Detection of Rogue Access Point in WLAN using Hopfield Neural Network

Menal Dahiya[1], Sumeet Gill[2]
[1]Department of Computer Science, Maharaja Surajmal Institute, C-4, Janakpuri, Delhi, 110058, India
[2]Department of Mathematics, Maharshi Dayanand University, Rohtak, Haryana 124001, India

Article Info

Article history:

Received Sep 29, 2016
Revised Nov 14, 2016
Accepted Jan 6, 2017

Keyword:

Hopfield algorithm
Neural network
Rogue access point
Wireless communication
WLAN

ABSTRACT

The serious issue in the field of wireless communication is the security and how an organization implements the steps against security breach. The major attack on any organization is Man in the Middle attack which is difficult to manage. This attack leads to number of unauthorized access points, called rogue access points which are not detected easily. In this paper, we proposed a Hopfield Neural Network approach for an automatic detection of these rogue access points in wireless networking. Here, we store the passwords of the authentic devices in the weight matrix format and match the patterns at the time of login. Simulation experiment shows that this method is more secure than the traditional one in WLAN.

Corresponding Author:

Menal Dahiya,
Department of Computer Science,
Maharaja Surajmal Institute,
C-4, Janakpuri, Delhi, 110058, India.
Email: menaldahiya@gmail.com

1. INTRODUCTION

Development in science increases the usage of technology day by day, especially in the area of wireless communication. Widespread use of the internet and the advancement of technology leads to the need of better security in wireless communication. They provide flexible workforce, mobility, freedom of access and other special characteristics that can achieved with suitable security and proper performance requirements [1]. Therefore, taking care of security is much more difficult in wireless networking as compared to the wired network. Wireless communication has gained popularity due to its ease of use, speed, flexibility, etc. So it is more important in case of wireless networking to provide authentic access control to the users and protect sensitive information against possible violations [2]. Security has become one of the vital aspect of any organization on which an organization builds, whether it is a security of a particular computer system or a security of the whole network. The rapid development in wireless communication is a big issue of security. As data is transferred over the waves every time, anybody can use it if he/she is in the range of wireless signals. Many of us use the protocols WPA and WEP in systems to provide encryption and decryption to the messages and protect unauthorized access of the resources [3]. Although they provide encryption and good authentication mechanisms to protect user and data eavesdropping, but not in a solid way. Even these protocols could not protect the organization from the unauthorized access by their own staff.

Wireless Network works in two modes of operation ad hoc and infrastructure mode. In ad hoc networking, wireless nodes communicate directly while in the infrastructure type of mode wireless nodes are communicating through a central point called a wireless access point (WAP) [4]. Maintaining security to both the modes is a very tedious task, as various attacks become possible in wireless networking. The most common attack is a Man-in-the-Middle-attack, where an intruder placed himself between sender and receiver are in the network [5]. In wired communication detection of this type of attack is easy while in the case of a wireless communication attacker easily access the information by making fake access points. These access points can be easily bought as they are inexpensive. These unauthorized access points are called rogue access points [6]. Any unsanctioned, unauthorized wireless point that connects to an enterprise's authorized network is defined as a rogue access point. It is one of the greatest risks to an organization's security. These access points work as an open network system for the outsiders which are ready to take out information from the enterprise's traffic.

We explore the use Hopfield neural network method of Artificial Neural Network for detecting the presence of rogue access points in the network. Hopfield network is trained with known patterns and recall that pattern at the time of execution. Section II focuses on different types of rogue access point present in any organization and their methods of detection. Section III describes the work done by different researchers and industry solutions related to this problem. Section IV explains Hopfield Neural Network and then the simulation part for detection of rogue access points using the neural network tool.

2. THE ROGUE ACCESS POINT

Rogue access point is a point connected to the network without the permission of the administrator. Therefore, it is called an unauthorized point which accesses the information. The unauthorized access point is divided into two categories: – 2.1) Rogue access point, 2.2) Fake access point.

2.1. Rogue access point

Rogue access point is an access point which is installed by not only the outsider, but also by the authorized user to take benefit of the network. There are four common types of the rogue access point:

2.1.1. Employee Rogue Access Point

This type of access point occurred when employee buy an access point and installs it on the company's LAN for its own benefit without permission. It also opens the way for outside attackers to access the network. This type of incident takes place where there is a lack of wireless security policies and lack of awareness in employees.

2.1.2. Attacker's external rogue access point

This type of access point is setup outside the organization by the attacker and does not connect to the network. It aims to allow the target employee to connect to a rogue access point. All user traffic is redirected through this rogue access point and attacker analyzes it. This attack is also called Man-in-the Middle Attack.

2.1.3. Attacker's internal rogue access point

In contrast, with above access point this access point is set up inside the organization by the attacker and does not connect to the network. But an attacker uses this rogue access point at a later time to access the internal LAN. Once it is successful, it would be a serious security breach.

2.1.4. Neighborhood rogue access point

As the name suggests, the access point is set-up by another company in the close vicinity.

2.2. Fake access point

Fake access point is an access point that is inserted by an outsider or an attacker without the permission of the authorized user of the network [7]. He used this access point for stealing information and for other false purposes. All these types of rogue access point are prevented by the two processes, first is rogue access point detection to identify the rogue access point and the second one has taken security measures to disable the rogue access point.

Most enterprises use WPA2 security for wireless communication. But even WPA2 also cannot protect from rogue access point, we can incorporate WPA2 only on managed access point, but the rogue access point is an unauthorized access point. So, we cannot enforce security control over it. Although detecting rogue access points is a challenging process. There are many existing techniques for detection of rogue access point [8],[9]. These techniques divided into following categories like traditional approach, client side approach, server side approach and hybrid approaches.

2.3. Traditional Approach

Traditional approach based on matching concept. It verifies the MAC address and SSID. If all the attributes are same then it is an authorized access point. Traditional approach isn't effective for authentication of authorized access points as the number of tools being available in the market that spoof MAC address and logical address. So, this approach is not sufficient.

2.4. Server side Approach

It is the central controller of the wired and wireless network. Software tools are installed on a centralized server and they analyze the whole network and detect rogue access point by performing some operations and if found something wrong, it checks the status of that particular access point.

2.5. Client Side Approach

This is a challenging process as there is former information about networks and about the access point list [10]. This approach takes services from server sides for detection of rogue access points or they also install software on their device of access point whether it is authorized or not.

3. LITERATURE SURVEY

Along with academic solutions there are some industrial solutions also exist which are as follows: Air Defense contains sensors deploy throughout the network [11]. Here, network manager handles the software tool, which detects attacker and attacks in the network. The only problem with this tool is that its response time is slow but it is a commercial product and easily available.

Air Magnet is also a commercial product and helps in detecting an unauthorized access point and denial of service attack by flooding [12]. This also requires a technical manager for detection. Jana et al.proposed server side solution using clock skews of the access point [13]. This approach is unable to detect MAC spoofing and has a lack of accuracy and speed in the calculation of clock skews. Kindberg et al. Proposed a model for security for public Wi-Fi network [14]. It uses encryption and authentication techniques to authenticate the access point in the wireless network.

Shivraj et al. Present server side Hidden Markov Model (HMM) based approach to detect rogue access point [15]. This technique achieves more than 80% accuracy and uses a variation in packet inter arrival time to detect authorized and unauthorized access points. This approach is also in denial of service attack. Kim et al. Proposed client side approach for detecting fake access point using the idea of received signal strength (RSS). They collect the signals, normalize it and then apply sequential hypothesis technique. They never consider the distance, as it affects the signal strength.

Kao et al. Proposed client side approach for detecting rogue access point [16]. It uses a passive packet analysis approach based on bandwidth estimation. Liran Ma et al. Proposed hybrid approach for detecting the rogue access points [17]. It is a cost effective solution and also the proposed model uses a traditional approach for detecting fake access points. Here, in this paper, we propose the automatic detection of rogue access points by using the Hopfield neural network approach. For accessing any service or resource of the network, the device must first communicate with another device, i.e. pair with each other for communication. A four way handshake protocol is involved in the set up of WLAN [18-21]. The main purpose of this process is to enable an access point to authenticate itself to the client and then use the user's login and password. In conventional method passwords are easily tracked by the intruders, but in this mechanism the tracking/cracking of password is impossible.

4. HOPFIELD NEURAL NETWORK

In the neural networks, when we use feed forward flow of information, i.e. one output vector is associated to every input vector is called a Feed Forward Network. But sometimes, it is possible where output value can return back to the input repeatedly, then; these types of situations come under feedback networks. John Hopfield proposed this type of concept of neural network in his paper which was published in 1982 [22]. Hopfield works on auto associative non-linear properties of the network. It is a fully connected or recurrent type of network in which each neuron is linked with each other but not with it. Neural networks are complex and non-linear in nature, so analyzing their behavior is difficult. Hopfield used non-linear dynamical system theory on neural networks. In the network architecture, he embedded the physical principle and set up an energy function. Hopfield Neural Network basically uses the concept of content addressable memory [23]. The network develops a number of stable points in state space and the other points in the state space move into the direction of stable points, here known as attractors which are energy minima. Attractors can also be applied when we reconstruct the missing information and often called associative memory on the Hopfield Neural Network. Operation of auto associative memory is rooted from reconstruction property in the sense that the new input states can be linked to the appropriate patterns already stored in the memory. The Hopfield Neural Network is a single layer, non-linear and constant addressable memory network and of two types:

discrete and continuous. The Hopfield Neural Network model consists of two layers, one is input and other is output, where each unit is connected to every other unit in the network other than itself [24], [25]. HPNN consists of neurons and each neuron can be in one of two states, i.e. $\underline{+}1$ and p bipolar patterns:

$$X^u = (x_1^u, x_2^u, \ldots\ldots, x_n^u)$$

The connection of weight matrix is square and symmetric i.e. all the diagonal elements of Hopfield network are zero:

$$W_{ij} = W_{ji} \text{ and } W_{ii} = 0$$

Hopfield specified the W_{ij} ·s by using the Hebbian rule [26] i.e

$$W_{ij} = \sum_{i=0}^{n} x_i^u \ x_j^u \quad (i \neq j)$$

Hopfield associated with a function known as an energy function or Lyapunov function. This function proves that the net will converge to a stable limit point. The energy function is given by:

$$E = -0.5 \sum_i \sum i \neq j \ w_{ij} y_i y_j - \sum x_i \ y_i + \sum i \ \theta_i$$

So for storing a pattern, the energy function should be minimized. Figure 1 shows a Hopfield model with eight nodes, where each node is connected to every other node in the network.

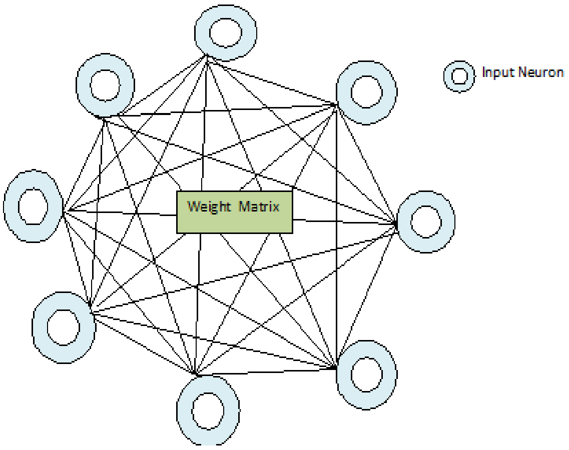

Figure1. Hopfield Neural Network

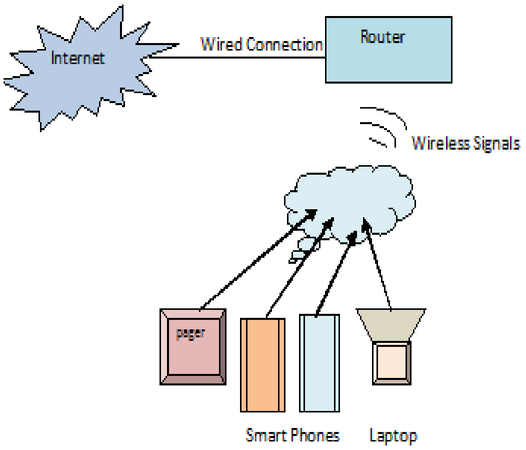

Figure 2. Devices paired with the router

5. SIMULATION DESIGN

Figure 2, depicts the connection of different devices to the router used for simulation implementation. In this segment we take eight entities, smart phones, laptops and other wireless devices that are connected to the Wi-Fi router. The passwords for accessing the devices are stored in the router and also known to the authentic user. Memory of router stores the different trained patterns of the passwords that are used by the organization staff and other authentic employees. The passwords are first normalized and then convert into bipolar, as Hopfield network take bipolar inputs. We train our network for a particular pattern and stored in the memory. By looking at network parameters, an attacker did not find out the design of the network. In this section, we are conducting the simulation of real test data [27]. Here we, memorize the input pattern sets by minimizing the error between the desired and actual output for 8 sets having 40 bits each.

Input data set

H = [0	0	1	1	0	0	0	1
0	0	0	0	1	1	1	1
0	0	0	0	1	1	0	1
1	0	1	0	1	1	1	1
0	1	1	1	1	0	1	0
0	0	0	1	0	1	1	0

```
     0      0      1      0      0      1      0      0
     1      1      0      1      0      1      1      0
     0      0      1      0      1      1      1      1
     1      1      0      1      0      0      1      0
     0      0      0      0      1      0      0      0
     1      0      0      0      0      0      0      0
     0      0      0      1      0      1      0      0
     0      0      1      0      0      1      0      0
     0      0      1      0      0      1      0      0
     1      1      1      0      0      0      1      1
     0      0      1      1      1      0      0      1
     1      0      0      1      1      1      1      0
     0      1      1      0      0      0      0      1
     1      0      0      0      1      0      1      1
     0      0      1      1      0      0      0      0
     0      1      0      0      1      0      1      0
     0      0      0      1      0      1      1      1
     1      1      0      0      1      1      1      1
     0      0      1      0      0      0      1      0
     0      1      0      1      1      1      0      1
     0      1      1      1      0      0      0      1
     0      0      0      0      1      1      1      0
     0      0      0      1      0      0      1      0
     1      1      1      0      1      1      0      1
     0      0      1      1      0      1      0      0
     1      1      0      0      1      1      0      1
     0      0      0      1      0      1      1      0
     1      0      1      0      0      0      1      0
     0      0      0      1      0      0      0      0
     0      0      1      0      0      0      1      1
     0      1      0      0      0      0      0      1
     0      0      1      1      0      1      1      0
     0      0      0      0      1      1      0      1
     1      1      1      0      1      1      1      0]
```

Output data set

```
 T = [0     0      1      1      0      0      0      1
     0      0      0      0      1      1      1      1
     0      0      0      0      1      1      0      1
     1      0      1      0      1      1      1      1
     0      1      1      1      1      0      1      0
     0      0      0      1      0      1      1      0
     0      0      1      0      0      1      0      0
     1      1      0      1      0      1      1      0
     0      0      1      0      1      1      1      1
     1      1      0      1      0      0      1      0
     0      0      0      0      1      0      0      0
     1      0      0      0      0      0      0      0
     0      0      0      1      0      1      0      0
     0      0      1      0      0      1      0      0
     0      0      1      0      0      1      0      0
     1      1      1      0      0      0      1      1
     0      0      1      1      1      0      0      1
     1      0      0      1      1      1      1      0
     0      1      1      0      0      0      0      1
     1      0      0      0      1      0      1      1
     0      0      1      1      0      0      0      0
     0      1      0      0      1      0      1      0
     0      0      0      1      0      1      1      1
     1      1      0      0      1      1      1      1
     0      0      1      0      0      0      1      0
     0      1      0      1      1      1      0      1
     0      1      1      1      0      0      0      1
     0      0      0      0      1      1      1      0
     0      0      0      1      0      0      1      0
     1      1      1      0      1      1      0      1
     0      0      1      1      0      1      0      0
     1      1      0      0      1      1      0      1
     0      0      0      1      0      1      1      0
     1      0      1      0      0      0      1      0
     0      0      0      1      0      0      0      0
     0      0      1      0      0      0      1      1
     0      1      0      0      0      0      0      1
```

0	0	1	1	0	1	1	0
0	0	0	0	1	1	0	1
1	1	1	0	1	1	1	0]

After converting inputs into bipolar

[-1	-1	1	1	-1	-1	-1	1
-1	-1	-1	-1	1	1	1	1
-1	-1	-1	-1	1	1	-1	1
1	-1	1	-1	1	1	1	1
-1	1	1	1	1	-1	1	-1
-1	-1	-1	1	-1	1	1	-1
-1	-1	1	-1	-1	1	-1	-1
1	1	-1	1	-1	1	1	-1
-1	-1	1	-1	1	1	1	1
1	1	-1	1	-1	-1	1	-1
-1	-1	-1	-1	1	-1	-1	-1
1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	1	-1	1	-1	-1
-1	-1	1	-1	-1	1	-1	-1
-1	-1	1	-1	-1	1	-1	-1
1	1	1	-1	-1	-1	1	1
-1	-1	1	1	1	-1	-1	1
1	-1	-1	1	1	1	1	-1
-1	1	1	-1	-1	-1	-1	1
1	-1	-1	-1	1	-1	1	1
-1	-1	1	1	-1	-1	-1	-1
-1	1	-1	-1	1	-1	1	-1
-1	-1	-1	1	-1	1	1	1
1	1	-1	-1	1	1	1	1
-1	-1	1	-1	-1	-1	1	-1
-1	1	-1	1	1	1	-1	1
-1	1	1	1	-1	-1	-1	1
-1	-1	-1	-1	1	1	1	-1
-1	-1	-1	1	-1	-1	1	-1
1	1	1	-1	1	1	-1	1
-1	-1	1	1	-1	1	-1	-1
1	1	-1	-1	1	1	-1	1
-1	-1	-1	1	-1	1	1	-1
1	-1	1	-1	-1	-1	1	-1
-1	-1	-1	1	-1	-1	-1	-1
-1	-1	1	-1	-1	-1	1	1
-1	1	-1	-1	-1	-1	-1	1
-1	-1	1	1	-1	1	1	-1
-1	-1	-1	-1	1	1	-1	1
1	1	1	-1	1	1	1	-1]

5.1. Training Using Neural Networks

Weights of layer1 from input 1

0.3927	-0.0182	0.0106	0.0119	-0.0335	-0.0151	-0.0122	-0.0869	0.0138	-0.0426
-0.0182	0.3964	0.1059	0.0663	-0.0220	0.0590	-0.0113	-0.0365	0.1329	-0.0655
0.0106	0.1059	0.3561	0.0487	-0.0645	0.0009	0.0142	-0.0635	0.0783	-0.1053
0.0119	0.0663	0.0487	0.3719	-0.0969	-0.0203	0.0353	-0.0664	0.0927	-0.0752
-0.0335	-0.0220	-0.0645	-0.0969	0.5181	-0.0110	-0.0280	-0.0555	0.0240	0.0185
-0.0151	0.0590	0.0009	-0.0203	-0.0110	0.3626	0.0152	0.0925	0.0200	0.0382
-0.0122	-0.0113	0.0142	0.0353	-0.0280	0.0152	0.3685	0.0030	0.0527	-0.0783
-0.0869	-0.0365	-0.0635	-0.0664	-0.0555	0.0925	0.0030	0.4055	-0.0944	0.1214
-0.0138	0.1329	0.0783	0.0927	0.0240	0.0200	0.0527	-0.0944	0.3905	-0.1127
-0.0426	-0.0655	-0.1053	-0.0752	0.0185	0.0382	-0.0783	0.1214	-0.1127	0.3757
-0.0248	0.0173	0.0389	0.0078	0.1096	-0.0504	-0.0313	-0.0909	0.0276	-0.0493
-0.0020	-0.0666	-0.0295	0.0560	-0.1209	-0.0403	-0.0175	0.0281	-0.0746	0.0375
0.0137	-0.0083	0.0280	-0.0379	-0.0535	0.0813	0.0408	0.0655	-0.0347	-0.0016
-0.0122	-0.0113	0.0142	0.0353	-0.0280	0.0152	0.1454	0.0030	0.0527	-0.0783
-0.0122	-0.0113	0.0142	0.0353	-0.0280	0.0152	0.1454	0.0030	0.0527	-0.0783
0.0111	-0.0089	-0.0669	0.0301	-0.0561	-0.0309	-0.0095	0.0254	0.0070	0.0509
0.1448	-0.0009	0.0495	0.0197	0.0761	-0.0655	-0.0435	-0.1778	0.0415	-0.0919
-0.0419	0.0097	0.0103	0.0434	-0.0223	0.0487	-0.0335	0.0296	-0.0270	0.0264
0.0419	-0.0097	-0.0103	-0.0434	0.0223	-0.0487	0.0335	-0.0296	0.0270	-0.0264
0.0241	0.0776	0.0345	0.1135	-0.0689	-0.0356	-0.1101	-0.0694	0.0400	0.0031
0.0899	-0.0778	-0.0416	-0.0202	0.0666	-0.0121	0.0236	-0.0534	-0.0185	-0.0177
-0.1234	0.0557	-0.0230	-0.0767	0.2284	0.0011	-0.0516	-0.0022	0.0425	0.0362
0.0646	0.1186	0.0531	0.0118	-0.1110	0.1364	-0.0206	0.0590	0.0523	0.0133
-0.0899	0.0778	0.0416	0.0202	-0.0666	0.0121	-0.0236	0.0534	0.0185	0.0177

0.0032	0.0270	-0.0547	0.0440	0.0885	0.0191	0.0385	-0.0310	0.0891	-0.0074
-0.0013	0.0396	0.0843	-0.1000	0.0324	0.0212	-0.0211	0.0029	-0.0145	-0.0302
0.0998	-0.0471	-0.0242	-0.0901	0.0429	-0.0217	-0.0070	-0.0250	-0.0259	0.0031
-0.0978	0.1137	0.0537	0.0342	0.0780	0.0620	0.0245	-0.0030	0.1006	-0.0406
0.0292	0.0300	-0.0410	-0.0292	0.0631	0.0851	-0.0661	0.0315	0.0017	0.0693
-0.0291	-0.0300	0.0410	0.0292	-0.0631	-0.0851	0.0661	-0.0315	-0.0017	-0.0693
0.0457	-0.0487	0.0003	-0.0114	-0.0075	0.0422	0.1049	0.0076	-0.0002	-0.0488
-0.0612	0.0104	0.0687	0.0027	-0.1091	-0.0461	0.0020	0.0264	-0.0361	-0.0221
-0.0151	0.0590	0.0009	-0.0203	-0.0110	0.1394	0.0152	0.0925	0.0200	0.0382
0.0013	-0.0396	-0.0843	0.1000	-0.0324	-0.0212	0.0211	-0.0029	0.0145	0.0302
0.0579	-0.0374	-0.0139	-0.0467	0.0206	0.0270	-0.0405	0.0046	-0.0530	0.0295
0.0829	0.0866	-0.0026	0.0761	-0.0115	0.0161	0.0027	-0.0644	0.1214	-0.0323
0.0099	0.0307	0.0174	-0.0699	-0.0237	-0.0097	-0.0306	0.0283	-0.0074	0.0208
0.0169	0.0186	-0.0268	0.0061	0.0350	0.1004	0.0793	0.0345	0.0544	-0.0090
0.0106	0.1059	0.1330	0.0487	-0.0645	0.0009	0.0142	-0.0635	0.0783	-0.1053
-0.1376	-0.0222	-0.0383	0.0146	0.0795	-0.0240	0.0763	0.0289	0.0206	-0.0046

Weights from Columns 11 through 20

-0.0248	-0.0020	0.0137	-0.0122	-0.0122	0.0111	0.1448	-0.0419	0.0419	0.0241
0.0173	-0.0666	-0.0083	-0.0113	-0.0113	-0.0089	-0.0009	0.0097	-0.0097	0.0776
0.0389	-0.0295	0.0280	0.0142	0.0142	-0.0669	0.0495	0.0103	-0.0103	0.0345
0.0078	0.0560	-0.0379	0.0353	0.0353	0.0301	0.0197	0.0434	-0.0434	0.1135
0.1096	-0.1209	-0.0535	-0.0280	-0.0280	-0.0561	0.0761	-0.0223	0.0223	-0.0689
-0.0504	-0.0403	0.0813	0.0152	0.0152	-0.0309	-0.0655	0.0487	-0.0487	-0.0356
-0.0313	-0.0175	0.0408	0.1454	0.1454	-0.0095	-0.0435	-0.0335	0.0335	-0.1101
-0.0909	0.0281	0.0655	0.0030	0.0030	0.0254	-0.1778	0.0296	-0.0296	-0.0694
0.0276	-0.0746	-0.0347	0.0527	0.0527	0.0070	0.0415	-0.0270	0.0270	0.0400
-0.0493	0.0375	-0.0016	-0.0783	-0.0783	0.0509	-0.0919	0.0264	-0.0264	0.0031
0.3518	-0.0198	-0.0287	-0.0313	-0.0313	-0.0999	0.1038	0.0584	-0.0584	0.0391
-0.0198	0.3537	-0.0032	-0.0175	-0.0175	0.0231	-0.0218	0.0705	-0.0705	0.0735
-0.0287	-0.0032	0.3407	0.0408	0.0408	-0.0888	-0.0151	0.0493	-0.0493	-0.0787
-0,0313	-0.0175	0.0408	0.3685	0.1454	-0.0095	-0.0435	-0.0335	0.0335	-0.1101
-0.0313	-0.0175	0.0408	0.1454	0.3685	-0.0095	-0.0435	-0.0335	0.0355	-0.1101
-0.0999	0.0231	-0.0888	-0.0095	-0.0095	0.4119	-0.0888	-0.1077	0.1077	0.0396
0.1038	-0.0218	-0.0151	-0.0435	-0.0435	-0.0888	0.4717	0.0165	-0.0165	0.0632
0.0584	0.0705	0.0493	-0.0335	-0.0335	-0.1077	0.0165	0.4007	-0.1776	0.0770
-0.0584	-0.0705	-0.0493	0.0335	0.0335	0.1077	-0.0165	-0.1776	0.4007	-0.0770
0.0391	0.0735	-0.0787	-0.1101	-0.1101	0.0396	0.0632	0.0770	-0.0770	0.4467
0.0233	-0.0017	0.0241	0.0236	0.0236	-0.0474	0.1132	0.0094	-0.0094	-0.0438
0.0863	-0.1192	-0.0776	-0.0516	-0.0516	-0.0087	-0.0371	-0.0318	0.0318	-0.0251
-0.0985	-0.0405	0.0709	-0.0206	-0.0206	0.0276	-0.0339	-0.0026	0.0026	0.0324
-0.0233	0.0017	-0.0241	-0.0236	-0.0236	0.0474	-0.1132	-0.0094	0.0094	0.0438
-0.0113	-0.0451	-0.0626	0.0385	0.0385	0.0739	-0.0081	-0.0373	0.0373	0.0050
0.0311	-0.0855	0.0659	-0.0211	-0.0211	-0.0970	0.0299	-0.0332	0.0332	-0.0789
-0.0455	-0.0642	0.0011	-0.0070	-0.0070	0.0444	0.0543	-0.1314	0.1314	-0.0831
0.0653	-0.0664	0.0021	0.0245	0.0245	-0.0674	-0.0325	0.0610	-0.0610	0.0097
-0.088	-0.0308	0.0141	-0.0661	-0.0661	-0.0054	0.0204	0.0455	-0.0455	0.0369
0.0088	0.0308	-0.0141	0.0661	0.0661	0.0054	-0.0204	-0.0455	0.0455	-0.0369
-0.0184	-0.0112	0.0912	0.1049	0.1049	-0.0729	0.0273	0.0126	-0.0126	-0.1163
-0.0016	0.0388	0.0122	0.0020	0.0020	-0.0106	-0.0628	-0.0088	0.0088	0.0007
-0.0504	-0.0403	0.0813	0.0152	0.0152	-0.0309	-0.0655	0.0487	-0.0487	-0.0356
-0.0311	0.0855	-0.0659	0.0211	0.0211	0.0970	-0.0299	0.0332	-0.0332	0.0789
0.0129	0.0062	0.0505	-0.0405	-0.0405	-0.0633	0.0708	0.0461	-00461	-0.0062
-0.0594	-0.0453	-0.0731	0.0027	0.0027	0.1324	0.0235	-0.0886	0.0886	0.0734
-0.0688	-0.0625	-0.0230	-0.0306	-0.0306	0.0917	-0.0589	-0.1409	0.1409	-0.0393
-0.0400	-0.0483	0.0549	0.0793	0.0793	-0.0149	-0.0231	0.0120	-0.0120	-0.0732
0.0389	-0.0295	0.0280	0.0142	0.0142	-0.0669	0.0495	0.0103	-0.0103	0.0345
0.0352	-0.0060	-0.0400	0.0763	0.0763	0.0048	-0.1024	0.0052	-0.0052	-0.0617

Weights from Columns 21 through 30

0.0899	-0.1234	0.0646	-0.0899	0.0032	-0.0013	0.0998	-0.0978	0.0292	-0.0292
-0.0778	0.0557	0.1186	0.0778	0.0270	0.0396	-0.0471	0.1137	0.0300	-0.0300
-0.0416	-0.0230	0.0531	0.0416	-0.0547	0.0843	-0.0242	0.0537	-0.0410	0.0410
-0.0202	-0.0767	0.0118	0.0202	0.0440	-0.1000	-0.0901	0.0342	-0.0292	0.0292
0.0666	0.2284	-0.1110	-0.0666	0.0885	0.0324	0.0429	0.0780	0.0631	-0.0631
-0.0121	0.0011	0.1364	0.0121	0.0191	0.0212	-0.0217	0.0620	0.0851	-0.0851
0.0236	-0.0516	-0.0206	-0.0236	0.0385	-0.0211	-0.0070	0.0245	-0.0661	0.0661
-0.0534	-0.0022	0.0590	0.0534	-0.0310	0.0029	-0.0250	-0.0030	0.0315	-0.0315
-0.0185	0.0425	0.0523	0.0185	0.0891	-0.0145	-0.0259	0.1006	0.0017	-0.0017
-0.0177	0.0362	0.0133	0.0177	-0.0074	-0.0302	0.0031	-0.0406	0.0693	-0.0693
0.0233	0.0863	-0.0985	-0.0233	-0.0113	0.0311	-0.0455	0.653	-0.0088	0.0088
-0.0017	-0.1192	-0.0405	0.0017	-0.0451	-0.0855	-0.0642	-0.0664	-0.0308	0.0308
0.0241	-0.0776	0.0709	-0.0241	-0.0626	0.0659	0.0011	0.0021	0.0141	-0.0141

0.0236	-0.0516	-0.0206	-0.0236	0.0385	-0.0211	-0.0070	0.0245	-0.0661	0.0661
0.0236	-0.0516	-0.0206	-0.0236	0.0385	-0.0211	-0.0070	0.0245	-0.0661	0.0661
-0.0474	-0.0087	0.0276	0.0474	0.0739	-0.0970	0.0444	-0.0674	-0.0054	0.0054
0.1132	-0.0371	-0.0339	-0.1132	-0.0081	0.0299	0.0543	-0.0325	0.0204	-0.0204
0.0094	-0.0318	-0.0026	-0.0094	-0.0373	-0.0332	-0.1314	0.0610	0.0455	-0.0455
-0.0094	0.0318	0.0026	0.0094	0.0373	0.0332	0.1314	-0.0610	-0.0455	0.0455
-0.0438	-0.0251	0.0324	0.0438	0.0055	-0.0789	-0.0831	0.0097	0.0369	-0.0369
0.3422	-0.0525	-0.0412	-0.1191	0.0231	-0.0213	0.0504	-0.0486	0.0236	-0.0236
-0.0525	0.5040	-0.0698	0.0525	0.0654	0.0537	-0.0075	0.1266	0.0395	-0.0395
-0.0412	-0.0698	0.4654	0.0412	-0.0008	0.0413	0.0277	0.0128	0.0907	-0.0907
-0.1191	0.0525	0.0412	0.3422	-0.0231	0.0213	-0.0504	0.0486	-0.0236	0.0236
0.0231	0.0654	-0.0008	-0.0213	0.3670	-0.0988	-0.0017	0.0468	0.0427	-0.0427
-0.0213	0.0537	0.0413	0.0213	-0.0988	0.0474	0.0660	0.0196	-0.0118	0.0118
0.0504	-0.0075	0.0277	-0.0504	-0.0017	0.0660	0.3838	-0.0965	0.0064	-0.0064
-0.0486	0.1266	0.0128	0.0486	0.0468	0.0196	-0.0965	0.3860	0.0245	-0.0245
0.0236	0.0395	0.0907	-0.0236	0.0427	-0.0118	0.0064	0.0245	0.1229	-0.1229
-0.0236	-0.0395	-0.0907	0.0236	-0.0427	0.0118	-0.0064	-0.0245	-0.3460	0.3460
0.0834	-0.0909	0.0045	-0.0834	-0.0005	0.0117	0.0233	-0.0110	-0.0142	0.0142
-0.0829	-0.0262	-0.0244	0.0829	-0.1048	0.0660	-0.0275	-0.0113	-0.0946	0.0946
-0.0121	0.0011	0.1364	0.0121	0.0191	0.0212	-0.0217	0.0620	0.0851	-0.0851
0.0213	-0.0537	-0.0413	-0.0213	0.0988	-0.1843	-0.0660	-0.0196	0.0118	-0.0118
0.0598	-0.0392	0.0251	-0.0598	-0.0390	0.0328	0.0293	-0.0355	0.0519	-0.0519
-0.0061	-0.0055	0.1050	0.0061	0.1240	-0.0787	0.0477	-0.0024	0.0482	-0.0482
-0.0687	0.0450	0.0689	0.0687	-0.0248	0.0873	0.1103	-0.0479	-0.0172	0.0172
0.0472	-0.0122	0.0701	-0.0472	0.0812	-0.0329	-0.0006	0.0489	0.0568	-0.0568
-0.0416	-0.0230	0.0531	0.0416	-0.0547	0.0843	-0.0242	0.0537	-0.0410	0.0410
-0.0307	0.1101	-0.1309	0.0307	0.0589	-0.0529	-0.0787	0.0847	-0.0575	0.0575

Weights from Columns 31 through 40

0.0457	-0.0612	-0.0151	0.0013	0.0579	0.0829	0.0099	0.0169	0.0106	-0.1376
-0.0487	0.0104	0.0590	-0.0396	-0.0374	0.0866	0.0307	0.0186	0.1059	-0.0222
0.0003	0.0687	0.0009	-0.0843	-0.0139	-0.0026	0.0174	-0.0268	0.1330	-0.0383
-0.0114	0.0027	-0.0203	0.1000	-0.0467	0.0761	-0.0699	0.0061	0.0487	0.0146
-0.0075	-0.1091	-0.0110	-0.0324	0.0206	-0.0115	-0.0237	0.0350	-0.0645	0.0795
0.0422	-0.0461	0.1394	-0.0212	0.0270	0.0161	-0.0097	0.1004	0.0009	-0.0240
0.1049	0.0020	0.0152	0.0211	-0.0405	0.0027	-0.0306	0.0793	0.0142	0.0763
0.0076	0.0264	0.0925	-0.0029	0.0046	-0.0644	0.0283	0.0345	-0.0635	0.0289
-0.0002	-0.0361	0.0200	0.0145	-0.0530	0.1214	-0.0074	0.0544	0.0783	0.0206
-0.0488	-0.0221	0.0382	0.0302	0.0295	-0.0323	0.0208	-0.0090	-0.1053	-0.0046
-0.0184	-0.0016	-0.0504	-0.0311	0.0129	-0.0594	-0.0688	-0.0400	0.0389	0.0352
-0.0112	0.0388	-0.0403	0.0855	0.0062	-0.0453	-0.0625	-0.0483	-0.0295	-0.0060
0.0912	0.0122	0.0813	-0.0659	0.0505	-0.0731	-0.0230	0.0549	0.0280	-0.0400
0.1049	0.0020	0.0152	0.0211	-0.0405	0.0027	-0.0306	0.0793	0.0142	0.0763
0.1049	0.0020	0.0152	0.0211	-0.0405	0.0027	-0.0306	0.0793	0.0142	0.0763
-0.0729	-0.0106	-0.0309	0.0970	-0.0633	0.1324	0.0917	-0.0149	-0.0669	0.0048
0.0273	-0.0628	-0.0655	-0.0299	0.0708	0.0235	-0.0589	-0.0231	0.0495	-0.1024
0.0126	-0.0088	0.0487	0.0332	0.0461	-0.0886	-0.1409	0.0120	0.0103	0.0052
-0.0126	0.0008	-0.0487	-0.0332	-0.0461	0.0886	0.1409	-0.0120	-0.0103	-0.0052
-0.1163	0.0007	-0.0356	0.0789	-0.0062	0.0734	-0.0393	-0.0732	0.0345	-0.0617
0.0834	-0.0829	-0.0121	0.0213	0.0598	-0.0061	-0.0687	0.0472	-0.0416	-0.0307
-0.0909	-0.0262	0.0011	-0.0537	-0.0392	-0.0055	0.0450	-0.0122	-0.0230	0.1101
0.0045	-0.0244	0.1364	-0.0413	0.0251	0.1050	0.0689	0.0701	0.0531	-0.1309
-0.0834	0.0829	0.0121	-0.0213	-0.0598	0.0061	0.0687	-0.0472	0.0416	0.0307
-0.0005	-0.1048	0.0191	0.0988	-0.0390	0.1240	-0.0248	0.0812	-0.0547	0.0589
0.0117	0.0660	0.0212	-0.1843	0.0328	-0.0787	0.0873	-0.0329	0.0843	-0.0529
0.0223	-0.0275	-0.0217	-0.0660	0.0293	0.0477	0.1103	-0.0006	-0.0242	-0.0787
-0.0110	-0.0113	0.0620	-0.0196	-0.0355	-0.0024	-0.0479	0.0489	0.0537	0.0847
-0.0142	-0.0946	0.0851	0.0118	0.0519	0.0482	-0.0172	0.0568	-0.0410	-0.0575
0.0142	0.0946	-0.0851	-0.0118	-0.0519	-0.0482	0.0172	-0.0568	0.0410	0.0575
0.3629	-0.0344	0.0422	-0.0117	0.0349	-0.0382	-0.0611	0.0907	0.0003	0.0028
-0.0344	0.3643	-0.0461	-0.0660	-0.0363	-0.0831	0.0554	-0.0926	0.0687	0.0146
0.0422	-0.0461	0.3626	-0.0212	0.0270	0.0161	-0.0097	0.1004	0.0009	-0.0240
-0.0117	-0.0660	-0.0212	0.4074	-0.0328	0.0787	-0.0873	0.0329	-0.0843	0.0529
0.0349	-0.0363	0.0270	-0.0328	0.2985	-0.0409	-0.0306	0.0114	-0.0139	-0.0735
-0.0382	-0.0831	0.0161	0.0787	-0.0409	0.4361	0.0537	0.0509	-0.0026	-0.0480
-0.0611	0.0554	-0.0097	-0.0873	-0.0306	0.0537	0.4022	-0.0478	0.0174	-0.0480
0.0907	-0.0926	0.1004	0.0329	0.0144	0.0509	-0.0478	0.3593	-0.0268	0.0189
0.0003	0.0687	0.0009	-0.0843	-0.0139	-0.0026	0.0174	-0.0268	0.3561	-0.0383
0.0028	0.0146	-0.0240	0.0529	-0.0735	-0.0480	-0.0480	0.0189	-0.0383	0.4036

Figure 3 shows the performance of network done by Hopfield Neural Network and Table 1 explains network parameters used by network for training. Figure 4 shows the performance of the network when input data is not similar to the previously stored data.

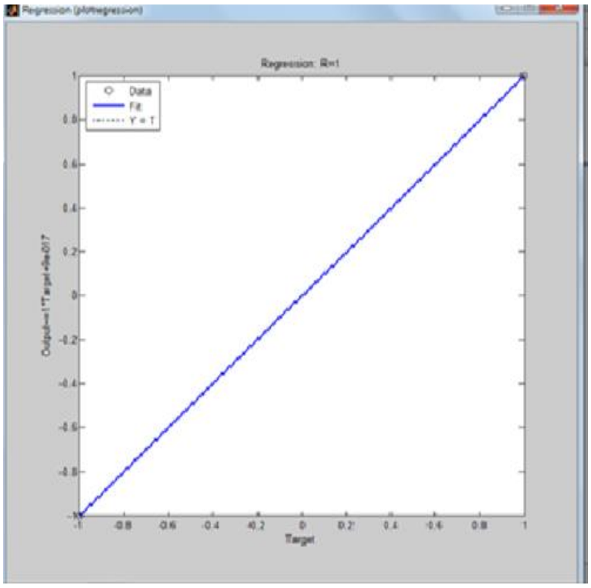

Figure 3. Training graph for Network Performance by Hopfield Neural Network

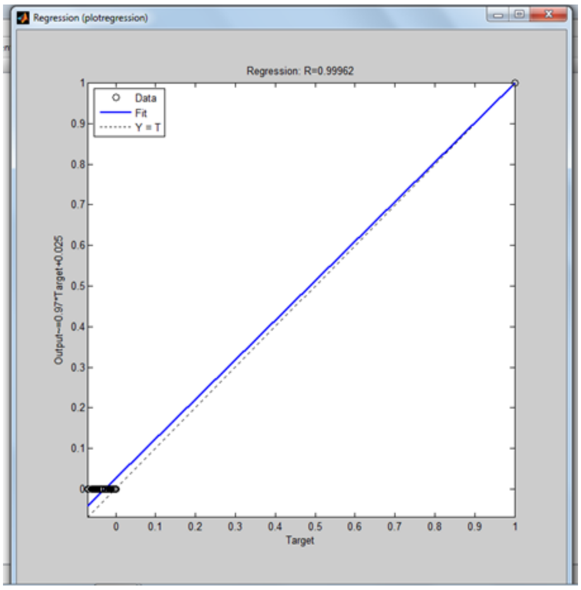

Figure 4. Graph showing variation in the input and output

Table 1. The parameters used for The Training of Hopfield Neural Network

Parameter	Value
Neurons in Input Layer	40
Neurons in Output Layer	40
Total Number of Patterns	8
Minimum Error Exist in the Network	12
Training Time	0.000003 sec
Initial Weight and biased term values	Values between 0 and 1

6. RESULT AND DISCUSSION

The simulation is carried out on MATLAB tool. The straight line of the graph in Figure 3 shows that the memory stores the passwords on the different devices in the form that no one can find out which training algorithm has been used for training the network. These network parameters are stored in the memory and when user access the network with the stored password, then system proves it as an authentic person who is in the premises of the organization. In contrast, if there are intruders who attack on the system security, i.e. perform man-in-the-middle-attack and want to hack the sensitive data from the organization, then the patterns that are stored in the network will give the network performance that is shown in Figure 4 by dotted line which shows some variation in the graph. Network parameters are also changes if some unauthentic users are interfering in the organization.

7. CONCLUSION

In this paper, we proposed an automatic detection of rogue access point which is based on a Hopfield Neural Network algorithm. Above simulations conclude that this neural network algorithm takes very less time for execution and stores number of patterns very accurately. The connections of different devices were stored in the form of network parameters which is hard to crack and the variation in the input, output data show the presence of unauthorized points i.e. rogue access points. The researchers generated a secret key for encryption and decryption of the messages using HPNN. Some of the researchers train the HPNN model that could recall the legal users and reject the illegal users correctly and covered registration and authorization phases. Different researchers implemented various approaches and methods, but none of them use Hopfield Neural Network Mechanism for detecting the rogue access points in wireless communication.This proposed neural network approach is a safe and easy operated method than the conventional method of encryption. The proposed model is designed for utilizing the existing WLAN infrastructure and there is no need of extra equipment for performing this detection.

REFERENCES

[1] A. Nayyar, "Security Issues on Converged Wifi & WiMAX Networks," *National Conference on Recent Advancements in Computer Science (RACS)*, 2011.

[2] H. Kim, *et al.*, "A Daily Activity Monitoring System for Internet of things-assisted Living in Home Area Network," *International Journal of Electrical and Computer Engineering (IJECE)*, vol/issue: 6(1), pp. 399-405, 2016.

[3] B. Jeon, *et al.*, *"Network management system for wireless LAN service,"* 10th International Conference on Telecommunications, vol. 2, pp. 948-953, 2003.

[4] G. Gopichand and R. K. Saravanaguru, "A Generic Review on effective Intrusion Detection in Ad Hoc Networks," *International Journal of Electrical and Computer Engineering (IJECE)*, vol/issue: 6(4), pp. 1779-1784.

[5] N. Prasad and A. Prasad, "WLAN Systems and Wireless IP for Next Generation Communications," Artech House, Inc. Noorwod, USA, 2002.

[6] "Rogue Access Point Detection‖ Automatically Detect and Manage Wireless Threats to Your Network," www.wavelink.com.

[7] T. Kim, *et al.*, "Online detection of fake access points using received signal strengths," *75th IEEE Vehicular Technology Conference (VTC Spring)*, 2012.

[8] R. Beyah, *et al.*, *"Rogue Access Point Detection using Temporal Traffic Characteristics,"* 4th IEEE Global Telecommunications Conference (GLOBECOM), 2004.

[9] S. B. Vanjale, *et al.*, "Detecting and Eliminating Rogue Access Point in IEEE 802.11WLAN," *International Journal of Smart Sensors and Ad Hoc Networks (IJSSAN)*, vol/issue: 1(1), pp. 108-112, 2011.

[10] S. Nikbakhsh, *et al.*, "A Novel approach for rogue access point detection on the client-side," *26th International conference on Advanced Information Networking and Applications Workshops (WAINA), Japan*, pp. 684-687, 2012.

[11] "Airdefense enterprise: WIPS," Available: *http://www.airdefense.net*, "Survey-Enterprise Wireless LAN Security & WLAN Monitoring," www.airdefense.net/products/survey/index.php.

[12] "Airmagnet," Available: http://www.airmagnet.com, "WLAN Security, Analysis and Wireless Network Design Tool," enterprise.netscout.com/enterprise –network/wireless-design-analysis-and-security.

[13] S. Jana and S. K. Kasera, "On fast and accurate detection of unauthorized wireless access points using clock skews," *IEEE Transactions on Mobile computing*, vol/issue: 9(3), pp. 449-462, 2010.

[14] T. Kindberg, *et al.*, *"Authenticating public wireless network with physical evidence,"* 5th IEEE International Conference on Wireless and Mobile Computing, Networking and Communication (WiMob), pp. 394-399, 2009.

[15] G. Shivraj, *et al.*, "A hidden markov model based approach to detect rogue access points," *IEEE Militery Communications Conference (MILCOM)*, 2008.

[16] K. Kao, *et al.*, "Detecting rogue access points using client-side bottleneck bandwidth analysis," *Computers & Security*, vol/issue: 28(3-4), pp. 144-152, 2009.

[17] Liran M., *et al.*, *"A Hybrid Rogue Access Point Protection Framework for Commodity Wi-Fi Networks,"* 27th IEEE Conference on Computer Communications (INFOCOM), pp. 1894-1902, 2008.

[18] H. Deng, *et al.*, "Routing Security in Wireless Ad Hoc Networks," *IEEE Communications Magazine*, vol/issue: 40(10), pp.70-75, 2002.

[19] J. P. HuBaux, *et al.*, *"The Quest for Security in mobile Ad Hoc Networks,"* 2nd ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC), pp. 146-155, 2001.

[20] H. Hsieh and R Sivakumar, "Transport over Wireless networks," Handbook of wireless Networks and Mobile Computing, 2002.

[21] Y. Hu, *et al.*, *"Packet Leashes: A Defense against Worm Hole Attacks in Wireless Ad Hoc Networks,"* Proceedings of IEEE INFORCOM, 2002.

[22] S. Kumar, "Neural Networks: A Classroom Approach," Tata McGraw-Hill Education, 2004.

[23] M. T. Hagan, *et al.*, "Neural Network Design," PWS Publishing, Boston, USA, 1996.

[24] B. D. C. N. Prasad, *et al.*, "A Study on Associative Neural Memories," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol/issue: 1(6), pp. 124-133, 2010.

[25] T. Schmidt, *et al.*, *"A Review of Applications Of Artificial Neural Networks In Cryptosystems,"* Seventh International Symposium on Neural Networks, China, 2010.

[26] P. Michiardi and R. Molva, "Ad Hoc Networks Security," John Wiley & Sons, Inc. New York, USA, 2003.

[27] J. Principe, *et al.*, "Neural and Adaptive System – Fundamentals through Simulations," John Wiley & Sons, Inc. New York, USA, 1999.

BIOGRAPHIES OF AUTHORS

Ms Menal Dahiya is Assistant Professor of Computer Science at Maharaja Surajmal Institute(GGSIP University,Delhi) and a Research Scholar of Maharshi Dayanand University,Rohtak in the Dept. Of Computer Science and Applications. She received her MPhil in Computer Science from Chaudhary Devi Lal University, Sirsa, India in 2007.Before she had studied at Guru Jambheshwar University of Science & Technology (GJU), Hisar and KUK, Kurukshetra,India. Her main research interest are Neural Network,Wireless Security and Wireless Communication. Several of her research papers have been published in international peer-reviewed journals indexed in Scopus, Copernicus and others.

Dr Sumeet Gill is Assistant Professor of Computer Science at Department of Mathematics at Maharshi Dayanand University, Rohatk, India. He received his Ph.D in Computer Science from Dr.B. R .Ambedgar University, Agra, India in 2009. Before he had done MSc (Physics) from KUK, Kurukshetra in 1999 and S.S. Plasma Astro Physics from Indian Institute of Science, Bangalore, India in 1998. His main research area are Neural Network, Wireless Communication. He is on Research Panel of various universities like Dr.B. R. Ambedger University, RTM University, SGV University etc.. He is also a Liaison officer of the USENIX Association, U.S.A. He published 2 Books and several Research Papers indexed in SCOPUS and Copernicus.