International Journal of Informatics and Communication Technology (IJ-ICT)
Vol. 5, No. 1, April 2016, pp. 11~20
ISSN: 2252-8776

Attack Detection in a Rule-Based System using Fuzzy Spiking Neural P System

FRufai Kazeem Idowu*, Ravie Chandren Muniyandi*, Zulaiha Ali Othman**
* SOFTAM Research Centret, Faculty of Information Science & Technology, Universiti Kebangsaan Malaysia
** CAIT Research Centre, Faculty of Information Science & Technology, Universiti Kebangsaan Malaysia.

Article Info

Article history:

Received Dec 15, 2015
Revised Mar 24, 2016
Accepted Mar 26, 2016

Keyword:

Fuzzy spiking neural P system
Parallel computation
Matrix-based fuzzy reasoning
membrane computing
Rule-based system

ABSTRACT

The virtual area of communication known as cyber space brought about by the debut of the internet has enabled some cyber crimes – 'Intrusion' inclusive. So, efforts are being geared towards ensuring that reliable and efficient Intrusion Detection Systems (IDSs) are developed to curtail this menace. However, Spiking Neural P (SN P) systems have been established as a class of distributed parallel computing models. So, in this paper, a novel network intrusion prediction model based on trapezoidal Fuzzy Reasoning Spiking Neural P (tFRSN P) system, is implemented for the very first time for the detection of intrusion. tFRSN P system is an extension of SN P system. It has a graphical modeling advantage which makes it well suited for fuzzy reasoning as well as fuzzy knowledge representation using If-Then rules. The dynamic firing power of neurons is harnessed in a simple parallel matrix-based fuzzy reasoning format to generate inferences. To establish the effectiveness of this approach especially in the area of speed of parallel reasoning and the handling of uncertainties, detection of Brute Force Attack (BFA) is used for demonstration. From the crisp results (0.0431, 0.0414, 0.4453, 0.1703 and 0.0414) obtained, it shows that the parallel processing capability of tFRSN P system could be used to rapidly reason and analyze the severity (possibility of an attack) from any network data.

Corresponding Author:

Rufai Kazeem I.,
SOFTAM Research Centret, Faculty of Information Science & Technology,
National University of Malaysia,
43600 UKM Bangi, Selangor, Malaysia.
Email: ruffyk2001@yahoo.com

1. INTRODUCTION

Globally, 'Intrusion' issue has become a major concern not only to the cyber security experts but to all the users of the internet. Most often, it is a haculean task to flag a user's behaviour as an attack. This is not unconnected to the fact that some degrees of uncertainties are involved. Hence, any mechanism which would venture into handling this type of security challenge must be capable of reasoning with uncertainty. Therefore, since fuzzy logic can reduce the false signal in determining intrusive activities by providing possibility instead of crispy decision, its usage with SN P which has a graphical modeling advantage, would engender quick and automatic attack detection decision.

There is no disputing the fact that Spiking Neural P (SN P) system is a well established class of Membrane Computing (MC). SN P system which is a biologically inspired distributed parallel computing device which functions by the way neurons communicate. Apart from other intrinsic advantages of SN P systems, they have been proved to be computationally complete [1], [2].

So far, some of the areas in which they have proved to be immensely beneficial include the efforts of Díaz-Pernil et al [3] in 2012. In their work, they used spiking neural P systems to solve the skeletonization problem. Based on such devices, they built a parallel software which was implemented within the Graphics

Processors Units (GPU) architecture. Furthermore, Tseren-Onolt et al [4] employed a variant of MC to specifically give a deterministic solution to each of the two well known PSPACE-complete problems: QSAT and Q3SAT. In the case of QSAT, they opined that the answer to any instance of the problem is computable in a time which is linear with respect to both the number *n* of Boolean variables and the number *m* of clauses that compose the instance. As regards Q3SAT, they postulated that the answer is computable in a time which is at most cubic in the number *n* of Boolean variables.

Most recently, precisely in 2014, Rufai et al [5] explored the parallelism advantage of MC for the feature selection in IDS. In the work, they applied MC to the Bee algorithm used for an anomaly-based IDS with a view to reducing minimally, the redundant features which adversely affect detection rate. Their approach consqently produced high detection and classification accuracy rates as well as reasonably decreased the false alarm rate. In a similar perspective, Thuzar [6] had earlier on in 2012 proposed an approach which used mutual correlation for feature election by reducing from 34 continuous attributes to 10. She subsequently used Fuzzy Decision Tree classifier for detection and diagnosis of attacks which yielded goo accuracy.

However, despite the tremendous achievements being recorded by SN P systems in different areas, a challenge of using it to handle uncertainty problems has arisen. There is therefore the constant need to extend it so as to handle emerging cases such as fuzzy problems. Hence, in this work, a trapezoidal Fuzzy Reasoning Spiking Neural P system (tFRSN P) system (as proposed by Wang et al 2013) [7] is used to represent the fuzzy production rules in a knowledge base of a rule-based intrusion detection system. With this application, the certainty factors of fuzzy production rules and the truth values of propositions are described by trapezoidal fuzzy numbers. In this work however, a trapezoidal Fuzzy Reasoning Spiking Neural P-Network Intrusion System (tFRSN P-NIDS) framework is hereby proposed. This applies a trapezoidal Fuzzy Reasoning Spiking Neural P system to detect intrusive traffic in a rule-based environment of a network detection intrusion system.

This is achieved by combining the (classical) dynamic firing mechanisms of neurons with fuzzy reasoning in a matrix-based form. By so doing, tFRSN P-NIDS would bring about much more enhanced inference ability in attack detection. This may be considered as a novel approach because going through the literature, it appears that this is the first time SN P system (an element of MC) is being applied to rule-based IDS.

The framework relies on the significant parameters of anomalous network packets, the statistics of system behavior, and the decision with threshold and fuzzy rule-based technique. With a set of fuzzy rules corresponding with the appropriate membership values, the example of Brute Force Attack (BFA) was implemented.

The rest of the paper is tructured under the following sections: Section 2 briefly discusses IDS and attack classifications. In section 3, Fuzzy Rule-base knowledge base IDS is presented with emphasis on trapezoidal fuzzy number arithmetics and generation of fuzzy production rules for network attack.While the fourth section dwells on SNP versus tFFRSNP systems, section 5 presents the proposed tFRSNP-NIDS framework. Sections 6 and 7 highlight the implementation, results and discussion. The final section draws the conclusion.

2. IDS AND ATTACK CLASSIFICATIONS

An intrusion is a security threat which is deliberately done to access and compromise the integrity and confidentiality of a resource and also to render an information system unreliable or unusable. [8] - [10]. Then, an IDS is a device which monitors the information syatem in order to check it against any potentially malicious activity and to report same to administrators for further investigation. IDSs are a critical component of any security infrastructure. Also, an Intrusion Detection System analyzes information from a computer or a network to detect malicious actions and behaviors that can compromise the security of a computer system [11], [12]. Therefore, it is a software product of hardware technology that automate a monitoring process of events which occur in a computer system or network with a view to analysing them for signs of intrusion.

In similar perspective, Debar et al [13] were of the view that an IDS is a system which dynamically monitors the action taken in a given environment, and decides whether or not these actions are symptomatic of an attack or constitute a legitimate use of the environment

Attacks are used to spread misinformation, cripple tactical services, access sensitive information, espionage, data theft and above all, financial losses [14]. Ordinarily, sets of network traffic should comprise of sets of normal traffic and four categories of attack. These categories of attack are:

1) *Denial of service* (DoS) attacks is an attack situation in which the attacker makes some computing or memory resource too busy to manage authentic requests. In other words, it is a scenario whereby an

attacker overwhelms a target machine with too much data and consequently disallowing it from executing its legitimate duties. It simply exhausts the network. Examples here include; Smurf, Teardrop, Neptune and TCP SYN flooding.

2) *User to root* (U2R) attacks: An attacker in this situation begins his dastardly act by accessing a normal user account and takes advantage of its vulnerabilities to gain unauthorized access to the root. Examples are; Buffer _overflow, loadmodule, and rootkit
3) *Remote to user* (R2L) attack occurs when an attacker who has the privilege of sending network packets to a machine, thereafter exploits the machine's vulnerabilities to gain local access. Examples are: Ftp_write, imap, multihop, phf, spy, warezclient, Brute Force Attack (BFA)
4) *Probing* (PROBE) As the name connotes, is a situation whereby an attacker examines a network for the sole aim of garnering vital information which may be used to circumvent its security controls. Examples are Satan, ipsweep, nmap, portsweep.

3. FUZZY RULE-BASE KNOWLEDGE BASE IDS

Fuzzy rules are normally created by network security experts based on their domain knowledge. In general therefore, the fuzzy rules given to the fuzzy system is done manually or by experts, who give the rules by analyzing intrusion behaviour [15]. However, the number of fuzzy rules should be reduced as much as possible. Also, the "*IF*" part of fuzzy rules should considerably be short [16], [17].

Fuzzy rules are desirable because of their interpretability by human experts. Based on the severity of an attack, fuzzy rules could be used to generate an alert which falls under either of *absolutely-false, very-low, low, medium-low, medium, medium-high, high, every-high or absolutely-high.*

3.1. Trapezoidal Fuzzy Number Arithmetic

Trapezoidal fuzzy set has been acknowledged to be highly useful because it allows full membership over any range in the universe of discourse and the range of the right and left tails can be adjusted, thus, providing great flexibility.

Trapezoid Fuzzy Number Ā, may be parameterized as a 4-tupple (*p, q, r, s*), as shown in Figure 1 below, where its membership function defined by:

$$\mu A(x) = \begin{cases} 0, x < p \\ \dfrac{x-p}{q-p}, p \le x \le q \\ \dfrac{s-x}{s-r}, r \le x \le s \\ 0, x > r \end{cases}$$

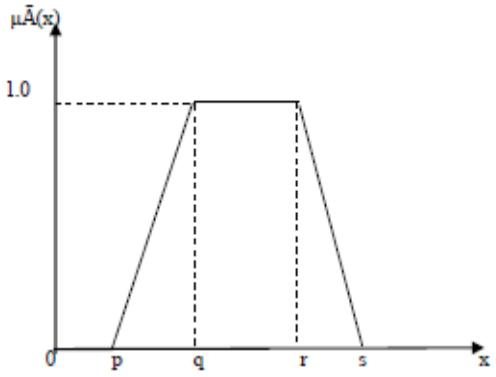

Figure 1. Graph of Trapezoidal numbers [18]

From table 1 below, for the num_failed_logins feature, the membership terms used are; Absolutely Small (AS), Very Small (VS), Small (S), Medium Small (MS), Medium(M), Medium Large (ME), Large (E), Very Large (VE), Absolutely Large (AE). However, the time interval between number of failed logins is represented with the terms; Absolutely Short (AT), Very Short (VT), Short (T), Medium Short (MT), Medium (M), Medium Long (MG), Long (G), Very Long (VG) and Absolutely Long (AG). Also, the

membership terms used for attack possibility are; Absolutely False (AF), Very Low (VL), Low (L), Medium Low (ML), Medium (M), Medium High (MH), High (H), Very High (VH) and Absolutely High (AH).
So, when applying tFSN P to attack detection, each input fuzzy term defined in the deterministic trapezoidal fuzzy system includes the following membership functions (AF, VL, L, ML, M, MH, H, VH, and AH) could be adopted.

Table 1. Numbers Defining Membership Terms

Linguistic Terms			Trapezoidal Fuzzy Numbers
No._failed_Logins	Time Interval	Attack Possibility	
AS	AT	AF	(0, 0, 0, 0)
VS	VT	VL	(0, 0, 0.02, 0.07)
S	T	L	(0.04, 0.1, 0.18, 0.23)
MS	MT	ML	(0.17, 0.22, 0.36, 0.42)
M	M	M	(0.32, 0.41, 0.58, 0.65)
ME	MG	MH	(0.58, 0.63, 0.80, 0.86)
E	G	H	(0.72, 0.78, 0.92, 0.97)
VE	VG	VH	(0.975, 0.98, 1, 1)
AE	AG	AH	(1, 1, 1, 1)

3.2. Generating Fuzzy Production Rule for Network Attack

Although, there are five basic types of fuzzy production rules, in this work however, we apply the type called composite conjunctive fuzzy production rule of the form [7], [19]:

$$R_i(c_i): p_1(\theta_1) \textcircled{\wedge} p_2(\theta_2) \textcircled{\wedge} \ldots \textcircled{\wedge} p_{k-1}(\theta_{k-1}) \rightarrow p_k(\theta_k); \quad \theta_k = (\theta_1 \wedge \theta_2 \wedge \ldots \wedge \theta_{k-1}) \textcircled{x} c_i$$

Here, R_i and c_i respectively represent the i^{th} fuzzy production rule and certainty factor. Whereas, P stands for the proposition and k for its number in a rule-based environment, θ is the truth value for the i^{th} proposition. However, the symbol " $\bigwedge$" represents the *AND* operator of trapezoidal fuzzy number in which " $\bigwedge$ " performs minimization operation.

A fuzzy set is a set which is defined by a membership function. A membership function assigns to each element in the set under consideration (the universal space) a membership grade, which is a value in the interval [0, 1]. Fuzzy "*if-then*" rules are often employed to capture the imprecise modes of reasoning which play an essential role in the human ability to make decisions in uncertain and imprecise environments.

The following rules which are formulated for Brute Force Attack (BFA) were done by adopting the simple fuzzy rules theory. BFA is a situation where an intruder tries to login with several users' passwords and fails. This attack can be identified by observing the number of login failures and the time interval between each failure. [20]

Rule 1: (CF = VH)
Symptom
(i) Set of num_failed_logins is VS
(ii) Time interval is VG
Probable Attack
Brute Force attack not suspected
Rule 2: (CF = H)
Symptom
(i) Set of num_failed_logins is VS
(ii) Time interval is T
Probable Attack
General failed login attempts
Rule 3: (CF = H)
Symptom
(i) Set of num_failed_logins is M
(ii) Time interval is M
Probable Attack
May and may not be BFA
Rule 4: (CF = H)

Symptom
(i) Set of num_failed_logins is VE
(ii) Time interval is T
Probable Attack
Serious Brute Force attack
Rule 5: (CF = H)
Symptom
(i) Set of num_failed_logins is VE
(ii) Time interval is VT
Probable Attack
Very severe Brute Force attack

4. SN P SYSTEM VERSUS tFRSN P

Efforts have been put up to extend the basic Spiking Neural P system like that of Wang H et al [7] where trapezoidal Fuzzy Reasoning Spiking Neural P system (tFRSN P) system was promulgated.

4.1. Basic SN P System

SN P system is class of distributed and parallel computing model which is inspired by the neurophysiological behaviour of neurons sending electrical impulses (spikes) to other *neurons*. The set of *neurons* are placed in the nodes of a graph which facilitate the movement of the spikes along the synapses (edges of the graph), under the control of firing rules. For the main purpose of communication, these *neurons* are connected to each other in an intricate pattern. They have three functionally distinct parts called ***dendrites, soma*** and ***axon***. Hence, when they interact, there is an exchange of spikes. In doing this though, pre-synaptic neuron is configured to have a kind of 'handshake' with the post-synaptic neuron at a junction known as ***synapse*** by means of specific rules.

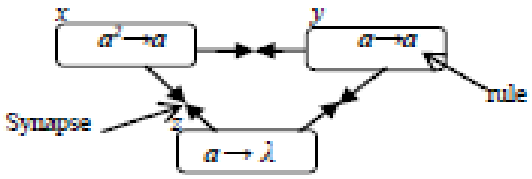

Figure 2. Schematic representation of how Neurons communicate [21]

Figure 2 above depicts a simple schematic representation of an SN P system with three neurons *x* , *y* and *z*. The spike, denoted as "*a"* which is the basic unit of information is stored in the neuron. While neuron *x* has rule $a^2 \rightarrow a$, *y* has rule $a \rightarrow a$ and *z* has rule $a \rightarrow \lambda$. The synapse is also captured.

Furthermore, when the rules (which may be used concurrently) are applied, the system is transformed. By assuming the presence of a global clock, the system is synchronized. Atimes, the cell sending out spikes is "closed" during a refractory period of a neuron. At this point, the neuron does not only closes to the acceptance of input, it also cannot fire spike again. Depending on the exact formalisation of the model, the notion of a successful computation is defined together with its output [22].

Simply put, Spiking Neural P system is a non deterministic class of membrane computing systems which is similar to other P system variants such as Tissue-like and Cell-like.

In general, an SN P system of degree $m \geq 1$ is a construct of the form:

$\prod = (O, \sigma_1 \ldots\ldots \sigma_m \text{ syn, out})$,

Where:

1) $O = \{a\}$ is the singleton alphabet. (a is called spike);
2) $\sigma_1, \ldots\ldots, \sigma_m$ are neurons, of the form $\sigma_1 = (n_i, R_i)$, $1 \leq i \leq m$, where:
   * $ni \geq 0$ is the initial number of spikes contained by the neuron;
   * R_i is a finite set of rules of the following two forms:
     a) $E/a^c \rightarrow a$;d, where E is a regular expression over *O, c ≥ 1, and d ≥ 0;*

b) $a^s \rightarrow \lambda$, for some s ≥1, with the restriction that as L∉ for no rule E/a^c a; d of type (1) from R_i;

3) syn ⊆ { 1, 2, ..., m } x { 1, 2, ..., *m* }with (*i, i*) ∈ syn, for 1 ≤ *i* ≤ m (synapses);

4) out ∉ { 1, 2, ..., m} indicates the output neuron

The rules of type (1) are called *spiking rules*, which is written in a shorthand notation as $a^c \rightarrow a^b$. The rules of type (2) are called *forgetting rules*. The application of the rules depends on the contents of the neuron. This implies that the applicability of a rule is established based on the total number of spikes contained in the neuron. If no firing rule can be applied in a neuron, there may be the possibility to apply a forgetting rule, which removes from the neuron a predefined number of spikes.

4.2. Trapezoidal Fuzzy Reasoning Spiking Neural P System

Normally, when the antecedent (condition) part of a rule is satisfied, the right hand side which is called the consequent is triggered/activated. Fuzzy Reasoning therefore, is the process of firing and execution of the fuzzy rule.

A tFRSN P system of degree m ≥ 1 [23], is a construct of the form

$\prod$ = (*O* , $\sigma_1, \ldots, \sigma_m$, s*yn, in out*)

where:

1) *O* = {*a*}is the singleton alphabet (*a* is called spike);
2) $\sigma_1, \ldots, \sigma_m$ are neurons of the same form
   ι. $\sigma_i = (\theta, c_i, r_i), 1 \leq i \leq m,$

   where:
   a) θ_i is the potential value of spikes (i.e. pulse value) contained in neuron σ_i , and it is expressed by a trapezoidal fuzzy number in [0,1];
   b) c_i can be understood as either the fuzzy truth value of a proposition (when σ_i corresponds to a proposition neuron) or the certainty factor of a production rule (when σ_i corresponds to a rule neuron), and it is expressed by a trapezoidal fuzzy number in [0,1];
   c) r_i represents a firing (spiking) rule contained in neuron σ_i with the form $E / a^{\theta} \rightarrow a^{\beta}$, where *E* ($E = a^n$) is the firing condition, and *n* is the number of presynaptic neurons connected to neuron σ_i which is expressed by an integer, θ and β are expressed by trapezoidal fuzzy numbers in [0, 1].
3) syn ⊆ {1,2,...,m}×{1,2,...,m} , with $i \neq j$ for all (*i, j*) ∈ *syn* ,1 ≤ *i*, *j* ≤ *m*, is a directed graph of synapses between the linked neurons;
4) *in, out* indicate the input neuron set and the output neuron set of $\prod$, respectively.

Suffice to note that when tFRSN P system is implemetend in a matrix reasoning format as it relates to the composite conjunctive fuzzy production rule being applied here, arithmetic multiplication ⓧ operator, matrices $\boldsymbol{\theta}^*$ and $\boldsymbol{\delta}^*$ need to be defined thus;

Definition 4.2.1: Given trapezoidal numbers

$\overline{P} = (p_1, p_2, p_3, p_4)$ and $\overline{Q} = (q_1, q_2, q_3, q_4)$,

$\overline{P}$ ⓧ $\overline{Q}$ = (p_1, p_2, p_3, p_4) ⓧ (q_1, q_2, q_3, q_4) = ($p_1 \times q_1$, $p_2 \times q_2$, $p_3 \times q_3$, $p_4 \times q_4$)

Definition 4.2.2: $\boldsymbol{\theta}^*$ is an *nx1* matrix containing the truth values of the proposition neurons expressed by trapezoidal fuzzy number in [0,1].

Definition 4.2.3: $\boldsymbol{\delta}^*$ is a *mx1* matrix containing the truth values of the rule neurons expressed by trapezoidal fuzzy number in [0,1]

5. THE PROPOSED tFRSN P – NIDS FRAMEWORK

This section discusses the architecture of the proposed framework for tFRSN P-NIDS.

The framework uses fuzzy rule-based system to detect intrusive traffics and to send signal to or alert the System Administrator (SA) about these attacks.

In the framework (fig. 3) below, the tFRSN P acts as the coordinating point of the fuzzified network traffic and the well-defined rules coming from the rule base. Infact, it is considered as the engine room because it is where decisions are taken. After performing fuzzy reasoning on it, tFRSN P system releases defuzzified information (that is, the detection results) to the outside world through the user interface. The

membership function is defined based on fuzzy logic of trapezoidal linguistic terms which falls within the trapezoidal number [0,1].

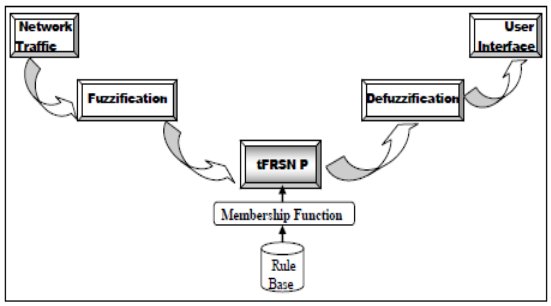

Figure 3. tFRSN P-IDS Framework

6. IMPLEMENTING tFRSN P IN ATTACK DETECTION

The knowledge base shown above can be modeled using tFRSN P system as captured in the Figure 4 below. The model contains 12 proposition neurons and 5 rule neurons. In the model, the initial trapezoidal linguistic truth values of inputs neurons σ_1 , σ_2 , σ_3 , σ_4, σ_5, σ_6 and σ_7 are "VG", "T", "M", "VT", "VS", "M" and "VE" respectively.

In summary therefore, the fuzzy production rules are defined as a construct:

$\prod = (O, \sigma_1, \ldots, \sigma_{12}, \sigma_{13}, \ldots, \sigma_{17}, syn, in, out)$

Where

(1) $O = \{a\}$
(2) $\sigma_1, \ldots, \sigma_{12}$ are proposition neurons having fuzzy truth values $p_1, \ldots, p_{12}$ respectively.
(3) $\sigma_{13}, \ldots, \sigma_{17}$ are "AND" –type rule neurons associated with production rules R_1 , ..., R_5 respectively.
(4) syn = {(1,13), (2,14), (2,16), (3,15), (4,17), (5,13), (5,14), (6,15), (7,16), (7,17), (13,8), (14,9), (15,10), (16,11), (17,12)}
(5) in = $\{\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5, \sigma_6, \sigma_7\}$
out =$\{\sigma_8, \sigma_9, \sigma_{10}, \sigma_{11}\sigma_{12}\}$

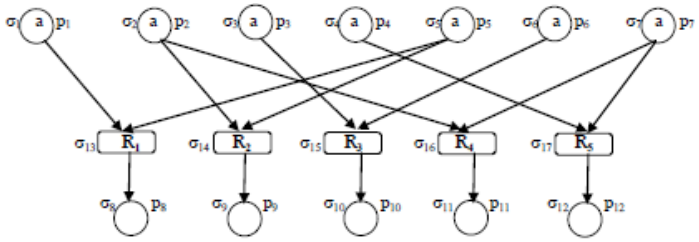

Figure 4. tFRSN P system Model for BFA

At the initial instance, i.e when $t = 0$, $\boldsymbol{\theta_0}$ and $\boldsymbol{\delta_0}$ which represent the initial values of proposition neurons and rule neurons respectively, are given in the following matrices:

$$\boldsymbol{\theta_0} = \begin{vmatrix} 0.975, 0.98, 1, 1 \\ 0.04, 0.1, 0.18, 0.23 \\ 0.32, 0.41, 0.58, 0.65 \\ 0, 0, 0.02, 0.07 \\ 0, 0, 0.02, 0.07 \\ 0.32, 0.41, 0.58, 0.65 \\ 0.975, 0.98, 1, 1 \\ \mathbf{0} \end{vmatrix}_{12x1} \qquad \boldsymbol{\delta_0} = [\mathbf{0}]_{5x1}$$

At step *t = 1*, after performing the operation Ⓐ within the five rule neurons and subsequently multiplying Ⓧ with their corresponding certainty factors (CF), we obtain the results:

$$\boldsymbol{\delta_1} = \begin{vmatrix} 0, 0, 0.02, 0.07 \\ 0, 0, 0.02, 0.07 \\ 0.32, 0.41, 0.58, 0.65 \\ 0.04, 0.1, 0.18, 0.23 \\ 0, 0, 0.02, 0.07 \end{vmatrix}_{5x1}$$

$$\boldsymbol{\theta_1} = \begin{vmatrix} \mathbf{0} \\ 0, 0, 0.02, 0.07 \\ 0, 0, 0.0184, 0.0679 \\ 0.2304, 0.3198, 0.5336, 0.6305 \\ 0.0288, 0.078, 0.1656, 0.2231 \\ 0, 0, 0.0184, 0.0679 \end{vmatrix}_{12x1}$$

At step t = 2, the reasoning process ends hence we obtain the results:

$$\boldsymbol{\delta_2} = \mathbf{[0]}_{5x1}$$

7. RESULT AND DISCUSSION

The computation halts since there are five reasoning results: (0, 0, 0.02, 0.07), (0, 0, 0.0184, 0.0679), (0.234, 0.3198, 0.5336, 0.6305), (0.0288, 0.078, 0.1656, 0.2231), and (0, 0, 0.184, 0.0679) corresponding to the five output neurons σ_8, σ_9, σ_{10}, σ_{11} and σ_{12}. Also, there are no further rules to be executed and is called stopping criteria (i.e $\boldsymbol{\delta_2} = \mathbf{0, 0, 0, 0)}$, which is an absolutely false condition [24]. These results express the confidence levels at which Brute Force could occur in a typical network environment.

Thereafter, the above is defuzzified. Defuzzification is a process which converts a fuzzy set or fuzzy number into a crisp value. Defuzzification is used in fuzzy modeling simply for the purpose of converting fuzzy outputs from the systems to crisp values (which are quantified by real-valued functions).

The computed defuzzified results then help to determine the severity of the attack.

$$\text{Crisp number (Cn)} = \frac{(s-e) + (r-e)}{((s-e) + (r-e)) - ((p-f) + (q-f))} \quad (3)$$

As adapted from [24], *e* and *f* are 0, 1 respectively being the two extreme values of the entire fuzzy set range. Also, (as shown in fig. 1 above) while *p* and *s* are the left and right width of the trapezoidal range, q and r stand for the interval at which the membership is 1. Hence, we obtained 0.0431 (4.31%), 0.0414 (4.14%), 0.4453 (44.53%), 0.1703 (17.03%) and 0.0414 (4.14%) respectivel.

Therefore, since the severity of none of these values is up to 0.5 (50%), we then conclude that BFA does not portend any danger or appears to be a threat in this scenario.

8. CONCLUSION

For the very first time, Spiking Neural P (SN P) system in conjunction with trapezoidal fuzzy logic system has successfully been applied to a rule-based Intrusion Detection system. It was able to flag the level at which the severity of the attack could or otherwise serve as a threat to the information system.

So, we have applied tFRSN P system to attack detection and have used it to model the knowledge base of a type of attack called BFA. Implementing this detection in a matrix format by incorporating the parallelism advantage of SN P system makes it to be very intuitive, simple and above all, fast. Furthermore, since this work has only been implemented for BFA, future works may be extended to include the application of tFRSN P system to other classes of attack such as Denial of Service (DoS).

REFERENCES

[1] A. Păun, Gh. Păun, "Small Universal Spiking Neural P Systems", Journal of Biosystems, Elsevier, Vol.90, (2007) Pp.48-60.

[2] M. Ionescu, Gh. Păun, T. Yokomori. "Spiking Neural P systems". Fundamenta Informaticae 71(2–3): pp. 279–308, 2006.

[3] D. Díaz-Pernil, P. Francisco, A. G. Migue. "A Parallel algorithm for skeletonizing images by using spiking neural P systems". Neurocomputing Vol. 115 (2013) Pp. 81–91

[4] I. Tseren-Onolt, A. Leporati, L. Pan, X. Zeng, X. Zhang. Deterministic solutions to QSAT and Q3SAT by Spiking Neural P systems with Pre-computed Resources. Theoretical Computer Science 411 (2010) 2345-2358

[5] K. I. Rufai, C. M. Ravie and Z. A. Othman. Improving Bee Algorithm Based Feature Selection in Intrusion Detection System Using Membrane Computing Journal of Networks, Vol 9, No 3 (2014), 523-529,

[6] H. Thuzar: "Feature Selection and Fuzzy Decision Tree for Network Intrusion Detection". International Journal of Informatics and Communication Technology (IJ-ICT) Vol.1, No.2, December 2012, pp. 109~118

[7] T. Wang, G. Zhang. "Application of Fuzzy Reasoning Spiking Neural P System to Fault Diagnosis". (2013).Asian Conference on Membrane Computing.

[8] M. S. Abadeh, H. Mohamadi , J. Habibi. Design and Analysis of Genetic Fuzzy Systems for Intrusion Detection in Computer Networks. Expert Systems with Applications 38 (2011) 7067–7075

[9] H. T. Elshoush, I. M. Osman. Alert Correlation in Collaborative Intelligent Intrusion Detection Systems — A Survey. Applied Soft Computing 11 (2011) 4349–4365

[10] A.N. Toosi, M. Kahani, A new Approach to Intrusion Detection Based on an Evolutionary Soft Computing Model using Neuro-Fuzzy Classifiers, Computer Communications 30 (2007) Pp 2201–2212.

[11] R. G. Bace, Intrusion Detection: Defining Intrusion Detection. Macmillan Technical Publishing, 2000

[12] A. Einipour. Intelligent Intrusion Detection in Computer Networks Using Fuzzy Systems. Global Journal of Computer Science and Technology Neural & Artificial Intelligence Volume 12 Issue 11 Version 1.0 (2012)

[13] H. Debar, M. Dacier and A. Wespi, "Towards a Taxonomy of Intrusion-Detection Systems" Computer Networks, 31 (8), pp. 805–822, 1999.

[14] M. Uma, G. Padmavathi. "A Survey on Various Cyber Attacks and their Classification" International Journal of Network Security, Vol.15, No.6, PP.391-397, Nov. 2013 391

[15] B. Shanmugam N. B. Idris. Improved Intrusion Detection System Using Fuzzy Logic for Detecting Anamoly and Misuse type of Attacks. International Conference of Soft Computing and Pattern Recognition (2009)

[16] O. Cordon, F. Gomide, F. Herrera, F. Hoffmann, L. Magdalena. "Ten years of genetic fuzzy systems: Current Framework and New Trends", Fuzzy Sets and Systems, vol.141, no.1, (2004) pp. 5–31.

[17] M. Saniee Abadeh, J. Habib and C. Lucas. "Intrusion detection using a fuzzy genetics-based learning algorithm", Journal of Network and Computer Applications, vol.30, no.1,,(2007) pp. 414–428

[18] W. H. Chen. Fault Section Estimation Using Matrix-based Reasoning Methods. IEEE Transactions on Power Delivery, 26(1), (2011), 205 - 213

[19] H. Peng, J. Wang, M. J. Perez-Jimenez, H. Wang, J. Shao, T. Wang. "Fuzzy Reasoning Spiking Neural P System for Fault Diagnosis". Information Sciences, 235 (2013) 106

[20] S. Sangeetha, S. Haripriya, S.G. Mohana Priya, V. Vaidehi,.N. Srinivasan. Fuzzy Rule-Base Based Intrusion Detection System on Application Layer CNSA 2010, CCIS 89, (2010) pp. 27–36.

[21] K. I. Rufai, C. M. Ravie and Z. A. Othman. The Prospects of Using Spiking Neural P System for Intrusion Detection : International Journal of Information & Network Security (IJINS) Vol.2, No.6 (2013), pp. 492~498 , ISSN: 2089-3299

[22] P. M. Venkata, et al., "Protocol Modeling in Spiking Neural P systems and Petri nets" International Journal of Computer Applications. Volume 1 – No. 24 , 2010

[23] Tao Wang, J. Wang, H. Peng, H. Wang. Knowledge Representation and Reasoning Based on FRSN P System (Proceedings of the 8th World Congress on Intelligent Control and Automation June 21-25 (2011), Taipei, Taiwan) 978-1-61284-700-9/11/$26.00 ©2011 IEEE

[24] G. Xiong, D. Shi, L. Zhu, X. Duan . A New Approach to Fault Diagnosis of Power Systems Using Fuzzy Reasoning Spiking Neural P Systems. Mathematical Problems in Engineering Volume 2013 (2013), http://dx.doi.org/10.1155/ 2013/

BIBLIOGRAPHY OF AUTHORS

Rufai I. Kazeem is a Nigerian academics who is currently pursuing a PhD degree at the National University of Malaysia. His research interest is in optimizing Intrusion Detection System using Membrane Computing paradigm.
Rufai has been involved in teaching and research for over a decade.
He is a member of the Computer Professional Registration Council of Nigeria(CPRN) which is the highest regulatory body in ICT in Nigeria.

Chandren M. Ravie (PhD) is a senior research fellow of the Faculty of Technology and Information Science. He is presently working on Information Retrieval, Programming and Membrane Computing.
He has published several articles in international journals and presented papers both at national and international conferences.

Zulaiha Ali Othman (PhD) is an Associate Professor. She bagged her Master and PhD degrees from the prestigious University of Sheffield, United Kingdom.
She specializes in Data Mining and Optimization. Zulaiha is a prolific writer of repute