ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 11, November 2013

A Survey on Machine Learning Techniques for Intrusion Detection Systems

Jayveer Singh[1], Manisha J. Nene[2]

Department of Computer Engineering, DIAT, Pune, India, 411025 [1, 2]

Abstract: The rapid development of computer networks in the past decades has created many security problems related to intrusions on computer and network systems. Intrusion Detection Systems IDSs incorporate methods that help to detect and identify intrusive and non-intrusive network packets. Most of the existing intrusion detection systems rely heavily on human analysts to analyze system logs or network traffic to differentiate between intrusive and non-intrusive network traffic. With the increase in data of network traffic, involvement of human in the detection system is a non-trivial problem. IDS's ability to perform based on human expertise brings limitations to the system's capability to perform autonomously over exponentially increasing data in the network. However, human expertise and their ability to analyze the system can be efficiently modeled using soft-computing techniques. Intrusion detection techniques based on machine learning and soft-computing techniques enable autonomous packet detections. They have the potential to analyze the data packets, autonomously. These techniques are heavily based on statistical analysis of data. The ability of the algorithms that handle these data-sets can use patterns found in previous data to make decisions for the new evolving data-patterns in the network traffic. In this paper, we present a rigorous survey study that envisages various soft-computing and machine learning techniques used to build autonomous IDSs. A robust IDSs system lays a foundation to build an efficient Intrusion Detection and Prevention System IDPS.

Keywords: Intrusion detection, IDS, signature, anomaly, machine learning, neural networks, fuzzy logics, genetic algorithms, Bayesian networks.

I. Introduction

Communication networks and information systems have become an essential factor in economic, social development and almost in every facet of our daily lives. But rapid development of internet services and communication networks, information systems are vulnerable to one or more types of cyber attacks. The security of communication networks and information systems and their availability in particular, is therefore an increasing concern for cyber-security officers, network administrators and end-users. In general, cyber security threats are increasing rapidly, the incidents range from defaced websites to theft of large volumes of intellectual property and money, to even Internet crimes, hence Intrusion Detection Prevention System IDPS plays an important role for detect of security violation. A robust IDSs system lays a foundation to build an efficient Intrusion Detection and Prevention System IDPS.

An intrusion attempt or a threat is a deliberate and unauthorized attempt to (i) access information, (ii) manipulate information, or (iii) render a system unreliable or unusable. For example, (a) Denial of Service (DoS) is an attempt to make a machine or network resource unavailable to its intended users, which are needed to function correctly during processing; (b) Worms and viruses exploit other hosts through the network; and (c) Compromises obtains privileged access to a host by taking advantages of known vulnerabilities [1]. Every attack leaves its signature. A signature in the world of cyber-crime can be defined as signature is a set of rules that an IDS and an IPS use to detect typical intrusive activity, such as DoS attacks. Some of the methods that can be used to identify signature are:

- Connection attempt from a reserved IP address. This is easily identified by checking the source address field in an IP header.
- Packet with an illegal TCP flag combination. This can be found by comparing the flags set in a TCP header against known good or bad flag combinations.
- Email containing a particular virus. The IDS can compare the subject of each email to the subject associated with the virus-laden email, or it can look for an attachment with a particular name.
- DNS buffer overflow attempt contained in the payload of a query. By parsing the DNS fields and checking the length of each of them, the IDS can identify an attempt to perform a buffer overflow using a DNS field. A different method would be to look for exploit shell code sequences in the payload.
- Denial of service attack on a POP3 server caused by issuing the same command thousands of times. One signature for this attack would be to keep track of how many

times the command is issued and to alert when that number exceeds a certain threshold.

- File access attack on an FTP server by issuing file and directory commands to it without first logging in. A state-tracking signature could be developed which would monitor FTP traffic for a successful login and would alert if certain commands were issued before the user had authenticated properly.

Most of the existing intrusion detection systems rely heavily on human analysts to differentiate between intrusive and non-intrusive network traffic [2]. Human interventions are required to create, test, and deploy the signatures on the analyzed data-sets/databases. Thus, it may take hours or days to detect/generate a new signature for an attack, which may involve infeasible time to deal with rapid attacks. Information security while using Internet is a primary concern to the system users and administrators.

Intrusion detection has evolved as a major research problem for researchers, cyber-security officers and network administrators, since the compromised systems lay a serious impact on business and personal networks. As there are many risks of network attacks under the Internet environment, there are various systems designed to block the Internet-based attacks. Particularly, IDSs aid the network to resist external attacks. That is, the goal of IDSs is to provide a wall of defense to confront the attacks on computer systems connected to Internet. IDSs can be used to detect different types of malicious network communications and computer systems usage, whereas the conventional firewall cannot perform these tasks. [7]

IDSs systems can be defined as attempts to identify intrusions, defined as unauthorized uses, misuses, or abuses of computer systems by either authorized or unauthorized users. Some IDSs monitor a single computer, while others monitor a particular network or many networks that form a Wide Area Network. IDSs detect intrusions by analyzing information about processes running in the computer as well as network traffic going in and out of the protected network. This type of information resides in audit trails, system tables, and network traffic summary logs. Intrusion detection is based on the assumption that the behavior of intruders differ from a legal user. In general, IDSs can be divided into two categories based on their detection approaches: ***Anomaly*** based IDSs and ***Signature*** based IDSs. Both the techniques use packet-feature selection in building efficient IDSs.

Signature Based Detection

Signature based methodology works by comparing observed signatures to the signatures on database. This database is a list of known attack signatures. Any signature pattern constructed through the feature extracted from the packets in monitored environment matches the signatures on database is flagged as a violation of the security policy or as an attack. The signature based IDPS has little overhead in computation and preprocessing so it does not inspect every activity or network traffic on the monitored environment. Instead it only searches for known signatures in the database or file.

The signature based methodology is easy to deploy since it does not need to learn the environment [8]. This methodology works by simply searching, inspecting, and comparing the contents of captured network packets for the known *threat-signatures*. It also compares behavior signatures against *allowed behavior signatures*. Signature based methodology also analyzes the systems calls for known ***threat-payloads.*** Signature based methodology is very effective against known attacks/violations but it cannot detect new attacks until it is updated with new signatures. Signature based IDPS are easy to evade since they are based on known attacks and are dependent on new signatures to be applied before they can detect new attacks. Signature based detection systems can be easily bypassed by attackers who modify known attacks and target systems that have not been updated with new signatures that detect the modification. Signature based methodology requires significant resources to keep up with the potential infinite number of modifications to known threats. Signature based methodology is simpler to modify and improve since its performance is mainly based on the signatures or rules deployed. [9] [10] [11]

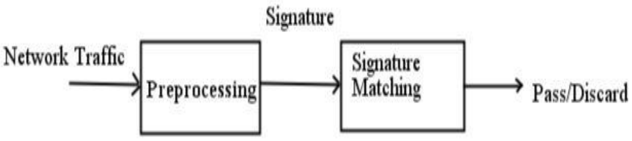

Fig.1 Signature Based Detection

The general architecture of a signature based methodology is shown in fig. 1. This architecture uses the preprocessing to find signature and compare activity signatures found in the monitored environment to the known signatures in the signature database. If a match is found, an alert is issued and there is no match the detector does nothing. Most commercial IDSs use this approach because known attacks can be detected economically and reliably with low false-positive errors.

Pseudo code for above detection technique is given below:

```
$INPUT$:
  $P$: $Packet$                          /
                   $*$ $incoming$
                   /$outgoing\ network\ packet$ $*/$
$OUTPUT$:
  $True\ alerts$: $set\ of\ alerts$
$PROCEDURE$:
$For\ each\ packet\ do$
```

```
Extract the information from packet and build signature S
  If (S == database Signature) then
Put an alarm                                  /
                * Rise an Alarm */
Add new signature into Signature database    /
                * Signature Database updation */
  End if
End for
```

Anomaly Based Detection

Also named as "Behavior- Based Detection" is described as a network IDS which models the behavior of the network, users, and computer systems and raises an alarm whenever there is a deviation from this normal behavior. Anomaly detection is assumed to be able to detect a new attack or any new potential attacks. It works on the principle of distance measurement by constructing profiles representing normal usage and then comparing it with the current behavior of data to find out a likely mismatch. It detects any action that significantly deviates from the normal behavior based on normal behavior, profile and thresholds calculated.

Anomaly based detection mechanism detects any traffic that is new or unusual, and is particularly good at identifying *sweeps* and *probes* towards network hardware. It can, therefore, give early warnings of potential intrusions because *probes* and *scans* are the predecessors of all attacks. And this applies equally to any new service installed on any item of hardware. For instance, if telnet is deployed on a network router for maintenance purposes and forgotten about when the maintenance was finished. An anomaly-based IDS is a perfect mechanism to detect anything from port anomalies and web anomalies to mis-formed attacks, where the URL is deliberately mistyped. But the problem occurs when an intrusion (noise) data in the training data, makes a misclassification which leads to generate a lot of false positive alarms.

The general architecture of a anomaly based methodology is shown in fig. 2. This architecture consists of preprocessing to extract information then build the pattern of the connection if it deviates from the normal behavior rise an alarm.

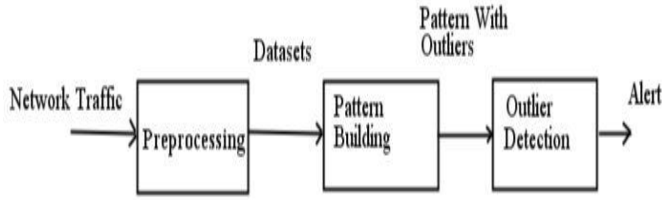

Fig.2 Anomaly Based Detection

However, based on the literature survey, each of the techniques has some limitations. Signature based detection techniques fail to detect zero-day attacks. Continuous updation in signature database is a major technical concern. In anomaly based detection techniques, different extraction notions of anomaly for different application domains produce high false alarm-rates.

In literature, numbers of anomaly detection systems are developed based on many different machine learning techniques. For example, some studies apply single learning techniques, such as neural networks, genetic algorithms, support vector machines, etc. On the other hand, some systems are based on combining different learning techniques, such as hybrid or ensemble techniques. In particular, these techniques are developed as classifiers, which are used to classify or recognize whether the incoming Internet access is the normal access or an attack. Machine learning techniques have ability to implement a system that can learn from data. For example, a machine learning system could be trained on incoming packets to learn to distinguish between intrusive and normal packet. After learning, it can then be used to classify new incoming packets into intrusive and normal packets.

II. MACHINE LEARNING TECHNIQUES

Signatures based IDSs rely on humans to create, test, and deploy the signatures. Thus, it may take hours or days to generate a new signature for an attack, which can be too long when dealing with rapid attacks. Nevertheless, in order to offer a human-independent solution to the above-mentioned problem, *anomaly* based IDSs based on machine learning techniques provide an added advantage. A*nomaly* based IDSs using machine learning techniques have the ability to implement a system that can learn from data (experience) and give the decision for unseen data. Fig 3 shows the mostly used machine learning techniques used to classify intrusive and non-intrusive behavior.

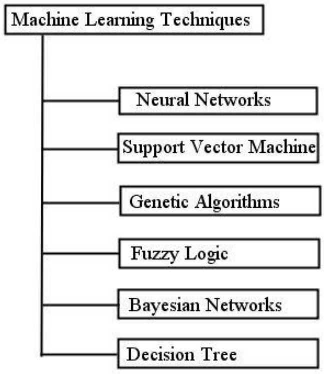

Fig. 3 Classification of machine learning techniques

Neural Networks

Artificial neural networks consists of a collection of processing elements that are highly interconnected and transform a set of inputs to a set of desired outputs that is inspired by the way biological nervous systems, such as the brain, process information. The technique of Neural Networks NN follows the same theories of how the human brains works. The Multilayer Perceptions MLP has been widely used neural network for intrusion detection. They are capable of approximating to arbitrary accuracy, any continuous function as long as they contain enough hidden units. This means that such models can form any classification decision boundary in feature space and thus act as non-linear discriminate function. When the NN is used for packets classification, there is one input node for each element of the feature vector. There is usually one output node for each class to which a feature may be assigned (shown in Fig. 4). The hidden nodes are connected to input nodes and some initial weight assigned to these connection. These weights are adjusted during the training process. MPL NN Structure One learning algorithm used for MLP is called *back-propagation* rule. This is a gradient descent method and based on an error function that represents the difference between the network's calculated output and the desired output. This error function is defined, based on the Mean Squared Error MSE. The MSE can be summed over the entire training set. In order to successfully learn, the network's true output should approach the desired output by continuously reducing the value of this error. The back-propagation rule calculates the error for a particular input and then back- Propagates the error from one layer to the previous one.

The connection weights, between the nodes, are adjusted according to the back-propagated error so that the error is reduced and the network learns. The input, output, and hidden layers neurons are variable. Input/output neurons are changed according to the input/output vector and hidden layer neurons are adjusted as per performance requirement, more hidden layers neurons more complex MLP.

The neural network intrusion detection system consists of three phases:

- Using automated parsers to process the raw TCP/IP dump data in to machine-readable form or use some benchmark dataset contain parsed connection records.

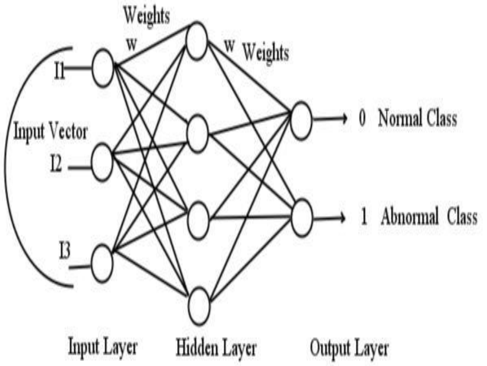

Fig. 4 Simple Architecture of MLP

- Training: neural network is trained on different types of attacks and normal data. The input has 41 features and the output assumes one of two values: intrusion (22 different attack types), and normal data.
- Testing: performed on the test dataset.

Neural network based intrusion detection system, intended to classify the normal and attack patterns and the type of the attack. It is observed that the long training time of the neural network was mostly due to the huge number of training vectors of computation facilities. However, when the neural network parameters were determined by training, classification of a single record was done in a negligible time. Therefore, the neural network based IDS can operate as an online classifier for the attack types that it has been trained for. The only factor that makes the neural network off-line is the time used for gathering information necessary to compute the features.[3][6]

Support Vector Machine

In classification and regression, Support Vector Machines SVM is the most common and popular method for machine learning tasks. In this method, a set of training examples is given with each example is marked belonging into one of two categories. Then, by using the Support Vector Machines algorithm, a model that can predict whether a new example falls into one categories or other is built [4]. For classification using SVM we need to select the training samples and carry out attribute extraction on samples, generally, we select a network connection as a sample or benchmark datasets like KDD99 which have collection of network connections attributes captured from variance networks. It is needed to define an input space x. For each network connection, choosing n attribute characteristics. The one-dimensional vector x can be Used to express a network connection, $x = \{x_1, x_2, \dots\dots\dots, x_n\ \}$ in which x_i, $i = 1,2, \dots\dots..n,$ refers to the i characteristic value of the sample x. Defining Y as output domain, as we only judge whether it is normal for each network connection, only two states are enough to express the problem, therefore, it can be defined that Y = (+1,−1) . When Y = +1 it is normal connection and when Y = −1 it is abnormal connection. A simple SVM classification diagram shown in Fig. 5.

In this only classes are there +1and -1 for normal and abnormal connections. It maps sample space to a higher dimensional even an infinite dimensional character space by the use of nonlinear mapping function ϕ (xi).

Support Vector Machine is a classification method possessing better learning ability for small samples, which has been widely applied in many fields such as Network Intrusion Detection, web page identification and face identification. Support Vector Machine applied in intrusion detection possesses such advantages as high training rate and decision rate, insensitiveness to dimension of input data, continuous correction of various parameters with increase in training data which endows the system with self-learning ability, and so on. Besides, it is also capable of solving many practical classification problems, such as problem involving small samples and non-linear problem. Therefore, Support Vector Machine will become increasingly popular in network security.

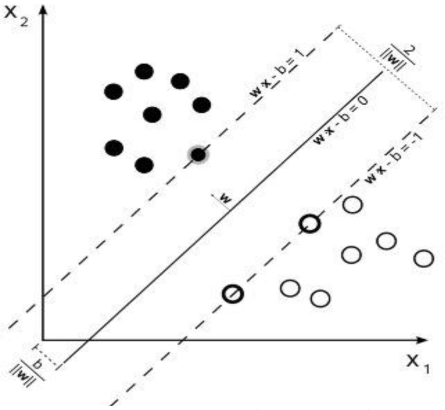

Fig.5 Simple Architecture of SVM classification

Decision Tree

Decision Tree algorithm is normally used for classification problem. In this algorithm, the data set is learnt and modeled. Therefore, whenever a new data item is given for classification, it will be classified accordingly learned from the previous dataset. Decision Tree algorithm can also be used for Intrusion Detection. For this reason, the algorithm will also learn and models data based on the training data. As a result, the model can classify which attack types does a future data belongs to based on the model built. One of the strength of Decision Tree is it can works well with huge data sets. This is important as large amount of data flow can be found across the computer networks. In real-time Intrusion Detection, it works well because Decision Tree gives the highest detection performance and can construct and interpret model easily. Another useful property of Decision Tree in Intrusion Detection model is its generalization accuracy. This is due to the trends in the future where there will always be some new attacks, and by having generalized accuracy provided by Decision Tree, these attacks can be detected [5].

Genetic Algorithm

Genetic algorithm GA is a search method that finds an approximate solution to an optimization task - inspired from biological, evolution process and natural genetics and proposed by Holland (1975). GA uses hill climbing method from an arbitrary selected number of genes. GA has four operators: ***initialization***, ***selection***, ***crossover*** and ***mutation***. GA has been used in different ways in IDS. There are many researchers that used evolutionary algorithms and especially GAs in IDS to detect malicious intrusion from normal use. Genetic algorithm based intrusion detection system is used to detect intrusion based on past behavior. A profile is created for the normal behavior based on that genetic algorithm learns and takes the decision for the unseen patterns. Genetic algorithms also used to develop rules for network intrusion detection. A chromosome in an individual contains genes corresponding to attributes such as the service, flags, logged in or not, and super-user attempts. These attacks that are common can be detected more accurately compared to uncommon attributes. Genetic algorithms are capable of deriving classification rules and selecting optimal parameters for detection process. The application of Genetic Algorithm to the network data consist primarily of the following steps:

- The Intrusion Detection System collects the information about the traffic passing through a particular network.
- The Intrusion Detection System then applies Genetic Algorithms which is trained with the classification rules learned from the information collected from the network analysis done by the Intrusion Detection System.
- The Intrusion Detection System then uses the set of rules to classify the incoming traffic as anomalous or normal based on their pattern.
- GA as evolutionary algorithms was successfully used in different types of IDS. Using GA returned impressive results, the best fitness value was very closely to the ideal fitness value. GA is a randomization search method often used for optimization problem. GA was successfully able to generate a model with the desired characteristics of high correct detection rate and low false positive rate for IDS. And it used successfully in IDS to distinguish the normal action and the intruded actions, and clustering GAs is a promising method for the detection of malicious intrusions into computer systems. [17][18] [19] [20] [21].

Fuzzy Logic

Fuzzy logic is derived from fuzzy set theory under which reasoning is approximate rather than precisely derived from classical predicate logic. Fuzzy techniques are thus used in the field of anomaly detection mainly because the features to be considered can be seen as fuzzy variables. With fuzzy spaces, fuzzy logic allows an object to belong to different classes at the same time. This concept is helpful when the difference between classes is not well defined. This is the case in the intrusion detection task, where the differences between the normal and abnormal classes are not well defined.

In intrusion detection, there are $m + 1$ classes where every object should be classified. Among these classes, there is one special class called the normal class and m classes called the abnormal classes (based on known intrusions or attacks). The data set used by the learning algorithms consists of a set of objects, each object with $n + 1$ attributes. The first n attributes define the object characteristics (monitored parameters) and the last attribute defines the class that the object belongs to (the classification attribute). Accordingly, fuzzy classifier system for solving intrusion detection problem should have a set of $m + 1$ rules, one for the normal class and m for the abnormal classes, where the condition part is defined by the monitored parameters and the consequent part is an atomic expression for the classification attribute.

The evolved fuzzy rules are not complex as no more than five attributes are used in each rule. It allows characterization of the normal and abnormal behaviors in a simple way. Simpler fuzzy rules have a clear advantage in real applications. First, they yield rules that are easier to interpret, hence score high on interpretability. Second, they yield a classifier rule that is faster in deployment. This is especially crucial for data involving a large number of attributes.

Although fuzzy logic has proved to be effective, especially against port scans and probes, its main disadvantage is the high resource consumption and large time consumed during the training. [12][14][19].

Bayesian Networks

A Bayesian network is a model that encodes probabilistic relationships among the variables of interest. This technique is generally used for intrusion detection in combination with statistical schemes. The naïve Bayesian (NB) algorithm is used for learning task, where a training set with target class is provided. Training set is described by attributed A1 through An, and each attribute is described by attribute values $a1, a2 \dots an$ associated with class C. The objective is to classify an unseen example, whose class value is unknown but attribute values are known. The Bayesian approach to classifying the unseen example is to assign the most probable target class. $Cmap$ Given the attribute values $(a_1, a_2, \dots \dots \dots, a_n)$ that describes the example. $C_{map} = argmax\ C_j\ \Sigma\ C_P\left(C_j \middle| a_1, a_2 \dots \dots \dots a_n\right)$ the expression can be rewrite using Bayesian theorem as

$$C_{map} = argmax\ C_j \Sigma\ C_P(a_1, a_2 \dots \dots \dots a_n \mid C_j) P(C_j) \qquad (1)$$

It is easy to estimate each of the $P\ (C_j)$ simply by counting the frequency with which each target class C_j occurs in the training set. The naïve Bayesian algorithm is based on the simplifying assumption that given the target class of the example, the probability of observing the conjunction $a1, a2 \dots an$ is just the product of the probabilities for the individual attributes: $P(a_1, a_2 \dots \dots \dots a_n | C_j) = \bullet\ i\ P(ai|C_j)$. Substituting this into equation 1, we get

$$C_{NB} = argmax\ C_j \Sigma\ C_P(C_j \bullet i\ P(a_i)|C_j) \qquad (2)$$

WhereC_{NB}, denote the target class predicate by the naïve Bayesian classifier. In naïve Bayesian algorithm, the probability values of equation 2 are estimated from the given training data. These estimated values are then used to classify unknown examples.

a procedure that yields several advantages, including the capability of encoding interdependencies between variables and of predicting events, as well as the ability to incorporate both prior knowledge and data [14][18].

III. COMPARISON OF THE REVIEWED SCHEMES

Though all above mentioned machine learning based intrusion detection schemes have tried to achieve high detection rate but each one have their own pros and cons. Following table described about this. [13][15]

TABLE I
C OMPARISON OF THE REVIEWED SCHEMES

S. No.	Machine Learning Technique	Pros	Cons
1	Neural Networks	Ability to generalize from limited, noisy and incomplete data. Does not need expert knowledge and it can find unknown or novel intrusions.	Slow training process so not suitable for real-time detection. Over-fitting may happen during neural network training.
2	Bayesian Network	Encodes probabilistic relationships among the variables of interest. Ability to incorporate both Prior knowledge and data.	Harder to handle continuous features. May not contain any good classifiers if prior knowledge is wrong.
3	Support Vector Machine	Better learning ability for small samples. High training rate and decision rate, insensitiveness to dimension of input data.	Training takes a long time. Mostly used binary classifier which cannot give additional information about detected type of attack.
4	Genetic Algorithm	Capable of deriving best classification rules and Selecting optimal parameters. Biologically inspired and employs evolutionary algorithm.	Genetic algorithm cannot assure constant optimization response times. Over-fitting.
5	Fuzzy Logic	Reasoning is Approximate rather than precise. Effective, especially against port scans and probes.	High resource consumption Involved. Reduced, relevant rule subset identification and dynamic rule updation at runtime is a difficult task.

6	Decision Tree	Decision Tree works well with huge data sets. High detection accuracy.	Building a decision tree is computationally intensive task.

IV. CONCLUSION

Machine learning for intrusion detection has received much attention in the computational intelligence community. In intrusion detection algorithm, huge volumes of audit data must be analyzed in order to construct new detection rules for increasing number of novel attacks in high speed network. Intrusion detection algorithm should consider the complex properties of attack behaviors to improve the detection speed and detection accuracy. Analyze the large volume of network dataset and improve the performance of detection accuracy, intrusion detection become an important research field for machine learning. Advantage of machine learning-based detection systems detect or categorize persistent features without any feedback from the environment. Disadvantages of learning-based detection systems are if a credible amount of normal traffic data is not available, the training of the techniques becomes very difficult. Each of the approaches to implement an intrusion detection system has its own advantages and disadvantages. This is apparent from the discussion of comparison among the various methods. Thus it is difficult to choose a particular method to implement an intrusion detection system over the other.

REFERENCES

[1] J. P. Anderson, “Computer Security Threat Monitoring and Surveillance,” James P Anderson Co, Fort Washington, Pennsylvania, Tech. Rep., April 1980.

[2] Chris Sinclair, Lyn Pierce, Sara Matzner “An Application of Machine Learning to Network Intrusion Detection”. Phoenix, AZ 06 Dec 1999-10 Dec 1999

[3] Hua TANG, Zhuolin CAO “Machine Learning-based Intrusion Detection Algorithms” Binary Information Press, December, 2009

[4] J. Burges, "A tutorial on support vector machines for pattern recognition" Data Mining and Knowledge Discovery, vol. 2, pp. 12 1- 167, 1998.

[5] Kamarularifin Abd Jalil, Muhammad Hilmi Kamarudin, Mohamad Noorman Masrek “Comparison of Machine Learning Algorithms Performance in Detecting Network Intrusion”, 2010

[6] D. Rumelhart, G. Hinton and R Williams, "Learning internal representations by back-propagating errors," Parallel Distributed Processing: Explorations in the Microstructure of Cognition, D. Rumelhart and 1. McClelland editors, vol. I, pp. 3 18-362, MIT Press, 1986.

[7] Dorothy, Denning. “An intrusion-detection model,” IEEE Transactions on Software Engineering, Vol. SE-13, No.2. February, 1987.

[8] Alfonso Valdes, Keith Skinner, “Probabilistic alert correlation,” 4th International Symposium on Recent Advances in Intrusion Detection (RAID2001), 2001, pp.54–68.

[9] Indraneel Mukhopadhyay, Mohuya Chakraborty and Satyajit Chakrabarti, "A Comparative Study of Related Technologies of Intrusion Detection & Prevention Systems," Journal of Information Security, Vol. 2 No. 1, pp. 28-38.

[10] Justin Lee, Stuart Moskovics, Lucas Silacci, “A Survey of Intrusion Detection Analysis Methods,” CSE 221, University of California, San Diego, spring 1999.

[11] B. Balajinath and S. V. Raghavan, “Intrusion detection through learning behavior model,” Computer Communications, vol. 24, no. 12, pp. 1202–1212, July 2001.

[12] M. S. A. Khan, “Rule based Network Intrusion Detection using Genetic Algorithm,” International J. Computer Applications, vol. 18, no. 8, pp. 26–29, March 2011.

[13] P Garcia Teodora, J Diaz Verdejo, G Macia Farnandez, and E Vazquez, "Anomaly-based network intrusion detection:Techniques,Systems and Challenges," *Journal of Computers & Security*, vol. 28, no. 1, pp. 18-28, February 2009.

[14] Rajdeep Borgohain, " FuGeIDS : Fuzzy Genetic paradigms in Intrusion Detection Systems," *International Journal of Advanced Networking and Applications*, vol. 3, no. 6, pp. 1409-1415, 2012

[15] Hua TANG†, Zhuolin CAO “Machine Learning-based Intrusion Detection Algorithms “Journal of Computational Information Systems5:6(2009) 1825-1831

[16] Taeshik Shon’, Yongdue Kim, Cheolwon Lee’, and Jongsub Moon”A Machine Learning Framework for Network Anomaly Detection using SVM and GA” 0-7803-9290-6/2005

[17] Dewan Md. Farid, Mohammad Zahidur Rahman “Learning Intrusion Detection Based on Adaptive Bayesian Algorithm” 1-4244-2136-7/2008

[18] Jonatan Gomez and Dipankar Dasgupta “Evolving Fuzzy Classifiers for Intrusion Detection” Workshop on Information Assurance United States Military Academy, West Point, NY June 2001

[19] A.A. Ojugo, A.O. Eboka, O.E. Okonta, R.E Yoro, F.O. Aghware “Genetic Algorithm Rule-Based Intrusion Detection System” (GAIDS), ISSN 2079-8407 VOL. 3, NO. 8 Aug, 2012

[20] J. L. Zhao, J. F. Zhao, and J. J. Li, ―Intrusion Detection Based on Clustering Genetic Algorithm‖, International Conference on Machine Learning and Cybernetics IEEE, Guangzhou, 2005, pp. 3911-3914.

[21] W. Spears, and V. Anand, ―A Study of Crossover Operators in Genetic Programming”, In Proceedings of the Sixth International Symposium on Methodologies for Intelligent Systems, Charlotte, NC. 1991, pp. 409-418.