

[Egyptian Informatics Journal (2014) 15, 37–50](http://dx.doi.org/10.1016/j.eij.2013.10.003)

Cairo University

Egyptian Informatics Journal

[www.elsevier.com/locate/eij](http://www.elsevier.com/locate/eij) [www.sciencedirect.com](http://www.sciencedirect.com/science/journal/11108665)

ORIGINAL ARTICLE

Effective approach toward Intrusion Detection System using data mining techniques

G.V. Nadiammai, M. Hemalatha [\*](#_bookmark0)

*Department of Computer Science, Karpagam University, Coimbatore 641021, Tamilnadu, India*

Received 22 May 2013; revised 29 August 2013; accepted 27 October 2013

Available online 5 December 2013

Abstract With the tremendous growth of the usage of computers over network and development in application running on various platform captures the attention toward network security. This paradigm exploits security vulnerabilities on all computer systems that are technically difficult and expensive to solve. Hence intrusion is used as a key to compromise the integrity, availability and confidentiality of a computer resource. The Intrusion Detection System (IDS) plays a vital role in detecting anomalies and attacks in the network. In this work, data mining concept is integrated with an IDS to identify the relevant, hidden data of interest for the user effectively and with less execution time. Four issues such as Classification of Data, High Level of Human Interaction, Lack of Labeled Data, and Effectiveness of Distributed Denial of Service Attack are being solved using the proposed algorithms like EDADT algorithm, Hybrid IDS model, Semi-Supervised Approach and Varying HOPERAA Algorithm respectively. Our proposed algorithm has been tested using KDD Cup dataset. All the proposed algorithm shows better accuracy and reduced false alarm rate when compared with existing algorithms.

© 2013 Production and hosting by Elsevier B.V. on behalf of Faculty of Computers and Information,

KEYWORDS

Anomaly based algorithm; Classification algorithms; Data communication; Denial of service attack; Intrusion detection

Cairo University.

1. Introduction

\* Corresponding author. Tel.: +91 9003702602.

E-mail addresses: [gvnadisri@gmail.com](mailto:gvnadisri@gmail.com) (G.V. Nadiammai), [csreaserch.](mailto:csreaserch.hema@gmail.com) [hema@gmail.com](mailto:csreaserch.hema@gmail.com) (M. Hemalatha).

Peer review under responsibility of Faculty of Computers and Information, Cairo University.

**Production and hosting by Elsevier**

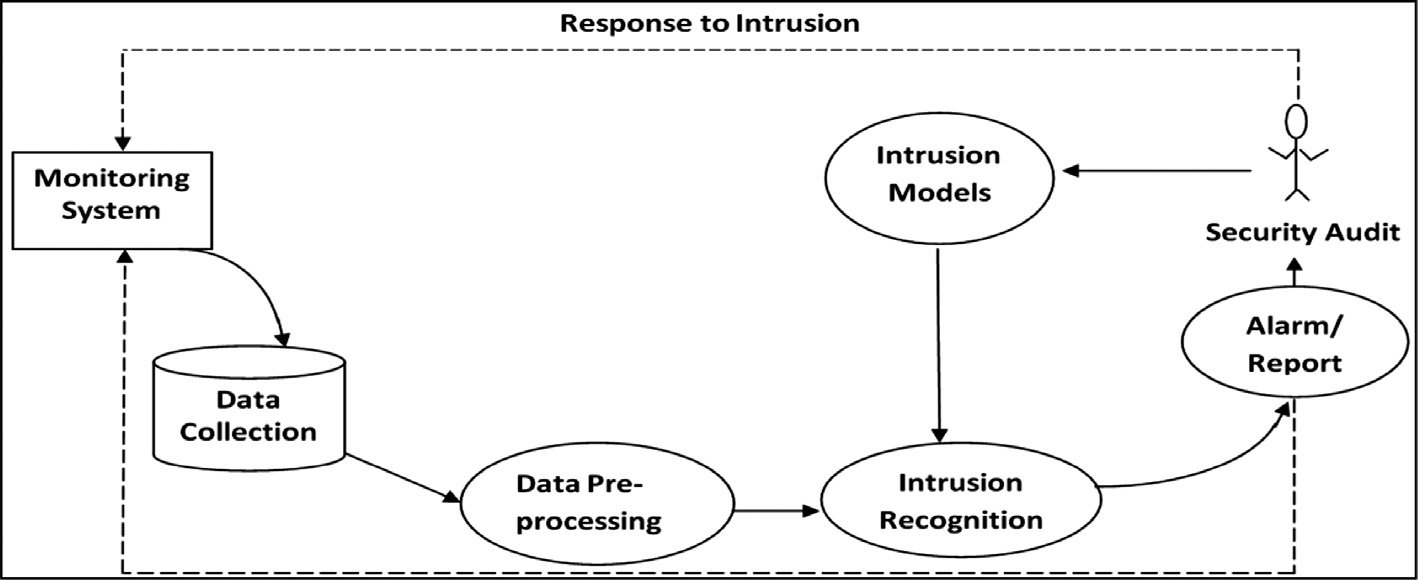
In this modern world intrusion occurs in a fraction of seconds. Intruders cleverly use the modified version of command and thereby erasing their footprints in audit and log files. Success- ful IDS intellectually differentiate both intrusive and nonin- trusive records. IDS was first introduced by James Anderson in the year 1980 [[1]](#_bookmark35). Most of the existing systems have security breaches that make them easily vulnerable and could not be solved. Moreover substantial research has been going on intru- sion detection technology which is still considered as immature and not a perfect tool against intrusion. It has also become a most priority and challenging tasks for network administrators

1110-8665 © 2013 Production and hosting by Elsevier B.V. on behalf of Faculty of Computers and Information, Cairo University. <http://dx.doi.org/10.1016/j.eij.2013.10.003>

38 G.V. Nadiammai, M. Hemalatha

Figure 1 Overall structure of Intrusion Detection System.

and security experts. So it cannot be replaced by more secure systems.



Data mining based IDS can efficiently identify these data of user interest and also predicts the results that can be utilized in the future. Data mining or knowledge discovery in databases has gained a great deal of attention in IT industry as well as in the society. Data mining has been involved to analyze the useful information from large volumes of data that are noisy, fuzzy and dynamic. [Fig. 1](#_bookmark1) illustrates the overall architecture of IDS. It has been placed centrally to capture all the incoming packets that are transmitted over the network. Data are col- lected and send for pre-processing to remove the noise; irrele- vant and missing attributes are replaced. Then the pre- processed data are analyzed and classified according to their severity measures. If the record is normal, then it does not re- quire any more change or else it send for report generation to raise alarms. Based on the state of the data, alarms are raised to make the administrator to handle the situation in advance. The attack is modeled so as to enable the classification of net- work data. All the above process continues as soon as the transmission starts.

Ektefa [[2]](#_bookmark36) compared C4.5 & SVM to show the performance of both algorithm and FAR values too. Among these two C4.5 works better compared to other. Since the performances of a classifier are often evaluated by an error rate and it does not suit the complex real problems, in particular multiclass. Holden [[3]](#_bookmark37) has proposed hybrid PSO algorithm that can deal with nominal attributes without going for the both conversion and nominal attribute values. To overcome the drawback (features) that the PSO/ACO algorithm lacks. The proposed method shows simple rule set efficiently to increase in accuracy. Likewise Ardjani [[4]](#_bookmark38) applied SVM with PSO as (PSO-SVM) to optimize the performance of SVM. 10-Fold cross validation is done to estimate the accuracy. It utilizes the advantage of minimum structural risk with global optimiz- ing features. The result shows better accuracy with high execu- tion time. Since there [[5]](#_bookmark39) is an existence of multidimensional data set, it is necessary to extract the features and also to remove the redundant and inconsistent features that affects classification. Based on this, information gain and genetic algorithm has been combined to select the significant features. This method shows better accuracy when features are selected than individually applied. Panda [[6]](#_bookmark40) uses two class classifica- tion method in terms of normal or attack. The combination of J48 and RBF shows more error prone and RMSE rate.

Compared to this, Nested Dichotomies and random forest method show 0.06% error with a 99% detection rate.

Petrussenko Denis [[7]](#_bookmark41) involves LERAD to detect attacks in network packets and TCP flow to capture a new form of attack patterns. He found LERAD performs well using a DARPA data set with high detection rate. In [[8]](#_bookmark33) Mahoney used anomaly detection methods like PHAD, ALAD, LERAD to model the application layer, data link layer and so on. Out of that LER- AD performs well. SNORT [[9]](#_bookmark33) is tested on IDEVAL data set and attacks are tabled daily nearly for one week. SNORT, SNORT with PHAD, SNORT with NETAD and SNORT is also combined with PHAD and NETAD as a preprocessor. As a result, SNORT + PHAD + NETAD detects nearly 146 attacks from 201 attacks respectively.

Unsupervised method [[10]](#_bookmark33) uses a huge set of data as pre-la- beled training data and produces less accuracy. To overcome this issue, a semi-supervised algorithm is used. Fuzzy Connect- edness based Clustering [[11]](#_bookmark33) approach is evaluated using both Euclidean distance and statistical properties of clusters. It facilitates the discovery of any shape and detects not only known but also its variants. Ching-Hao et al. [[12]](#_bookmark33) proposed a co-training framework to leverage unlabeled data to improve intrusion detection. This framework provides lower error rate than single view method and thereby incorporating an active learning method to enhance the performance. In [[13]](#_bookmark33) the semi supervised learning mechanism is used to build an alter filter to reduce the false alarm ratio and provides high detection rate. Where the features of both supervised and semi supervised learning are same in nature.

A Tri-Training SVM algorithm is used to improve the accu- racy rate and speed [[14]](#_bookmark33). Monowar H. Bhuyan [[15]](#_bookmark33) present a tree based clustering technique to find clusters among intrusion detection data set without using any labeled data. The data set can be labeled using cluster labeling technique based on a Tree CLUS algorithm. It works faster for the numeric and a mixed category of network data.

Partially Observable Markov Decision Process [[16]](#_bookmark34) involves the combination of both misuse and anomaly detection to determine the cost function. Semi-Supervised strategy is ap- plied to three different SVM classifiers and to the same PSVM classifiers.

Distributed Denial of Service Attacks [[17]](#_bookmark42) has been achieved tremendous growth in recent years since it is difficult to solve. It has been obtained through two ways. First by filter- ing the legitimate traffic from malicious traffic and second by

Effective approach toward Intrusion Detection System using data mining techniques 39

degrading the performance of legitimate traffic gradually. Denial of capability attack is one of the major causes for the existence of DDoS attacks. DDoS attacks can be prevented by denial of the capability approach by Sink tree model [[18]](#_bookmark43) representing quota assigned to each domain on the network.

Distributed Denial of Service Attacks are not only suited for the specified target machine but also compromises the whole network. Based on this perspective proactive algorithm has proposed by Zhang et al. [[19]](#_bookmark44). The network is divided into a set of clusters. Packets need permission to enter, exit or pass through other clusters. High speed traffic [[20]](#_bookmark45) is monitored by IP prefix based aggregation method to detect DDoS related anomalies facilitated in a streaming fashion. In [[21]](#_bookmark46) the author presents the synchronous communication based on acknowl- edgment based and in the presence of fixed clock drift. The cli- ent clock relates to that of the server. But any acknowledgment loss and faster or slower clock drift of client enables the adver- sary to cause a direct attack on it.

Preprocessing of network data [[22]](#_bookmark47) consumes time and hard for network administrator to solve. Even though the prepro- cessing is made successfully, Classification of Data and label- ing of unlabeled data seems to be a challenging task. Based on this, four issues such as Classification of Data, High Level of Preprocessing, Semi-Supervised Approach and Mitigating Distributed Denial of Service Attack has been solved using proposed approaches that help IDS to efficiently identify the attacks accurately with reduced false alarm rate.

This paper is organized as follows. Section [2](#_bookmark3) includes the motivation of the work. Section [3](#_bookmark4) describes the problem state- ment. Section [4](#_bookmark2) explains the methodology of the proposed work. Section [5](#_bookmark9) describes the data set used and its features in detail. Section [6](#_bookmark11) includes the details of performance evalua- tion based on the experimental study. Section [7](#_bookmark31) refers to con- clusion and future enhancement.

1. Motivation

This research focuses on solving the issues in intrusion detec- tion communities that can help the administrator to make pre-processing, classification, labeling of data and to mitigate the outcome of Distributed Denial of Service Attacks. Since, the network administrator feels difficult to pre-process the data. Due to the overwhelming growth of attacks which makes the task hard, attacks can be identified only after it happens. To overcome this situation, frequent updating of profiles is needed. Reduced workload of administrator increases the detection of attacks. Data mining includes many different algorithms to accomplish the desired tasks. All of these algo- rithms aims to fit a model to the prescribed data and even ana- lyzes the data and simulate a model which is closest to the data being analyzed.

1. Problem statement

Data mining approaches have been implemented by many authors to solve the detection problem. This implies that we are close to the solution. Since pattern signature approach is currently utilized only by network administrators. The fact is that the existing works deal with the subset of problem that are needed for achieving intrusion detection and not others.

To solve the above issues the following solutions were made,

* To solve the problem of Classification of Data, an enhanced data adapted decision tree algorithm is pro-

posed. This algorithm works different normal decision tree algorithm. It efficiently classifies the data into normal and attack without any misclassification.

* To minimize the workload of network administrator,

SNORT is combined with anomaly based approaches.

Using this technique, a Hybrid IDS model is proposed. This technique automatically classifies the data based on the predefined rules within it.

* The problem of implementing supervised and unsuper-

vised method can be solved by using Semi-Supervised

Approach where with small amount of labeled data, the large amount of unlabeled data can be labeled.

* Distributed Denial of Service Attack can be greatly

reduced using varying clock drift, with the help of vary-

ing clock drift in network based application, the adver- sary finds difficult to access the port that has been used by the legitimate client. At the same time, any client can communicate with the server for longer time inter- vals without any interruption.

Hence the proposed approaches address the issues and effi- ciently identifies any kind of attack. All these four solutions have been discussed in the following section in detail.

1. Methodology

Some of the open issues have been taken to detect attacks over the network. To achieve this, the framework of the proposed methods has been discussed below. The entire framework of the proposed methodology in Intrusion Detection System is described in [Fig. 2](#_bookmark5).

* 1. *Framework of proposed EDADT (Efficient Data Adapted Decision Tree) algorithm*

The pseudo code of the proposed EDADT algorithm shown in [Fig. 3](#_bookmark6) utilizes the hybrid PSO technique to identify the local and global best values for n number of iterations to obtain the optimal solution. The best solution is obtained by calculat- ing the average value and by finding the exact efficient features from the given training data set. For each attribute a, select all unique values of a to find the unique values belong to the same class label. If *n* unique values belong to the same class label, split them into *m* intervals, and *m* must be less than *n*. If the unique values belong to different c class label, check whether the probability of the value belongs to same class. If it is found then change the class label of values with the class label of highest probability. Split the unique values as c interval then repeat checking of unique values in the class label for all values in the data set. Find out the normalized information gain for each attribute and decision node forms a best attribute with the highest normalized information gain. Sublists are gener- ated using best attributes and those nodes forms the child nodes. These processes continue until the data set converges. At last, train the EDADT model.

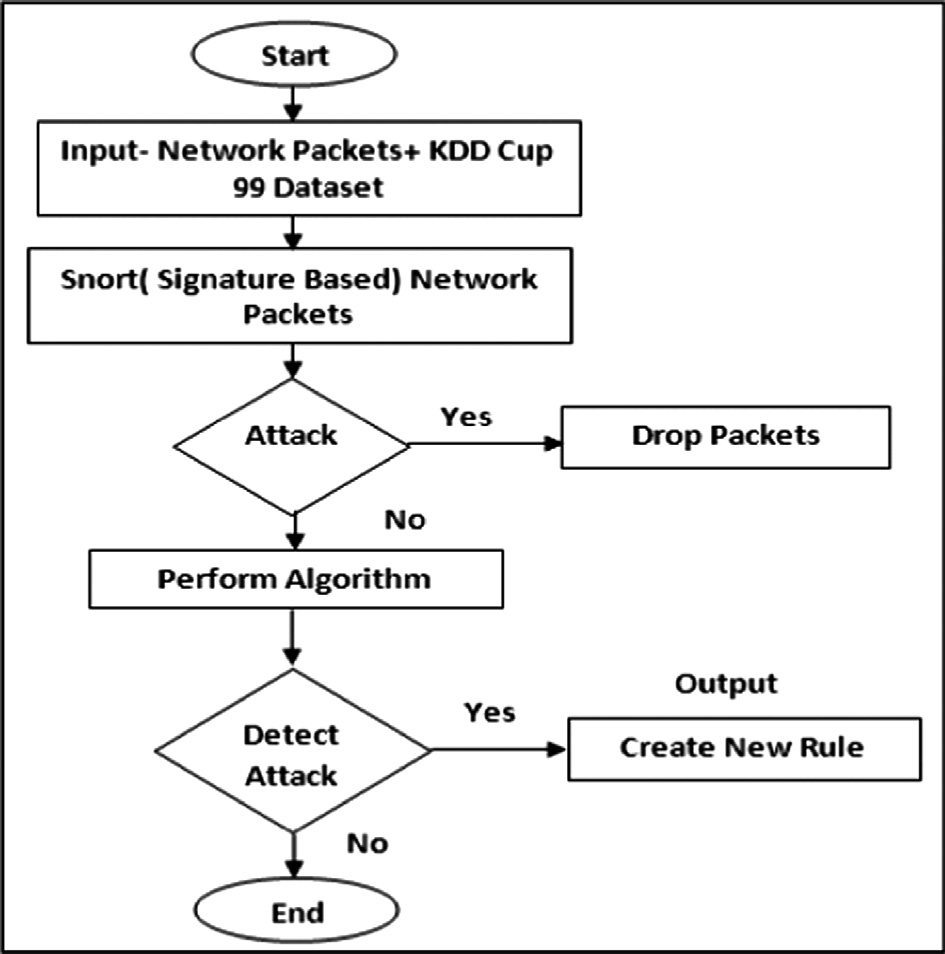
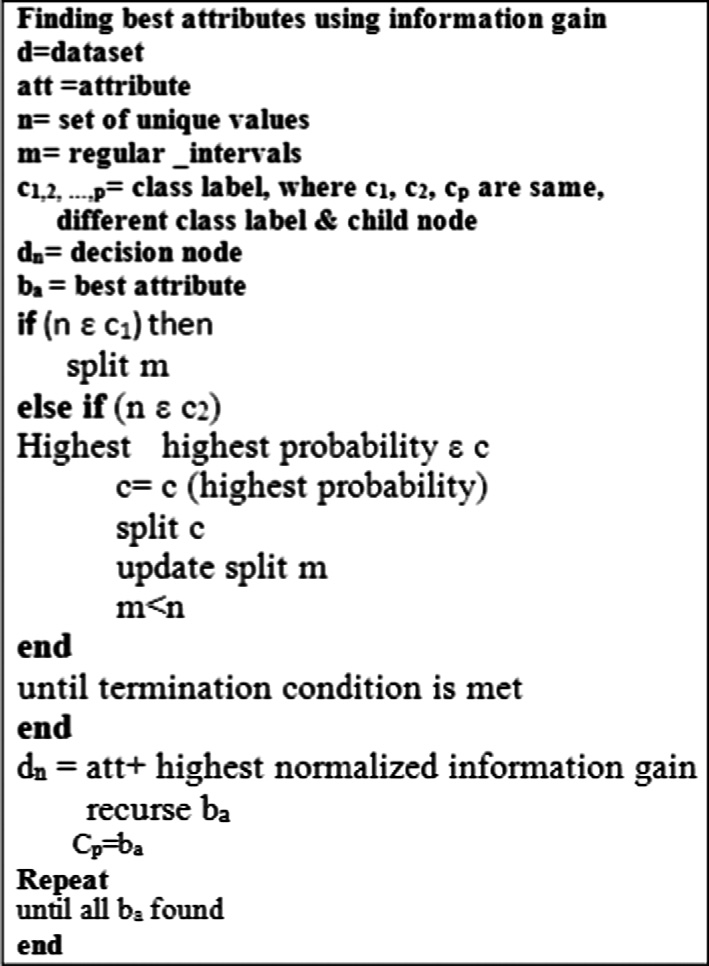
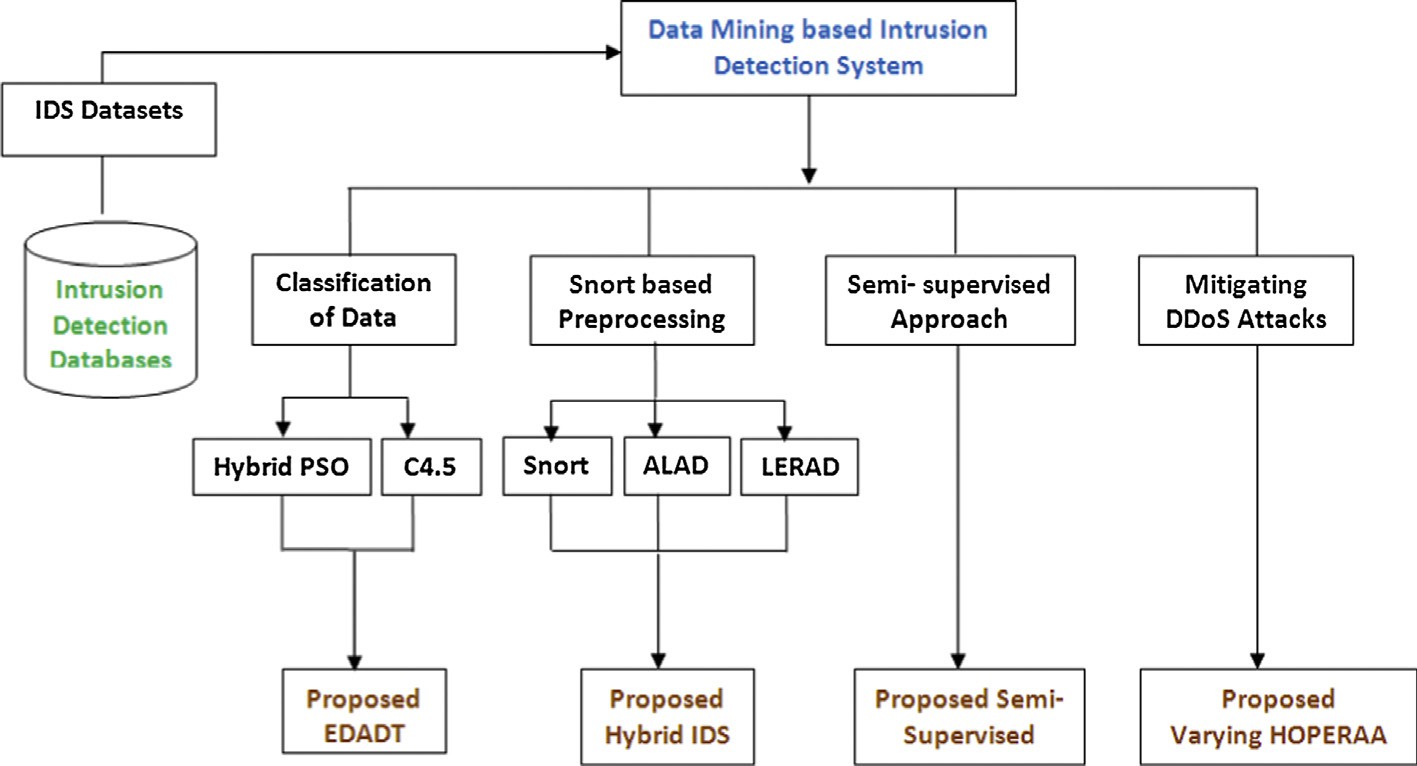
40 G.V. Nadiammai, M. Hemalatha

Figure 2 Entire framework of proposed methodology in Intrusion Detection System.

Figure 3 Pseudocode of proposed EDADT algorithm.

Figure 4 Framework of proposed Hybrid Intrusion Detection System.

* 1. *Framework of proposed Hybrid Intrusion Detection System*



SNORT shown in [Fig. 4](#_bookmark7) is installed to capture the network packets in real time and also KDD Cup 99 dataset is used. SNORT is a signature based method because it detects the at- tack based on the set of rules that are predefined within the SNORT. If any attack data is found, it automatically drops the packet otherwise the particular record is considered as a normal one. Since SNORT detects only profile based attacks some of the anomaly based approaches such as Packet Header Anomaly Detection, Network Traffic Anomaly Detector, Application Layer Anomaly Detector, Learning Rules for Anomaly Detection have been used to perform better predic- tion. Based on the new attack, it is updated in the profile as a new rule. By using this approach any kind of attacks can also be found.

* 1. *Framework of proposed Semi-Supervised Approach*

In this approach, the dataset is divided into training and test- ing data. First, training data includes both the labeled data and unlabeled data. Using the labeled data the unlabeled data can be labeled. Hence this kind of approach is said to be Semi- Supervised Approach. The labeled training data are applied to the SVM classifier and the model is generated. Then, change the SVM parameters by applying the Radial Basis Function kernel function and generate the model for each tuning pro- cess. Apply the training unlabeled data to SVM model as test data and results are generated for all models. Check majority voting for all models. Drop records which do not satisfy the voting results. Include the changed label as predicted labels. Randomly generate 1000 data points to find the vector dis- tance between each support vector and the data points. This

Effective approach toward Intrusion Detection System using data mining techniques 41

process enables the most confidential data. Provide these new data instances with the trained labeled data. At last, include the unlabeled test data to the classifier and check the accuracy rate and its corresponding false alarm rate respectively. This approach has been successfully tested using KDD Cup 99 data set and the results obtained have been compared with the exist- ing algorithms as shown in [Fig. 5](#_bookmark8).

* 1. *Framework of proposed varying HOPERAA (Hopping Period Alignment and Adjustment) algorithm*

To mitigate the Effectiveness of Distributed Denial of Service Attack Varying HOPERAA Algorithm is proposed. Based on the previous work done by Zhang Fu, et al. [[22]](#_bookmark47), in our ap- proach, a variable clock drift method is proposed to avoid the client waiting time for server and at the same time message loss is avoided greatly. It includes three aspects,

* Contact initiation part.
* Data transmission part.
* Varying HOPERAA Algorithm.
  + 1. *Contact initiation part*

This activity takes place on the client side to initiate a request to the server for connection and further communication. The server divides the range of port numbers into intervals and

ports are evenly split for every interval and these ports changes every s time unit. Port sequence is only known by the client and server which is independent from other clients. Suppose if any message is lost then the port remains open which can be disabled by the adversary and starts accessing the server by pretending as the legitimate client. To overcome this issue, the pseudorandom function and index value can be issued by the server to intimate the client to choose the next set of ports through this the bandwidth would be maintained.

Once the server receives the contact initiation message it sends the varying clock value of the client and the server’s clock value is *t*1 .. . *tn* which denotes the arriving time of the same message at different rates. Then,

*Vhc* (*t*1)= {low rate (L); medium rate (M); high rate (H)} (1)

It would be stored by the client to estimate the variable clock

drift. The client waits for the acknowledgment from the server for a specific period of time say 2l + *L*. After that, it chooses another interval and starts sending messages. It may take any a number of trials to get access to the server. In our proposed algorithm, the number of trails made by the client in contact initiation part has been minimized so as to improve the reli- ability of the application.

Once the message is received, the server waits for the port to open. As soon as the port is open it sends the acknowledgment with pseudorandom function, index value and varying time and *t*1, *t*2 to the client.

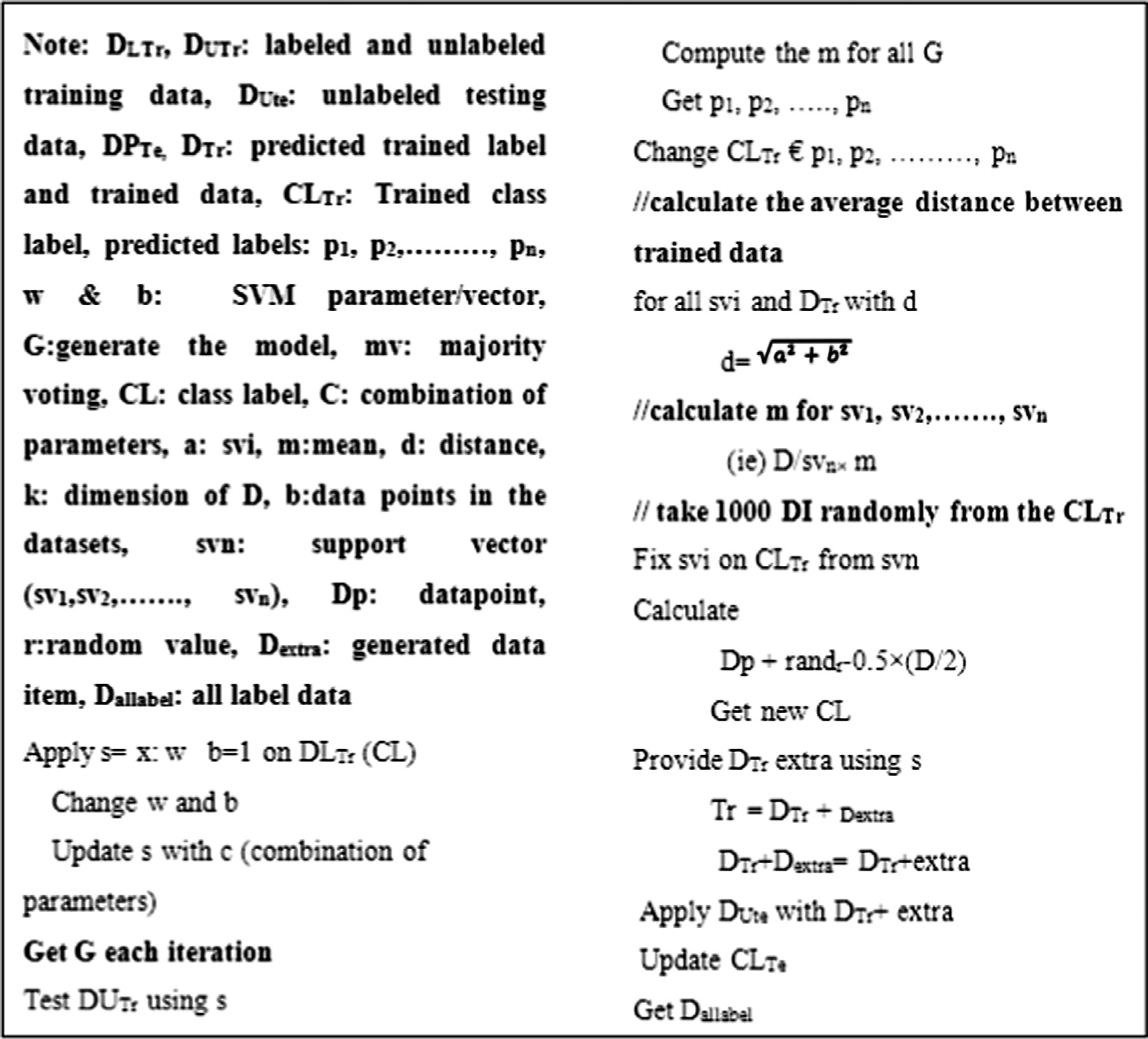


Figure 5 Pseudocode of Semi-Supervised Approach.

42 G.V. Nadiammai, M. Hemalatha

* + 1. *Data transmission part*

In this part, the client starts to send messages to sever s. Once the client receives a reply from the client during contact initia- tion part it got the port hopping sequence. According to the state of the message the port sequence varies. The opening time of the port is *L* + l time unit where *L* is greater than l. As soon as the port pi opens when it reaches ltime units, the new subsequent port *pi* + 1 opens simultaneously like that the port grows according to the message length. In our algo- rithm the message delivery latency has been improved for bet- ter performance. The timer at client c will be placed zero when it receives the reply message from the server.

* + 1. *Varying HOPERAA Algorithm*

The Varying HOPERAA Algorithm has been illustrated using [Fig. 6](#_bookmark10) in which *Lc* does not drift apart from the server. Since the varying clock drift is maintained depending on the length of the message. In this section, client *c* has a variable clock drift *Vhc* related to the server. The server maintains threshold value to estimate the nature of the message. If the client sends data con- tinuously it has taken as high rate, if messages send moderately it will be taken as medium rate and not frequently then it is taken as low rate. The server keeps the part of a HOPERAA algorithm to estimate the clock drift and also evaluates the state of message consequently. At the maximum the client runs Vary- ing HOPERAA one time to estimate the clock drift but in case of fixed clock drift the client runs HOPERAA over three times to know the clock drift is slower or faster than the server. Through the proposed Varying HOPERAA Algorithm growth the growth of intervals is extremely reduced.

1. Data set description

Attacks can be described as

* *Dos attack* – It is a kind of attack where the attacker makes processing time of the resources and memory

busy so as to avoid legitimate user from accessing those resources.

* *U2R attack* – Here the attacker sniffs the password or makes some kind of attack to access the particular host

in a network as a legitimate user. They can even promote some vulnerability to gain the root access of the system.

* *R2L attack* – Here the attacker sends a message to the

host in a network over remote system and makes some

vulnerability.

* *Probe attack* – Attacker will scan the network to gather information and would make some violation in the

future.

KDD Cup 99 data set [[23]](#_bookmark48) contains 23 attack types and their names are shown in [Table 1](#_bookmark12) and its features are grouped as,

1. *Basic features*

It encompasses all the attributes of TCP/IP connection and leads to delay in detection.

1. *Traffic features*

It is evaluated in accordance with window interval & two features as same host and same service.

* 1. *Same host feature*

It examines the number of connections for the past 2 s that too from the same destination host. In other words, the probability of connections will be done in a specific time interval.

* 1. *Same service feature*

It examines the number of connections in a partic- ular time interval that too posses same service.

1. *Content features*

Dos & probe attack have frequent intrusion sequen- tial patterns than the R2L & U2R. Because these two attacks include many connections to several hosts at a particular time period whereas R2L and U2R perform only a single connection. To detect these types of attacks, domain knowledge is impor- tant to access the data portion of the TCP packets. Ex. Failed login, etc. these features are called as con- tent features.

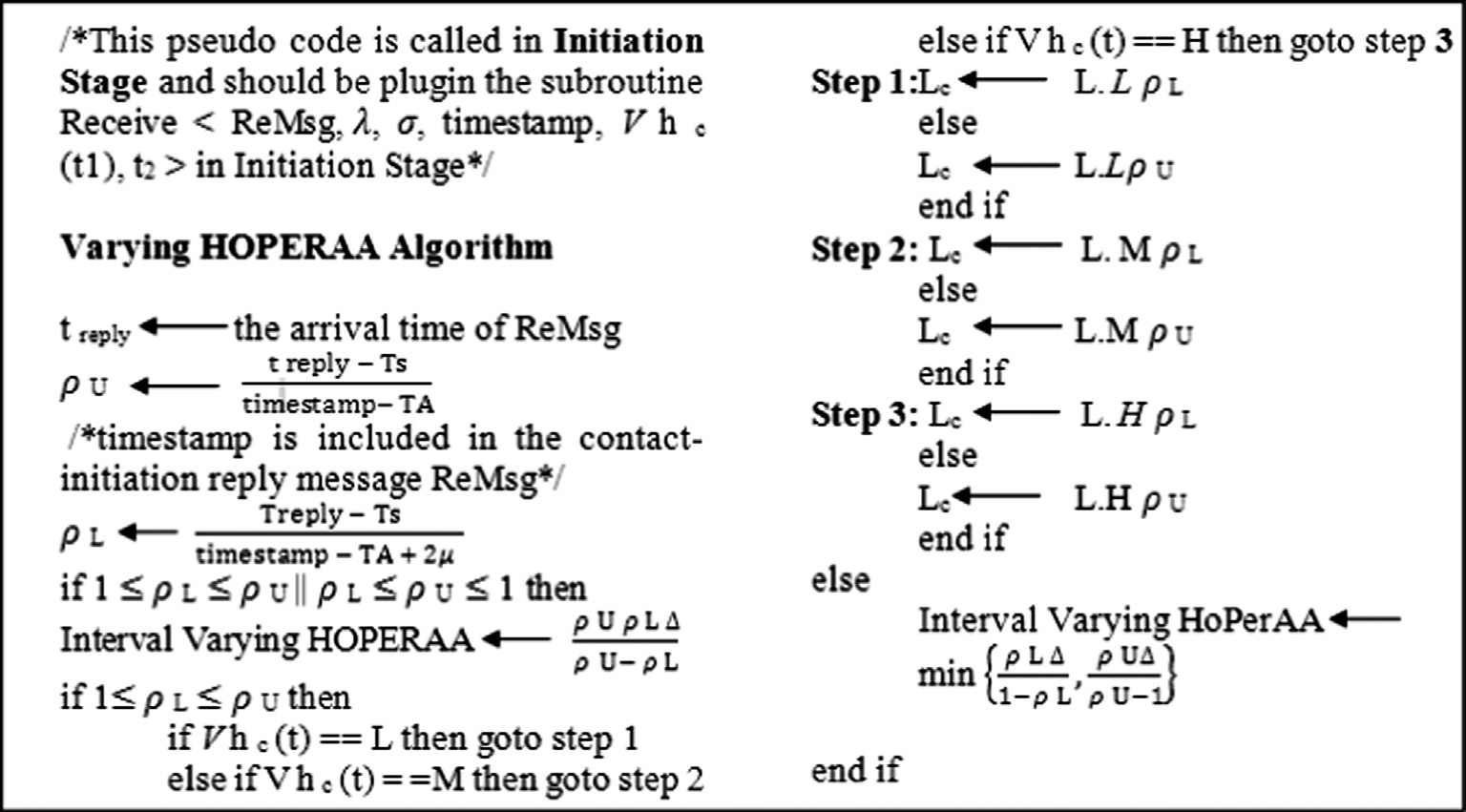


Figure 6 Pseudocode of Varying HOPERAA Algorithm.

Effective approach toward Intrusion Detection System using data mining techniques 43

Table 1 Name of the attacks classified under 4 groups.

Denial of service

Probes

Remote to local User to root

Back, land, neptune, pod, smurf, teardrop

Satan, ipsweep, nmap, port sweep

ftp\_write, imap, guess\_passwd, phf, spy, warezclient, multihop, warezmaster Buffer\_overflow, load module, Perl, root kit

1. Results and discussions

|  |  |
| --- | --- |
| Table 2 Reduced feature set for each attack after discriminate analysis. | |
| Attack name | Related features |
| Back | 27, 28, 31, 36, 38, 39, 40, 41 |
| Buffer overflow | 14, 16, 36, 37 |
| ftp\_write | 1, 5, 6, 9, 10, 12, 13, 16, 17, 19, 22, 31, 34, 35, 36, 37 |
| guess\_pwd | 27, 28, 31, 38, 39, 40, 41 |
| Imap | 8, 12, 24, 25, 28, 38 |
| ipsweep | 5, 27, 28, 31, 34, 35, 36, 37, 40, 41 |
| Land | 1, 6, 15, 17, 34, 35, 38, 39 |
| load module | 12, 23, 25, 36, 38, 40, 41 |
| multihop | 1, 5, 6, 10, 12, 17, 22, 31, 34, 35, 36, 40 |
| Neptune | 29, 30, 34, 35 |
| Nmap | 5, 25, 26, 34, 35, 36, 38, 39 |
| Normal | 31, 34, 35, 36, 37 |
| Perl | 1, 5, 34 |
| Phf | 5, 10, 15, 24 |
| Pod | 34, 35, 36 |
| port sweep | 1, 25, 26, 27, 28, 29, 30, 34, 35, 36, 38, 39, 40, 41 |
| root kit | 9, 10, 11, 12, 13, 16, 17, 341 |
| Satan | 1, 6, 12, 27, 29, 30, 34, 35, 36, 39, 40 |
| Smurf | 1, 5, 6, 12, 37 |
| Spy | 12, 15, 17, 18, 19, 34, 39 |
| teardrop | 5, 26, 30, 34 |
| warezclient | 5, 15, 28 |
| warezmaster | 1, 34 |
|  |  |

KDD Cup 99 data set has been used in this research of which 60% is treated as training data and 40% is considered as test- ing data. The proposed framework has been implemented in MatLab10 and Java using data mining techniques. Perfor- mance of four proposed methods such as,

* Classification of network data using EDADT algorithm.
* Proposed Hybrid IDS.
* Performance of Semi-Supervised Approach for IDS and,
* Mitigating DDoS attacks using Varying Clock Drift Mechanism.

has been evaluated in terms of accuracy and FAR values as shown below.

* 1. *Classification of network data using EDADT algorithm*

Data mining technology to Intrusion Detection Systems can mine the features of new and unknown attacks well, which is a maximal help to the dynamic defense of Intrusion Detection System. This work is performed using Machine learning tool with 5000 records of KDD Cup 99 data set to analyze the effectiveness between our proposed method and the traditional algorithms. The performance of the var- ious algorithms measured in terms of accuracy, Sensitivity, Specificity and false alarm rate. [Table 2](#_bookmark13) represents the related features for a particular attack. For all 23 attacks, the related features are calculated by enabling the threshold value. If the attribute satisfies the specified constraints then the attribute is chosen as the related features of particular attack.

[Table 3](#_bookmark14) represents the rule structure for the KDD Cup 99 data set. Using this rule structure the data set can be easily classified in the future. If any new type of attack is found it can also be added in the in this profile for better classification results. [Fig. 7](#_bookmark15) represents the overview of EDADT algorithm. The advantage of using this algorithm is that it splits the re- cord into the attack or normal without leaving any data as unclassified. Moreover the data set can be classified as normal or attack record efficiently.

Since many data mining algorithms are available this is an innovative and efficient work toward IDS to detect the ongoing attacks over a large network. However, this algorithm reduces the space occupied by the dataset. So it would be useful for the network administrator/experts to avoid the delay between the arrival and the detection time of the attacks respectively. It also produces a less false alarm rate and computation time in real time. Performance of various existing algorithms is compared with the proposed EDADT algorithm for Intrusion Detection

Systems. Information gain is applied to improve the accuracy of Intrusion Detection System through feature extraction. Then, select the training and testing data set. After a EDADT model is generated the test data set is applied on it to evaluate the Detection Rate (DR) and false alarm rate (FAR) values. Hence, compare the values obtained in the previous step and finally represent the results by measures of performance of the model. [Table 4](#_bookmark16) represents the accuracy, sensitivity and spec- ificity values for C4.5, SVM, C4.5 + ACO, SVM + ACO, C4.5 + PSO, SVM + PSO and Improved EDADT algo- rithms. The ROC curve is shown for a graphical plot of sensitivity and specificity.

Based on values obtained, the accuracy of C4.5 is 93.23%, the accuracy of SVM is 87.18%, the accuracy of C4.5 + ACO is 95.06%, the accuracy of SVM + ACO is 90.82%, the accuracy of C4.5 + PSO is 95.37%, the accuracy of SVM + PSO is 91.57% and the accuracy of Improved EDADT is 98.12%. Finally, an improved EDADT took high- est accuracy percentage when compared to all six classifica- tion based algorithms. [Fig. 8](#_bookmark17) specifies the corresponding chart for the result obtained in [Table 4](#_bookmark16). [Fig. 9](#_bookmark18) illustrates the build time of C4.5, SVM, C4.5 + ACO, SVM + ACO, C4.5 + PSO, SVM + PSO and Improved EDADT algo- rithms. C4.5 + PSO take more time to build the model. SVM + ACO takes less time than the proposed algorithm but provide less accuracy percentage than the other. However

44 G.V. Nadiammai, M. Hemalatha

Table 3 Rule structure for KDD Cup 99 dataset.

Rule no. Attack description Attack type

1. protocol = ICMP, service = ecr\_i, src\_byte = 1032, flag = SF, host\_count = 255
2. protocol = tcp, service = private or ctf, flag = SO or SF, serror\_rate = 1, srv\_serror\_rate = 1
3. protocol = ICMP, service = SF or SH, src\_byte = 8, same\_srv\_rate = 1, srv\_diff\_host\_rate = 1
4. protocol = tcp, service = http, flag = SF or RSTFR,

src\_byte = 54540, dst\_byte = 7300 or 8314, same\_srv\_rate = 1, srv\_count P 5

1. protocol = UDP, service = private, flag = SF, src\_byte = 1, dst\_host\_count = 255, dst\_host\_same\_src\_port\_rate = 1
2. protocol = UDP, service = SF, src\_byte = 28, wrong fragment = 3, dst\_host\_count = 255
3. protocol = icmp, service = eco\_i, flag = SF, src\_byte = 18, count = 1, dst\_host\_count = 1
4. protocol = TCP, service = Private or remote\_ic, dst\_host\_count = 255, dst\_host\_srv\_count = 1
5. duration = 26 or 134, protocol = tcp, service = FTP or login, flag = SF, logged\_in = 1
6. protocol = tcp, service = telnet, flag = RSTO, src\_byte = 125 or 126, dst\_byte = 179, hot = 1, num\_failed\_login = 1
7. service = imap4, count 6 4, dst\_host\_same\_srv\_rate = 1,

dst\_host\_srv\_count < = 1

1. service = tcp, flag = telnet or ftp\_data, flag = SF, dst\_host\_srv\_count 6 3, dst\_host\_same\_src\_port\_rate = 1
2. duration = 377or 299, service = tcp, flag = telnet, dst\_host\_count = 255, dst\_host\_diff\_srv\_rate = 0.01
3. protocol = tcp, service = ftp, flag = SF, src\_byte > 980, dst\_byte P 1202, hot P 3, dst\_host\_count = 255
4. protocol = tcp, service = telnet or ftp\_data, flag = SF, loggin\_in = 1, dst\_host\_same\_srv\_rate = 1
5. duration P 2, protocol = tcp, service = ftp or ftp\_data, flag = SF, dst\_host\_count > 2, dst\_host\_srv\_count P 1
6. protocol = tcp, service = telnet, flag = SF, dst\_host\_count = 1, dst\_host\_same\_src\_port\_rate = 1
7. duration P 25, protocol = tcp, service = telnet, flag = SF, logged\_in = 1, dst\_host\_srv\_count 6 2, dst\_host\_diff\_srv\_rate 6 0.07
8. protocol = tcp, service = telnet or ftp, flag = SF, dst\_host\_count = 255, dst\_host\_diff\_srv\_rate = 0.02
9. protocol = tcp, service = finger, flag = SO, land = 1, srv\_count = 2, dst\_host\_srv\_serror\_rate P 0.17
10. protocol = ICMP, service = ecr\_i, flag = SF, src\_byte = 1480, wrong\_fragment = 1, dst\_host\_count = 255, dst\_host\_diff\_srv\_rate = 0.02
11. protocol = tcp, service = telnet, flag = SF, dst\_host\_count = 255, dst\_host\_serror rate = 0.02

Smurf Neptune Nmap Back Satan teardrop ipsweep portsweep ftp\_write

guess\_passwd Imap Multihop Spy warezclient

Buffer\_overflow warezmaster load module Perl

root kit Land Pod Phf

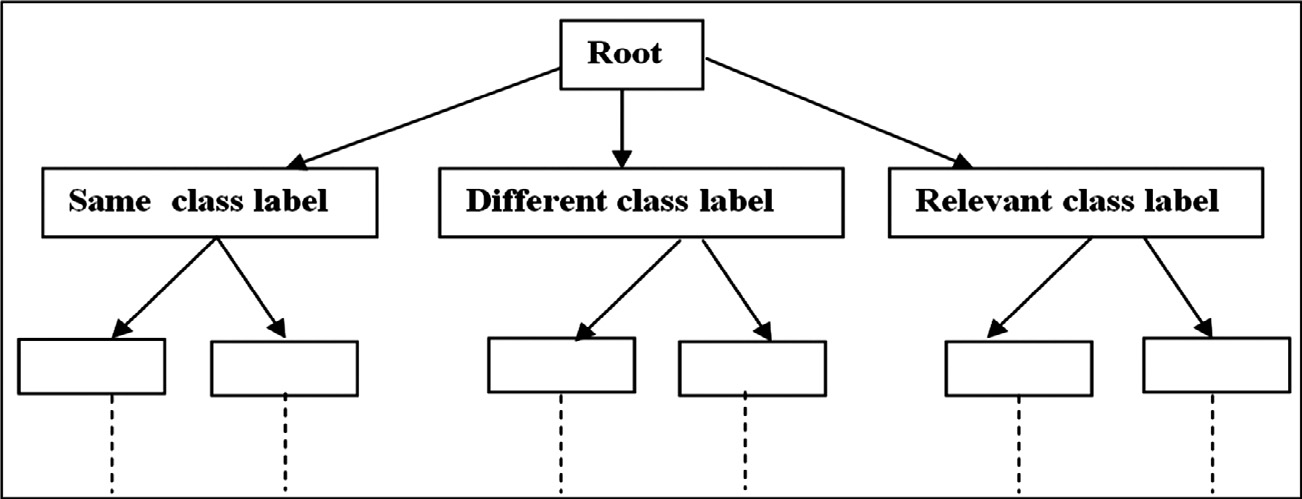


Figure 7 Overview of proposed EDADT algorithm.

Effective approach toward Intrusion Detection System using data mining techniques 45

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Table 4 Performance of proposed EDADT vs. existing algorithms. | | | | |
| Algorithms | Sensitivity (%) | Specificity (%) | Accuracy (%) | FAR (%) |
| C4.5 | 86.57 | 82.00 | 93.23 | 1.56 |
| SVM | 83.82 | 64.29 | 87.18 | 3.2 |
| C4.5 + ACO | 89.26 | 85.42 | 95.06 | 0.87 |
| SVM + ACO | 87.42 | 67.95 | 90.82 | 2.42 |
| C4.5 + PSO | 92.51 | 88.39 | 95.37 | 0.72 |
| SVM + PSO | 90.06 | 70.80 | 91.57 | 1.94 |
| Proposed EDADT | 96.86 | 92.36 | 98.12 | 0.18 |
|  |  |  |  |  |

[Fig. 10](#_bookmark19) shows the performance of existing and proposed EDADT algorithms based on false alarm rate (FAR). Thus the proposed EDADT Algorithm effectively detects attack with less computational time and false alarm rate.

* 1. *Proposed Hybrid Intrusion Detection System*

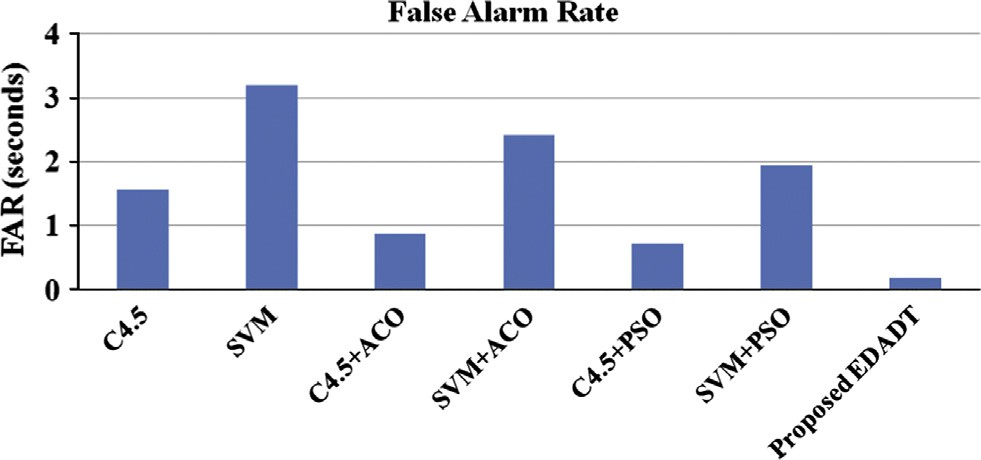
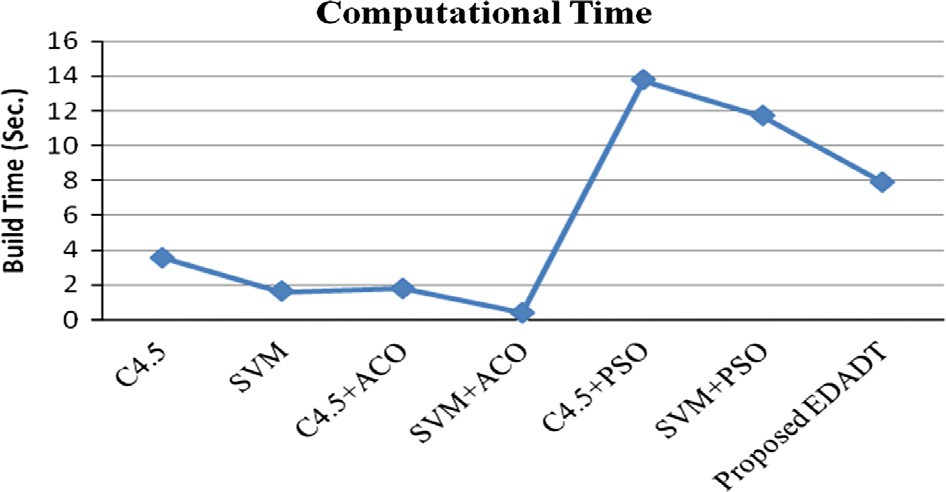
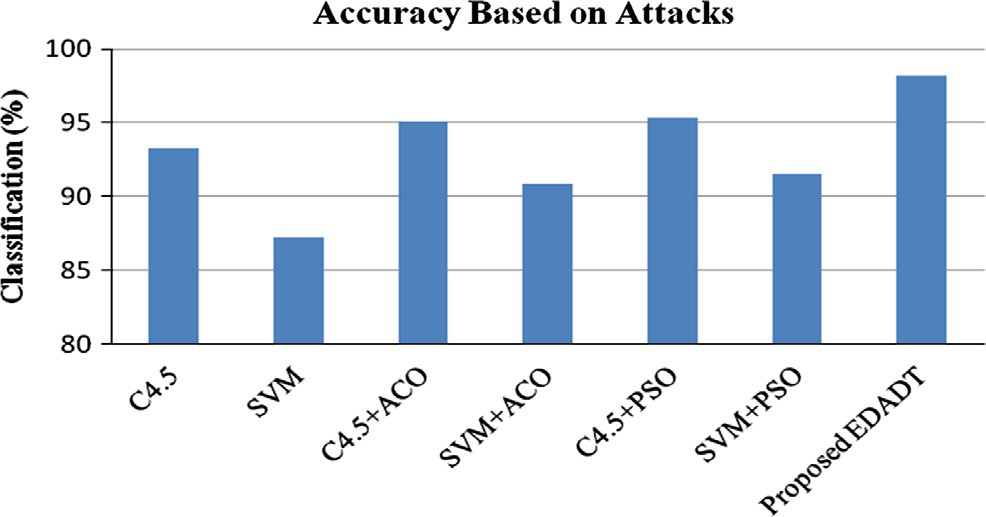


Figure 8 Results of EDADT algorithm vs. existing algorithms.

Figure 9 Computational time taken for the existing and proposed EDADT algorithms.

Figure 10 False alarm rate of proposed EDADT algorithm vs. existing algorithms.

the improved EDADT takes less time when compared to C4.5 + PSO and SVM + PSO and provides better accuracy in terms of all existing algorithms.

The misuse based and anomaly based approach has been taken for the study. Hybrid IDS is developed to overcome the human interaction toward pre-processing. Most of the evaluation of intrusion detection is based on proprietary data and results are not reproducible. To solve this problem, KDD Cup 99 (2009) has been used. Public data availability is one of the ma- jor issues during evaluation of Intrusion Detection System. Mixed data set (real time + simulated) has been used for this study. Out of 500 data instances, 320 instances involved in the training phase and remaining 180 instances are taken for test- ing phase. Each data include the source IP address, destination IP address, state of the packet and so on. Data can be analyzed with the help of the snort rules that are predefined within it and anomaly score for each packet. Under anomaly based ap- proach, we have four types of statistical methods like Packet Header Anomaly Detection (PHAD), Network Traffic Anom- aly Detector (NETAD), Application Layer Anomaly Detector (ALAD), Learning Rules for Anomaly Detection (LERAD) respectively. Analysis is done based on the scenarios given below:

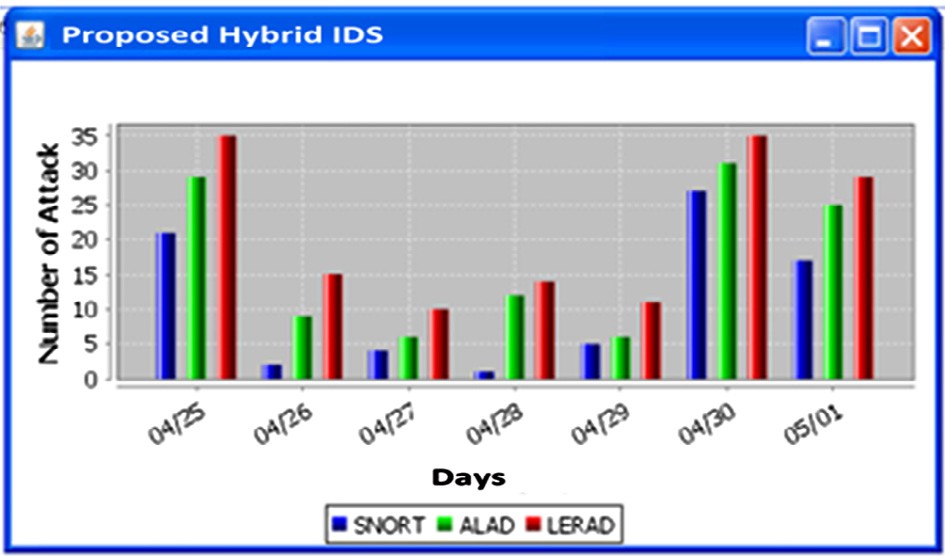
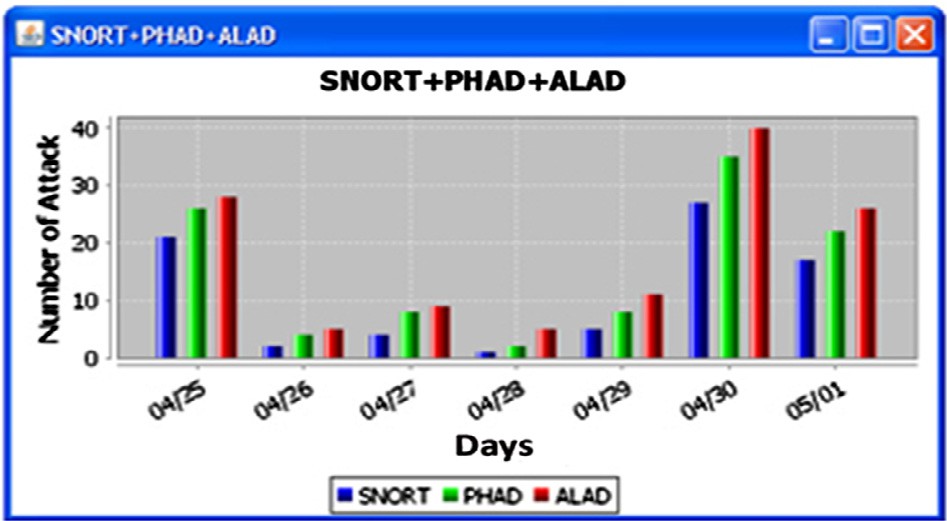
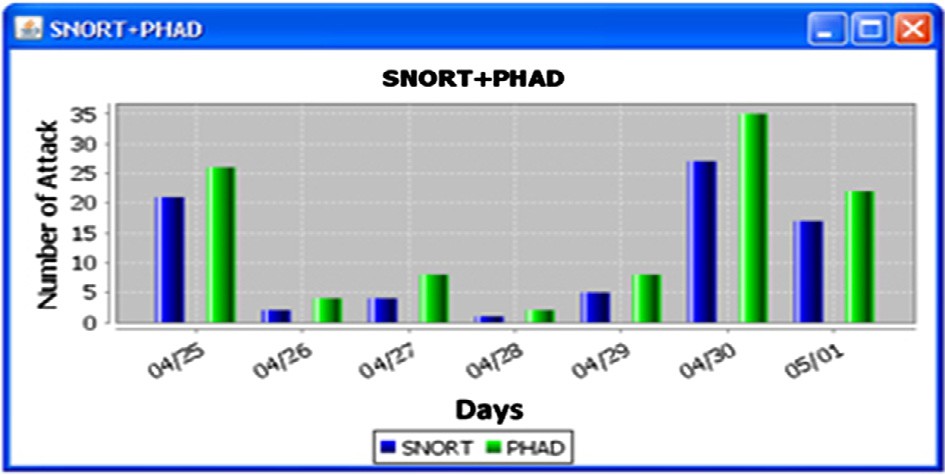
* Based on SNORT.
* Based on SNORT + PHAD.
* Based on SNORT + PHAD + ALAD.
* Based on SNORT + ALAD + LERAD.
  + 1. *Performance of SNORT*

SNORT is tested on real time traffic and simulated data set (one week data including attack) and attacks detected are listed day by day. The files have been downloaded from and LAN network. Attack detected on a daily basis is shown in [Fig. 11](#_bookmark23). SNORT has detected 77 attacks out of 180 attacks without adding any anomaly based approaches.

* + 1. *Performance of SNORT + PHAD*

Attacks detected by SNORT and PHAD on their own and results in the Hybrid Intrusion Detection System are shown in [Fig. 12](#_bookmark20). It is understood that after adding PHAD with Snort it detects better than before. The number of attacks detected by SNORT increased from 77 to 105 in SNORT + PHAD.

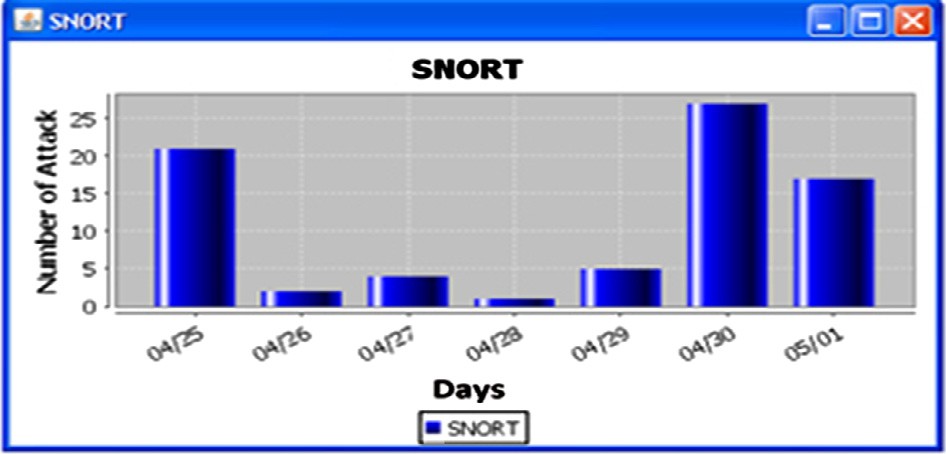
46 G.V. Nadiammai, M. Hemalatha



* + 1. *Performance of SNORT + PHAD + ALAD*

When PHAD and ALAD are added to the snort it detects more attacks than before. It is clearly shown in the graph [Fig. 13](#_bookmark21). The number of attacks increases while adding PHAD and ALAD with SNORT the IDS becomes powerful. The number of attacks detected by SNORT + PHAD increased from 105 to 124 in SNORT + PHAD + ALAD version of IDS. The main reason is Snort detects the attacks based on rule definition files but PHAD and ALAD detect using packet header and network protocol.

* + 1. *Proposed Hybrid IDS (SNORT + ALAD + LERAD)*



Attacks detected by SNORT + ALAD + LERAD on their own and results in the Hybrid Intrusion Detection System (SNOR- T + ALAD + LERAD) is shown in [Fig. 14](#_bookmark22). After adding SNORT + ALAD + LERAD, the IDS gives better results when compare with other methods. The number of attacks detected by SNORT + PHAD + ALAD increased from 124 to 149 in SNORT + ALAD + LERAD (Hybrid IDS) version of the IDS.

* 1. *Performance of Semi-Supervised Approach for IDS*

It is very hard for IDS to collect and analyze the data. Based on this issue, rule based technique has been applied. But if there is any little change in the data then the rule seems to be meaningless. To accomplish this task, we go for Semi- Supervised Approach. In supervised approach labeled data can be taken for training phase and unlabeled data have been taken for testing phase. Usually the network data are unla- beled. It needs the security experts to label the unlabeled data which is expensive and time consuming. Because supervised approach needs the formal labeling of data to analyze whether the testing data are attacked or a normal one. But it is not real- istic in real time. So Semi-Supervised Approach is considered as most significant one. It requires only a small quantity of la- beled data with large amount of unlabeled data. This method is done on the assumption. By analyzing the distance between the data points labeling is done. These data points are consid- ered as most confident data. In turn, these data are taken as training data and corresponding testing data are applied to label the unlabeled one. The parameters of SVM are tuned between 0 and 1. This process continues till the bias value is same for many trials. Totally 5000 datasets are taken in this approach. Training phase includes both the labeled and unla- beled data together and testing with unlabeled data.

Based on the predicted label, the accuracy will be calcu- lated. The four classes of the data set namely Dos, Probe,

Figure 11 Attacks detected by SNORT on daily basis.

Figure 12 Attacks detected by SNORT + PHAD on daily basis.

Figure 13 Attacks detected by SNORT + PHAD + ALAD on daily basis.

Figure 14 Attacks detected by proposed IDS on daily basis.

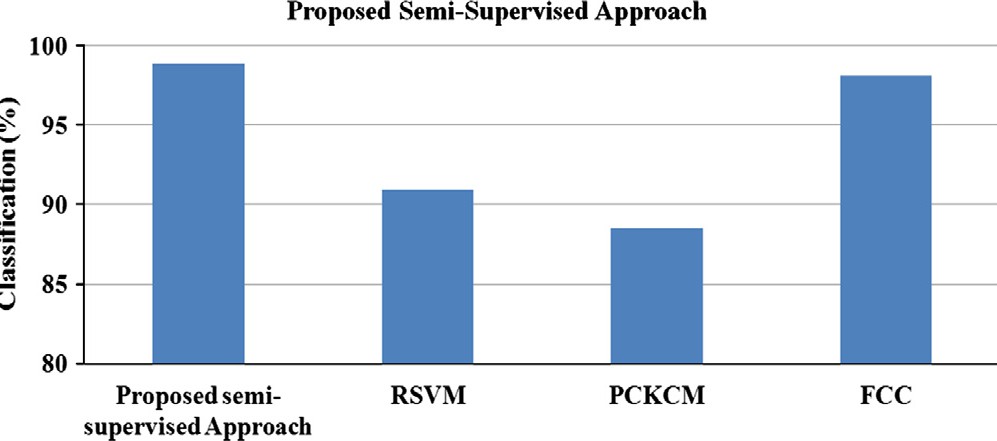
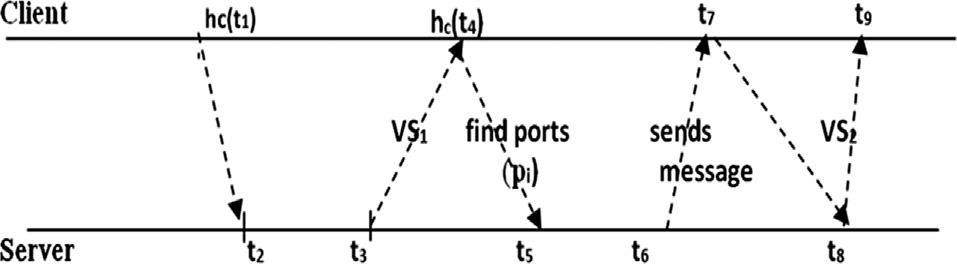


Figure 15 Performance of proposed Semi–Supervised Approach.

Effective approach toward Intrusion Detection System using data mining techniques 47



R2L and U2R label have been labeled as 1, 2, 3 and 4 respec- tively. In [Fig. 15](#_bookmark24), the proposed Semi-Supervised Approach is compared with the existing algorithms like Reduced Support Vector Machine, Semi-Supervised clustering algorithm (PCKCM) and Fuzzy Connectedness based Clustering. Among these, the proposed Semi-Supervised Approach shows 98.88% in terms of accuracy and [Fig. 16](#_bookmark26) illustrates the false alarm rate out of 6 existing methods such as RSVM, one step Markov, order 10 Markov, Markov chain + drift, PCKCM and FCC. In these comparisons, the proposed semi supervised approach shows 0.5% false alarm rate respectively.

* 1. *Mitigating DDoS attacks using Varying Clock Drift Mechanism*

By performing DoS attack the adversary captures zombies of machine in the network and explores attack by flooding pack- ets to make the server busy, which are harder to deflect and promotes congestion thereby the victim resources will be flooded with request from hundreds and thousands of multiple sources. Port hopping mechanism is carried out by. In the author presents the synchronous communication based on acknowledgment in the presence of fixed clock drifts, main- tained by Hopping Period Alignment and Adjustment (HOP- ERAA) algorithm where the client clock relates to that of the server. But any acknowledgment loss and faster or slower clock drift of client enables the adversary to cause a direct at- tack on it. In the existing work the interval of HOPERAA

Figure 17 Client and server communication in execution of Varying HOPERAA Algorithm.

* Computation ratio is calculated to prove the efficiency of the server’s receiving capability.
* Due to the variable clock drift the message loss do not cause severe damage.

Lemma. Suppose when we use server’s clock as reference clock and the client sends the request message at time *t*1 with timestamp *hc* (*t*1) and the message received by the server at time *t*2 and reply to the message with variable clock drift time at time *t*4 as found in [Fig. 22](#_bookmark32). The client sets the clock to Vs1 and executes the Varying HOPERAA Algorithm to adjust the hopping period and receives the port sequence randomly. At time *t*5, the client searches, finds the port and starts sending the message to the specific sequence. In time *t*7 the server analyzes the state of the message and sends a reply at *t*7 and this process continues till *t*8 respectively. Then we have,

grows gradually over time and opens ports seems to be easy cause for the adversary to enable possible attacks. Moreover,

the servers need not to worry about synchronization of clock

*Vhc*(*t*9)— *Vhc*(*t*4)

(*t*8)— (*t*5)+ 2l

6 *pc*

6 *Vhc*(*t*9)— *Vhc*(*t*4)

(*t*8)— (*t*5)

(2)

during multiparty communication. To overcome this issue, the Varying HOPERAA Algorithm has been proposed to ad- just the deviation in terms of hopping between client and ser- ver with varying clock drift. The message overhead between client and server & its average value is observed to evaluate the performance of the proposed algorithm.

Based on the previous work we also made three experi- ments to validate the efficiency of the proposed algorithm. It includes,

* The average number of contact initiation trails that the cli- ent need to communicate with the server.
* The growth of Varying HOPERAA execution intervals is greatly reduced that are made to estimate the client’s clock drift.

From this lemma, it can be proved that the message transfer

delay can be avoided by selecting the appropriate port based on the length of the data. Thus, the proposed Varying HOPERAA can reduces message transfer delay and also de- creases the execution time to fractions of seconds as shown in [Fig. 17](#_bookmark25).

[Fig. 18](#_bookmark27)(a and b) indicates the average number of the con- tact–initiation part in various time unit say for example 1000 and 5000 ms. For 1000 ms the number of trails is estimated likewise for 5000 ms. Compared to the existing results, the pro- posed algorithm provides 3.6 trails for 9000 ports at 1000 ms and 3.2 trails at 5000 ms. Even if we have 10,000–50,000 ports the average number of trails would be reduced from existing algorithm. The message can be of 30, 40, 50 byte and so on. The average time spent in this part will be less than 3 s.

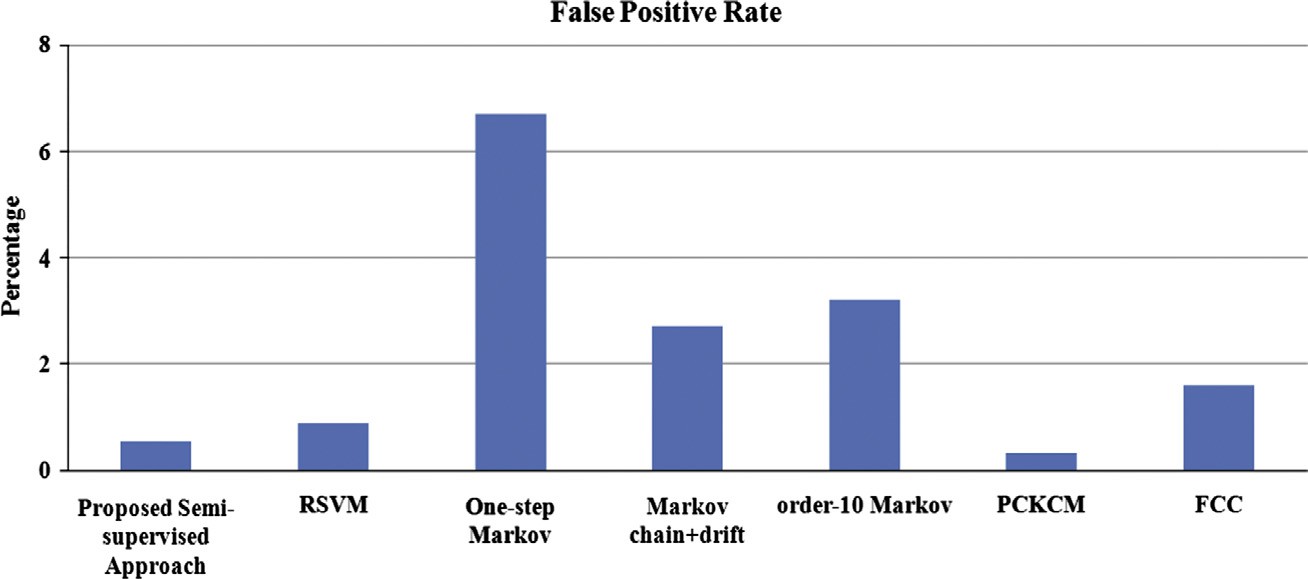


Figure 16 Performance of false alarm rate of proposed Semi-Supervised Approach vs. existing methods.

48 G.V. Nadiammai, M. Hemalatha

Figure 18 Average number of contacts–initiation trial in (a) 1000 and (b) 5000 ms in one contact initiation part.

Figure 19 Length of Varying HOPERAA execution intervals vs. number of Varying HOPERAA executions.

[Fig. 19](#_bookmark28) shows the length of the HOPERAA execution inter- val is estimated with the hopping period of the server. At D = 0.3 L and *pc* = {0.7, 0.9, 1.1, 1.3} the Varying HOPER- AA execution intervals are greatly reduced at every stage and after 10 executions the client keeps sending data within

few minutes. At *pc* = 0.7 the execution interval seems to be 14 ms and if *pc* = 0.9 the execution interval seems to be is 13 ms, and if *pc* = 1.1 the execution interval seems to 10 ms and at *pc* = 1.3 the HOPERAA execution interval is over 9 ms approximately. If the value of the *pc* increases then the number of executions reduces. After one Varying HOPERAA execution, the client will know whether the clock rate is faster or slower than the server because it receives the varying clock drift details from the server according to the state of the mes- sages. Hence the hopping strategy is reduced within short time sequences respectively.

[Fig. 20](#_bookmark29)(a) and (b) shows the receiving capability of server with various clock drift of the client is taken into account. The clock drift of the client pc is {0.6, 0.7, 0.8, 1.1, 1.2, 1.3,

1.4, 1.5} D = {0.1 L, 0.2 L, 0.3 L} and l = 40 and 100 ms.

In this simulation scenario the client executes Varying HOP- ERAA Algorithm 10 times and server keeps the state of the message in the form of threshold value to issue the range of port numbers in accordance with the message sending ratio. Thus, the Varying HOPERAA algorithm shows 99% as receiving capacity at the 100 ms rate and does not show much deviation for 40 ms rate as shown in existing algorithm. In the previous work, only security issues in terms of contact

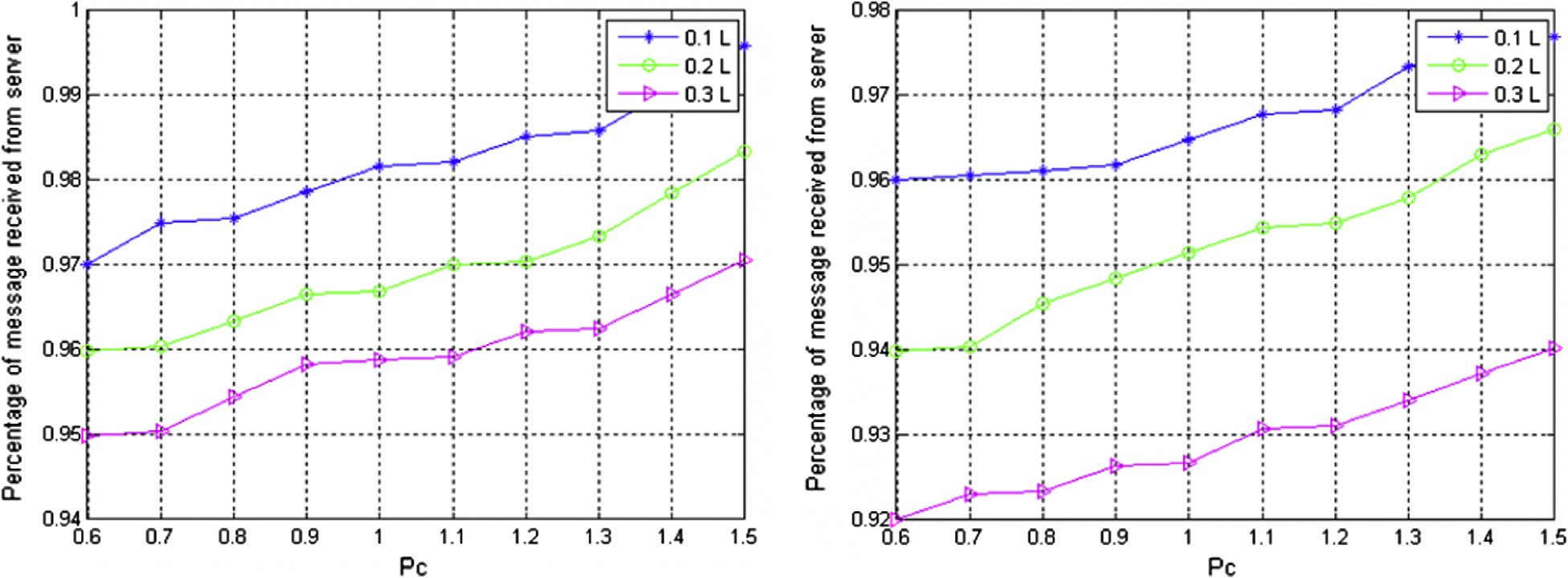
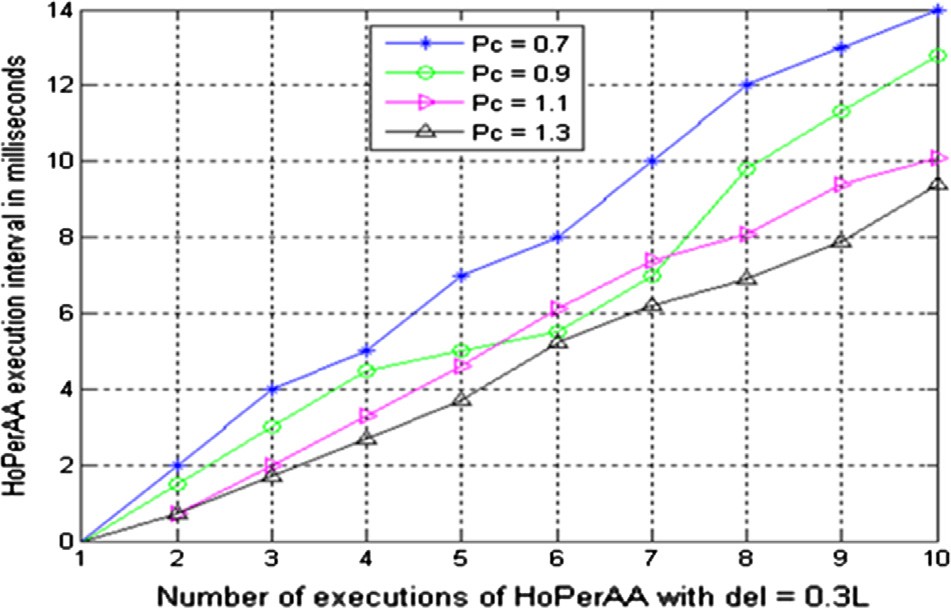
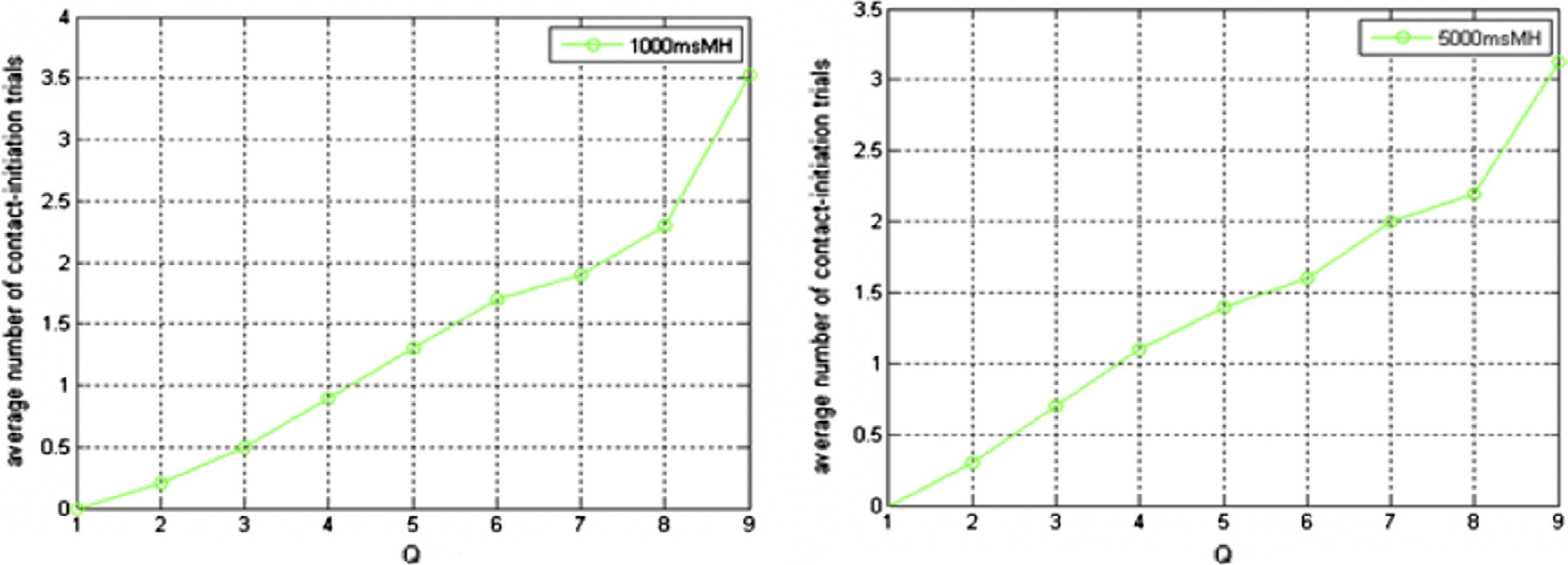


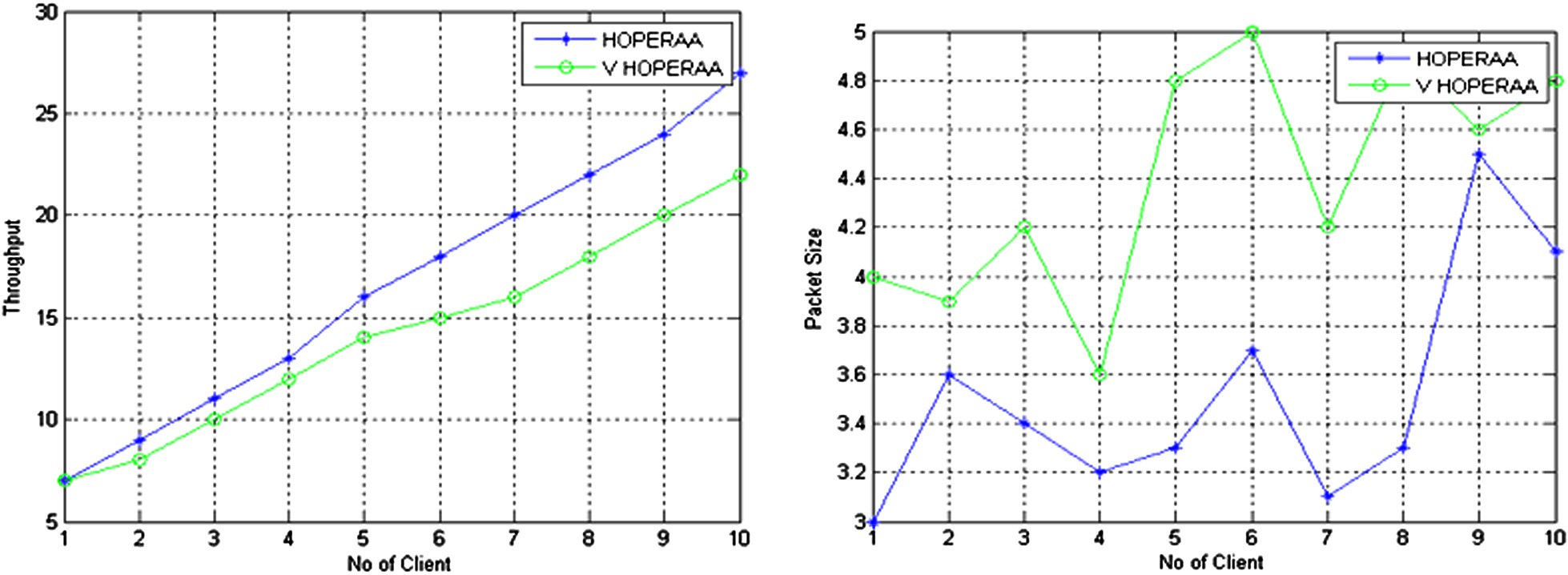
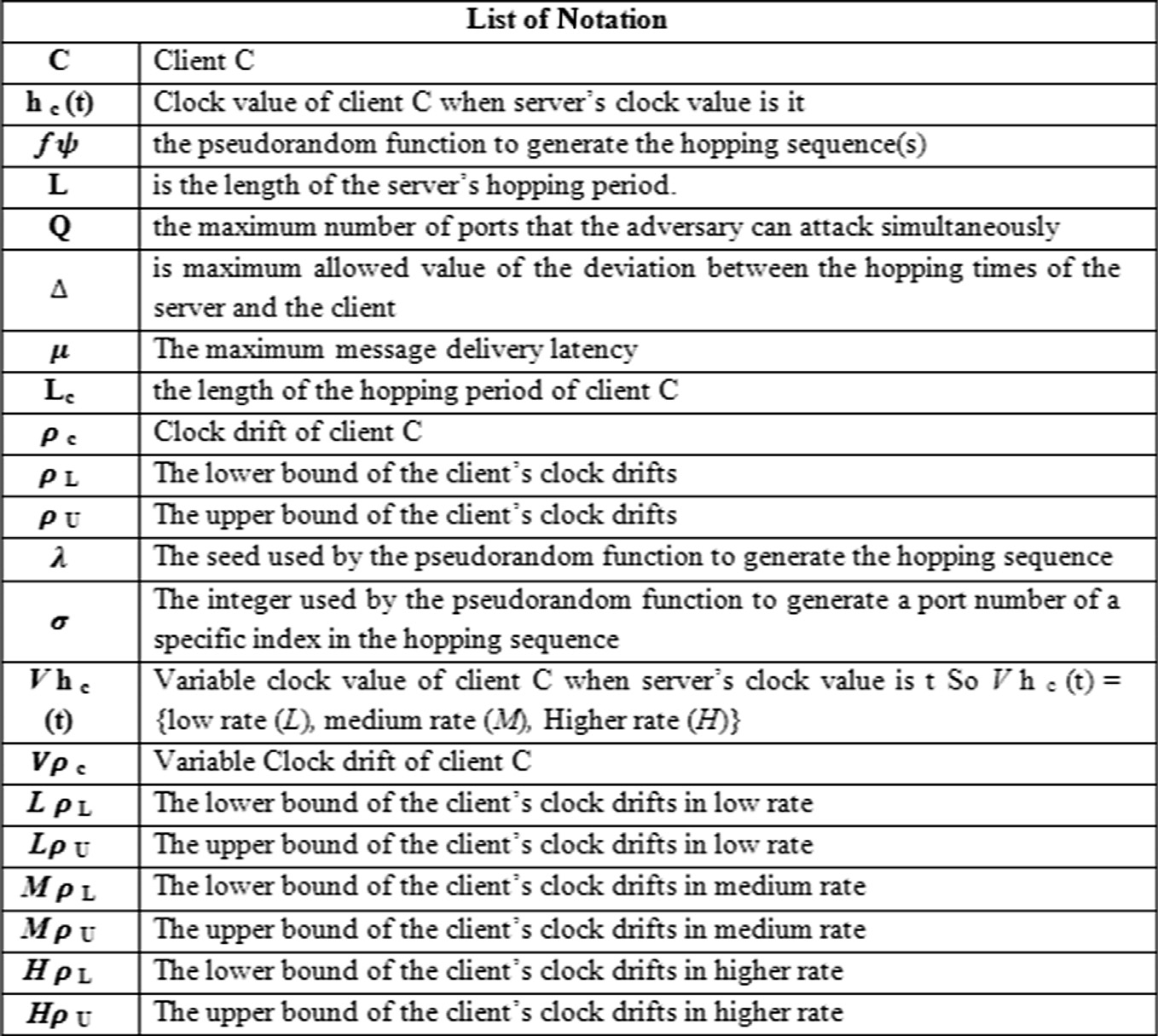
Figure 20 Receiving percentage of the server with l is set to (a) 40 and (b) 100 ms.

Effective approach toward Intrusion Detection System using data mining techniques 49

Figure 21 Performance of proposed varying HOPERAA and existing HOPERAA algorithm in terms of throughput, packet size total and number of clients.

Figure 22 List of notation used in proposed Varying HOPERAA Algorithm.

initiation trials for milliseconds, and sever’s receiving capacity are taken into account. Apart from that, in our work the performance measures also evaluated using throughput and packet size metrics. If number of client increases then through- put seems to be increasing gradually. The comparison has been made with the existing HOPERAA and proposed varying HOPERAA algorithm. The result shows that the throughput increases as 5, 8, 11, 13, 16, 18, 20, 22, 24 and 27 for 10 clients and corresponding packet size also shown in [Fig. 21](#_bookmark30) respectively.



1. Conclusion and future work

Based on the first issue, the proposed EDADT algorithm re- duces the actual size of the dataset and helps the administrator to analyze the ongoing attacks efficiently with less false alarm rate respectively. It shows 19.4% better than C4.5, 18.8% bet- ter than SVM, 19.6% better than C4.5 + ACO, 19.2% better than SVM + ACO, 19.7% better than C4.5 + PSO, 19.3% better than SVM + PSO in terms of accuracy. Based on the second issue, the proposed Hybrid IDS performs well by

50 G.V. Nadiammai, M. Hemalatha

detecting 149 attacks out of 180 (83%) attacks after training in one week attack free traffic data. This approach helps to over- come the human interaction toward preprocessing. Regarding third issue, the proposed Semi-Supervised approach is 18.1% better than RSVM, 18.9% better than PCKCM, 19.9% better than the FCC. To solve the overwhelming problem of super- vised and unsupervised methods, the semi supervised approach has been carried out. Finally, based on the mitigation of DDoS attack scenario, the port hopping concept is used depending upon the message length. Hence the message loss is greatly re- duced and it does not create severe damage if happens. Both the security and performance measures with a variable clock drift mechanism have been evaluated. With the help of varying clock drift, the client can easily communicate with the server with minimum contact initiation trails and the improved max- imum delivery latency has been achieved. Varying HOPERAA algorithm shows 99% as receiving capacity at 100 ms and slightly deviates for 40 ms rate with better throughput capac- ity. Our experimental results proved that the proposed algo- rithms solves the above mentioned issues and detects the attacks in an effective manner compared with other existing works. Thus, it will pave the way for an effective means of intrusion detection with better accuracy and reduced false alarm rates.

In the future, two issues such as lack of resource consump- tion information and lack of model adjustment information techniques have been deployed to achieve an automatic Intru- sion Detection System.

References

1. [Anderson JP. Computer security threat monitoring and surveil-](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0005) [lance. In: Technical report. Fort Washington, Pennsylvania:](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0005) [James P Anderson co.; 1980](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0005).
2. Mohammadreza Ektefa, Sara Memar, Fatimah Sidi, Lilly Suriani Affendey. Intrusion detection using data mining techniques. In: International conference on information retrieval and knowledge management; 2010. p. 200–4.
3. [Holden Alex Nicholas, Freitas A. A hybrid PSO/ACO algorithm](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0015) [for discovering classification rules in data mining. J Artif Evol](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0015) [Appl 2008;2:1–11](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0015).
4. [Ardjani Fatima, Sadouni Kaddour. Optimization of SVM mul-](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0020) [ticlass by particle swarm (PSO-SVM). J Mod Educ Comput Sci](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0020) [2010;2:32–8](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0020).
5. [Sethuramalingam S. Hybrid feature selection for network intru-](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0025) [sion. Int J Comput Sci Eng 2011;3(5):1773–9](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0025).
6. Mrutyunaya Panda, Ajith Abraham, Manas Ranjan Patra. A hybrid intelligent approach for network intrusion detection. In: International conference on communication technology and system design, procedia engineering, vol. 30; 2011. p. 1–9.
7. [Petrussenko Denis. Incrementally learning rules for anomaly](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0035) [detection. CS-2009-02. Florida: Florida Institute of Technology](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0035) [Melbourne; 2009](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0035).
8. Matthew Vincent Mahoney. A machine learning approach to detecting attacks by identifying anomalies in network traffic. TR- CS-2003-13, Melbourne, Florida; 2003.
9. [Mahoney Matthew V, Chan Philip K. PHAD: packet header](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0045) [anomaly detection for identifying hostile network traffic. In:](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0045) [Technical report CS-2001-04. Florida Institute of Technology; 2001](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0045).
10. Gao Xiang, Wang Min. Applying semi-supervised cluster algo- rithm for anomaly detection. In: IEEE international symposium on information processing, no. 3; 2010. p. 43–5.
11. Qiang Wang, Vasileios Megalooikonomou. A clustering algo- rithm for intrusion detection. In: International conference on data mining, intrusion detection, information assurance, and data networks, security, 5(12), 2005, p. 31–8.
12. Ching-Hao, Hahn-Ming L, Devi P, Tsuhan C, Si-Yu H. Semi- supervised co-training and active learning based approach for multi-view intrusion detection. In: ACM symposium on applied computing, no. 9; 2009. p. 2042–7.
13. Chien-Yi Chiu, Yuh-Jye Lee, Chien-Chung Chang. Semi- super-vised learning for false alarm reduction. In: Industrial conference on data mining, no. 10; 2010. p. 595–605.
14. [Li Jimin, Zhang Wei, KunLun Li. A novel semi-supervised SVM](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0080) [based on tri-training for intrusion detection. J Comput 2010;5(4):](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0080) [638–45](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0080).
15. Monowar H. Bhuyan, Bhattacharyya DK, Kalita JK. An effective unsupervised network anomaly detection method. In: Interna- tional conference on advances in computing, communications and informatics, no. 1; 2012. p. 533–9.
16. Lane T. A decision-theoretic, semi-supervised model for intrusion detection. In: International conference on machine learning and data mining for computer security; 2006. p. 157–77.
17. Zhang Fu, Marina Papatriantafilou, Philippas Tsigas. Off-the- wall: lightweight distributed filtering to mitigate distributed denial of service attacks. In: IEEE international symposium on reliable distributed systems, no. 31; 2012. p. 207–12.
18. Zhang Fu. Marina Papatriantafilou, Philippas Tsigas, Wei Wei. Mitigating denial of capability attacks using sink tree based quota allocation. In: ACM symposium on applied computing, no. 25; 2010. p. 713–18.
19. Zhang Fu. Marina Papatriantafilou, Philippas Tsigas. CluB: a cluster based framework for mitigating distributed denial of service attacks. In: ACM symposium on applied computing, no. 26; 2011. p. 520–27.
20. Vincenzo Gulisano, Zhang Fu, Mar Callau-Zori, Ricardo Jim Enez-Peris, Marina Papatriantafilou, Marta Patino-Martınez. STONE: a stream-based DDoS defense framework. In: Technical report no. 2012-07, ISSN 1652-926X, Chalmers University of Technology; 2012.
21. [Zhang Fu, Marina Papatrianta Filou, Philippas Tsigas. Mitigat-](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0105) [ing distributed denial of service attacks in multiparty applications](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0105) [in the presence of clock drifts. IEEE Trans Depend Secure](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0105) [Comput 2012;9(3):401–13](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0105).
22. [Catania Carlos A, Garino Carlos Garcı´a. Automatic network](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0110) [intrusion detection: current techniques and open issues. Elsevier](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0110) [Comput Electr Eng 2012;38(5):1062–72](http://refhub.elsevier.com/S1110-8665(13)00041-8/h0110).
23. KDD Cup99 intrusion Detection Dataset. Available from:

<[http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html](http://www.kdd.ics.uci.edu/databases/kddcup99/kddcup99.html)>.