# Optimization of Rules for Intrusion Detection System (Org-Ids)

**S. Latha, V. Sinthu Janita Prakash**

***Abstract:*** ***Computer Networks are prone to be attacked by a number of network attacks. To protect an individual system or the entire network from the malicious behaviour, a high level security system is needed. Intrusion detection system (IDS) is a system which give such protection to the network from the intrusions like misuse, unauthorised access etc. Even though many forms of new attacks come into practice, providing the security for the system from the known attack is also a challenging task. The solution is a Signature based IDS which is a potential tool to identify the known attack, sending alert and protect the networks. So a novel signature based IDS(ORG-IDS) with four phases such as Feature Selection, Classification, Optimized Rule generation and Pattern matching is proposed. For any efficient signature based IDS, it should have the signature rules in less number but it should be effective in identifying attacks with good time and memory complexity. In this paper, a new algorithm is proposed for Rule generation phase of proposed IDS to configure the rules by implementing Ant Colony Optimization Technique with Association Rule Mining . The parameters like number of rules, running time and memory utilization are measured and proved that this proposed algorithm outperforms the other existing algorithms.***

***KEYWORDS: Intrusion Detection System, Ant Colony Optimization, Association Rule Mining, Apriori algorithm, Aho-Corasick Pattern algorithm.***

## I. INTRODUCTION

The network attacks are in different types. The attack may be an activity which is malicious in nature to compromise the security of a network. The attacks may be either active or passive in nature and the attacker involves in such activity may be a inside or outside the network. The system which identify the threats and protect the network is an IDS. It protects an individual system as Host-based IDS(HIDS) or an entire network as Network based IDS(NIDS) [1].

When an attack is identified by the IDS, the response of the IDS may be a procedural notification(Alert message) or action of termination or session recording for further study to make them as evidences.

Manuscript published on 30 September 2019
* Correspondence Author

**S. Latha***, Department of Computer Science, Cauvery college for women, Trichy, TamilNadu, India, Email: latha.it@cauverycollege.ac.in , slathanagaraj@gmail.com

**Dr. V. Sinthu Janita Prakash,** Department of Computer Science, Cauvery college for women, Trichy, TamilNadu, India, Email:sinthujanita.cs@cauverycollege.ac.in

For identifying the attack, IDS can follow any of two methods as Signature-Based or Anomaly-Based. When there is any significant deviation from normal behaviour of system, the anomaly based IDS reacts. In signature based IDS, it works on the basis of the attack signatures/rules which are stored as patterns. For identifying attacks, it examines the network traffic or incoming packets and react accordingly.

For designing a Signature based IDS, Log files like Web server log, Error log, Sys log, Event log etc. may be considered.

In this paper, a novel NIDS is designed with four phases. They are Feature selection, Classification, Rule Generation and Pattern Matching. A new algorithm (ORGA) is proposed for generating the optimised rules in the Rule generation phase of new IDS(ORG-IDS).

As a flow of this paper, Literature survey on techniques like Feature Selection, Classification, Rule generation and Pattern matching algorithms is presented in section 2. The proposed algorithm ORGA for rule generation phase of ORG-IDS is explained in section 3. The experimental results and discussion are in section 4 and the conclusion is in section 5.

## II. LITERATURE SURVEY

As per Jianglong Song, Wentao Zhao, Qiang Liu and Xin Wang[2], redundant and irrelevant features worsen the performance of IDS. They presented a novel technique wherein, the principal phase conducts an initial quest for an ideal subset of features using Chi-square feature selection. Then those selected features were augmented using the Random Forest (RF).

M.R. Gauthama Raman, et al.[ [3] presented a novel approach based on Helly property of Hypergraph and Arithmetic Residual based Probabilistic Neural Network (HG AR - PNN) to address the classification problem in IDS.

Longjie Li, et al.[ [4] presented a novel hybrid model to detect network intrusion effectively. Gini index was used to select the optimal subset of features, the Gradient Boosted Decision Tree (GBDT) algorithm was adopted to detect network attacks, and the Particle Swarm Optimization (PSO) algorithm was utilized to optimize the parameters of GBDT.

Zbeel BM[5] used Genetic algorithm(GA) for designing IDS where network connection information were considered as chromosomes. DARPA data set was analysed by the network sniffers and results were fed into GA for fitness evaluation. Finally he obtained attack rules for network based IDS.

Wei Li[6] applied Genetic Algorithm (GA) to design a network Intrusion Detection System. He also presented a brief overview on IDS, genetic algorithm, and related detection techniques. Parameters and evolution process for GA were used in detail.

Sadiq AS et al.[7] used Particle Swarm Optimization(PSO) to perform optimized classification with Artificial Neural Network (ANN) classifier for classifying the features of KDD CUP data set into normal and abnormal. Also proved that their algorithm is better than ANN, GA and PSO.

Hajimirzaei et. Al.[8] designed a new IDS based on a combination of a Multi Layer Perceptron (MLP) network, and Artificial Bee Colony (ABC) and fuzzy clustering algorithms. They also suggested that the combination of meta heuristic methods and the genetic algorithm can be used for better results.

## III. ORG-IDS: A SIGNATURE BASED IDS WITH OPTIMISED RULE GENERATION ALGORITHM-ORGA

### A. Phases of ORG-IDS:

In the proposed Signature based IDS(ORG-IDS), there are four phases namely Feature selection, Classification, Rule Generation and Pattern Matching.

- **Feature Selection:**

It is a process of removing insignificant , irrelevant and redundant features in the data set. It simplifies the task, time and increase the efficiency. A high pertinent feature selection mechanism-HPFSM[9] is used in this phase and a novel IDS, IDSFS[10] is developed.

- **Classification:**

It is a function that categorises the items in the dataset to different categories or classes. Enhanced Artificial Network-EANN is involved in classification process of IDSFS and arrived an IDS, IDSFSC[11] with better efficiency.

- **Optimized Rule Generation:**

The classified items-Abnormal and Unknown classes, from the above phase are given as input to this phase. By performing an association analysis between various features of the data set, different association rules can be formed. These rules are important as they are the attack patterns, used by the intruders. To get these attack signatures in proper number, an optimization technique must be used. Here the Ant Colony Optimization(ACO) based Association Rule mining(APriori algorithm) is proposed to generate the optimized rules.

- **Pattern Matching:**

For every IDS, some method is needed to check the attack signatures in the incoming network packets. In this proposed IDS, a multi key word patterN matching algorithm Aho-Corosick is used.

### B. Proposed Rule Structure Generation Algorithm(ORGA):

- ***Association Rule Mining***

Association Rule Mining (ARM) [12] is a method to regulate the manageable association rules for predictabilities among the items in comprehensive swapping information recorded. Let I=$I_1$, $I_2$, ….$I_m$ be a set of m targeted attributes and T be a transaction that contains a group of objects such that T→I. D is a database with exclusive transaction files. An association rule is a repercussion of type *X→Y* where *X* and *Y* are attributes and X ∩ Y = ø. X is known as the antecedent event and Y is known as the consequent. So, the two important principles for association rule mining are *Support* (S) and *Confidence* (C), which designates how often items are in the database and how many times the item sets are presented, correspondingly. The succeeding includes some key classifications in ARM.

***Definition 1:*** Given a collection of n transactions T= {$t_1$, …$t_n$} and m items I = {$i_1$,… $i_m$}, an association rule is expressed in the form:

$$X\ (Antecedent) \rightarrow Y\ (Consequent) \qquad (1)$$

where $X, Y \subseteq I, X \cap Y = \Phi$, the left hand and right side rules are the antecedents and the consequents, respectively.

***Definition 2:*** Support(X) describes the proportion of transactions in T including X.

$$Support\ (X) = \frac{Number\ of\ Transaction\ Containing\ X}{Total\ Number\ of\ Transactions}, X \in T \qquad (2)$$

**Definition 3:** If $Support\ (S) \geq Min\_support$ then S is known as frequent item set where $Min_support$is a threshold value described by users.

**Definition 4:** Transactions Count is $N = |T|$

**Definition 5:**Largest transaction length is *E=Max (|ti|).*

**Definition 6:** The rule confidence is the proportion of transactions in T including item set X which also include item set Y.

$$Confidence\ (X \rightarrow Y) = \frac{Support\ (X \cup Y)}{|Support\ (x)|} \qquad (3)$$

Rules with both Support (X→Y) ≥ Min_Support and Confidence (X→Y) ≥ Min_Confidence are called strong rules. These thresholds values are fixed by users.

- ***Proposed Ant Colony Optimization based Association Rule Mining Algorithm (ORGA)***

The simple idea of Ant Colony Optimization based Association Rule Mining Method (ORGA) is to originate the rules based on Ant Colony Optimization. The rules which have less sustenance and are not necessary will be removed. Only frequent rules are retained and exhibited. It creates a candidate item set and frequent item set by combining and trimming the data. In the algorithm, the maintenance and assurance for the rules are achieved through Apriori algorithm and the ability of rules are obtained through the pheromone migration using Ant Colony Optimization. The fitness function of rules and pheromone of the ants are then updated iteratively in anticipation of the comprehensive optimum solution is extended. As a result, the remaining rules are optimized. It comes from the fact that assessing the rules which have less fitness value is momentous and essential to be transferred. For every occurrence, possibility of rules is updated and next position for the movement is assessed. In this method, those rules which are less fit will be primarily progressed to an enhanced place.

In specific, the proposed ORGA comprises two parts: Rules calculation and Rule optimization. In the first part, data are transmuted to dualistic principles for creating rules using Apriori algorithm.

This sequentially accelerates the database scanning process and consequences in speedy scheming of the backing value. Later, the fitness value for every rule is calculated based on the Support and Confidence values. Similarly, the Local best values are measured and from that Global best rule is obtained. The fitness exploration is subjugated to decide rules that are to be amended using ACO.

The proposed ORGA precedes the result acquired from the Apriori algorithm as input to produce the elevated association rules, which are supportive to scrutinize the recurrent sign to determine the mysterious attack in the network and to discover the interesting rules with recognized attack patterns. The fitness value for every rule is calculated to approximate the prominence of each rule. The fitness value of any rule governed on the backing and assurance which are statistically articulated below:

$$Fit\ (X) = Con\ (X) \times \log(Sup\ (X) \times Length\ (x) + 1) \tag{4}$$

In the above equation, $Fit\ (X)$ is the fitness value of rule type *x*, *Sup(x)* and *Con(x)*are discussed in equations(2) and (3), and *Length(x)* is the length of rule type *x*. ORGA maximizes the fitness value function since big support and confidence values result in great association, which embodies substantial rules. The ant of ACO that has the maximum fitness value is nominated as lbest, and its support and confidence are hired as its sharp edges.

In ORGA each ant portrays a rule and each rule comprises of a sequence of decision variables which can note the position of every item in the rule. Each ant in ORGA has a population old pheromone trail, where population is exemplified as a solution recommended by the ant and pheromone is the rate of fluctuations to the next position with reverence to current position. The population and pheromone are subjectively modified in ORGA thus comprising a pool of random population i.e. rules. Throughout each iteration, all populations are restructured by local best ($Pb_i$) and global best ($Gb_i$) values. Herein, lbest précises the finest result it has accomplished so far. Then, the population apprises its pheromone trail using the following formula.

$$PTOld\,(i,j) = \frac{[\tau_{i,j}]^{\propto}.[\eta_{i,j}]^{\beta}}{\sum_{v \in S}[\tau_{i,v}]^{\propto}.[\eta_{i,v}]^{\beta}} \tag{5}$$

$\tau_{i,j}$ is the amount of pheromone present and $\eta_{i,j}$ is the heuristic function value between the nodes i and j , $\alpha$ and $\beta$ *are the parameters which gives the importance of the pheromone and heuristic values.* control variables in which $\alpha$ control the relative importance of trail and $\beta$ control the visibility. $v$ represent the vertices of the entire ant colony system and S gives the nodes which are in connection with node I but which are not yet visited by the ant.

$$PT_{new}(i,j) = PTold\ (i,j) + FF\ \times PT(p(b),j) - FF \times PT(g(b),j) \tag{6}$$

The calculation of data fitness is done by using De-Jong's function.

$$DPTnew(i,j) = \frac{PTnew\ (i,j)}{\sqrt{\sum_{k=1}^{n}((PTnew(i,j))k)^2}} \tag{7}$$

The position of a population is updated at each iteration as follows:

$$Pnew = Pold + DPTnew\,(i,j) + FF \tag{8}$$

where FF is the fitness function for evaporation of pheromone trail, between 0 and 1, P is the population position, j is the current population, Pb is the Personal best value of a population, Gb is the global best value, and Pold(i, j) is the pheromone trail of i<sup>th</sup> population. By the above tactics, rules are augmented in an efficient way.

- ***Algorithm for Ant Colony Optimization based Association Rule Mining Method***

```
Input: C– Total number of transactions (t1,t2,..tn), i- current rule, Na = number of ants, , xi – i^th position of the ant,  PTi – the pheromone trail of the i^th ant,  Pbi – Personal  best of the i^th ant, $Gb_i$- Global best ant
Output: Optimized Rule Set
Step 1: Rule Generation by using Apriori Algorithm
    Step 1.1: Generate the frequent item sets and rules
    Step 1.2: Calculate support and confidence values of all features in the dataset
Step 2: Initialisation for finding the Personal  best and  Global best Rules
    Step 2.1: Randomly initialize t =0, a=0
    Step 2.2: Initialize $x_i$ where i $\in \{1,2,\dots,N_a\}$, $x_i$ is the ith position of the ant
    Step 2.3: Initialize $PT_i$ where i $\in \{1,2,\dots,N_a\}$, $PT_i$ is the pheromone of the ith ant.
    Step 2.4: $Pb_i \leftarrow x_i$ where $Pb_i$ is the Personal  best of the i^th ant.(Initially $Pb_1$=$x_1$)
    Step 2.5: $Gb_i \leftarrow x_i$ where $Gb_i$ is the  Global best ant.
Step 3:  Finding  & Updating the Personal  best and  Global best values of Rules
    Step 3.1: for t=1 to C
    Step 3.2: for i=1 to $N_a$
    Step 3.3:  FF($x_i$) = Confidence ($x_i$) x log(Support ($x_i$) x length(x) +1)
    Step 3.4:  If (FF($x_i$) < $Pb_i$)
    Step 3.5:  $Pb_i$=FF($x_i$) //Update the Personal  best
    Step 3.6:  Next  i
    Step 3.7:     $Gb_i \leftarrow \min(Pb_1, Pb_2, \dots, Pb_{N_a})$// Update the Global best.
    Step 3.8: for i=1 to $N_a$
    Step 3.9: j=i+1
Step 3.10: Initialize $PT_{new}$=t, $PT_{old}$=t-1
Step 3.12: Calculation of pheromone trail by the equation
```

$$PTOld\,(i,j) = \frac{[\tau_{i,j}]^{\propto}.[\eta_{i,j}]^{\beta}}{\sum_{v \in S}[\tau_{i,v}]^{\propto}.[\eta_{i,v}]^{\beta}}$$

Step 3.13: Update the pheromone trail by the following equation

$$PT_{new}(i,j) = PTold(i,j) + FF \times PT(Pb_i, j) - FF \times PT(Gb_i, j)$$

Step 3.14: Calculate the De-Jong Function for solving ACO problems by

$$DPTnew(i,j) = \frac{PTnew(i,j)}{\sqrt{\sum_{k=1}^{n}((PTnew(i,j))k)^2}}$$

Step 3.15: Finding the new position by the following equation

$$Pnew = Pold + DPTnew(i,j) + FF$$

Step 3.16: Updating the Local best and Global best position

If ($Pnew > Pb_i$ )

Step 3.17: $Pb_i = Pnew$

Step 3.18: Update the Global best position by the equation

$$Gb_i = \min(Pb_i, Gb_i)$$

Step 3.19: Next i

Step 3.20: Next t

## IV. RESULT AND DISCUSSION

A huge amount of intrusion-detection audit data was found in the US government agency DARPA. This data set is taken for the complete process of ORG-IDS.

After classification of HPFSM processed features into Abnormal, Normal and Unknown, the features of Abnormal and Unknown are considered for generating the attack rules. In the Rule generation phase, the proposed ORGA algorithm plays an important role in generation of less number of rules, minimum running time and reduction in memory utilisation. These parameters are measured in proposed ORGA and it shows that this proposed ORGA outperforms the other existing algorithms.

### A. Effect of ORGA in Rule Generation of ORG-IDS:

- **Number of Rules generated:**

The Table 1 depicts the number of rules generated by ARM, ACO and ORGA for given reduced dataset obtained from proposed HPFSM with various Support and Confidence values. The same is graphically represented in Figure 1. It shows that the number of rules are considerably reduced when ORGA algorithm is used.

**Table 1: Number of Rules obtained by ARM, ACO and ORGA**

| Support and Confidence Value | Number of Rules obtained | | |
|---|---|---|---|
| | ARM | ACO | ORGA |
| 1.0 | 308 | 318 | 217 |
| 0.9 | 527 | 421 | 256 |
| 0.8 | 568 | 432 | 289 |
| 0.7 | 632 | 478 | 312 |
| 0.6 | 717 | 496 | 378 |
| 0.5 | 798 | 498 | 419 |
| 0.4 | 841 | 504 | 458 |
| 0.3 | 892 | 542 | 502 |
| 0.2 | 936 | 612 | 565 |
| 0.1 | 978 | 647 | 614 |

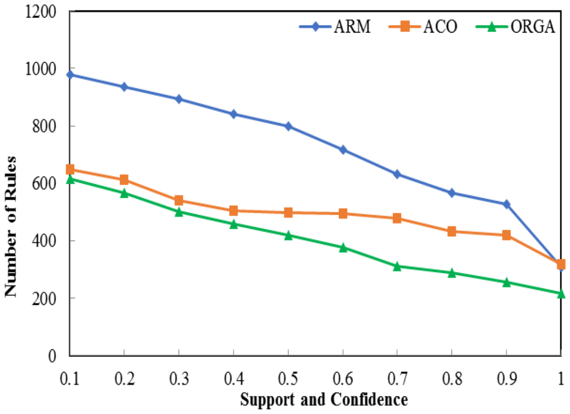

**Fig 1: Number of Rules by ARM, ACO and ORGA**

- **Running time:**

he Running time is the time taken for frequent item set generation with respect to different Support and Confidence values. It is measured in terms of milliseconds (ms) using the formula as follows ,

$$RT = n \times T(n)$$

where *RT* is the Running time, *n* represents the number of frequent item sets generated, and *T(n)* represented time taken for frequent item set generations. The Running time for frequent item set generation must be low, for any efficient method..

Table 2 and Figure 2 depict the total Running time with ARM, ACO and ORGA. It is clear that the proposed ORGA requires less running time than other two existing algorithms.

Table 2: Running Time (ms) of ARM, ACO and ORGA

| Support and Confidence Value | Total Running Time (ms) | | |
|---|---|---|---|
| | ARM | ACO | ORGA |
| 1.0 | 78 | 98 | 39 |
| 0.9 | 151 | 187 | 126 |
| 0.8 | 185 | 185 | 174 |
| 0.7 | 174 | 214 | 167 |
| 0.6 | 216 | 231 | 187 |
| 0.5 | 232 | 267 | 201 |
| 0.4 | 269 | 276 | 231 |
| 0.3 | 312 | 321 | 213 |
| 0.2 | 323 | 343 | 275 |
| 0.1 | 341 | 341 | 301 |

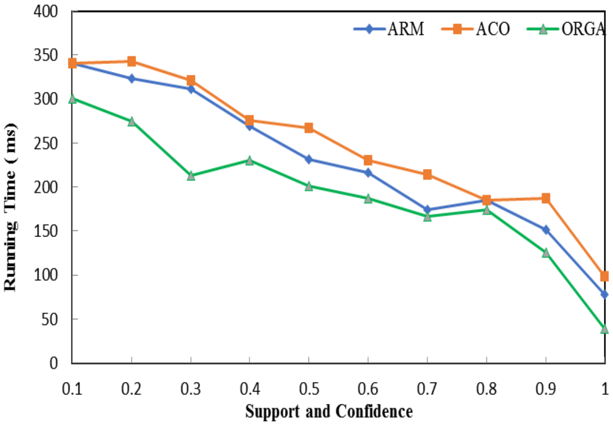

**Figure 2: : Running Time (ms) of ARM, ACO and ORGA**

- **Memory Consumption:**

Table 3 depicts the total memory consumption in Mega Bytes for ARM, ACO and proposed ORGA and the same is graphically represented in Figure 3. From this it is understood the proposed ORGA required less total memory consumption than ARM and ACO.

**Table 3: Memory Consumption(MB) of ARM, ACO and ORGA**

| Support and Confidence Value | Memory Consumption(MB) | | |
|---|---|---|---|
| | ARM | ACO | ORGA |
| 1.0 | 9.5 | 7.8 | 3.1 |
| 0.9 | 17.9 | 14.2 | 11.2 |
| 0.8 | 19.4 | 18.5 | 13.9 |
| 0.7 | 21.1 | 18.5 | 16.5 |
| 0.6 | 29.5 | 24.2 | 19.5 |
| 0.5 | 27.8 | 24.1 | 21.2 |
| 0.4 | 33.1 | 30.1 | 22.0 |
| 0.3 | 31.2 | 33.2 | 22.4 |
| 0.2 | 34.1 | 33.4 | 30.3 |
| 0.1 | 45.0 | 45.2 | 41.2 |

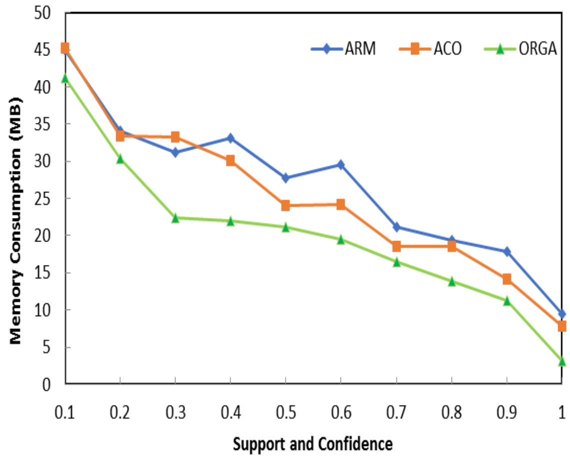

**Figure 3: Memory Consumption by ARM, ACO and proposed ORGA**

**B. Effect of ORGA in Pattern Matching phase of ORG-IDS:**

- **4.2.1 Running time(sec) of Pattern Matching Algorithm for HPFSM processed dataset**

Table 4 depicts the running time taken by Aho-Corasick pattern matching algorithm for checking the incoming packet's inputs(Bytes) with ARM, ACO and proposed ORGA. The same is graphically represented in Figure 4. From this it is understood the proposed ORGA required less running time than ARM and ACO.

**Table 4: Running Time(sec) of Pattern matching Algorithm with ARM, ACO and ORGA**

| Input Size (Bytes) | Running time in seconds | | |
|---|---|---|---|
| | ARM | ACO | ORGA |
| 1000 | 14 | 15 | 12 |
| 2000 | 44 | 51 | 40 |
| 3000 | 67 | 60 | 55 |
| 4000 | 90 | 88 | 79 |
| 5000 | 109 | 99 | 85 |
| 6000 | 119 | 108 | 87 |
| 7000 | 125 | 119 | 98 |
| 8000 | 140 | 123 | 101 |
| 9000 | 156 | 129 | 107 |

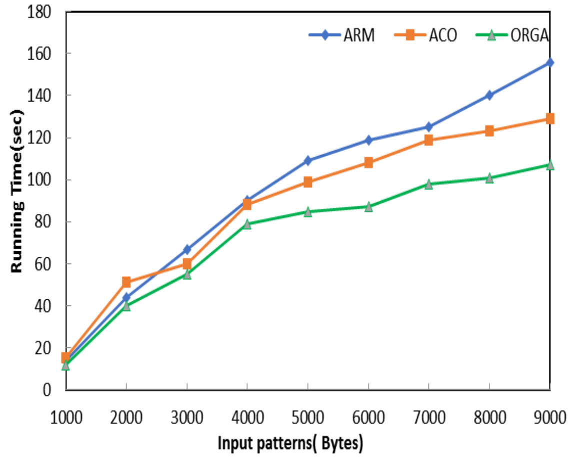

**Figure 4: Running Time of the Pattern matching with ARM, ACO and ORGA**

## V. CONCLUSION

The proposed work ORG-IDS with Feature Selection, Classification, Optimized Rule Generation and Pattern matching phases performs well in producing the important metrics. As ACO based ARM is used for generating optimized rules, the proposed IDS reduces the number of rules , works in less Running time and consumes minimum memory. It is also good in checking signatures in pattern matching algorithm by consuming less running time. So it is clear that the proposed algorithm produces better result in all metrics than existing algorithms.

## REFERENCES

1. Latha, S., and Sinthu Janita Prakash, "A survey on network attacks and Intrusion detection systems.", 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS). IEEE, Aug 2017
2. Jianglong Song, Wentao Zhao, Qiang Liu and Xin Wang, "Hybrid Feature Selection for Supporting Light Weight Intrusion Detection Systems," IOP Conference Series, Journal of Physics, Conference Series: 887, pp. 1-7, Aug 2017
3. Raman, M.R. Gauthama, et. al., "A hyper graph and Arithmetic Residual based Probabilistic Neural Network for classification in intrusion detection systems. "Neural Networks, Vol.92, Pp. 89-97, Aug 2017
4. Li, Longjie, et al., "Towards Effective Network Intrusion Detection: A Hybrid Model Integrating Gini Index and GBDT with PSO," Journal of Sensors, 2018.
5. Zbeel BM, " Using Genetic Algorithm for Network Intrusion Detection", kufa studies center journal, Vol.1, Iss.29, pp. 209-224, 2013
6. Li W, " Using genetic algorithm for network intrusion detection", Proceedings of the United States Department of Energy Cyber Security Group, pp. 1-8, May 2004
7. Sadiq AS, Alkazemi B, Mirjalili S, Ahmed N, Khan S, Ali I, Pathan AS and Ghafoor KZ, "An efficient ids using hybrid magnetic swarm optimization in wanets" , IEEE Access, Vol. 6, pp.29041-53, 2018
8. Hajimirzaei, Bahram, and Nima Jafari Navimipour, " Intrusion detection for cloud computing using neural networks and artificial bee colony optimization algorithm", ICT Express, Vol. 5, Iss. 1, pp. 56-59, Mar. 2019
9. Latha, S & Prakash, S.J., "HPFSM-A high pertinent feature selection mechanism for intrusion detection system",International Journal of Pure and Applied Mathematics, 118. 77-83, 2018.

10. S. Latha and Sinthu Janita Prakash, " IDSFS: A Signature Based Intrusion Detection System with High Pertinent Feature Selection Method", Asian Journal of Computer Science and Technology(AJCST), Vol. 8, Iss. 2, April-June2019, pp. 25-31
11. S. Latha and Sinthu Janita Prakash, "A Signature Based Intrusion Detection System with HPFSM and Fuzzy Based Classification Method (IDSFSC)", Asian Journal of Engineering and Applied Technology(AJEAT), Vol. 8, Iss. 2, April-June2019, pp. 23-29
12. Sathya M & Thangadurai K. (2016). Association Rule Generation Using E-ACO Algorithm. International Journal of Control Theory and Applications, 27(9), 513-521.
13. Santosh Kumar Sahu, "A Detail Analysis on Intrusion Detection Datasets," IEEE International Advance Computing Conference (IACC), pp.1348-1353, Feb 2014.

## AUTHORS PROFILE

**S. Latha** qualified with Master of Computer Applications, M.Phil and NET. She is currently working as an Assistant Professor in the Department of Computer Science, Cauvery college for women, Tiruchirapalli. She is a life time member of Indian Society of Systems for Science and Engineering(ISSE). She has published nearly 5 research papers in reputed international journals and conferences including IEEE and Springer. Her interest in research includes the area of Communication Networks, Network Security and Data Mining. She has 13 years of teaching experience and 9 years of Research experience in guiding MPhil students.

**Dr. V. Sinthu Janita** pursued Master of Computer Applications and Ph.D from Bharathidasan University. She is currently working as a Professor and Head in the PG & Research Department of Computer Science, Cauvery college for women, Tiruchirapalli. She is a member of IEEE, Computer Society of India and a life time member of Indian Society of Systems for Science and Engineering(ISSE). She has published more than 35 research papers in reputed international journals including Thomson Reuters (SCI & Web of Science) and conferences including IEEE, Elsevier, ACM, Springer and it's also available online. Her main research work focuses on Communication Networks, Network Security, Big Data Analytics, Data Mining and Mobile communications. She has 23 years of teaching experience and 14 years of Research experience.

*Retrieval Number: B3373078219/19©BEIESP*
*DOI:10.35940/ijrte.B3373.098319*
*Journal Website: www.ijrte.org*

*Published By:*
*Blue Eyes Intelligence Engineering*
*& Sciences Publication*