A LIGHT WEIGHT SOLUTION FOR DETECTING DE-AUTHENTICATION ATTACK

Rajinder Singh[1] and Satish Kumar[2]

[1]Department of Computer Science and Applications, PUSSGRC Hoshiarpur, Punjab, India

[2]Department of Computer Science and Applications, PUSSGRC Hoshiarpur, Punjab, India

ABSTRACT

Nowadays Wireless local area networks (WLANs) are growing very rapidly. Due to the popularity of 802.11 networks, possibilities of various attacks to the wireless network have also increased. In this paper, a special type of attack De-Authentication/disassociation attack has been investigated. In a normal scenario, a wireless client or user sends a de-authentication frame when it wants to terminate the connection. These frames are in plain text and are not encrypted. These are not authenticated by the access point. Attackers take advantage of this, and spoof these packets and disable the communication between the connected client and access point. In this paper, an algorithm based on radio-tap header information is suggested to identify whether there is a De-Authentication attack on the client or not.

KEYWORDS

De-Authentication Attack, Kali Linux, Scapy, Python

1. INTRODUCTION

Wireless LANs have become much more popular recently. These networks are growing fast, gaining popularity rapidly in various big organizations as well as in small industries. Wireless LANs have many advantages over the traditional LANs, such as mobility and cost savings. Due to the low cost of the wireless network hardware and easy procedure of installing the wireless network, anyone can install the wireless network. The two main components of a WLAN are wireless Access Point (AP) and a network interface card (NIC). Wireless clients such as Laptops, Smart-phones, printers, tabs, etc. connect to AP through the antenna and AP is connected to the distributed network either by cables or wireless. NIC in wireless clients is used to connect AP in WLANs. Nowadays, the majority of wireless LAN networks are based on IEEE 802.11 standards. But with the increase in popularity of WLAN, WLAN networks are also becoming prone to various types of attacks, because this standard has many security issues. With the recent development of wireless network various threats, risks and vulnerabilities have also increased. Due to the broadcasting nature of the wireless network all the devices which are present in the range of the wireless network can send and receive data. Therefore security of the wireless network is very important. There is a need to secure the wireless network with proper methods. To increase the security of the wireless network many security protocols and standard were developed such as WEP, WPA, and WPA2, EAP and 802.11i etc. However, there are still many vulnerabilities and flaws which are not addressed by these protocols and later these protocols were found to be vulnerable to many attacks. S ecurity of the wireless network is the primary concern for the organization nowadays. The main pillar for the wireless network security is given below. Confidentiality: Confidentiality means that transmitted information is not disclosed to any

unauthorized users. Many different algorithms are used to provide the confidentiality of the data. Authentication: It is the verification of a user's identity as well as the sender's identity. Access Control: It means controlling the access of a resource across the network and also limiting its use for the authenticated user. Integrity Control: It means that the data sent by an authorized user is not altered during the transmission [1].

Rest of the paper consists of following important sections. In section-II main vulnerabilities of WLAN are discussed. In section-III an introduction about De-Authentication attack and necessary tools which are used to carry this attack have been discussed. In section-IV, a literature review of this attack has been discussed. In section V and VII, architecture and research methodology are discussed. Discussion related to result is given in section VII and Conclusion is drawn in section VIII.

2. MAIN VULNERABILITIES OF WLAN

Main reasons due to which WLAN becomes vulnerable are given below:

1) Transfer of data in open air: Major difference between a wired network and WLAN is that in case of WLAN data is transmitted through air using radio signals instead of transmitting electrical signals over the wire. So anyone with the use of current wireless technology can intercept radio signals and can monitor data [2].
2) Flaws in WEP: Wired Equivalent Privacy (WEP) security algorithm was introduced in IEEE 802.11 to provide user data confidentiality. But sooner many security flaws [3] were found in this protocol. According to Fluhrer, Mantin and Shamir [4] an attacker can find the secret key, if he is able to eavesdrop a large amount of the frames. Another issue with the WEP is that it is optional and many users ignore to turn on this encryption facility.
3) Flaws in WPA: To fix the various weaknesses discovered in WEP, a new protocol Wi-Fi Protected Access (WPA) was came into the picture and it was released in 2003. WPA is considered stronger than WEP, but later it is also found vulnerable to many attacks such as dictionary and brute force attacks [5].
4) No encryption of the Management Frames: Data frames which are sent across the network are encrypted but the management frames are not encrypted. So all 802.11 control frames and management frames are vulnerable to various threats [6].
5) MAC Access Control List (ACL): WLANs can use this security mechanism so that only authorized users are allowed to use the network and its resources. But in case of WLAN the ACL contain both the SSID as well as the client MAC address. So MAC ACLs are vulnerable to MAC spoofing because transmission is in the open air through radio waves and hackers can easily sniff the AP's SSID and MAC address of the clients [7].
6) Rogue access points: An attacker sets up an unauthorized access point having specifications as that of original access point without the permission of a network administrator. In case of IEEE 802.11 standard a user having a wireless device will always be connected to Access point which has strong signal strength. Attacker fools the client by using an antenna having strong signals. This causes the wireless client to disconnect from the legal AP and connects to RAP. So an attacker can read the data and can get the sensitive information [8]. RAP can also be used by the attacker to carry out Man in the middle attack [9].

3. PROBLEM STATEMENT (DE-AUTHENTICATION ATTACK)

De-Authentication attack falls under the category of management frame attack. This attack can also be used as Denial of Service attack. Under normal circumstances, De-Authentication frame is used to gracefully terminate a connection between the connected client and Access point. A wireless client can disconnect itself from the network by sending the De-Authentication frame to

the connected AP. After receiving this frame Access Point also sends the De-Authentication frame back to the client. This is the normal process for De-Authentication, but an attacker can take advantage of this process. The attacker first waits for a client to authenticate to the AP. The attacker then spoofs the MAC address of this client and sends the De-Authentication frame to AP. So the connection with the connected client is broken as shown in Figure1. There are a number of free tools available to carry out this attack.
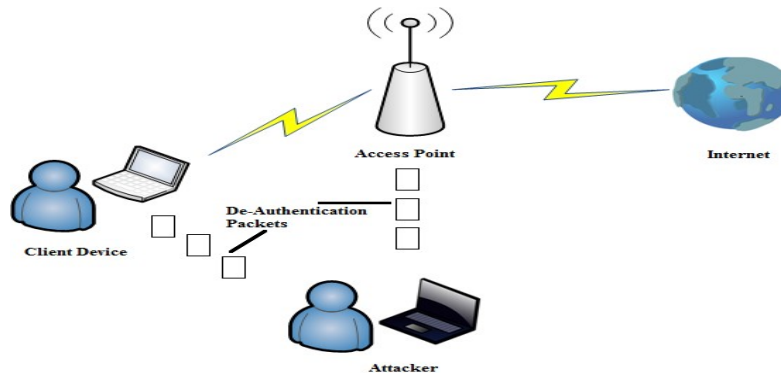

Figure 1 De-Authentication Attack

In this paper main two tools which are used to create the spoofed De-Authentication frames are i)

Kali Linux ii) Scapy

Kali Linux: Kali Linux is Debian based Linux distribution. It has more than five hundred penetration testing programs. Main tools which are offered by the Kali Linux are Nmap, Wireshark, John the Ripper (password cracker tool) and aircrack-ng. One of the main advantages of Kali Linux is that it can be run within a virtual machine also. Many packages of Kali Linux are imported from Debian repositories. Kali Linux contains many well-known security tools such as Nmap, aircrack-ng, kismet etc. In this paper, we have used the aircrack-ng tool for our research. It is written in C language. It is a software suite that can detect, sniffs packet and can crack the passwords. It can be used for sniffing 802.11a, 802.11b, and 802.11g traffic. This tool can be run on Linux as well as on windows operating systems. This tool has the following two powerful utility airman-ng which can be used to place different wireless network cards into monitor modes and replay-ng which can be used as packet injector [19].

SCAPY: Scapy is a powerful and interactive python based packet manipulation program which can send and sniff network packets. It can be used to generate and manipulate network traffic. It can be used as a construction tool that can scan probe or attack networks. Scapy can be used to decode a large number of packets of various protocols. Main tasks which are carried out with Scapy are i) scanning ii) fingerprinting iii) testing iv) attacking v) sniffing vi) packet forging. Scapy can also be used to inject your own 802.11 frames in the network. Scapy also help users to create their own packets. Users can put any value in any field. So users are able to make new tools for network by writing small lines of code. It can run interactively from the command prompt or it can be used as a library in a Python script. The main foundation of Scapy tool is python language which can be easily run on any system [20].

4. LITERATURE REVIEW

Nguyen et al.[10] developed a protocol which uses the one-way hard function to identify whether the De-Authenticated frames are from the legitimate users/stations or not. But the disadvantage of this method is that Proposed modified device drivers are hardware Specific. Moreover, the open source implementation of these solutions is also not given. Authors [11] used a scripting language to find the De-Authentication attack. They used a threshold number and if the count of De-

Authentication frames increases the threshold number, then it is De-Authentication attack. But this method considers only one parameter and other parameters related to the wireless network are ignored, leading a lot of false positives. Authors [12] proposed a scheme for detecting De-Authentication attack. They also proposed a method for reducing the DoS attack. This scheme uses the delegation concept and random number rate control mechanism. According to their method under De-Authentication attack, AP is not in a position to handle large authentication requests. So this AP forwards these packets to Authentication Server (AS). After verifying the hash, AS communicates back with the AP, for checking whether there is DoS attack or not. But this method is helpful only in IEEE 802.11i standard and it is not helpful for other standards. Authors [13] proposed an IDS/IPS for detecting the de-authentication DoS attack. They also suggested a solution from recovering the attack. But one drawback of this method is that it used a threshold number and value of this number can change a number of clients increases in the network or leave the network. The research only considers the arrival of a number of frames and do not consider MAC header and/or radio-tap header parameters for detection of a de-authentication attack. Authors [14] analyzed the traffic pattern during De-Authentication attack and suggested countermeasures: 1) To increase the beacon time interval 2) mapping MAC address. But the main drawback of this is that they considered a number of different types of packets transferred on the network. Authors in [15] proposed a new technique for preventing a de-authentication attack. This tool can be used as personal WIDS. It can detect De-Authentication attack and disassociation attack in case of 802.11a/b/g/n standards. But the main drawback of this method is it considered Radio tap header only. Authors in [16] proposed Machine Learning (ML) approach to detect the De-Authentication attack in case of Wi-Fi network. Their method used machine learning-based classifiers for detecting De-Authentication attack. Authors [17] implemented test beds to conduct two types of attack de-authentication and evil twin attack to learn their behavior. Then attack signatures and techniques are used to detect these attacks. But only reason code parameter is considered to verify the De-Authentication attack. Other parameters of the MAC header are not considered.

A tabular form of the methods used by the Authors and main limitation of their methods is given in the Table 1.

Table 1

Sr. No.	Author and Year	Method Used	Limitations
1	D. Nguyen et al(2008)	Suggested Letter envelop protocol based on one way hard function to check whether Authentication frame or disassociation frame is from a valid user or not.	Proposed device drivers are hardware Specific.
2	Agarwal et al(2013)	Authors suggested IDS and IPS that is based on throughput of the network, de-authentication frames from each station and total number of de-authentication frames in the network. Then they used a static/dynamic threshold value to check the attack.	Used a threshold number for de-authentication frames to find out the threat, but the threshold number can vary with a number of clients. The research only considers the arrival of a number of frames and do not consider MAC header and/or radio-tap header parameters for De-Authentication attack.
3	Hafiz et al (2014)	Authors collected wireless traffic and analyzed the traffic pattern in normal scenario and then studied the network behaviour in case of de-authentication attack.	It concentrated on the number of different types of packets transferred on the network.

4	Baharudin et al(2015)	This research proposed a WIDS consist of three modules. Monitor mode is used to capture all the wireless traffic, detect module for capturing de-authentication frames and alarm module to generate alarms.	Considered Radio tap header only
5	Agarwal et al.(2015)	The authors suggested Machine Learning (ML) approach to detect the de-authentication DOS attack on Wi-Fi network	Here machine learning based approach is used to detect the attack.
6	Afzal et al (2016)	The authors implemented test beds to conduct de-authentication attack and evil twin attack to learn attack behaviour.	Only reason code parameter is considered to verify the De-Authentication attack. Other parameters of the MAC header are not considered.

5. ARCHITECTURE

The test bed for the De-Authentication diagram is shown in Figure 2. In this case, testbed consists of a wireless network, an AP and a client. The AP connects to the internet, and the client connects to the AP. It provides all the services to the connected client. The attacker machine uses Kali Linux. Kali Linux machine can be used to launch the De-Authentication attack. Aircrack-ng tool available in Kali Linux is used to launch the attack. After a successful attack, the client station is disconnected from the wireless network and is not connected to it until the attacker stops the attack.

The same procedure is repeated to carry out the De-Authentication Attack with Scapy tool.
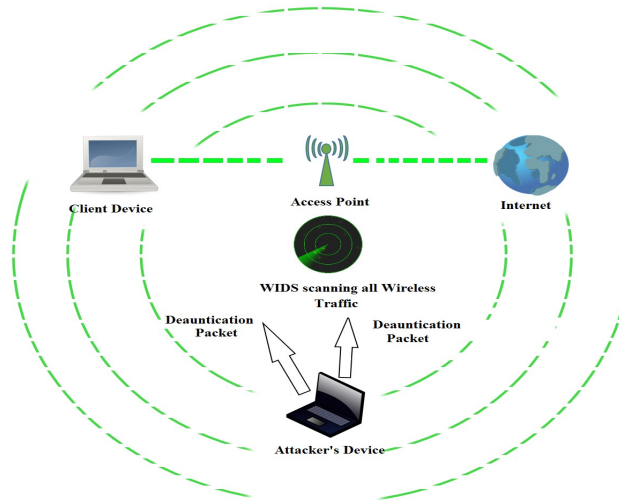

Figure 2 Test Bed for detection of De-Authentication Attack

6. RESEARCH METHODOLOGY

6.1 DESIGN

The proposed lightweight WIDS module is stored in a monitoring station that scans all the wireless traffic data and if it senses the de-authentication packets it sets alerts to the security about the possible De-Authentication attack (Figure 3). For the station to be cost-effective and also

powerful, we use the embedded hardware to create these stations. These stations are programmed in Python Language because of its ability to run on the new embedded devices. Also, Python Language is a high-level language and has many libraries to work with the wireless packets. In this case, we have used Python's PyShark library to dissect and check the packets arriving in the area of the WIDS. For sensing the packets in the area, the WIDS requires a network card to be operating in the monitor mode so that it can collect and dissect all the packets coming from various devices. For a small network like Home or Small offices, one such monitoring station is efficient. For large networks, multiple stations are required based on the size of the network.
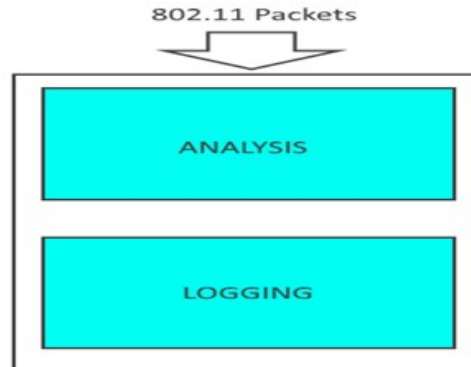
Figure 3 Research Methodology

6.2. WORKING

When the WIDS starts operating, it starts capturing all the 802.11 frames and dissects and inspects the packets. Along with these tests the WIDS also maintains the logs of these packets for future studies. It also keeps track of all the neighbouring BSSID, SSID, RSSI values and Channel number. If the WIDS detects an attack during scanning it generate alerts and maintains the details in the LOG files.

6.3. DETECTION OF ATTACK

Main parameters which are considered in our algorithm for the detection of this attack are i) Reason Code ii) MAC timestamp. Reason Code: According to paper [17] the majority of the De-Authentication attack tool use reason code 7 in all the de-authentication frames. Same reason code may also be used by the legitimate de-authentication frame, but the same reason code in many de-authentication frames is not normal. Therefore, considering only this parameter can increase the rate of false positives. ii) MAC timestamp: During the experiments a very little change in the value of this field noted. Since the attacker is using a tool to generate de-authentication frames and this tool create many de-authentication frames in the very small time period, so there is a very small change in the value of this field between successive de-authentication frames. To detect a De-Authentication attack, we use the criteria of mac timestamp of de-authentication packets in the wireless network and their reason code. In most de-authentication attack tools like Aircrack-ng, the standard reason code generated by the tool is 7. Also, in most of the cases to launch De-Authentication attack the attacker uses the reason code as 7. The second indicator is the mac timestamp of the de-authentication frame. After analysis of the many de-authentication attack scenarios, we found out that in most cases the there is a very small change in the value of mac timestamp in successive generated de-authentication packets. As shown in the Figures (4-6) given below:
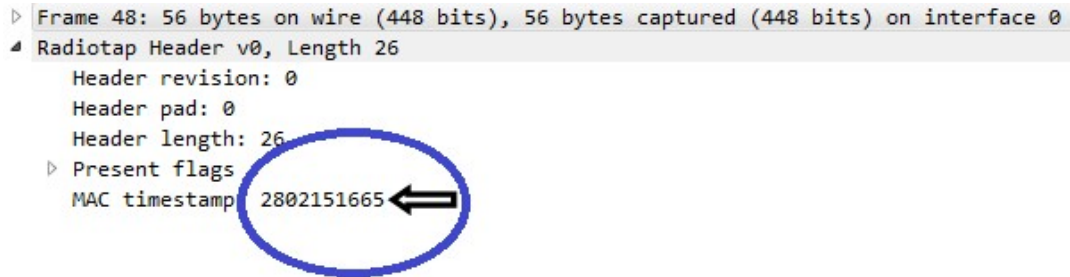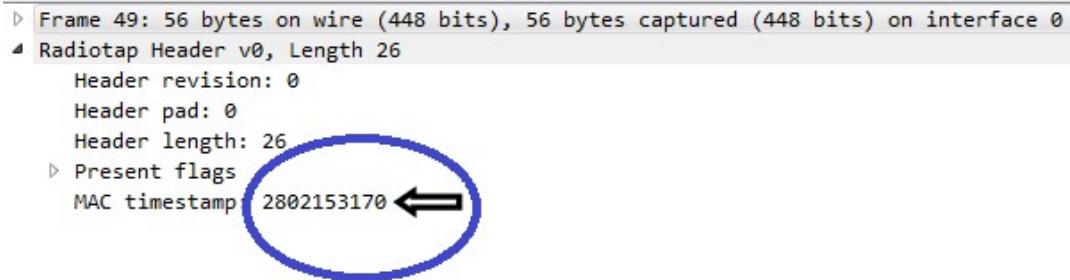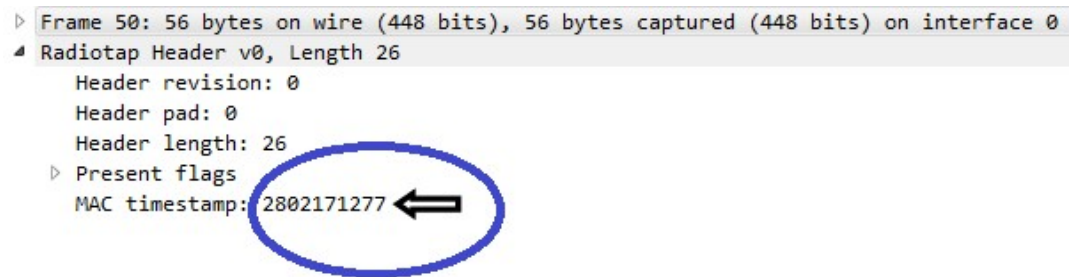
Figure 4

Figure 5

Figure 6

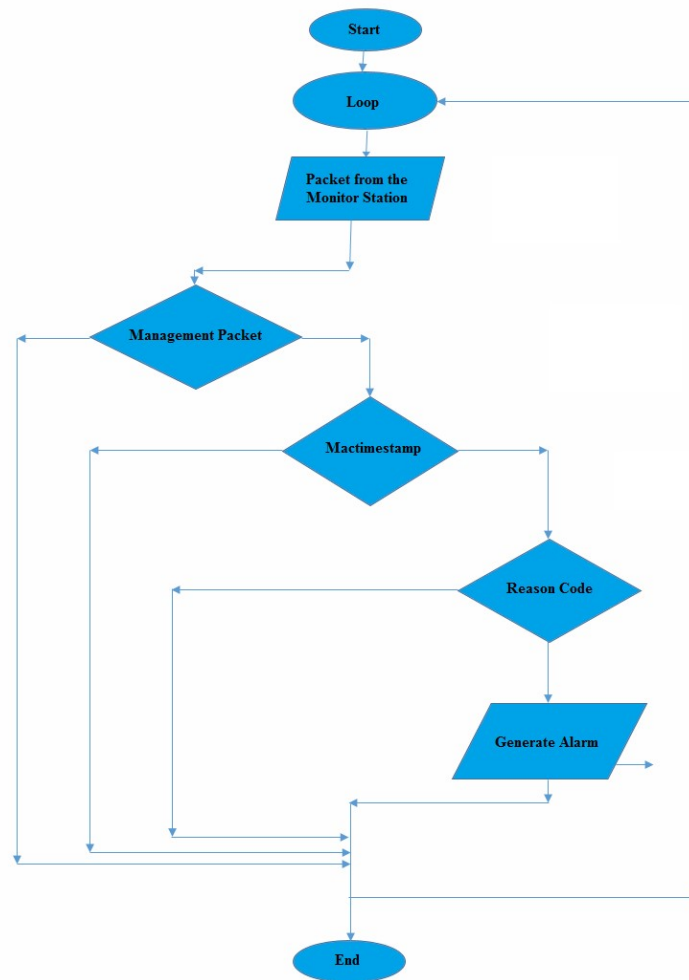A flowchart for the detection of the attack is given below.

Figure 7 Flowchart for detecting De-authentication Attack

Algorithm for Detection of Deauthentication attack.

DetectDeauth(Scanned_Packets)
//This algorithm detects the De-Authentication attack. Here Scanned Packets is the packets
//received while monitoring the network.
Output: Alert

```
check if management frames
check subtype
if(subtype == deauth) then
extractmactime
if(mactime[intervals] is constant) then
        check reason_code
        if(reason_code == same)
                alert()
        else if ( data_frames_after_deauth )
                alert()
```

```
        end if
else if ( data_frames_after_deauth )
        alert()
end if
else
return
end if
```

7. RESULT AND DISCUSSION

We conducted extensive experiments to study the behaviour of nodes under normal as well as under de-authentication attack conditions. Most of the attack tools use reason code 7 in all the de-authentication frames, but the same reason code can also be used by the legitimate de-authentication frame. So this parameter alone is not sufficient to detect this attack
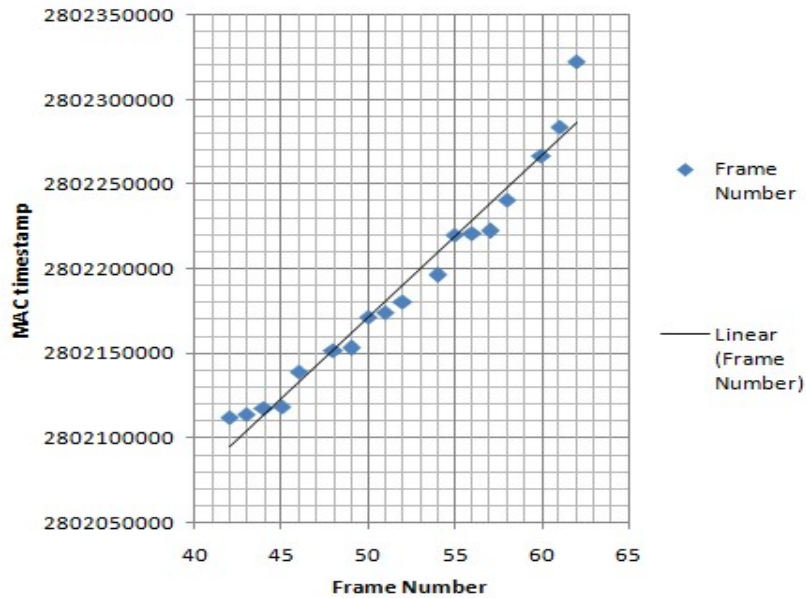

Figure 7

In the Figure 7 we have plotted the scattered graph between MAC timestamp field and frame number received. From the graph it is clear that there is a very small change in the value of MAC timestamp field between the successive de-authentication frames. The same kind of observation was made for the second node within same network as shown below (Figure 8).
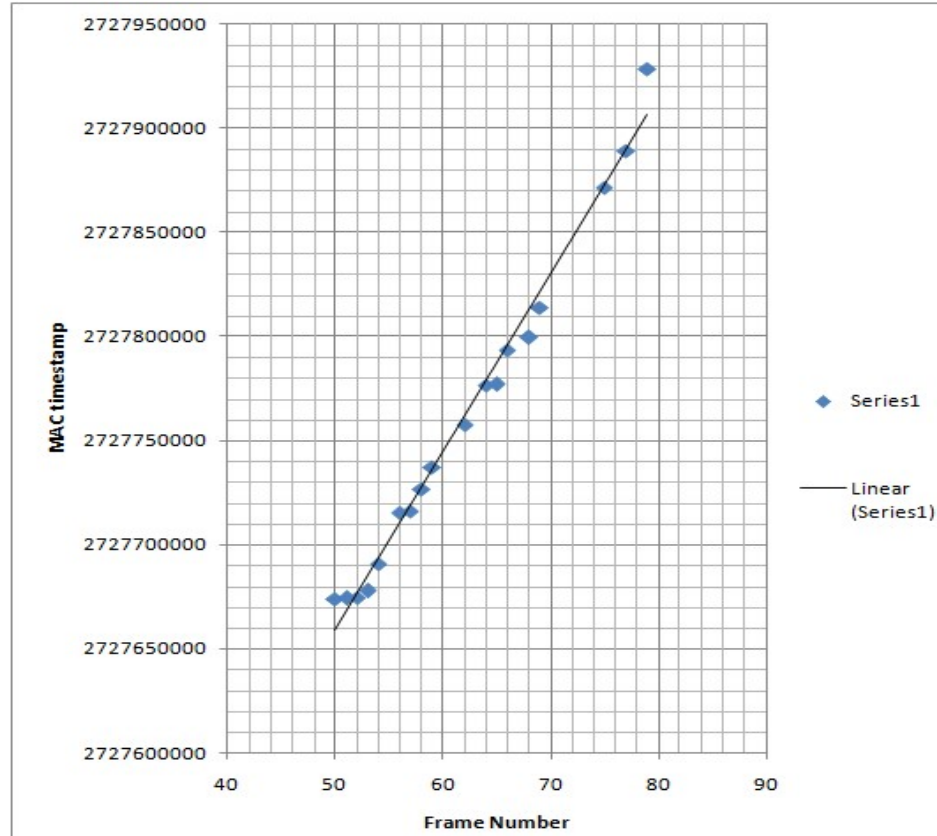
Figure 8

We have tested our algorithm for five clients and it successfully detected the De-Authentication attack. Result of detecting the live de-authentication attack during live scenario using our algorithm is given below (Figure 9).

Figure 9

Comparison with the other methods: Authors [13] gave a method for detecting the De-Authentication Attack. But this method is helpful when there are many clients in the network connected by an AP. Therefore, this method depends upon only one parameter threshold number of the die-authentication frames, which can lead to an increase in the false positive rate. Our method focuses on a single client and it uses combinations of three parameters do decrease the false positive rate. Authors [15] used a combination of the radio-tap header information for the detection of de-authentication attacks such as channel type, dB Antenna signal, and rate. Authors

[16] used only reason code parameter in the MAC header to detect the attack. Authors [18] considered a number of different frames as compared to our method which considered only De-Authentication frame. Authors [16] proposed a Machine Learning (ML) based Intrusion Detection System (IDS) to detect the de-authentication DoS attack in a Wi-Fi network. But our method does not follow the machine learning approach for detecting the De-Authentication attack.

8. CONCLUSIONS

In this paper, we described a lightweight solution for the detection of De-Authentication Attack. Practical implementation of this solution is suitable for a wireless environment and it can be easily installed on the wireless networks. This algorithm uses reason code and MAC timestamp parameters. These two parameters can be easily used to reduce false positive rates.

REFERENCES

[1] http://www.cis.temple.edu/~jiewu/research/publications/Publication_files/WLANsec.pdf

[2] http://www.sans.org/readingroom/whitepapers/wireless/security-vulnerabilities-wireless-lan-technology-1629

[3] Bittau, A., Handley, M., Lackey, J., "The final nail in WEP's coffin", IEEE Symposium on Security and Privacy, pp. 386–400, (2006).

[4] Fluhrer, S., Mantin, I., Shamir, A., "Weaknesses in the key scheduling algorithm of RC4", LNCS, vol. 2259, pp. 1–24, (2001).

[5] T. Stimpson, L. Liu, J., Zhang, R. Hill, W. Liu, Y. Zhan, "Assessment of Security and Vulnerability of Home Wireless Networks", IEEE 9th International Conference on Fuzzy Systems and Knowledge Discovery, Chongqing, China, 29-31 May, 2012, pp. 2133-2137.

[6] http://www.sans.org/readingroom/whitepapers/wireless/security-vulnerabilities-wireless-lan-technology-1629

[7] Waliullah, Md, and Diane Gan. "Wireless LAN Security Threats & Vulnerabilities: A Literature Review." International Journal of Advanced Computer Science & Applications 5.1 (2014).

[8] http://en.wikipedia.org/wiki/Rogue_access_point

[9] http://en.wikipedia.org/wiki/Man-in-the-middle_attack

[10] Nguyen, Thuc D., et al. "A lightweight solution for defending against deauthentication/disassociation attacks on 802.11 networks." 2008 Proceedings of 17th International Conference on Computer Communications and Networks. IEEE, 2008.

[11] TJ OConnor. Detecting and responding to data link layer attacks. SANS Institute InfoSec Reading Room, 13, 2010.

[12] Singh, Rajeev, and Teek Parval Sharma. "Detecting and reducing the denial of Service attacks in WLANs." Information and Communication Technologies (WICT), 2011 World Congress on. IEEE, 2011.

[13] Agarwal, Mayank, Santosh Biswas, and Sukumar Nandi. "Detection of De-Authentication Denial of Service attack in 802.11 networks." 2013 Annual IEEE India Conference (INDICON). IEEE, 2013

[14] Hafiz, Mohd Yusof Mohammad, and Fakariah Hani Mohd Ali. "Profiling and mitigating brute force attack in home wireless LAN." Computational Science and Technology (ICCST), 2014 International Conference on. IEEE, 2014.

[15] Baharudin, Norzaidi, et al. "Wireless Intruder Detection System (WIDS) in Detecting De-Authentication and Disassociation Attacks in IEEE 802.11." IT Convergence and Security (ICITCS), 2015 5th International Conference on. IEEE, 2015.

[16] Agarwal, Mayank, Santosh Biswas, and Sukumar Nandi. "Detection of De-Authentication DoS Attacks in Wi-Fi Networks: A Machine Learning Approach." Systems, Man, and Cybernetics (SMC), 2015 IEEE International Conference on. IEEE, 2015.

[17] Z. Afzal, J. Rossebø, B. Talha and M. Chowdhury, "A Wireless Intrusion Detection System for 802.11 networks," 2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), Chennai, 2016, pp. 828-834.

[18] Hafiz, Mohd Yusof Mohammad, and Fakariah Hani Mohd Ali. "Profiling and mitigating brute force attack in home wireless LAN." Computational Science and Technology (ICCST), 2014 International Conference on. IEEE, 2014.

[19] https://en.wikipedia.org/wiki/Kali_Linux

[20] https://scapy.net/

AUTHORS

Rajinder Singh is an Assist. Professor in DCSA, PUSSGRC, Hoshiarpur, Punjab, India. He has more than fifteen years of experience of teaching post-graduate classes. His areas of interest are Wireless Network Security, Cyber Security, Artificial Intelligence and Android Security. He is currently pursuing His Ph.D. Degree from P.U. Chandigarh, India.

Dr. Satish Kumar is Associate Professor in Department of Computer Science and Applications in Panjab University (PU), Chandigarh (India), currently posted at Panjab University SSG Regional Centre, Hoshiarpur, Punjab, India (a multi faculty prestigious campus of PU). He has more than fifteen years experience of teaching post-graduate classes. His areas of interest are Image Processing, Pattern Recognition, computer graphics and Artificial Intelligence.