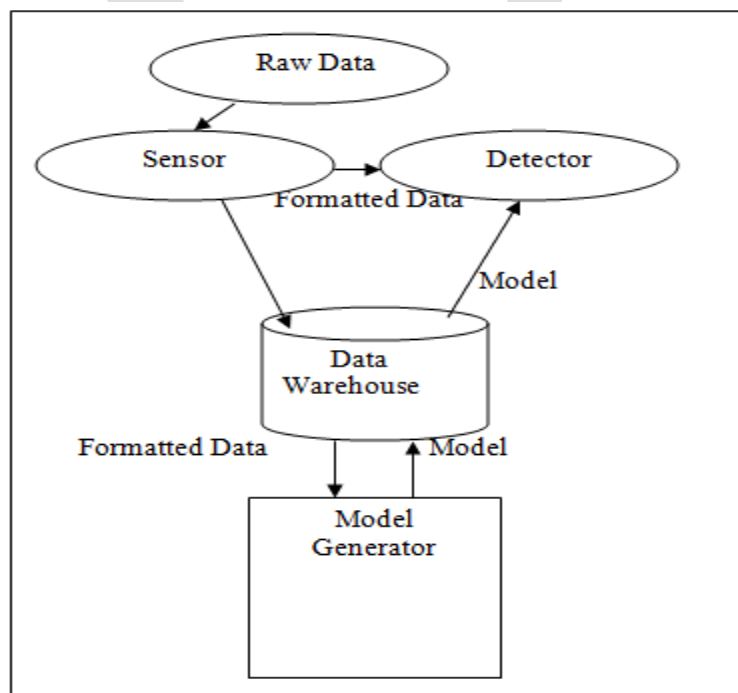
1. Introduction

Intrusion detection System monitors the violation of management and security policy and malicious activities in the computerized network. Traditional methods for intrusion detection are simple which are based on extensive knowledge of signatures of known attacks and Monitored events are matched against the signatures to detect intrusions. These methods extract features from various audit streams, and detect intrusions by comparing the new

generated values to a exist values of attack signatures provided by human experts. But now a days different kinds of attacks generated which can't detect without any tools or techniques. We use clustering and classification algorithms for detecting different kinds of attacks[5]. If we detect the attack once it comes into the network, a response can be initiated to prevent or minimize the damage to the system. It also helps prevention techniques improve by providing information about intrusion techniques and what kind of of attack has occurred. IDS can use different data sources which are the inputs to the system: system logs, network packets, etc. If an IDS monitors activities on a host and detects violations on the host, it is called host-based IDS (HIDS). [1]An IDS that monitors network packets and detects network attacks is called network-based IDS (NIDS).

2. Intrusion detection System Architecture

The overall system architecture is designed to support a data mining-based IDS with the properties described throughout this paper. As shown in Figure, the architecture consists of sensors, detectors, a data warehouse, and a model generation component. This architecture is capable of supporting not only data gathering, sharing, and analysis, but also data archiving and model generation and distribution. The system is designed to be independent of the sensor data format and model representation. A piece of sensor data can contain an arbitrary number of features.[2]

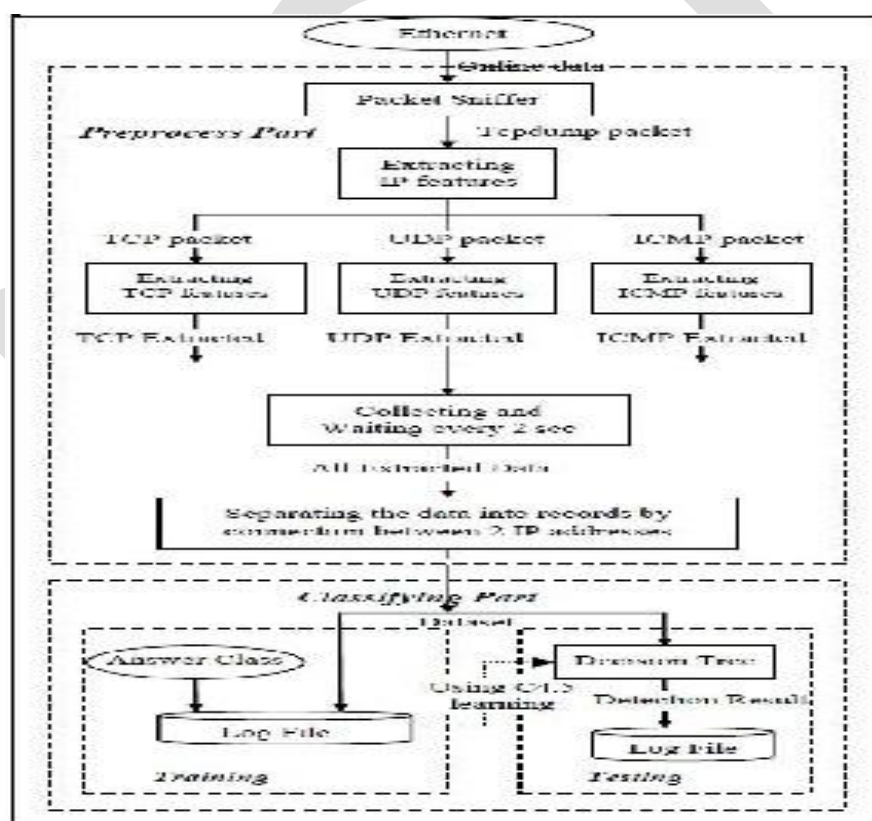


3. The KDD Cup 99 dataset

The KDD Cup 99 dataset has been the point of attraction for many researchers in the field of intrusion detection from the last decade. Many researchers have contributed their efforts to analyze the dataset by different techniques. The dataset was a collection of simulated raw TCP dump data over a period of nine weeks on a local area Network.[5] 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 data sets. 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

The following diagram shows the sources of input data ,extraction and classification.



4. K- Mean Clustering Algorithm

The clustering algorithm divides the training data into K clusters, but does not determine if a cluster reflect time intervals of normal or anomalous traffic.[7] An

essential problem of the K-means clustering method is to define an appropriate number of clusters K. As initial value, we chose $K = 2$, assuming that normal and anomalous traffic in the training data form two different clusters. Obviously, a different number of clusters may result in better clusters, e.g. if the considered service already shows distinct periods of very low and very high traffic volume under normal conditions .

Steps of k-mean algorithm [15]:

- 1) Define the number of clusters K.
- 2) Initialize the K cluster centroids. This can be done by arbitrarily dividing all objects into K-Clusters, computing their centroids, and verifying that all centroids are different from each other. Alternatively, the centroids can be initialized to K arbitrarily chosen, different objects.
- 3) Iterate over all objects and compute the distances to the centroids of all clusters. Assign each object to the cluster with the nearest centroid.
- 4) Recalculate the centroids of both modified clusters.
- 5) Repeat step 3 until the centroids do not change any more.

5. Naive Bayes Classification

Naïve Bayesian classifiers assume that the effect of an attribute value on a given class is independent of the values of the other attributes. This assumption is called class conditional independence. It is made to simplify the computations involved and, in this sense, is consider “Naive”. Naïve Bayesian classifiers allow the representation of dependencies among subsets of attributes. [6] Though the use of Bayesian networks has proved to be effective in certain situations, the results obtained are highly dependent on the assumptions about the behavior of the target system, and so a deviation in these hypotheses leads to detection errors, attributable to the model considered [6].

The naive Bayesian classifier works as follows:

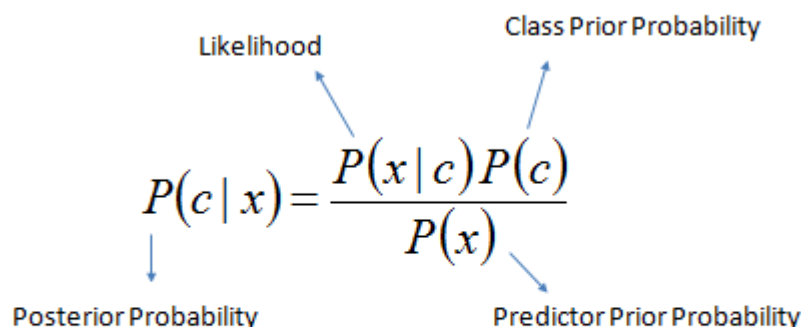
1. Let T be a training set of samples, each with their class labels. There are k classes, Each sample is represented by an n-dimensional vector $X = \{ \}$, depicting n measured values of the n attributes, , respectively.

2. Given a sample X, the classifier will predict that X belongs to the class having the highest a posteriori probability, conditioned on X. That is X is predicted to belong to the class.
3. As $P(X)$ is the same for all classes, only need be maximized. If the class a priori probabilities are not known, then it is commonly assumed that the classes are equally.
4. Given data sets with many attributes, it would be computationally expensive to compute. In order to reduce computation in evaluating. The naïve assumption of class conditional independence is made[5]. This presumes that the values of the attributes are conditionally independent of one another, given the class label of the sample.
5. In order to predict the class label of X, is evaluated for each class. The classifier predicts that the class label of X is if and only if it is the class that maximizes.

Algorithm

The Naive Bayesian classifier is based on Bayes' theorem with independence assumptions between predictors. A Naive Bayesian model is easy to build, with no complicated iterative parameter estimation which makes it particularly useful for very large datasets. Despite its simplicity, the Naive Bayesian classifier often does surprisingly well and is widely use because it often outperforms more sophisticated classification methods.

Bayes theorem provides a way of calculating the posterior probability, $P(c|x)$, from $P(c)$, $P(x)$, and $P(x|c)$. Naive Bayes classifier assume that the effect of the value of a predictor (x) on a given class (c) is independent of the values of other predictors. This assumption is called class conditional independence.

$$P(c | x) = \frac{P(x | c)P(c)}{P(x)}$$


$$P(c | X) = P(x_1 | c) \times P(x_2 | c) \times \dots \times P(x_n | c) \times P(c)$$

False Alarm= FP/ TN + FP

mean	0	0
std. dev.	0.0017	0.0017
weight sum	0	0
precision	0.01	0.01

7. References

1. **An Improved Techniques Based on Naive Bayesian for Attack Detection [41]**
International Journal of Emerging Technology and Advanced Engineering Website:
www.ijetae.com (ISSN 2250-2459, Volume 2, Issue 1, January 2012)
2. **Evaluation of Fuzzy K-Means And K-Means Clustering Algorithms In Intrusion Detection Systems [40]**
Farhad Soleimani Gharehchopogh, Neda Jabbari, Zeinab Ghaffari Azar,
international journal of scientific & technology research volume 1, issue 11, december
2012 ISSN 2277-8616
3. **Data Mining for Network Intrusion Detection [42]**
Paul Dokas, Levent Ertoz, Vipin Kumar, Aleksandar Lazarevic, Jaideep Srivastava,
Pang-Nig Tan Computer Science Department, 200 Union Street SE, 4-192, EE/CSC
Building University of Minnesota, Minneapolis, MN 55455, USA
4. **An Overview on Intrusion Detection System and Types of Attacks It Can Detect Considering Different Protocols**
Amrita Anand* , Brajesh Patel Volume 2, Issue 8, August 2012 ISSN: 2277 128X
5. **Intrusion Detection based on Boosting and Naïve Bayesian Classifier [43]**
International Journal of Computer Applications (0975 – 8887) Volume 24– No.3,
June 2011.
6. **Analysis of KDD '99 Intrusion Detection Dataset for Selection of Relevance Features** Adetunmbi A.Olusola., Adeola S.Oladele. and Daramola O.Abosede
Proceedings of the World Congress on Engineering and Computer Science 2010 Vol I
WCECS 2010, October 20-22, 2010, San Francisco, USA