Hybrid Fuzzy-PSO Based Approach to Detect Intrusion in Networks

Thesis submitted in partial fulfilment of the requirements for the

award of degree of

Master of Technology

In

Computer Applications

Submitted By

Sonam Rani

(601634016)

Under the supervision of:
Dr. Sushma Jain
Assistant Professor

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT

THAPAR INSTITUTE OF ENGINEERING AND TECHNOLOGY
PATIALA – 147004

JUNE 2018

CERTIFICATE

I hereby certify that the work which is being presented in the thesis entitled, **"Hybrid Fuzzy-PSO Based Approach to Detect Intrusion in Networks"**, in partial fulfillment of the requirements for the award of degree of Master of Engineering in *Computer Science and Engineering* submitted in Computer Science and Engineering Department of Thapar Institute of Engineering and Technology, Patiala, is an authentic record of my own work carried out under the supervision of ***Dr. Sushma Jain*** and refers other researcher's work which are duly listed in the reference section.

The matter presented in the thesis has not been submitted for award of any other degree of this or any other University.

Signature:

(Sonam Rani)

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

(Dr. Sushma Jain)
Assistant Professor
Computer Science and Engineering Department

ACKNOWLEDGEMENT

The successful completion of any task would be incomplete without acknowledging the people who made it possible and whose constant guidance and encouragement secured the success.

First of all, I wish to acknowledge the benevolence of omnipotent God who gave me strength and courage to overcome all obstacles and showed me the silver lining in the dark clouds with the profound sense of gratitude and heartiest regard. I express my sincere feelings of indebtedness to my guide **Dr. Sushma Jain** for their positive attitude, excellent guidance, constant encouragement, keen interest, invaluable co-operation, generous attitude and above all their blessings. She has been a source of inspiration for me.

I am grateful to **Dr. Maninder Singh**, Head of Department and **Dr. Sanmeet Bhatia**, P.G. Coordinator, Computer Science and Engineering Department, Thapar Institute of Engineering and Technology for the motivation and inspiration for the completion of this thesis. I will be failing in my duty if I don't express my gratitude to **Dr. S.S. Bhatia**, Senior Professor and Dean of Academics Affairs in the institute for making provisions of infrastructure such as Library facilities, Computer Lab equipped with internet facility immensely useful for the learners to equip themselves with latest in the field.

Last but not the least I would like to express my heartfelt thanks to my parents and my friends who with their thought provoking views, veracity and whole hearted cooperation helped in doing this thesis.

Sonam Rani

Sonam Rani

(601634016)

ABSTRACT

In Internet-based communication, various types of attacks have been evolved. Hence, attacker easily breaches the securities. Traditional techniques of intrusion detection have failed to observe these attacks and thus hefty systems are required to remove these attacks before uncovering of the entire network by attacks. The power of artificial intelligence systems is used, to make high computational speed, boost fault tolerance and error resistance against turbulent data. Therefore, fuzzy-PSO (particle swarm optimization) based hybrid approach is designed. The fuzzy logic based on the degree of truth whereas PSO algorithm depends on stochastic technique that helps in learning from the scenario, thus their combination increases the toughness of intrusion detection system. The proposed network intrusion detection system will be able to classify normal as well as anomalism behavior in the network. DARPA-KDD99 dataset examined on this system to address the behavior of each connection on the network and compared with an existing system. This approach improves the result on the aspects of precision, recall, and F1-score.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1 Introduction

1.1 Intrusion

In modern days, use of the Internet has expanded greatly. Its adoption has escalation to the core of most of the trades. The connectedness it contributes, grants enterprises to broaden their action and boost productiveness. After all, the internet advanced by swift degree, its scope expanded enormously. Nowadays, access to the internet is very easy, and considerably at low prices. It describes that anyone can get access of it and approach anybody over the network virtually. This luxury of openness created an unusual type of misbehaviour: virtual-misbehave [1]. This kind of misbehave matured aggressively from past twenty years, primarily by reason of the democratize of the internet [2]. Intrusion is an action attempting to break into or misuse one's system in violation of an established policy like remote root compromise, distribution of pirated software etc. There are two threats to the security of any system first one is malware and second is the intruder. Intruders, who are spread across the Internet, have become a major threat in the world. It is important to resolve this issue as soon as possible. Hence for its solution, a count of techniques such as firewall, encryption to prevent such penetration and to protect the infrastructure of computers are implemented, but with this, the intruders managed to penetrate the computers. In this chapter, we discuss the intrusion detection system (IDS). This is used to protect or prevent various penetration and inner structure of computers. This system consists of many hardware and software which are used to determine unexpected events which are going to give an indication like the attack is going to happen, attack is happening in the system or attack is happened in the system. These are such indication given by the IDS. It can be classified as its working type of the system like it warns before the attack or it can warn while the attack is in the process or it warns after the attack. There are three component of intrusion detection system.

- **Sensor: -** it is used to generate events and to sense the traffic network or activity of the system.
- **Console: -** It is used to control sensor, events, and alerts.

- **Detection Engine: -** It issued for generation of alerts after receiving variation from security events and also used to maintain the data of sensors' events in any database.

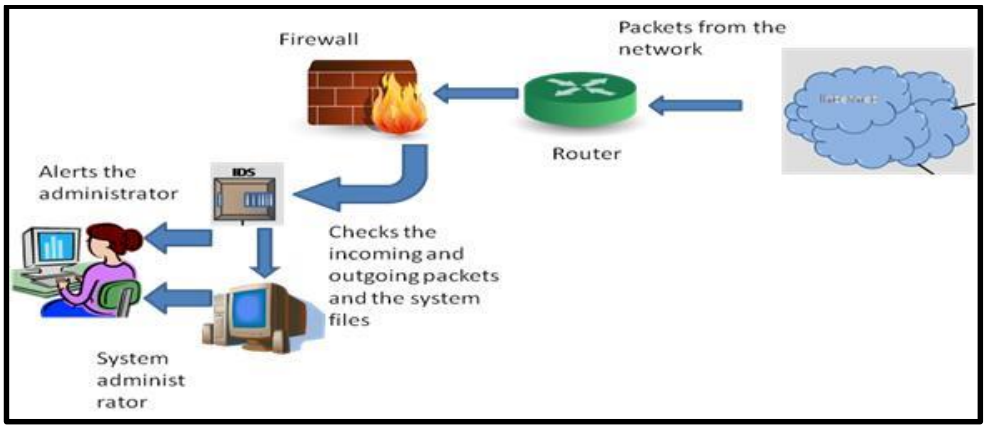

Figure 1.1 Intrusion Detection System

Andersons studied on intrusion detection system and concluded that the task of detecting a misfeasor was more difficult, in that the uniqueness between abnormal and normal behavior might be small. It also gave a differentiation between masquerade and the legitimate users were detected by observing past history.

- **Types of Intrusion Detection System**

There are three types of intrusion detection system:

Network based Intrusion Detection System: - Snort is the best example of network intrusion detection system. These systems used to observe network traffic and to control multiple hosts. There is a development in the execution of this system by using a technique where traffic networks are connected to the hub, network tap.

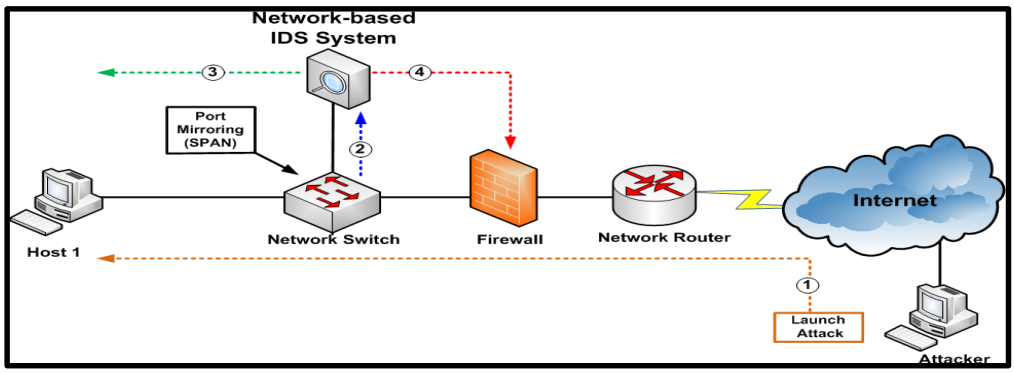

Figure 1.2 Network-based IDS System

Host based Intrusion Detection System: **-** This system detects the intrusion with the help of analyses of system calls, application logs, modification in file-system (binaries, a file having passwords etc.). All these detections are possible due to an agent in a host which is the key component of the host-based intrusion detection system. As the name indicates, this system is really based on the host for the purpose of detection.

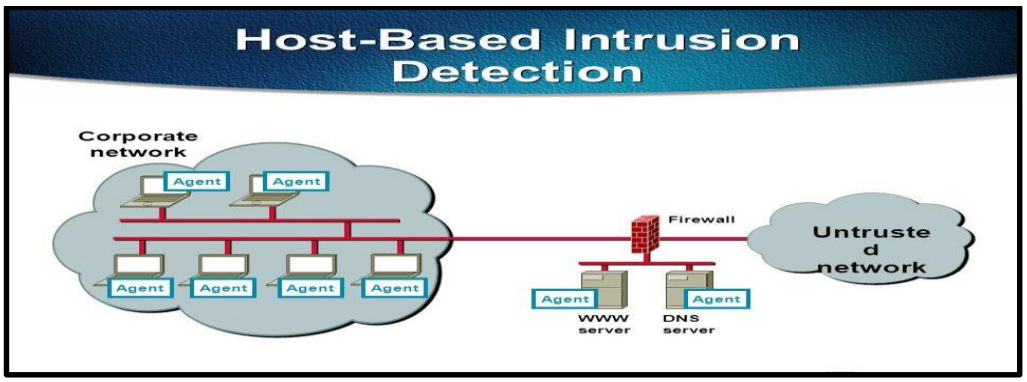

Figure 1.3 Host -based IDS System

Hybrid Intrusion Detection System: **-** It is a combination of two or more systems. Here it is a combination of above two systems. To frame a comprehensive view of the network, there is a combination of agent attached to the host with network information. The common example of hybrid IDS is prelude.

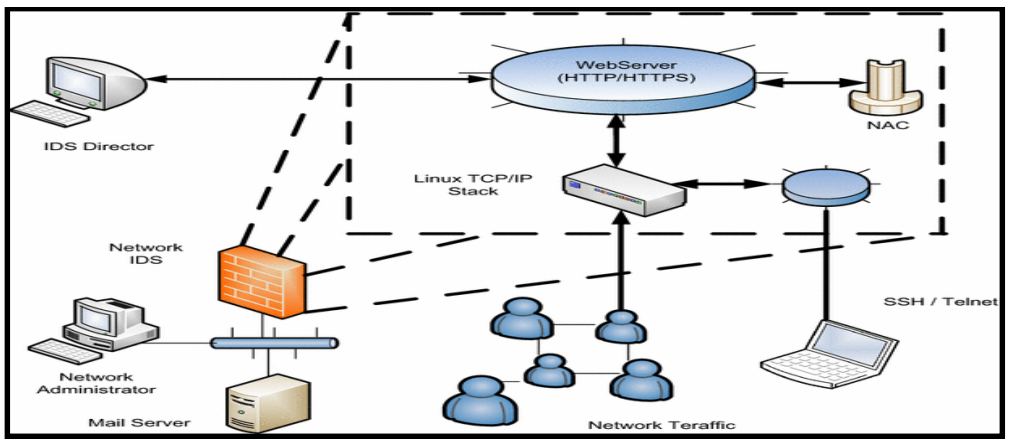

Figure 1.4 Hybrid IDS System

1.2 Network Attack

Network attack is usually defined as an intrusion on the network infrastructure that will first analyze the traffic on network and collect information in order to exploit the existing open ports or vulnerabilities as well as do unauthorized access to particular resources of the tenants. In such cases where the purpose of the attack is only to learn and get some information from particular system but the system resources are not altered or disabled in any way, known as passive attack. The active attack occurs where the perpetrator accesses and either alter, disables or destroys the resources or data. An attack can be performed either from outside of the organization by unauthorized entity (outside attack) or from within the company by an insider that already has certain access to the network (inside attack).

- **Types of attack**

A useful meaning of classifying security attack is in terms of active attack and passive attack. A passive attack attempts to monitor the information from the system but does not affect system resources. Active attack attempts to harm system resources and their operations. Classes of attack might include passive monitoring of communications, active network attacks, close-in attacks, exploitation by insiders, and attacks through the service provider. Information systems and networks offer attractive targets for exploitation.

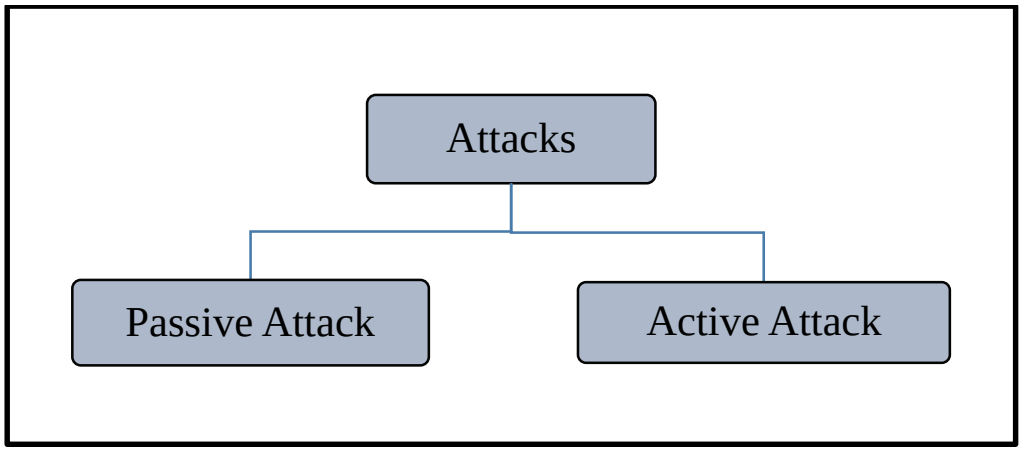

Figure 1.5 Types of Attacks

Passive attacks: **-** Passive attacks include traffic analysis, eavesdropping or monitoring of unprotected communications, decrypting weakly encrypted traffic, and capturing authentication information such as passwords. Passive interception of network operations enables adversaries to see upcoming

actions. It results in the disclosure of information or data files to an attacker without the consent or knowledge of the user.

Active attack: **-** It involves some modification of the data Stream or creation of the false stream. The attacker tries to bypass or break into secured systems. This can be done through stealth, viruses, worms, or Trojan horses. Active attacks include attempts to circumvent or break protection features, to introduce malicious code, to steal or modify information. These attacks are mounted against a network backbone, exploit information in transit, electronically penetrate an enclave, or attack an authorized remote user during an attempt to connect to an enclave. Active attacks subdivided into four categories; masquerade, replay, modification of message, and denial of service.

Distributed attack: **-** It requires the adversary introduce code, such as a Trojan horse or back-door program to a trusted component or software that will later be distributed to many other companies and users. Distribution attacks focus on the malicious modification of hardware or software at the factory or during distribution. These attacks introduce malicious code such as a backdoor to a product to gain unauthorized access to information or to a system function at a later date.

Insider attacks*:* **-** These are the most difficult to detect and prevent. It involves someone from the inside, such as a disloyal employee, attacking the network. Insider attacks can be malicious or no malicious. Malicious insiders intentionally eavesdrop, steal, or damage information; use information in a fraudulent manner; or deny access to other authorized users. No malicious attacks typically result from carelessness, lack of knowledge, or intentional circumvention of security for such reasons as performing a task.

Hijack attack: **-** In a hijack attack, a hacker takes over a session between tenants and disconnects the one tenant from other and hacker participate in the communication. Individual believe that communication is with the original party and may send private information to the hacker by accident.

Spoof attack: **-** In a spoof attack, the hacker tries to access the network's IP address. After gaining access to the network with a valid IP address, the attacker can modify, reroute, or delete the data.

Buffer overflow: **-** A buffer overflow attack happens when the attacker sends more data to system than expected. A buffer overflow attack usually results in the attacker gaining administrative access to the system through a command prompt or shell.

Exploit attack: **-** In exploit attack, the attacker knows about the security problem within an operating system or a piece of software and leverages that knowledge by exploiting the vulnerability.

Password attack: **-** An attacker tries to crack the passwords stored in a network account database or a password-protected file. There are three major types of password attacks: a dictionary attack, a brute-force attack, and a hybrid attack.

- **Security Goals**

In order to maintain a secure networking environment, below goals of security should be there.

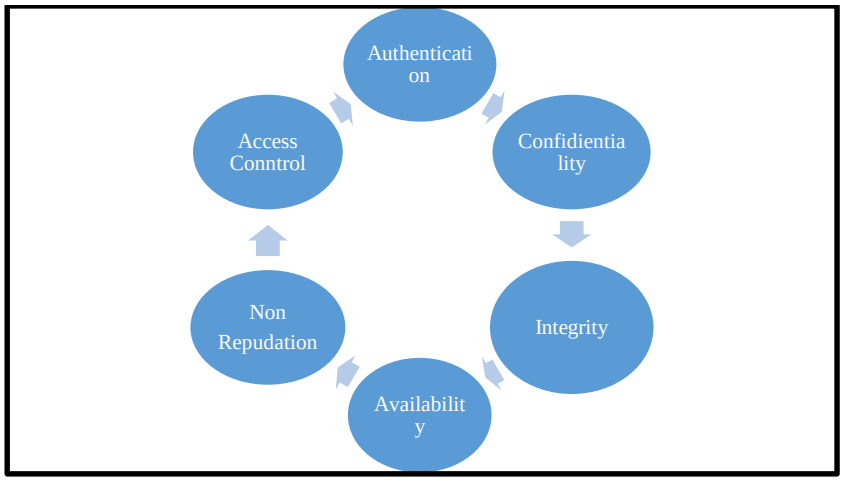

Figure 1.6 Security Goals

Authentication: **-** In order to have a secure system, authentication is must. The prime focus of this service is to ensure the sender and the receiver that the corresponding user is authentic and not the fake user. Authentication can be done by giving a user name, or a password to individual users or by using cryptographic techniques.

Confidentially: **-** In it, the data is send in an unreadable form for unauthorized users. By doing this if some unauthorized user wants to read the data then user must know the key. So data should be converted in non-readable form by

using cryptographic techniques. Confidentiality makes the backbone of security function.

Integrity: - The goal of integrity is to make sure that the data is not altered during transmission. For this, service should be provided using cryptographic techniques. Data should be encrypted, in non-readable form while transferring to the other user. This survive forms a very important part of network security.

Availability: - The goal of it is to make sure that the next hop should always be available to the source user when required. In case, when the desired hop or destination is not available then the information will be lost and hence the security will be compromised.

Non-repudiation: - The main purpose of it is to make sure that a particular message is sent by a particular system and an individual should not refuse that the message is not send by them. These kind of problems can be remove with the help of digital signatures or with the help of private encryption key.

Access Control: - This security goal depends on authentication. The whole point of access control is that some unwanted user or unauthorized user cannot access to the system and cannot use the services provided by the network. This can be done by assigning user name and password to individual authorized or authentic users.

1.3 Fuzzy Logic

Fuzzy logic approach has been proposed in 1965 by Zadeh by using the fuzzy rule concept. It has been used in various fields for classification and control applications. Following processes and concept are occurred in the fuzzy system: -

Crisp Set: - These set represents the binary case i.e. either 0 (False) or 1 (True) implication, that the element is either belongs to a crisp set, or not to crisp set. This value is used for variable as input for fuzzification process.

Fuzzy Sets: - It is multivalent logic that permit to hold values in range of 0 to 1 which shows the individual's reasoning. After the fuzzification process, crisp sets are converted to fuzzy sets.

Membership Function: - It gives the measure of degree of sameness of an elements in the universe's disclosure to a fuzzy set. It isn't same as the

probability however shows the participation in an enigmatically defined set. It is showed by $\mu_A(x)$

For sets, the membership function scope comes in range of 0-1.

$$\mu_A(x) \in [0, 1]$$

For crisp sets,

$$\mu_A(x) = \begin{Bmatrix} 0 & if\ x \in X \\ 1, & otherwise \end{Bmatrix}$$

Fuzzification: - The procedure of changing over crisp set into fuzzy set by utilizing participation work is called fuzzification which is lying in the vicinity of 0 and 1.

De-fuzzification: - The way toward changing over the fuzzy sets aback to crisp qualities or set is called defuzzification.

Fuzzy rule based system: - It is a gathering of propositions containing linguistics variable and the rules are described as X1 is A1 then Y is B1.

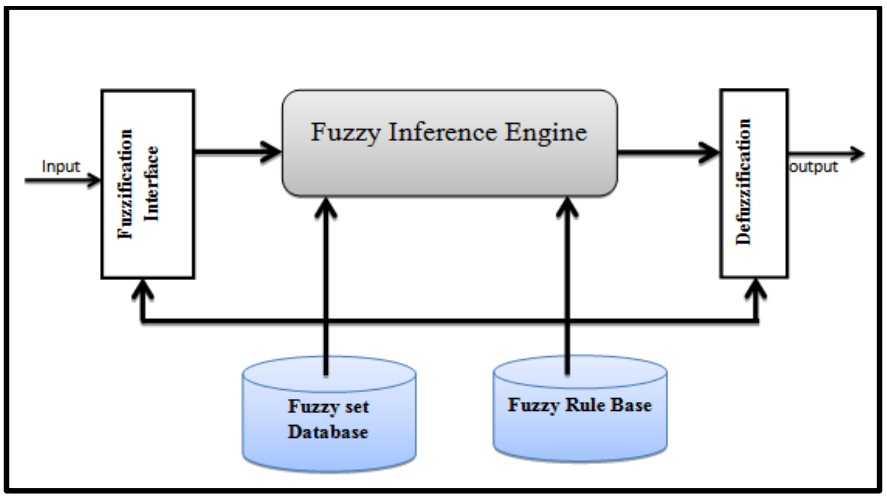

Figure 1.7 Fuzzy System

1.4 Fuzzy System

The rule based on fuzzy framework or systems are comprises of fuzzy if-then rules and have been utilized as a part of different sample classification problems [3]. It comprises of following parts.

- **Learning Base:** - The Learning Base is additionally separated into two sorts.

Rule Base: -It comprise of set of rules or standards. The principles can be described in various ways. Here the rule is of the accompanying form-

R^k- If X_1 is A_1^k and ..so on X_n is A_n^k then class has C^k with CD^k.

where each input variable X_i has an incentive from set of linguistic labels $A_i^k \left[L_i^1 \text{ or } L_i^2 \ldots\ldots \text{or } L_i^i\right]$, C^k represents the ensuing class and CD^k is the confidence degree or certainty ($CD^k \in 0,1$) . Certainty, or sureness degree define how adequately a fuzzy rule identifier allocates a pattern of an input to a class's label [8].

$$CD_r = \frac{\beta_{class\ h_r}(R_j) - \overline{\beta}}{\sum_{h=1}^{c} \beta_{class\ h_r}(R_j)} \quad (1)$$

Where $\beta_{class\ h_r}(R_j)$ shows the membership function's aggregation for training pattern in class h and $\overline{\beta}$ shows the aggregate of membership function of training pattern that don't have a place with class h.

Database: -The fuzzy set identified with linguistic terms which are utilized as a part of Rule Base is characterized in Database (DB). Hence the quantity of linguistic marks (l_i) for every factor X_i and the fuzzy set deals with linguistic terms with membership function are indicated. Here every factor has been referring five linguistic marks (i.e. very low, low, medium, high, exceptionally high) and every factor definition interval has been consistently divided under triangular fuzzy sets as appeared in below figure.

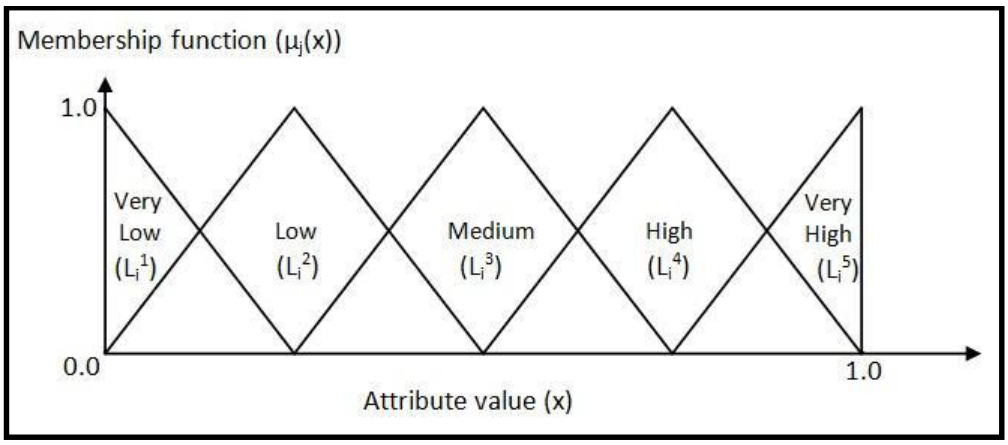

Figure 1.8 Triangular Membership Functions of Fuzzy Sets

- **Fuzzy Inference Engine*:*** - It is an analysis method that determines analysis or inference from fuzzy if-then rules. Here the input pattern is coordinated with the predecessors of rules and the pattern is classified by the rule that show subsequent by using the maximum matching property.

  So to classify an unlabeled pattern, $X_p = \{X_{p1}, X_{p2}, \ldots . X_{pn}\}$ from a given set of rule, the pattern of rule with the rule set is coordinated and the accompanying methodology is used.

  The class of the undefined particles is controlled by the accompanying conditions: -

$$C = \max_{h=1,2\ldots c} (\tau_h) \tag{2}$$

  where

$$\tau_h = \max_{\substack{r=1..N \\ C_r \epsilon h}} \left\{ \mu_{A_r(x_p).} CD_r \right\} \tag{3}$$

  In this manner, the effective rule is the one which has maximum τ_h but having more than one class has maximum τ_h or if $\tau_h = 0$ then that rule is rejected.

Thus, a rule set is created and the unrecognized particles are categorized into their particular classes according to their sureness or certainty degree.

Henceforth, fuzzy rule based system give a best platform to manage boisterous or loose data and accomplish two objectives: precision and interpretability.

1.5 Particle Swarm Optimization

Numerous optimization procedures exist. There are deterministic procedures for optimization, for example, interim optimization [4], branch-and-bound [5], and algebraic systems [6]. Further, there are stochastic improvement methods, for example, simulated annealing [7], Monte Carlo sampling of dataset [8], stochastic-tunneling [9], and parallel-tempering [10]. Moreover, there are techniques for heuristic and metaheuristic improvement, for example, genetic optimization [11], developmental techniques [12], transformative programming [13], PSO [14], colony optimization [15], memetic calculations [16] and cuckoo seek [17]

PSO was presented by Kennedy and Eberhart [14]. The conduct of PSO can be imagined by comparing it with bird swarm search, looking for ideal sustenance

sources, where the course in which a bird moves is impacted by its present movement, and the best sustenance source any bird in the swarm at any point experienced. It also can be described as that birds are driven by their inertia, their personal experience, and the learning of the swarm. As far as PSO, the particle's movement is impacted by their inertia, its personal best position, and the global best position.

PSO has various particles, and each particle comprises of its present target value, its position, its velocity, its personal best value, that is the best target value the particle at any experienced, also, its personal best position, with the help of this position, an individual's best value has been construct. Along with this, PSO keeps up the value for global best, that is the best target value has experienced by any particle, and the global best position, with the help of this position, value for global best has been construct. PSO [28] utilizes the accompanying emphasis to move the particles:

$$x^{(i)}(n\ +1)\ =\ x^{(i)}(n) + v^{(i)}(n+1), n = 0,1,2\ldots., N-1, \qquad (4)$$

where $x^{(i)}$ represents the position of particle i, n represents the iteration, $n = 0$ alludes to the initialization, N is the total count of iterations, and $v^{(i)}$ is the particle's velocity i. In established PSO, the velocity of the particle is resolved utilizing the accompanying iterations:

$$\mathrm{v^{(i)}(n+1)} = \omega \mathrm{v^{(i)}(n)} + \mathrm{c_1 r_1^{(i)}(n)}\left[\mathrm{x_p^{(i)}(n) - x^{(i)}(n)}\right] + \mathrm{c_2 r_2^{(i)}(n)}\left[\mathrm{x_g(n) - x^{(i)}(n)}\right],$$

$$n = 0,1,2,3\ldots, N-1, \qquad (5)$$

The inertia weight ω are the velocity weights, the individual best weight c1, and the global best weight c2.

1.6 Hybrid Fuzzy PSO System

The framework of PSO with fuzzy logic is a fuzzy system expanded with stochastic technique to expand the learning limit of the framework or system, hence producing powerful search abilities in a complex environment. The principle work is to learn from scenario which can consequently produce /plan learning base and this way help in search and improvement issues. The following segment plainly clarifies how the PSO based on learning scenario helps in learning process.

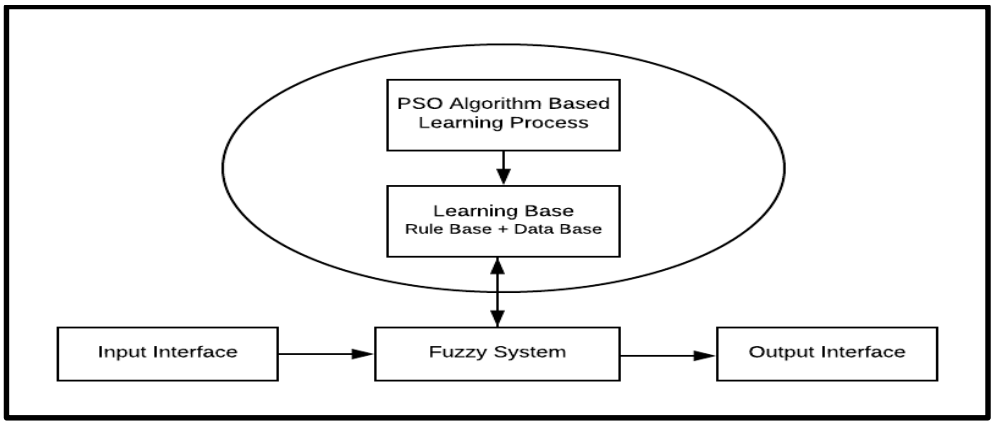

Figure 1.9 Hybrid Fuzzy-PSO System

Chapter 2 State of Art

2.1 Origin of Intrusion Detection System

An intrusion is characterized as an infringement to a human's security or organization's imperative data. Previously different regular strategies, for example, firewall, encryption strategies have been utilized to keep the PC frameworks from unapproved use. In any case, these systems were not sufficiently enough to prevent the interruption or intrusion as attacker and hacker grew effectively and could discover vulnerabilities in the system, subsequently breaching PC security policies. Thus, an additional tool i.e. intrusion detection system was set up and it turned into a most important component in the field of security infrastructure. The basic intrusion detection model had given by Denning in 1987 [18]. From that point, numerous intrusion detection models have been built to decide the conduct of the system precisely and in an effective way.

Porras *et al.* [19] proposed three measures to assess the working of intrusion detection frameworks. These are: -

- **Efficiency:** - The interruptions must be appropriately identified with no false alerts i.e. they must not be misclassified or mistreated.
- **Integrity:** - It shows the capacity to recognize all the interruptions. Hence, the intrusions detection system should continue to refreshing itself and should have detailed information about the interruptions.
- **Performance:** - It shows the rate of handling of audit information. For real-time recognition, performance of the IDS should be high.

The IDS has been categorized into different classes by Debar *et.al.* [20, 21] as described in figure 2.1.

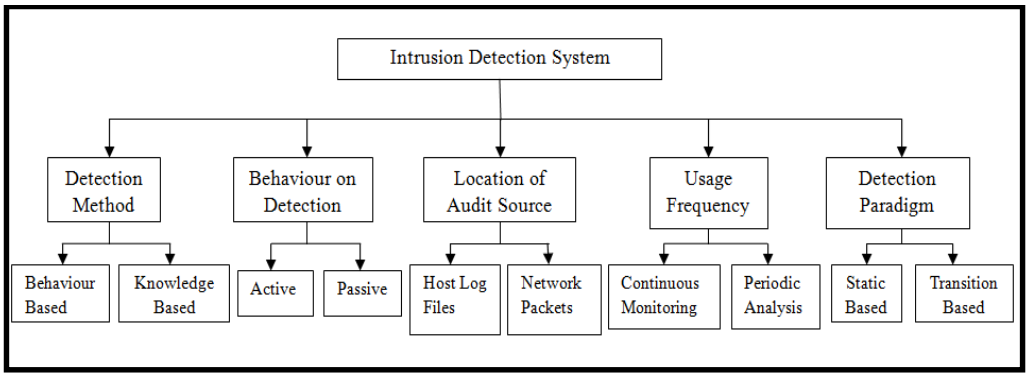

Figure 2.1 Classification of IDS

Following is the concise detail of each class of IDS-

Detection Method: -The attributes or the properties of the identifier or the detector is shown in this strategy. It comprises of two strategies: -

a) which fetch the data about the intrusion from the database and generate an alert when vulnerability is discovered such as misuse-detection.
b) when the control of normal behavior has kept and any anomaly from the control is identified as a malicious movement such as anomaly detection.

Behavior Detection: - It describes the conduct or reaction of the intrusion detection framework or system. When alerts have activated and corresponding to it no action has made, then the IDS considered to be passive but when a few countermeasures to destroy or control these interruption has taken, known as active.

Area of Audit Source: - The input data is analyzed and on the premise of which the interruption detection system is categorized or classified. The data can be in the form of system packet, host log information, or application log records.

Usage of Frequency: - It infers the analysis of the time of the interruption detection system in a specific situation. Continuous observing performs a continuous time analysis to get information about the ongoing activities while the periodic examination or analysis alludes to examining the exercises on a periodic basis.

Detection Paradigm: - It explains the technique used to recognize the interruptions in IDS. It can be either state base or changes with respect to the basis of transition from one state to another.

Afterwards, different artificial intelligent, machine learning and mathematical procedures were practiced on different interruption detection models to get exact outcomes while encountering different issues, for example, loose data, traffic on network, continuous adaption to evolving environment.

2.2 Dataset for Intrusion Detection System

For assessing the performance of interruption detection models, datasets have been needed which can definitely indicate whether the Intrusion Detection System can agree with the rules or not. The information can be gathered from different sources, for example information packets, log records, sequences of commands and so forth. To the great extent publically accessible and most utilized, two datasets are: - DARPA 98 Lincoln and KDD 99 dataset. KDD99 dataset was inherited from network-traffic dataset i.e. DARPA98 by ACM SIG-KDD in 1999. It contains TCP oriented links or connections as well as has datasets of nine weeks for training and two weeks' datasets for testing to detect intrusion and every single connection contains 41 attributes.

2.3 Analysis of Fuzzy Rule-based System

Fuzzy system has the logical capability to produce better accuracy and precision to build the real world systems. They are in view of fuzzy rule and predicates which incorporates the knowledge of human specialists furthermore, strengthen symbolic and numerical representation of information.

The framework which was introduced by madami known as mamdani fuzzy rule-based framework, comprises of two sections i.e fuzzy information base and fuzzy inference framework. Fuzzy rule-based framework in which a fuzzy rule structure is coordinated against the patterns and corresponding to it a class and a distinct value is given as a consequent proposed by Ishibuchi. The fuzzy rule-based categorized framework can be differentiated based on certainty factor or the confidence association to the class in the consequent in the accompanying ways: -

- Based on class name.
- Based on class name with the confidence factor.
- Based on only confidence factor.

The major disadvantage of Mamdani Fuzzy rule-based framework is absence of precision in case of high dimensional, complex dataset because of lack of adaptability in linguistic variables. Consequently, different augmentations were given with a specific goal to increase the accuracy of Mamdani fuzzy rule-based framework. Some of them incorporate with scatter fuzzy partitions, disjunctive typical frame, weighted rules or standards, and so forth.

A completely educated PSO had presented in [22]. The paper recommended to group particles into k bunches based on their position after initialization. The neighbors' size was expanded as the calculation proceeds, trying to accomplish a completely connected swarm, i.e., one bunch or cluster, after 80% of the total count of iterations N. The local best PSO studied comparisons with the weight factor free standard PSO and completely educated PSO for various topologies; i.e., neighborhood sizes, and neighborhood structures, in [23]. Weight factor free standard PSO was not utilized any weights; i.e., no inertia weight, no individual best weight, and no global best weight. The paper demonstrated that extensive neighborhoods involved in convergence, while little neighborhoods involved in diversity [24].

The co-operative combinatorial PSO [22] produced solutions for mix-integer and combinatorial optimization issues. Co-operative combinatorial PSO utilized different swarms to enhanced the diverse parts of the problem by moving particles towards the global best position of their swarm, and the global best position of neighboring swarms [25].

Hybridization of PSO strategy described combination of PSO with at least one optimization technique. The mimetic PSO consolidated with linear search procedures with PSO [26]. Essentially, fuzzy adaptive Neldermead PSO utilizes fuzzy adaptive PSO as a global search method and applies a Neldermead simplex search to the global best position after each iteration [22].

With the collaboration of fuzzy logic and genetic algorithm based system, an effective approach created to observe R2L and U2R intrusions with immense recall value while existing approaches flopped to do because there was less number of connections in training dataset proposed by Chadha [27].

2.4 Computational-intelligence for Intrusion Detection

Bezdeck [28] described computational intelligence and he laid out it as: A computationally intelligent framework is the one which is deal with low level information, consolidate constituents of reorganization of patterns, and does not use a knowledge in the artificial intelligence sense; and shows main characteristics' which are given below: -

- computational-adaptability
- rate of errors that is practically close to human performance
- fault tolerance of computations
- speed similarity turnaround like humans

Computational Intelligence comprises of four fundamental ideal models as shown in figure 2.2: -

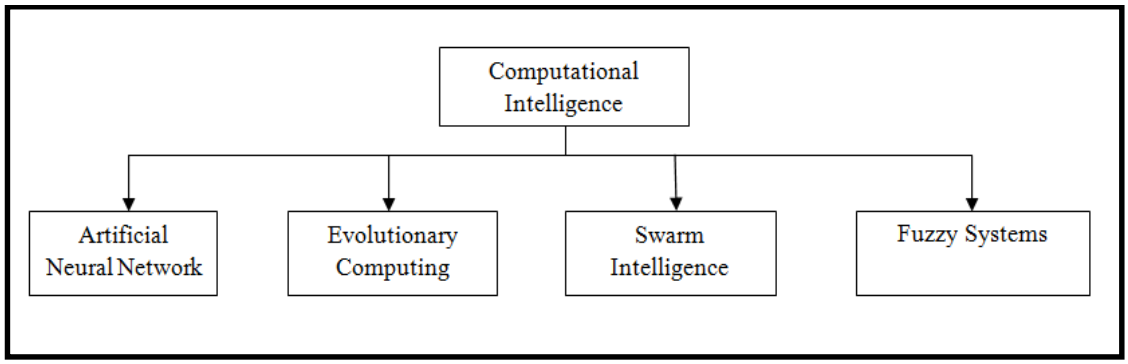

Figure 2.2 Computational Intelligence Models

Artificial Neural Network: - It is a data processing model encouraged from the biological neurons. It is an interconnected gathering of artificial neurons working as one to take care of complex issues. It is considered as the weighted coordinated diagram in which the hubs portray the neurons while the edges shows the interconnections between the neurons. Counterfeit neural systems are largely utilized because of their capacity to adjust, display adaptation to non-critical failure, real time operation, and self-organization.

Evolutionary Computing: - The evolutionary calculation depends on the components of genetic qualities and natural-selection is utilized as a part of examine and improvement issues. It utilizes an iterative approach where a populace is chosen from arbitrary space and diverse activities are connected to get a desired outcome. It comprises of genetic programming given by Koza,

development methodologies given by Rechen berg, transformative programming recommended by Fogel and genetic calculations given by Holland.

Swarm Intelligence: - Swarm intelligence is neighbor based framework encouraged from natural living beings where simple specialists show intelligence to solve of complex problems of neighbor.

Fuzzy Systems: - Fuzzy frameworks are based on fuzzy-logic and deals with noisy and inaccurate information to get enhanced outcomes. Mostly, They are used for control and characterization or classification problems.

Various techniques are proposed to assess KDD 99 dataset based on computational knowledge and give a productive and anomaly tolerant interruption detection framework.

Liao and Vumeri [29] proposed k-closest neighbor technique for selection of features in KDD99 dataset. Jiang et al., Zhang [30] researched in artificial neuron system on KDD-99 dataset and demonstrated that back-propagation as a piece of detection for misuse anomalism has a slightly better performance over Radial Basis Function for recognition of anomalism in terms of detection rate however more training time is needed

Hofman et al. [31] gave transformative wrapper approach (joining radial premise function with genetic calculation) and determined a similar conclusion on DARPA-98 dataset. The wrapper approach utilizes genetic calculation for feature-selection while functions based on radial system are used for optimization and this approach gives better rules for classification.

The 41 attributes carried out by KDD99 dataset were organized by Mukkamala and Sung [32] through vector-support machines(SVM). They again consolidated SVM with network of neurons in [33] to organize the attributes but the time-complexity of the entire framework increased and the classification was not precise.

Selection of features are form on fuzzy set to process feature-selection, and genetic calculation to make ideal subsets was presented by Wang in [34]. The calculation was explored for utilizing SVM classifier for the assessment of performance.

A multilayered Self Organizing Map was created by Rhodes [35], and Sarasamma *et al.* making determination that distinctive feature subsets were sufficiently proficient to distinguish distinctive intrusion.

Decision-trees, associative rules and fuzzy significance table were utilized to produce fuzzy logics. The combine effort of Hidden Markov Model and fuzzy-inference framework was constructed by Cho [36] to distinguish typical associations or connections. Fuzzy C-Medoids and Fuzzy C-Means calculations were two grouping approaches, used to recognize unusual conduct through the idea of exceptions. Gomez *et al.* [37] showed the work of fuzzy classifiers on intrusion detection system.

Parvat [38] proposed the approach to recognize payload traffic using Deep Packet Inspection(DPI) and improved security of network, quality of service (QoS) and privacy. It improved the performance of IDS using in/out based attributes of records. Aggarwal *et al*. [39] proposed an idea that intrusion detection method depend on the data, used for the detection because the quality of data was also a major concern for improvising the execution of offline interruption detection. An analysis had been done on the four factors of IDS that were basic, content, traffic, and host for detection rate and false alarm Rate.

Tavallaee *et al.* [40] compared the existing approaches and found most frequent issues in the space of anomaly based mistreatment detection that needed to be resolved. Chou *et al*. [41] conferred an information hypothetic feature selection rule on each low and high dimension feature areas with analysis of correlation to verified accuracy of the IDS employing a combination of Damper-Shafer theory, nearest neighbor of k and fuzzy cluster. A hybrid fuzzy firm primarily based on parallel genetic rule approach was taken into account to deal with the necessary features in construction of IDS was considered by Mahmud [42].

Patra *et al.* [43] gave the powerful approach named as hybrid clustering for improvising of mistreatment of COBWEB and Fast Fourier Transform (FFT) cluster used in detection of various intrusions. Wang *et al.* [44] implemented a brand new meta learner with collaboration of fuzzy aggregation approach for mistreatment detection by usage of artificial neural networks (ANNs) along with fuzzy clump and claimed that proposed approach provide stability and higher detection rate as compared to back propagation neural network, naïve bayes and decision tree.

Panda [45] used a hybrid naive bayesian decision tree (NBDT) along with adaboost approach for production of the most effective rate of detection and false positive rate for implementation of IDS. An intelligent method of automatic teller agglomeration and filtration in analysis of intrusion tellers was conferred by Siraj *et al.* [46] in two aspects i.e processing time and efficiency in classification.

Guo proposed the distance sum along with support vector machine (DSSVM) approach for effective detection of attacks and also introduced sum of distance, correlation between the samples and clusters centers to detect pattern reorganization problems [47]. Ghanem introduced a hybrid approach for massive scale dataset using multi-start meta heuristic methodology based intrusion detector and genetic algorithms to remove duplicate detections and thus, to decrease the time taken for abnormal detection for live connections [48].

Ahmed *et al.* [49] represented that the optimal feature subset selection was used to improvise the execution of mistreatment unmasking over the network. For optimal feature, principal component analysis (PCA) and genetic algorithm (GA) were used, GA was used to catch the genetic principal components that provided a subgroup of features with the highest discriminatory potential and excellent sensitivity and better performance for detection of intruder's connections and able to minimize count of features and boost rate of revelation.

Erfani *et al.* [50] implemented an unsupervised anomaly detection method for high-dimensional massive unlabelled datasets. The method was a combination of a dimensionality reduction algorithm and one-class support vector machine(SVM) which offered an effective, scalable and accurate anomaly detection rate.

Pateria *et al.* [51] suggested that in network-based IDS, agent based systems hold a distributed processing architecture to collect information from various components over the network to avoid the upcoming network traffic, boost fault tolerance. With the collaboration of fuzzy logic and genetic algorithm based system, an effective approach was created to observe R2L and U2R intrusions with immense recall value while existing approaches flopped to do because there were less number of connections in training dataset proposed by Chadha [27].

Desai et al. [52] implemented a hybrid intrusion detection system to identify internal and external attacks. The internal attacks were examined by using signature matching algorithms while the external attacks examined using fuzzy genetic algorithm. The proposed hybrid system was compatible in online as well as offline environment and improved accuracy and recall value.

Ravale *et al.* [53] proposed a hybrid approach of data mining, having K-mean clustering algorithm combined with the RBF kernel function which belongs to the SVM classification method to reduce the number of the related attribute with each data and results in better performance and provide good accuracy rate.

Bamakan *et al.* [54] proposed a new method for intrusion detection which has based on the linear programming and particle swarm optimization (PSO) to enhance the accuracy of attack detection. Linear programming method based on the mathematical programming which solves the data mining problems. PSO has easy to implement and gives the better performance with multiple criteria linear programming classifiers, and results in high detection rate.

Chapter 3 Problem Statement

3.1 Gap Analysis

The KDD-99 dataset composes of 5 primary classes. Yet, two of five classes especially R2L and U2R classes count are limited for training dataset. Hence, these two classes aren't accurately trained to achieve precise outcomes and perform inadequately in testing datasets which comprises of eleven kinds of attacks. These intrusions are extremely hard to recognized also which leads to a big disadvantage in existing IDS.

Wu *et al*. proposed diverse methodologies and demonstrated that soft computing procedures perform superior to other procedures. The computational intelligence approach performed superior to decision trees. The developed classification rules did not perform well when overlying happens, the information can't be isolated into two different classes. Additionally, it is simpler to capture the fuzzy standards or rules alone.

Self-constructed maps got experienced from problems, for example, higher discovery rates having false positives, high dimensionality, and computational overhead. Evolutionary processing strategies don't have reasonable termination criteria and have not produce precise outcomes about when the information distribution is not balanced.

Swarm Intelligence methods are principally used to learn groups and classification rules yet turn out to be a limitation in high dimensional system and can't separate dissimilar items.

In this manner, cooperation of different soft computing procedures is required which have capacity to learn in an ambiguous and uncertain network. It encases all the integral features of various strategies and constructs a powerful and fault tolerant framework.

3.2 Problem Statement

The system having computational intelligence can adjust, resilience of faults against loose data, show adaptation to internal failure, and high computational speed. The fuzzy rules develop accurate and adaptable patterns while the particle swarm

optimization based on stochastic calculation helps in accomplishing an optimal solution, hence their coordinated effort will build powerful intrusion detection system. By this reason, hybrid swarm fuzzy logic based system have been proposed. The proposed IDS will have the capacity to classify TCP connections according to their behaviour over the network.

The system based on fuzzy rules, performs well in a dubious and uncertain condition, builds up more compact and malleable patterns which improves the modification capacity and strength of the intrusion detection system and perform classification for every connection, either in normal or in abnormal behaviour accurately.

3.3 Objectives of the Proposed Work

The fundamental objective behind the proposed work is to construct a model which has high detection rate and low or negligible false alert rate. It must be precise and have capability to classify each attacks in their actual classes and should show the property of high flexibility i.e. the capacity to modify as indicated by changing conduct of the clients over the network and altering itself for proper working.

By this reason, the main objectives of the proposed IDS are

- To generate classification rules using fuzzy rules or logic.
- To optimized rules using PSO
- To validate proposed model with existing one.

Chapter 4 Implementation

The presented approach is combination of the fuzzy system with swarm optimization build a hybridized swarm fuzzy rule-based system which gives vigorous platform to distinguish interruptions existing in the network distinctively and classify them into normal and distinctive types of attacks as per their marks. The proposed work is performed on KDD-99 data set which is a standard dataset used to recognize interruptions in the system.

4.1 Evaluation of KDD-99 Dataset

The KDD-1999 IDS dataset utilizes a form of database that was composed in 1998 DARPA Intrusion Detection Evaluation Program to assess their exploration in interruption recognition. It comprised of nine weeks of crude TCP dump connections (data) as training dataset and two weeks of testing dataset. The KDD-99 dataset was utilized as a part of Third International Knowledge Discovery and Data Mining Tools Contest to set up an interruption identifier which can distinguish normal or abnormal links (connections) [56]. This dataset contains standard review information including a huge count of intrusion.

The dataset comprises of connections and every connection is a grouping of TCP packets. Every connection contains 41 attributes and labeled with either normal or type of attack. In proposed research, 10% of the KDD-1999 dataset is utilized to assess the entire procedure.

The 10% KDD dataset comprises of 4,94,021 connections /records for training dataset. Training dataset comprises of various types of 24 training attacks which comes under 4 main classes of attacks while the testing dataset comprises of 14 more attacks to distinguish numerous signatures and check whether these variations are caught or not to increase proficiency.

The following is the table which comprises of classes in which the entire training dataset is categorizing and related to each class the subclasses are specified.

Table 4.1 Classification of Classes

S.No	Class Label	Sub Classes
1	Denial of service (DoS)	land, smurf, neptune back, pod, teardrop
2	Remote to User (R2L)	phf,spy,guess_passwd,ftp_write,imap, warezmaster, warezclient,
3	User to Root (U2R)	loadmodule, multihop, rootkit,perl, buffer_overflow,
4	Probe	ipsweep,portsweep, satan, Nmap
5	Normal	NA

The attributes of KDD-99 dataset as characterized by Stolfo *et al.* [55] are classified into following classes

Essential Attributes: - This classification includes all the basic properties that can be recovered from an individual TCP connection which include 9 attributes.

Time Based Traffic Attributes: - It incorporates features which are computed on the premise of time-interval and is subdivided into two kinds: -

a) **Same Host Attributes:** - It analyzes only the connections in the previous two seconds that have an indistinguishable destination host as the present connection.
b) **Same Service Attributes:** - It inspects only the connections in the previous two seconds that have an indistinguishable service from the present connection.

Host Based Traffic Attributes: - In it, attributes are created by utilizing a window of 100 connections with a similar host rather than a period window of 2 seconds.

Content Based Attributes: -It comprises of 13 attributes that are extricated from domain information and are utilized to show suspicious conduct in the system or unstructured information bytes in the packets.

Table 4.2 depicts the 41 attributes of the KDD-Cup 99 dataset and indicates whether the attributes is of continuous type or of symbolic type.

Table 4.2 KDD-99 CUP Dataset

S.No	Attributes	Type
1	Duration	Continuous
2	Protocol_type	Symbolic
3	Service	Symbolic
4	Flag	Symbolic
5	Src_bytes	Continuous
6	Dst_bytes	Continuous
7	Land	Continuous
8	Wrong_fragment	Continuous
9	Urgent	Continuous
10	Hot	Continuous
11	Num_failed_logins	Continuous
12	Logged_in	Symbolic
13	Num_compromised	Continuous
14	Root_shell	Continuous
15	Su_attempted	Continuous
16	Num_root	Continuous
17	Num_file_creations	Continuous
18	Num_shells	Continuous
19	Num_access_files	Continuous
20	Num_outbound_cmds	Continuous
21	Is_host_login	Continuous

22	Is_guest_login	Continuous
23	Count	Symbolic
24	Srv_count	Continuous
25	Serror_rate	Continuous
26	Srv_serror_rate	Continuous
27	Rerror_rate	Continuous
28	Srv_rerror_rate	Continuous
29	Same_srv_rate	Continuous
30	Diff_srv_rate	Continuous
31	Srv_diff_host_rate	Continuous
32	Dst_host_count	Continuous
33	Dst_host_srv_count	Continuous
34	Dst_host_same_srv_rate	Continuous
35	Dst_host_diff_srv_rate	Continuous
36	Dst_host_same_src_port_rate	Continuous
37	Dst_host_srv_diff_host_rate	Continuous
38	Dst_host_serror_rate	Continuous
39	Dst_host_srv_serror_rate	Continuous
40	Dst_host_rerror_rate	Continuous
41	Dst_host_srv_rerror_rate	Continuous

These 41 attributes give the descriptive information about each connection. With the help of these attributes, identification of type of attack is possible.

4.2 Fuzzy Rule Based System

In the presented work, there are three major components- fuzzy rule-based systems, particle swarm optimization and computational model. In this segment the fuzzy rule-based system to identify interruptions is examined.

Step 1-Representation of data

The training dataset of 10% KDD-99 dataset comprises of 4, 94,021 records or connections. 750 random samples are considered and each connection contains 41 attributes. In this manner, fuzzy rule-based system has m i.e. 750 labeled patterns having n=41 dimensionality, therefore every pattern is described as

$$X_p = \{X_{p1}, X_{p2}, X_{p3,} X_{p4,} X_{p5,} X_{p6,\ldots\ldots\ldots\ldots,} X_{p39,} X_{p40,} X_{p41}\}$$

where p represents training dataset i.e $p = 1,2\ldots, m$ and every pattern is desired to be characterized into main c classes i.e. 5 classes. The attributes whose type are continuous, considered for further handling.

Five linguistic variables (Li) for each attribute, or features are considered (i.e. Low, Low, Medium, High, and Very High). Each attribute definition interim is consistently partitioned by utilizing triangular fuzzy sets.

Accordingly the total count of fuzzy if then rules produced is 5^n where n defines the pattern of n dimensions. In this case, count of attributes is high ($n = 41$), it is difficult to create 5^{41} rules. By this reason, heuristic approach is used to generate fuzzy if-then rules.

Step2-Normalization of data

Since the area of pattern considered is $[0,1]^n$, hence the value of attributes of every pattern is normalized in the scope of $[0,1]$ $i.e. X_{pi} \in [0,1]$ where the normalization formula is described by:-

$$y = \frac{x - min}{max - min} \tag{6}$$

Where x defines the actual value of attribute, min defines the lower limit value i.e. 0, max defines the upper limit value i.e. 1 and y defines normalized value i.e. $0 \leq y \leq 1$.

Step3-Calculating membership function

After normalization of the dataset, the membership function of each training pattern is calculated by the accompanying equation: -

$$\mu_j(x) = \max\left\{0,1 - \frac{|x - x_j|}{v}\right\} \quad (7)$$

where x defines every attribute's normalized value,

$$x_j = \frac{j-1}{L-1} \quad (8)$$

where $j = 1,2,3,\ldots L$,

and

$$v = \frac{1}{L-1} \quad (9)$$

where L defines the count of linguistic labels and changes from 1-5 that helps in calculation of membership function of each feature or attribute.

Step4-Determining every training pattern's compatibility

Every training pattern's compatibility x_p is computed with the help of fuzzy if-then rules R_j by using the accompanying formula:-

$$\mu_j(x_p) = \prod_{i=1}^{n} \mu_{ji}(x_{pi}) \quad (10)$$

where p represents training pattern, $p = 1,2\ldots,m$ and $\mu_{ji}(x_{pi})$ is the membership function of the i^{th} feature of p^{th} pattern.

Step5- Computing compatibility of every class

After getting the compatibility of every training pattern, relative sum of all compatibility grades of the training patterns having rule R_j, used for each class is computed which is given by:-

$$\beta_{class\ h}(R_j) = \frac{\sum x_{p\,\epsilon\,Class\ h}\mu_j(x_p)}{N_{Class\ h}} \quad (11)$$

where $\beta_{\text{class h}}(\text{R}_\text{j})$ defines the mean sum of compatibility grades in Class h with fuzzy if-then rule, $N_{Class\ h}$ represents the count of training patterns are used relating to each $class\ h$ and h =1,2,… ..,c. where c=5. Here five major classes i.e normal, u2r attacks, dos attacks, r2l attacks and probe attacks.

Step6-Selection fuzzy if-then rule for a specific class

The rules based on fuzzy if-then are chosen by the given training patterns. The subsequent class C_j for the stated R_j computed as below:

$$\beta_{\text{Class h}}(\text{R}_\text{j}) = \max_{\text{h}=1,\ldots,\text{c}}\left(\beta_{\text{Class h}}(\text{R}_\text{j})\right) \tag{12}$$

The maximal value of $\beta_{Class\ h}(R_j)$ is calculated and the one having maximal value is thought to be the class of the fuzzy if-then rule R_j. If maximal value get out as valid or true for more than one class, at that point the resulting class C_j can't be decided particularly and is taken as φ and the relating rule is rejected.

Step7-Assessing the confidence degree CD_j

The certainty, or confidence degree alludes to how exactly an input pattern is categorized by the fuzzy rule-based framework i.e. deciding the genuineness of the class with a specific certainty value. It is computed by the below formula: -

$$CD_j = \frac{\beta_{Class\ h_j}(R_j) - \bar{\beta}}{\sum_{h=1}^{c} \beta_{Class\ h_r}(R_j)} \tag{13}$$

where

$$\bar{\beta} = \frac{\sum_{\text{h}\neq \text{h}_\text{r}} \beta_{\text{Class h}}(\text{R}_\text{j})}{\text{c} - 1} \tag{14}$$

The confidence degree comes in the interval [0, 1]. If subsequent class C_j is φ , at that point the certainty is likewise φ, this decides the absence of legitimacy of that class in the specific rule.

The estimation of $CD_j = 1$ shows high certainty which indicates that the rule belongs with the particular class.

Step8-Generation of fuzzy if-then rules

Hence, fuzzy if-then rules has been created in the below way: -

Rule R_j = If x_1 is A_{j1} and x_2 is A_{j2} and x_3 is A_{j3} and … , x_n is A_{jn}, then class is C_j with CD_j, j = 1,2,3 ….. N

where R_j is the name or label of the j^{th} fuzzy if-then rule or logic, means the A_{j1}, A_{j2}, A_{j3} represents predecessor fuzzy sets, C_j shows the resulting class, CD_j alludes to the confidence degree or certainty in the class name or label and N is the count of rules.

For instance- If protocol value is 1.0,....., and dst_host_srv_count value is 1.0,....., and dst_host_srv_diff_host_rate value is 0.92,..... and dst_host_srv_rerror_rate value is 1.0 , at that point the class determined is r2l attack with confidence degree =1.0.

4.3 Particle Swarm Optimization

Particle swarm algorithm is outstanding in optimization and search problems. This algorithm imposes on recorded data to extract optimized outcome.

- **Formulation**

In the present work, researcher reduces the feature set by feature selection and feature extraction with the help of Fuzzy logic and PSO. In this work author solves the problem related to feature selection which is not based on the information gain and correlation. That's why in previous methods of feature selection is not able to provide knowledge of different types of intrusion.

- To reduce the feature set by feature selection by Fuzzy logic and Particle swarm optimization.
- To implement linear classifier on given feature of dataset.
- To analyze the classifier models on the basis of accuracy, precision, recall and F1-score.

- **Algorithm**

1: Generate the Fuzzy rules and then input them as a particle in PSO
2: In PSO model for each particle i in S do
3: for each dimension d in D do

4: //initialize each particle's position and velocity

5: $x_{i,d} = Rnd(x_{max}, x_{min})$

6: $v_{i,d} = Rnd(-v_{max}/3, v_{max}/3)$

7: end for

8: //initialize particle's best position and velocity

$v_i(k+1) = v_i(k) + \gamma 1_i(p_i - x_i(k)) + \gamma_{2i}(G - x_i(k))$

New velocity

$x_i(k+1) = x_i(k) + v_i(k+1)$

Where

i- particle index

k- discrete time index

v_i –velocity of i^{th} particle

x_i – position of i^{th} particle

P_i- best position found by i^{th} particle (personal best)

G- best position found by swarm (global best, best of personal bests)

$G_{(1,2)i}$- random number on the interval[0,1]applied to the i^{th} particle

9: $pb_i = x_i$

10: // update global best position

11: *if* $f(pb_i) < f(gb)$

12: $gb = pb_i$

13: end if

14: end for

Initialization: In this work, firstly initial fuzzy rules are generated and given as input to the PSO. PSO initialize the location and velocity of the particles and then calculates the fitness value of each particle.

Processing: Check the current fitness value is better than Pbest if yes then keeps it otherwise assign the current fitness as new fitness.

a) After this assign the best particle P_{best} to G_{best} and calculate the velocity of each particle. The velocity value of particle is used for updation.

b) Replace the population with new rule and apply heuristics. After this construct the math vertical model for new rule.

4.4 Mathematical Model

After execution all the above steps, to acquire more optimized and accurate outcomes, a mathematical model is implemented. The main focus is on that the created rules cover-up all the samples of training dataset.

Here are the following parameters and sets are accessible based on which more patterns will be covert in the fuzzy if-then rules and accuracy will be augmented as given by [36]: -

- C - defines set of classes
- F- set of attributes
- S- defines set of samples
- R- defines set of rules
- μ_{sr}- defines membership function of sample s for rule r.
- α_r- defines accuracy value for rule r.
- C_s- represents the label of class for sample s.
- M- a very huge number

Variables for decision are represented by:-

- $x_r = \begin{cases} 1 \text{ if rule r is selected} \\ 0 \qquad\qquad \text{else} \end{cases}$ (15)
- $y_{sc} = \begin{cases} 1 \; if \; sample \; s \; is \; classified \; as \; class \; c \\ 0 \qquad\qquad\qquad\qquad\qquad else \end{cases}$ (16)

The model is given by: -

Step1-Classification of samples

More count of samples would be classified effectively. In this manner samples should be classified into their right class else ways any attack could be misclassified as normal and may represent a genuine danger to the network and this is completed by boosting the count of correct classification.

$$\text{Maximize } Z = \sum_{s\in S, c\in C, C_s=c} y_{sc} \tag{17}$$

while satisfying the accompanying conditions :-

$$\sum_{r\in R: C_r=c} x_r \geq 1 \; \forall c \in C \tag{18}$$

which defines that no less than one rule is chosen from each class and the value of $x_r = \{0,1\} \forall r \in R$

$$\sum_{c \in C} y_{sc} \leq 1 \; \forall s \in S \tag{19}$$

Equation 19 examines that each sample must be classified exactly in one class and must not show ambivalent behavior.

To get the value of y_{sc} steps2 and step3 are taken.

Step2-Calculating accuracy of the rule

To get the accuracy of each chosen rule, the accuracy is figured as:-

$$\text{Accuracy}_s \geq \sum_{r \in R: C_r = c} \mu_{sr} \alpha_r x_r \; \forall s \in S, \forall c \in C \tag{20}$$

where $\text{Accuracy}_s \geq 0 \; \forall s \in S$ is an constraint.

Step3-Assurance of right class for each sample

Below equation guarantees that for each rule, the correspondent class chosen is correct.

$$y_{sc} \leq 1 - \left(\frac{1}{M}\right) \left(Accuracy_{sc} - \sum_{r \in R: C_{r=c}} \mu_{sr} \alpha_r x_r\right) \forall s \in S, \forall c \in C \tag{21}$$

For the chosen class, $\text{Accuracy}_{sc} - \sum_{r \in R: C_{r=c}} \mu_{sr} \alpha_r x_r = 0, \text{and } y_{sc} \leq 1.$

Yet, for the unselected class, $\text{Accuracy}_{sc} - \sum_{r \in R: C_{r=c}} \mu_{sr} \alpha_r x_r \geq 0$. Subsequently, $y_{sc} < 1$. By the integrality constraint $y_{sc} = \{0,1\} \forall s \in S, \forall c \in C$, therefore for the unselected rule, $y_{sc} = 0$ and for the chosen rule $y_{sc} = 1$.

The requirement $y_{sc} \leq \left(\sum_{r \in R: C_{r=c}} x_r\right) \forall s \in S, \forall c \in C$ keeps the classification of unselected rule for each class.

Subsequently the primary target is to escalates Z and cover up all training sample. Hence, the possibility of acquiring the best set of rule will be amplified which will enhance the classification of attacks in testing dataset as more attributes will be accessible for classification.

The overall working of system is given in the accompanying flowchart in Figure 4.1.

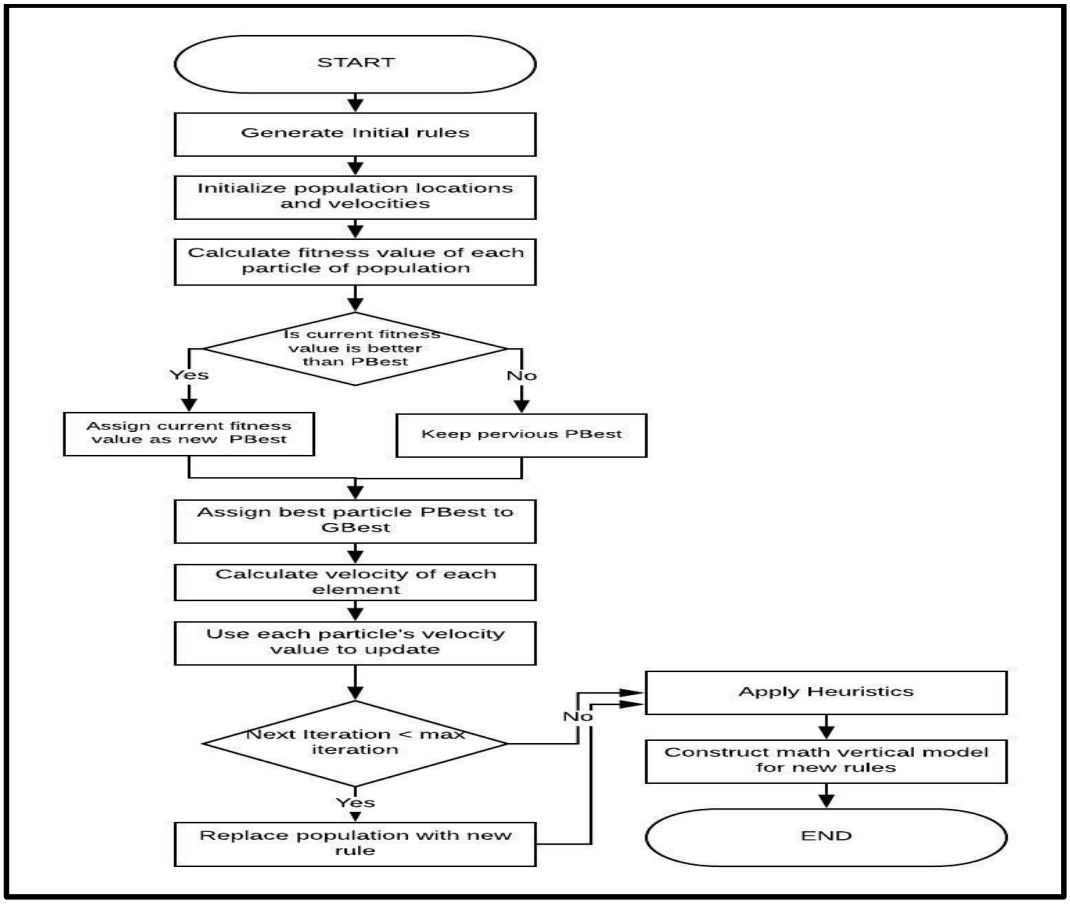

Figure 4.1 Flow Chart of Hybrid Fuzzy PSO System

Chapter 5 Results and Discussion

The analysis has been set down on KDD-99 dataset to set up an intrusion detection system and was implemented in Python 3.6. This dataset contains standard review information including a huge count of intrusion. In this proposed approach, 10% of the KDD99 dataset has been utilized i.e. the total count of records are 494,021. The proposed hybrid swarm fuzzy rule based system comprises of rules, or standard which are created by using training dataset while the tests are performed on testing dataset to approve the performance of the rules.

In the given practice, 750 samples were randomly selected from the training dataset furthermore, the value of each feature was normalized and standardized in the unit interval [0,1]. Fuzzy if-then rules were created of a specific size as an initial population and after that different swarm operations were executed to acquire the best standards or rules. The procedure proceeds until a end condition is fulfilled. The rules are examined randomly chosen testing connections and the conclusions were drawn about the feasibility of the swarm fuzzy interruption detection framework through different parameters.

5.1 Evaluation Parameters

The IDS is essentially assessed through the confusion matrix or possibility table which was given by Kohavi and Provost. The matrix contains data about the predicted and actual classification done by the system. It comprises of the accompanying records: -

- **True Negative (TN)** – It alludes to the count of right events which anticipated them as true associations or connections.
- **False Positive (FN)** – It infers the count of wrong predictions which analyzed bona fide events as phony events.
- **False Negative (FN)** – The count of wrong predictions which assessed false connections as true connections.
- **True Positive (TP)** – It alludes to the count of true predictions which foresee that the connection is fake or peculiar.

Here is the confusion matrix which certainly determines the results.

Table 5. 1 Confusion Matrix

		Predicted Classes	
		Negative	**Positive**
Actual Classes	**Negative**	**TN**	**FP**
	Positive	**FN**	**TP**

Thusly for this confusion matrix, different norms have been assessed to measure the performance of IDS.

- **Accuracy**- It is depicted as the ratio of total count of connections that are correct and is given by the formula: -

$$\text{Accuracy} = \frac{\text{TN} + \text{TP}}{\text{TN} + \text{FP} + \text{FN} + \text{TP}} \quad (22)$$

- **True Negative Rate**- It is characterized as the ratio of false or negative connections that are classified precisely. It is also known as specificity and is computed as: -

$$\text{TNR} = \frac{\text{TN}}{\text{TN} + \text{FP}} \quad (23)$$

- **True Positive Rate**- It is the ratio of actual events that are classified correctly. It is otherwise called as discovery rate or sensitivity or recall. It is given as: -

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (24)$$

- **False Positive Rate**- It is the proportion of authentic events that are misclassified in another class. It is additionally alluded to as False Alarm Rate and is figured as: -

$$\text{FPR} = \frac{\text{FP}}{\text{TN} + \text{FP}} \quad (25)$$

Or

$$\text{FPR} = 1 - \text{TNR} \quad (26)$$

- **False Negative Rate**-It is the ratio of fake connections that were mistakenly classified as normal events and is indicated by: -

$$\text{FNR} = \frac{\text{FN}}{\text{TP} + \text{FN}} \tag{27}$$

Or

$$\text{FNR} = 1 - \text{TPR} \tag{28}$$

5.2 Results

The proposed approach is created and applied on testing dataset to test and analyze the accuracy and robustness of the hybrid Fuzzy-PSO IDS.

Following are the results obtained from the confusion matrix: -

Table 5. 2 Confusion Matrix of the Proposed Approach

Actual Class	Predicted Class					
	Normal	DoS	U2R	R2L	Probe	Total
Normal	2263	8799	0	12	86	11160
Dos	1662	6350	0	9	52	8073
U2R	1	2	5	0	4	12
R2L	5	14	0	14185	0	14204
Probe	67	284	0	0	1123	1474

The obtained results are analyzed for different classes based on various parameters and are given as: -

Table 5. 3 Outcomes of Different Parameters for Various Classes

Classes	TPR	TNR	FPR	FNR	Accuracy
Normal	95.98	97.23	2.3	4.01	96.38
DoS	98.91	99.78	0.21	1.08	97.19
U2R	90.54	99.75	0.24	9.45	99.75
R2L	96.98	99.33	0.66	3.01	99.12
Probe	97.54	98.45	1.54	2.45	98.76

By applying the proposed approach, recall or detection rate for every class has better results, especially for the DoS, R2L and Probe than existing approaches. The false positive rate, or false alarm rate is not exceeding 2.3. Hence, it gives more accuracy and less misclassification of connections.

The precision for each class of IDS has been evaluated using 10,000 randomly selected records or connections of the training dataset. The results of existing approaches are compared with the current approach and validated to check whether the outcomes are more accurate and better than existing IDS. Following is the graph which represents precision on each class of existing and proposed approaches.

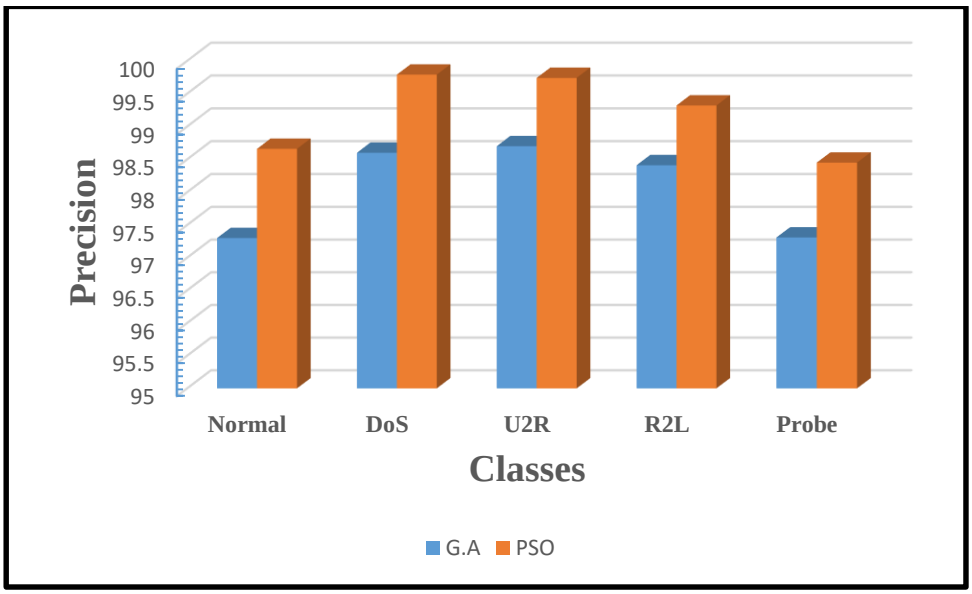

Figure 5.1 Precision on Each Class of Existing and Proposed Approach

Figure 5.1 depicts the result of precision of five different classes that are Normal, DoS, U2R, R2L and Probe classes of existing and proposed approach. The value of DoS has been improved from 98.60 to 99.79, R2L has been evaluated with better precision 99.32 i.e. higher than existing approach where precision for R2L is 98.4, the recall value for Probe class is slightly improved from 97.3 to 98.45 with the help of current approach.

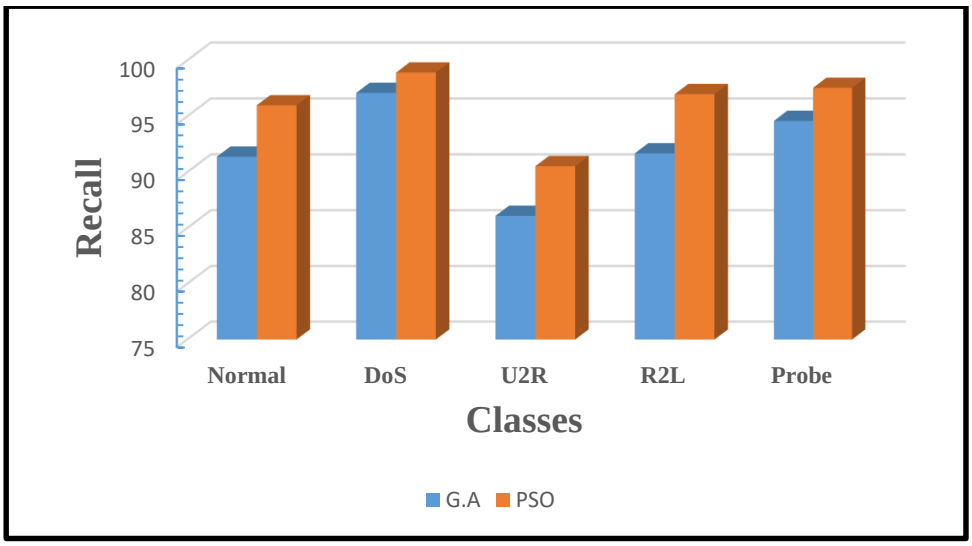

Figure 5.2 Comparisons of recall values for Existing and Proposed Approach

The Figure 5.2 shows the results on different classes of existing and proposed IDS on the basis of recall value. The value of DoS has been improved from 97.1 to 98.92, R2L has been evaluated with better recall value 96.99 i.e. higher than existing approach where recall for R2L is 91.7, the recall value for Probe class is slightly improved from 94.6 to 97.55 with the help of current approach.

The below Figure 5.3 depicts the results on different classes of existing and proposed IDS on the basis of F1-Score value.

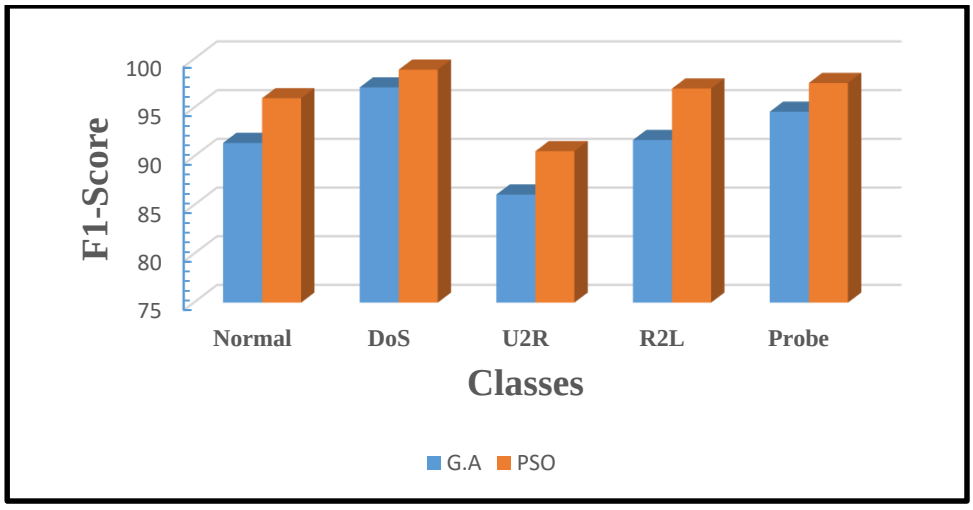

Figure 5.3 F1-Score on Each Class of Existing and Proposed Approach

The value of DoS has been improved from 97.1 to 98.92, R2L has been evaluated with better F1-Score 96.99 i.e. higher than existing approach where F1-score for R2L is 91.7, the F1-score for Probe class is slightly improved from 94.6 to 97.55 with the help of current approach.

Chapter 6 Conclusion and Future Scope

6.1 Conclusion

The system based on hybridization of fuzzy logic with particle swarm algorithm, a potent method is created for detection of attacks in an effectual way and compared with genetic algorithm.

This approach is capable in detection of DoS, R2L and probe intrusions with best F1-score, recall value and precision value while the existing approaches fail to do because training dataset has less number of records as shown in Figure 5.3, 5.2, 5.1.

Due to successful implementation of computational model, best logics and rules are preserved. Hence, providing a better approach for detection of attacks. The rules based on classification are capable to differentiate normal and anomaly behavior of traffic over the network with better accuracy is shown in Table 5.3.

The swarm operators ensure significant particle, as the class of produced rules is matched with their parent class. If they are same, at that point created rule is acknowledged, else it is rejected and the entire procedure is repeated. This approach reduces misclassification of rules and accordingly increase accuracy.

The swarm calculations are proceeded for some specific particles which give greater validity and accuracy to the rules. The fuzzy if-then else rules perform exceedingly well on loose and uncertain information, therefore the system is fault tolerant.

As the rules continuously updating themselves in the database with the changing connections, hence, the intrusion detection system has the property of high adaptability.

In this way, the classification rules have ability to classify the normal and anomaly behavior in the system with high accuracy, thus leaving less loopholes for the misinterpretation.

6.2 Summary

The proposed approach hybrid the features of fuzzy rule based system and particle approach to construct a strong and powerful intrusion detection system.

The fuzzy if-then rules fetch attributes or features even from such small and uncertain dataset to extract significant features which will help in further classification. The particle swarm algorithm helps in acquiring optimized rules because of their ability in

exploiting historical data and yielding better outcomes.

The swarm fuzzy rule-based system can work in high dimensional system and can deal with large amount of audit data. KDD-99 data set is a high dimensional dataset with 41 attributes or features containing more than 4, 00, 000 records of connections. So the IDS have the capacity to deal with huge amount of information easily.

Swarm calculations are utilized as a part of blend with fuzzy if-then rules to prevent overlapping and thusly every variable is unique. For every connection of dataset, compatibility factor is calculated at the phase of velocity updation and afterward is classified into particular class which has higher compatibility factor.

Additionally, the mathematical model tries to cover each sample of the training dataset by classifying them through rule set, thus boosting accuracy.

6.3 Future Scope

Reducing the false identification rate and generously increasing the recall is one of the major concerns of any effective and successful intrusion detection system. The proposed approach has a high detection rate yet a little higher false alert rate of around 1.27%, so the primary aim would be decrementing the false alert rate to avoid misclassification of connections.

The count of rules to be considered and kept in the database has not yet been assessed. Hence the scope of this work can be extended by working on creating compact and concise rules. In this manner, it would help in maximum classification in least count of rules.

This approach can also be improved more successfully by using better local search algorithms to produce more minimized and precise rules as these algorithms would look more into local optima, consequently yielding different local optimal solutions which would maintain diversity and validity in rules.

References

[1] N. I. of Standards & Technology, "An Introduction to Computer Security", *The NIST Handbook, Ed. U.S. Department of Commerce*, 2006.

[2] J. Grossklags, N. Christin, and J. Chuang, "Security and insurance management in networks with heterogeneous agents," *in Annual International Conference on Electronic Commerce*, pp. 160–169, 2008.

[3] H. Ishibuchi, and T. Yamamoto, "Fuzzy rule selection by Data Mining Criteria and Genetic Algorithms", *in International Conference on Genetic and Evolutionary Computation*, pp. 399-406, 2002.

[4] S.H. Chen, J. Wu, and Y.D. Chen. "Interval optimization for uncertain structures", *International Journal of Finite Elements' Analysis and Design*, vol. 40, no. 11, pp. 1379–1398, 2004.

[5] R. Vetschera. "A general branch-and-bound algorithm for fair division problems", *in Proceeding of International Conference on Computers Network and Operations Research*, pp. 234-239, 2010.

[6] M. Veltman. "Algebraic techniques. Computer Physics Communications", *International Journal of Computer Communication*, vol. 3, no. 10, pp. 75–78, 1972.

[7] X. Geng, J. Xu, J. Xiao, and L. Pan, "A simple simulated annealing algorithm for the maximum clique problem", *International Journal of Information Sciences*, vol. 22, no. 177, pp. 5064–5071, 2007.

[8] Eidehall, and L. Petersson, "Threat assessment for general road scenes using Monte Carlo sampling". *International Conference on Intelligent Transportation Systems Conference*, pp. 30-37, 2006.

[9] M. Lin and J. Wawrzynek, "Improving FPGA placement with dynamically adaptive stochastic tunneling", International Journal of Computer-Aided Design of Integrated Circuits and Systems, vol. 29, no. 12, pp. 1858–1869, 2010.

[10] J. Machta, "Strengths and weaknesses of parallel tempering", *International Journal of E - Statistical, Nonlinear, and Soft Matter Physics*, vol. 80, no. 5, pp. 123-129, 2009.

[11] Engelbrecht, "Computational Intelligence — An Introduction 2nd Edition", 2007.

[12] Pan, C. Xu, and G. Li, “Differential evolutionary strategies for global optimization”, *in Journal of Computational intelligence*, vol. 25, no. 2, pp. 211–215, 2008.

[13] Blaha, and D. Wunsch., “Evolutionary programming to optimize an assembly program”, *in Proceedings of Conference on Evolutionary Computation*, pp. 1901–1903, 2002.

[14] J.Kennedy, and R.Eberhart, “Particle swarm optimization”, *in Proceedings of International Conference on Neural Networks*, pp. 1942–1948, 1995.

[15] C. Wu, “Ant colony multilevel path optimize tactic based on information consistence optimize”, *International Conference on Computer Application and System Modeling (ICCASM)*, pp. 533–536, 2010.

[16] J. Digalakis and K. Margaritis. “Performance comparison of memetic algorithms”, *in International Conference on Applied Mathematics and Computation*, pp. 237–252, 2004.

[17] X. Yang and S. Deb, “Cuckoo search via lights”, *in Proceedings of the World Congress on Nature, Biologically Inspired Computing (NaBIC)*, pp. 210-214, 2009.

[18] DE. Denning, “An intrusion-detection model”, *International Journal on Software Engineering*, vol. 13, no. 2, pp. 222-232, 1987.

[19] PA. Porras, and A. Valdes, "Live Traffic Analysis of TCP/IP Gateways." *in Proceedings of International Conference on Network and Distributed System Security Symposium*, pp. 222-227, 1998.

[20] H.Debar, M.Dacier, and A.Wespi, “Towards a taxonomy of intrusion detection systems”, *International Journal on Computer Networks*, vol. 31, no. 8, pp. 805-822, 1999.

[21] H.Debar, M.Dacier, and A.Wespi, “A revised taxonomy for intrusion detection systems”, *International Journal on Computer Security*, vol. 55, no. 7, pp. 361-378, 2000.

[22] J. Jordan, S. Helwig, and R. Wanka, “Social interaction in particle swarm optimization, the ranked and adaptive multi-swarms”, *in Proceedings of the Annual Conference on Genetic and Evolutionary Computation*, pp. 49–56, 2008.

[23] J. Kennedy, and R. Eberhart. “Particle swarm optimization”, *in Proceeding of International Conference on Neural Networks*, pp. 1942–1948, 1995.

[24] J. Kennedy, and R. Mendes, “Neighborhood topologies in fully-informed and best of neighborhood particle swarms”, *in Proceedings of the International Conference on Soft Computing in Industrial Applications*, pp. 45–50, 2003.

[25] G. Lapizco-Encinas, C. Kingsford, and J. Reggia., “A cooperative combinatorial particle swarm optimization algorithm for side-chain packing”, *in Proceeding of International Conference on Swarm Intelligence Symposium,* pp. 111-116, 2009.

[26] O. Schutze, E. Talbi, C.C. Coello, L.V. Santana-Quintero, and G.T. Pulido, “A memetic pso algorithm for scalar optimization problems”, *in Proceedings of the International Conference on Swarm Intelligence Symposium,* pp. 128–134, 2007.

[27] K. Chadha and S. Jain, "Hybrid Genetic Fuzzy Rule Based Inference Engine to Detect Intrusion in Networks", *in the Proceedings of International Conference on Intelligent Distributed Computing, Advances in Intelligent Systems and Computing*, pp. 231-235, 2015.

[28] J.C Bezdek, “Computational Intelligence Defined-by Everyone”, *in Proceeding of the International Conference on Computational Intelligence: Soft Computing and Fuzzy-Neuro Integration with Applications*, pp. 10-37, 1998.

[29] Y. Liao, and V. R. Vemuri, “Use of k-nearest neighbor classifier for intrusion detection”, *International Journal on Computers and Security*, vol. 21, no. 5, pp. 439-448, 2002.

[30] C. Zhang, J. Jiang, and M. Kamel, “Intrusion detection using hierarchical neural networks”, *International Journal on Pattern Recognition Letters*, vol. 6, no. 6, pp. 779-791, 2005.

[31] Hofmann, T. Horeis, and B. Sick, “Feature selection for intrusion detection: an evolutionary wrapper approach”, *in Proceedings of the International Joint Conference in Neural Networks*, pp. 1563-1568, 2004.

[32] H. Sung, and S. Mukkamala, “Feature ranking and selection for intrusion detection systems using support vector machines”, *in Proceedings of the Second Digital Forensic Research Workshop*, 2002.

[33] S. Mukkamala, G. Janoski, and A. Sung, “Intrusion detection using neural networks and support vector machines”, *in Proceedings of the International Joint Conference in Neural Networks*, pp. 1702-1707, 2002.

[34] X. Wang, J. Yang, X. Teng, W. Xia, and R. Jensen, “Feature selection based on rough sets and particle swarm optimization”, *International Journal on Pattern Recognition Letters*, vol. 28, no. 4, pp. 459-471, 2007.

[35] B. C. Rhodes, J. A. Mahaffey, and J. D. Cannady, “Multiple self-organizing maps for intrusion detection”, *in Proceedings of the Conference on National Information Systems Security*, pp. 16-19, 2000.

[36] S. B. Cho, “Incorporating soft computing techniques into a probabilistic intrusion detection system”, *International Journal on Systems, Man, and Cybernetics*, vol. 32, no. 2, pp. 154-160, 2002.

[37] J. Gomez, and D. Dasgupta, “Evolving fuzzy classifiers for intrusion detection”, *in Proceedings of the International Workshop on Information Assurance*, pp. 321-323, 2002.

[38] P. Thaksen, and P. Chandra. "A Novel approach to deep packet inspection for intrusion detection.", *International Journal on Computer Science*, vol. 45, no. 5, pp. 506-513, 2015.

[39] P. Aggarwal, and S. Kumar, "Analysis of KDD dataset attributes-class wise for intrusion detection", *International Journal on Computer Science*, vol. 57, no. 4, pp. 842-851,2015.

[40] T. Sharma, and F. Ghorbani, “Towards credible evaluation of anomaly based intrusion detection methods”, *International Journal on System, Man and Cybernetics*, vol. 40, no. 5, pp. 516-518, 2010.

[41] S. Chan, K. Yen, and J. Luo, “Network intrusion detection design using feature selection of soft computing paradigms”, *In the Proceedings of International journal of computational intelligence*, vol. 4, no. 9, pp.196-208, 2008.

[42] S. Mahmud, H. Agiza, and E. Radwan., “Intrusion detection using rough sets based parallel genetic algorithm hybrid model”, *In the Proceeding of International Conference on Network Security*, pp. 123-128, 2009.

[43] M. Panda, and P. Gupta, “A Hybrid clustering approach for network intrusion detection using cobweb and FFT”, *In the Proceedings of Journal of Intelligent systems*,vol. 18, no. 7, pp. 229-233, 2009.

[44] G. Wang , J. Hao and L. Huang , “A new approach to intrusion detection using ANN and fuzzy clustering”, *In the International Journal on Expert systems with application*, vol. 20, no. 5, pp. 220-224, 2010.

[45] M. Panda, and H. Patra, “Semi Naïve Bayesian method for anomaly based network intrusion detection*”, In the Proceedings of International Conference on Computer networks and Security*, pp. 614-621, 2009.

[46] M. Siraj, M.Sharma, and K. Hashim, “A hybrid intelligent approach for automated alert clustering and filterin inintrusion alert analysis”, *In the International Journal of computer theory and engineering*, vol.1, no. 7, pp.539-45,2009.

[47] G. Chun, Y. Zhou, and Y. Ping, "A distance sum-based hybrid method for intrusion detection." *in Proceeding of International Conference on Applied intelligence*, pp. 178-188, 2014.

[48] G. Tamer, S. Elkilani, and M. Kader. "A hybrid approach for efficient anomaly detection using metaheuristic methods." *in the Proceedings of Journal of advanced research*, vol. 6, no. 7, pp. 609-619, 2015.

[49] S. Ahmad, and P. Iftikhar, "Enhancing SVM performance in intrusion detection using optimal feature subset selection based on genetic principal components." *in the Proceedings of International Conference on Neural Computing and Applications*, pp. 1671-1682, 2014.

[50] S. Erfani, and M. Sarah, "High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning.", *in the proceeding International Conference on Pattern Recognition*, pp. 121-126, 2016.

[51] P. Jain, S. Raghuwanshi, and P. Gupta, “New Mobile Agent Based Intrusion Detection Systems for Distributed Networks.”, *in International Journal of Wireless Communication*, vol. 1, no. 7, pp. 165-171, 2011.

[52] B. Desai, and P. Gaikwad, "Real time hybrid intrusion detection system using signature matching algorithm and fuzzy-GA," *in the Proceedings of International Conference on Advances in Electronics, Communication and Computer Technology (ICAECCT)*, pp. 291-294, 2016.

[53] Ravale, Ujwala, Nilesh Marathe, and Puja Padiya, "Feature selection based hybrid anomaly intrusion detection system using K means and RBF kernel function.", in Proceeding of International Conference on Procedia Computer Science, pp. 428-435, 2015.

[54] Bamakan, Seyed Mojtaba, and Hosseini, "A new intrusion detection approach using PSO based multiple criteria linear programming.", *in Proceeding of International Conference on Procedia Computer Science*, pp. 231-237, 2016.

[55] E. K. Aydogan, I. Karaoglan, and P. M. Pardalos, “HGA: Hybrid genetic algorithm in fuzzy rule-based classification systems for high-dimensional problems”, *International Journal on Applied Soft Computing*, vol. 12, no. 2, pp. 800-806, 2012.

[56] http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

Appendix A
List of Publications

[1] S. Rani and S. Jain, "Hybrid Approach to Detect Network based Intrusion", *In the Proceedings of IEEE Second International Conference on Computing Communication Control and automation (ICCUBEA)*, held at Pune, Maharashtra.

Hybrid Fuzzy-PSO Based Approach to Detect Intrusion in Networks

ORIGINALITY REPORT

14%	9%	7%	7%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to KDU College Sdn Bhd Student Paper	<1%
2	Wang, Anna, Jinbo Wang, Biao Wu, and Chenglong Shi. "Structural Optimization of the Permanent Magnet Drive Based on Artificial Neural Network and Particle Swarm Optimization", 2011 Third International Conference on Intelligent Human-Machine Systems and Cybernetics, 2011. Publication	<1%
3	esatjournals.net Int ernet Source	<1%
4	iyokan.lib.ehime-u.ac.jp Int ernet Source	<1%
5	www.nou.edu.ng Int ernet Source	<1%
6	psrcentre.org Int ernet Source	<1%
7	Submitted to Maulana Azad National Institute	