Detection and Prevention of De-authentication Attack in Real-time Scenario

Shweta Sharma, Meenakshi Mittal

***Abstract**: Wireless Local Area Network (WLAN) is an infrastructure network in which nodes are connected to a centralized system to provide Internet access to mobile users by radio waves. But WLANs are vulnerable to Medium Access Control (MAC) layer Denial of Service (DoS) attacks due to the susceptibility of the management frames. An attacker can spoof the MAC address of the legitimate client and perform de-authentication attack to disconnect WLANs users from the access point. Many free tools are available in Kali Linux Operating System (OS) by which this attack can be performed and cause a security threat to WLAN users. The consequences of de-authentication DoS attack are frequent disconnection from Internet, traffic redirection, man-in-the-middle attack, and congestion. Despite enormous efforts in combating de-authentication DoS attack in the past decade, this attack is still a serious threat to the security of the cyber world. Medium Access Control Spoof Detection and Prevention (MAC SDP) DoS algorithm performs detection and prevention of de-authentication attack caused by spoofing MAC address. This algorithm is modified to make it more immune to the de-authentication attack and implemented in real-time scenario. The results show that the proposed technique increases the packet flow rate by 20.36%, reduces the packet loss by 95.71%, and reduces the down time and recovery time by 0.39 sec and 0.9 sec respectively as compared to MAC SDP DoS algorithm.*

***Keywords**: WLAN, DoS attack, De-authentication attack, MAC layer, and Access Point.*

I. INTRODUCTION

Wireless Local Area Networks (WLANs) use radio waves to connect mobile users to the Internet over the wireless medium [1]. It permits two or more devices to communicate and access the Internet through an Access Point (AP). A WLAN covers an area that can range from a small office to a large campus and permits users to connect laptops, computers, and mobile phones with Internet. It uses radio frequency to send out and receive data over the channel, where radio waves are transmitted ranging from extremely low frequency (3Hz to 3KHz) to extremely high frequency (30GHz to 300GHz).

Table- I: Management Frames

Type value	Sub-Frame	Sub-type Value
00	Probe request	0100
00	Probe response	0101
00	Authentication request	1011
00	**De-authentication request**	**1100**
00	Association request	0000
00	Association response	0001
00	Re-association request	0010
00	Re-association response	0011
00	Disassociation	1010
00	Beacon	1000

WLANs are based on the IEEE 802.11 standard which is a standard to specify WLAN communication in 2.4GHz and 5GHz frequency bands [2]. The frame format of MAC layer of 802.11 standard consists of management frames to establish and maintain the connections between stations and APs. The stations use management frames to connect and disconnect to the wireless network and move associations from one AP to another AP [3]. The value of the type field for the management frame is 00 as shown in Table- I. The communication process between a client and an AP is shown in Fig. 1 and described as follows:

Step 1: The client searches for network by sending a probe request frame and supported rates on multiple channels.

Step 2: APs after receiving the probe request will send a probe response to the client. The client connects to the AP which has the strongest channel.

Step 3: To prevent illegal clients from accessing the network, authentication is needed between client and AP. So client sends an authentication request to AP.

Step 4: AP response to the client by sending authentication response with a status code.

Step 5: After authentication, the client sends an association request frame to the AP to access a wireless network via an AP.

Step 6: Then, the AP sends an association response to the client and saves the client information in its own database.

Step 7: After this connection, communication will take place and the client is can send data to the AP and vice versa.

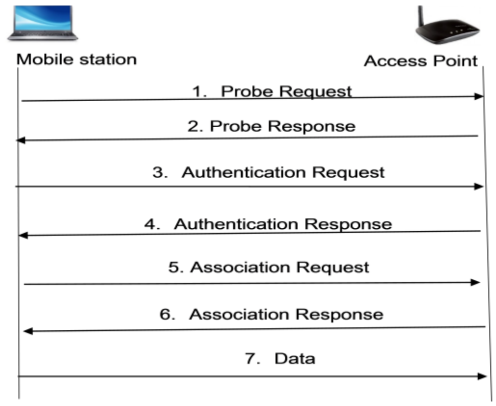


Fig. 1. Communication Process between client and AP

Manuscript published on 30 August 2019.
***Correspondence Author(s)**

Shweta Sharma*, Department of Computer Science and Engineering, NITTTR, Chandigarh, India. Email: shweta.rt.sharma@gmail.com

Meenakshi Mittal, Department of Computer Science and Technology, Central University of Punjab, Bathinda, India. Email: meenakshi.cup@gmail.com

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC-BY-NC-ND license http://creativecommons.org/licenses/by-nc-nd/4.0/

Retrieval Number: J12170881019/19©BEIESP
DOI: 10.35940/ijitee.J1217.0881019
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

This communication process can be thwarted by attackers after launching de-authentication attack which is a category of masquerading Denial of Service (DoS) attack. Thus, there is a requirement of defense mechanism to mitigate this attack. This paper contributes by proposing a framework for detection and prevention of de-authentication attack and performs implementation in real-time scenario.

This paper is organized as follows: Section II discusses de-authentication attack. Section III describes related work on mitigation of de-authentication attack. Section IV discusses proposed framework to detect and prevent de-authentication attack. Section V provides experimental setup and implementation details. Section VI discusses results obtained after implementing proposed technique. Section VII concludes.

II. De-Authentication Attack

De-authentication messages are used if client and AP wants to de-authenticate from each other. But these messages are not authenticated itself by any cryptographic procedure. Thus, an attacker can send de-authentication messages on users' behalf and launch de-authentication DoS attack. De-authentication attack is a MAC layer DoS attack in which an attacker spoofs the MAC address of legitimate user and send de-authentication packets to the AP. The AP will disconnect the legitimate user after getting request from spoofed MAC address. The attacker's motivation behind this attack is to keep everyone off from connecting to a specific AP. The impact of de-authentication attack is denial of service where the clients cannot connect to the internal WLAN network and the productivity will be lost [4]. This attack can disconnect all legitimate users from the AP with just one command. The Aireplay-ng tool under Aircrack-ng package is available in kali Linux OS to perform this attack.

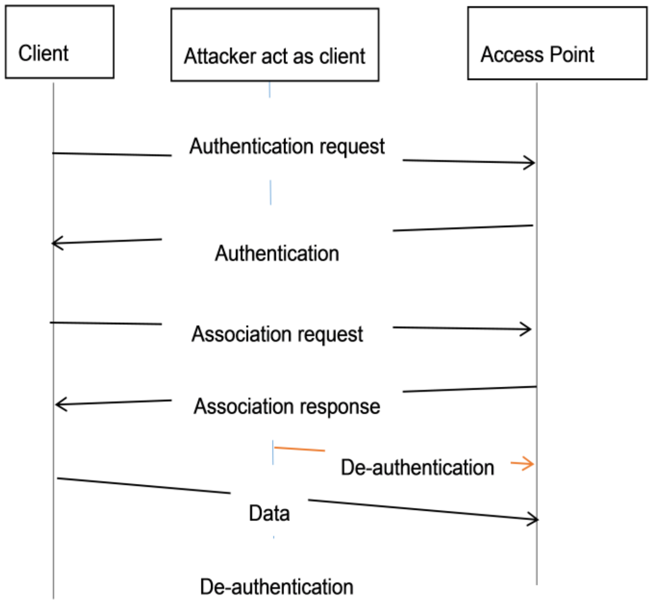


Fig. 2. De-authentication attack

Fig. 2 shows a scenario in which an attacker transmits de-authentication messages to an AP by spoofing MAC address of its legitimate clients. Since de-authentication is a notification, it can't be rejected, therefore, when an AP receives this message, it will de-authenticate the legitimate client whose MAC address is specified in the de-authentication message.

III. RELATED WORK

The MAC layer DoS attacks consumes resources of the network. Therefore, an attacker sends illegitimate de-authentication packets to the AP after spoofing MAC address of the legitimate users. When an AP receives these packets, it will de-authenticate the legitimate clients whose MAC address is mentioned in the de-authentication packets. The AP can't deny de-authentication notification. The existing countermeasures to detect the de-authentication DoS attack are discussed as follows: Bellardo & Savage [5], deferred the effects of de-authentication requests for 5-10 seconds by putting each request in a queue. If data packets arrived after queuing of de-authentication request, then the de-authentication request was rejected because a legitimate user never send packets in this order. But this technique did not work when mobile stations roam between the APs because deferring of management frames creates association problems and handoff issues. The author also suggested another approach to prevent from de-authentication attack by authenticating all management frames, but it required firmware up gradation and additional cost on client and AP side. Wright [6], performed an experiment by detecting anomalies in sequence number which had a sequential pattern to identify illegitimate de-authentication frames. But if the client moves out of the range of IDS, then it is not possible to monitor the sequence number patterns. The layer 2 wireless IDS can't monitor the clients who left the range of monitoring, so when the clients came back within the range of IDS, the sequence number appeared irregular due to a large gap in sequence values. Bicakci & Uzunay [7], proposed a scheme to use dynamically changing MAC address where each MAC address can be used for only one session. Moreover, the MAC address for next session can't be calculated from previous sessions. The limitation of this approach is the lost frame problem in which if a node updated its MAC address and then sent the frame which got lost and did not contact AP. Thus, the AP will not be able to update its database and used the old MAC address. Nguyen *et al.* [8], used letter-envelop protocol (LEP) to authenticate management frames. The unprotected management frames carried the MAC address of source, so an attacker can easily perform de-authentication and disassociation attacks by spoofing MAC address. This algorithm prevented the attack at the association level, but the algorithm will not work if the attacker launched the attack at the authentication level.

Cheema *et al.* [9], implemented de-authentication attack on the real wireless mesh test-bed and analyzed the impact of attack on victim's throughput and bandwidth which reaches to zero during the attack. They proposed an algorithm to determine whether the attack was performed by the attacker or the legitimate client wanted to de-authenticate from the AP. But this method did not provide any preventive measures against the attack. Wang & Srinivasan [10], proposed a hybrid mechanism to prevent from de-authentication attack.

In the first mechanism, if the de-auth frame was received by an authenticator before generation of Pre-shared Temporal Key (PTK), then they delayed the execution of these frames by queuing these requests for 5 seconds.

Also, these frames were protected by the sequence number to prevent replay attacks. In the second mechanism, the authenticator sent 128 bits cipher text HMACk (message) hashed by a secret key to the supplicant. This mechanism was used when an authenticator wanted to broadcast de-authentication frames in the WLAN. The key can be a nonce and generated every time for each broadcast to prevent it from forging.Arockiam & Vani [11], proposed MAC SDP DoS algorithm to detect and prevent MAC spoofing DoS attack by exchanging passkey values. The algorithm was simulated in NS2. But the key which were shared among clients and AP for authentication was sent in plain text. An attacker can easily spoof the key and launch the de-authentication attack. Agarwal *et al.* [12], proposed an Intrusion Detection System (IDS) with Intrusion Prevention System (IPS) to detect de-authentication attack. The IDS sniffed and transferred the packets to the analysis engine to filter malicious packets. After detecting de-authentication attack, the analysis engine informed IPS about the attack which in turn informed the AP to ignore the de-auth frames directed to the victim station. But the AP also dropped the legitimate de-authentication frame sent by the victim.

Shahidan & Mohammed [13] analyzed the de-authentication attack in Wireshark tool and performed detection with a portable tool in Kali Linux OS. Aung & Thant [14] performed detection of de-authentication attack with active and passive finger prints and performed implementation in real-time scenario. But these techniques did not work on prevention of de-authentication attack. Arora [15] worked on prevention of de-authentication attack by checking the source of the de-authentication frames. They assigned a unique ID to these frames which was hashed with secure hash algorithm. But it required firmware up gradation and additional cost on client and AP side.

IV. PROPOSED TECHNIQUE

In the proposed work, MAC SDP DoS algorithm [11] has been modified to detect and prevent the de-authentication attack which is launched by spoofing the MAC address of the legitimate client. The flow chart of the proposed technique is presented in Fig. 3 where the changes to the existing algorithm are highlighted in red color. The modified algorithm has been implemented in real-time and is explained as follows:

Step 1: Connect victim’s machine and attacker’s machine to the access point machine.

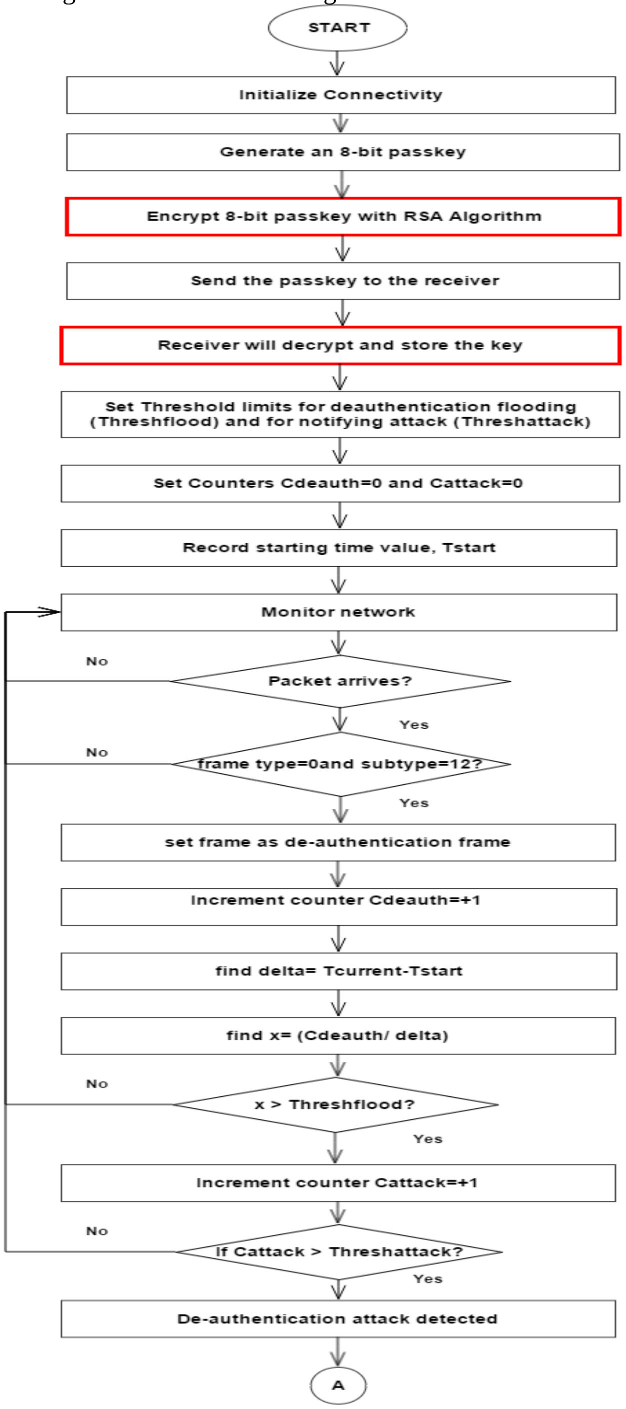
Step 2: The AP will generate an 8-bit pass-key and encrypt it with Rivest–Shamir–Adleman (RSA) algorithm to send it to the victim.

Step 3: The victim will decrypt the key by using RSA algorithm. The pass-key is stored by both the sender (AP) and the receiver (client).

Step 4: When an attacker performs the de-authentication attack on the victim’s machine, the de-authentication packets occurrence are detected by AP machine by checking threshold value. The threshold value is selected as 5 because at this value, the AP starts de-authenticating the client.

Step 5: After detection of de-authentication attack, AP will immediately stop sending Transmission Control Protocol (TCP) packets to the victim. The AP will request the victim to send pass-key. The victim will send encrypted key to the AP where the AP will decrypt and match the key with the available key.

Step 6: After successful verification of the key, the AP will check whether it has stopped receiving the de-authentication packets. If the AP stops receiving de-authentication packets, then AP will ask victim to send sequence number of last TCP packet. AP will send packets from last received legitimate sequence number to the victim. By this way, a victim can be prevented from de-authentication attack launched by spoofing MAC address of the legitimate client.


START
Initialize Connectivity
Generate an 8-bit passkey
Encrypt 8-bit passkey with RSA Algorithm
Send the passkey to the receiver
Receiver will decrypt and store the key
Set Threshold limits for deauthentication flooding (Threshflood) and for notifying attack (Threshattack)
Set Counters Cdeauth=0 and Cattack=0
Record starting time value, Tstart
Monitor network
No
Packet arrives?
Yes
No
frame type=0and subtype=12?
Yes
set frame as de-authentication frame
Increment counter Cdeauth=+1
find delta= Tcurrent-Tstart
find x= (Cdeauth/ delta)
No
x > Threshflood?
Yes
Increment counter Cattack=+1
No
If Cattack > Threshattack?
Yes
De-authentication attack detected
A

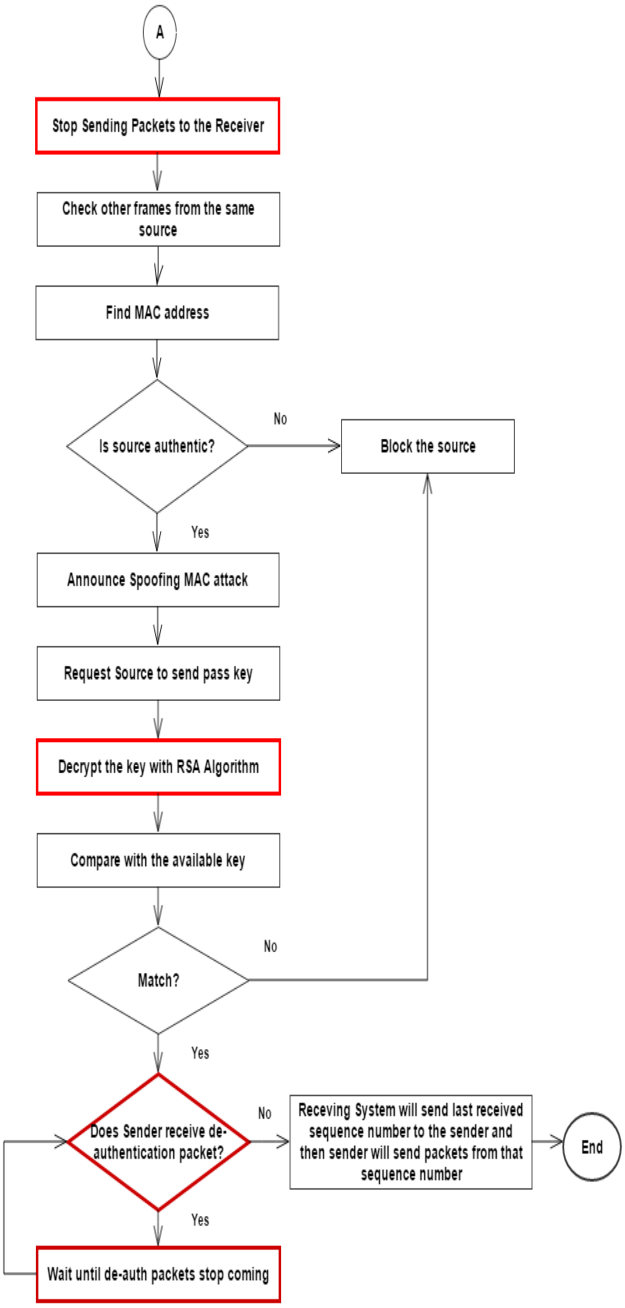

A
Stop Sending Packets to the Receiver
Check other frames from the same source
Find MAC address
Is source authentic?
No
Block the source
Yes
Announce Spoofing MAC attack
Request Source to send pass key
Decrypt the key with RSA Algorithm
Compare with the available key
Match?
No
Yes
Does Sender receive de-authentication packet?
No
Receving System will send last received sequence number to the sender and then sender will send packets from that sequence number
End
Yes
Wait until de-auth packets stop coming

Fig. 3. Proposed Framework

V. Experimental Setup and Implementation Details

The experimental environment is setup by taking three system (Access Point, Victim, and attacker machine) connected through WLANs as shown in Fig. 4. These systems run on Kali Linux OS. The configuration of these systems are shown in Table- II.

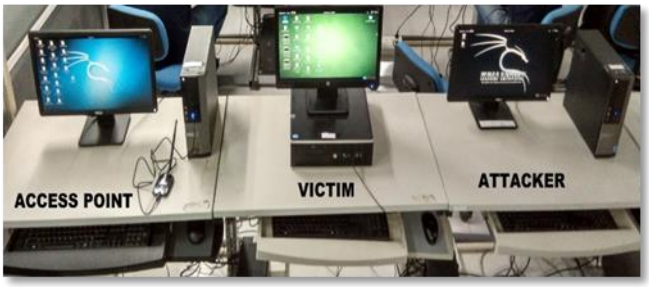


Fig. 4. Experimental Setup

Table- II: Configuration of Systems

	Machines		
	Access Point	**Victim's**	**Attacker's**
OS	Kali Linux 2.0 (sana)	Kali Linux 1.0	Kali Linux 1.0
Processor	Intel®Core™i7 -4770 CPU @ 3.40GHz x 8	Intel®Core™i7 -3770 CPU @ 3.40GHz x 8	Intel®Core™i7-4 770 CPU @ 3.40GHz x 8
RAM	4 GB	4 GB	4 GB
System type	64-bit OS	32-bit OS	32-bit OS

Table- III: Configuration of Network Adapters

Network Adapter	**Chipset**	**Standards**	**Frequency Range**	**Data Rate**
Alfa AWUS036NH A	Atheros AR9271	IEEE 802.11 b/g/n	2.412- 2.484 GHz	up to 150Mbps
D-link DWA-121	Realtek	IEEE 802.11 b/g/n	2.4- 2.4835 GHz	up to 150Mbps

The AP machine uses three network adapters– one Alfa network adapter and two D-link adapters. The purpose to use Alfa Network adapter is to create an AP on the system. The attacker and victim machine uses D-link adapters to access the network. The configurations of these adapters are shown in Table- III.

De-authentication attack has been performed on victim's machine from attacker's machine using "aircrack-ng" package which contains airmon-ng, airodump-ng, and aireplay-ng tools [16] to de-authenticate victim from AP. The analysis of de-authentication frames has been done with Wireshark tool [17]. The Socket programming has been written in Python language to exchange the key between AP and victim. The Scapy tool [18] has been run on both AP and victim's machine to send and stop the packets to victim's machine. Detection of de-authentication packets has been programmed in "Python" language on AP machine with Kali Linux OS (See Appendix).

VI. RESULTS AND DISCUSSIONS

We implemented the MAC SDP DoS algorithm and modified MAC SDP DoS algorithm (proposed) on AP in real-time scenario. We also performed a comparison between MAC SDP DoS algorithm and proposed technique by analyzing parameters, namely, number of TCP packets received by victim machine, packet flow rate, down time, recovery time, number of packets lost by AP, and number of packets resend by AP. We repeated each experiment 5 times and calculated the average values to display the results. The analysis of calculated parameters are discussed as follows:

A. Number of TCP packets received by victim's machine

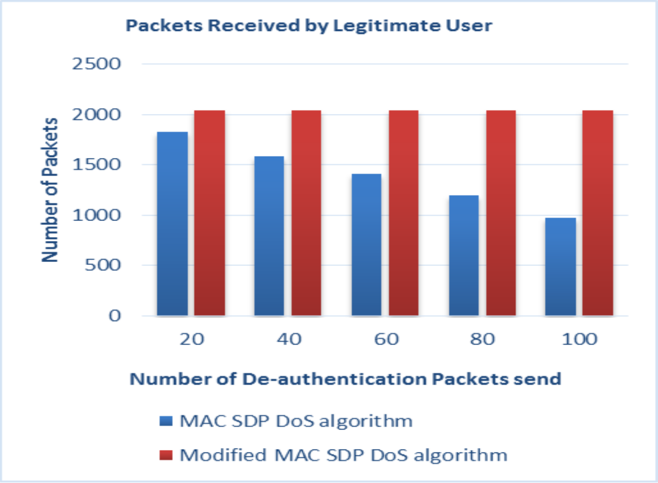


Fig. 5. Packets received by legitimate user under de-authentication attack

Analysis:

- In MAC SDP DoS algorithm, during de-authentication attack, all packets sent by AP are lost. That's why, the number of TCP packets received by the legitimate users decrease, as de-authentication packets increase.
- On the other side, 100% packets were received under proposed technique. This is because when AP detects de-authentication attack, then it stops transmitting packets immediately. After the attack, AP resends packets from last legitimate sequence number received by the victim.
- Fig. 5 shows that number of TCP packets received by the victim under proposed technique is more than the MAC SDP DoS algorithm. This is because AP resends all packets which were dropped during the de-authentication attack.

B. Packet flow rate

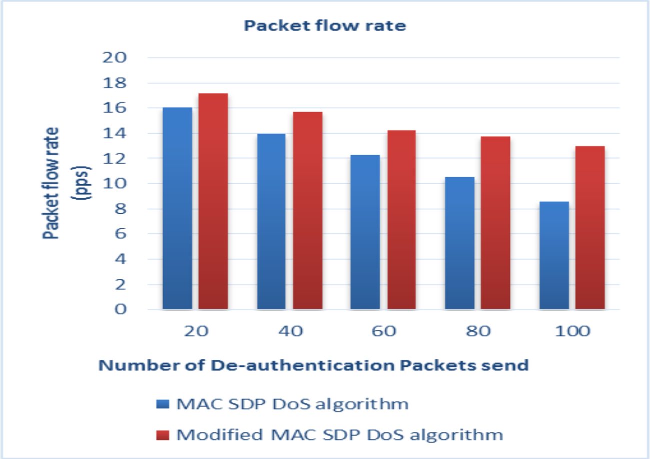


Fig. 6. Packet flow rate

Analysis:

- In MAC SDP DoS algorithm, the AP sends TCP packets to the victim's machine during the attack which will be lost and will not reach to the victim's machine when number of packets are large.
- In the proposed technique, AP immediately stops sending packets after detecting de-authentication attack. When the AP detects that it stops receiving de-authentication packets, then it will start sending TCP packets from the last legitimate packet sequence number received from the victim.
- Fig. 6 shows that packet flow rate of victim's machine under proposed technique is more than MAC SDP DoS Algorithm.

C. Down Time and Recovery Time

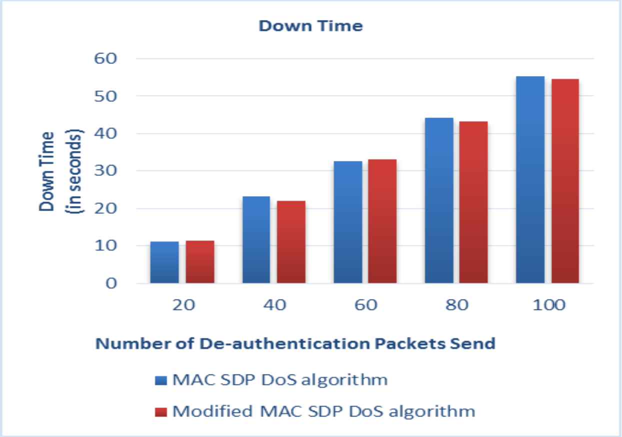


Fig. 7. Down Time

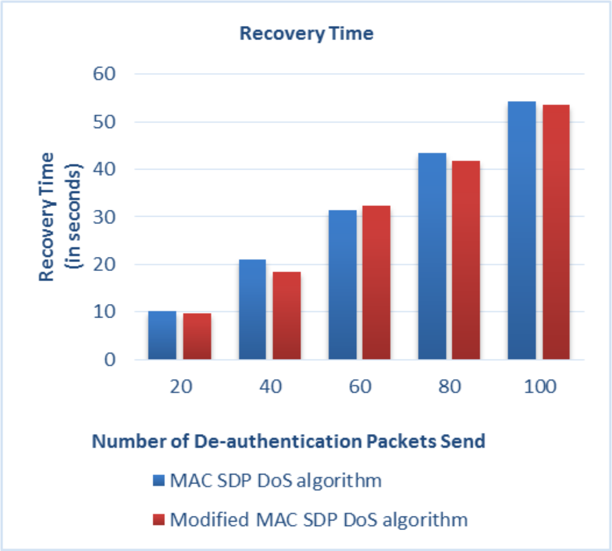


Fig. 8. Recovery Time

Analysis:

- As the de-authentication packets increase, the victim's machine network goes down for more period of time. This is because AP will de-authenticate the victim for the time it is receiving de-authentication packets. So the victim's machine goes down for more period of time and takes more time to recover.
- Fig. 7 and Fig. 8 shows that the down time and recovery time shows small improvement as compared to the MAC SDP DoS algorithm.

D. Packet Loss

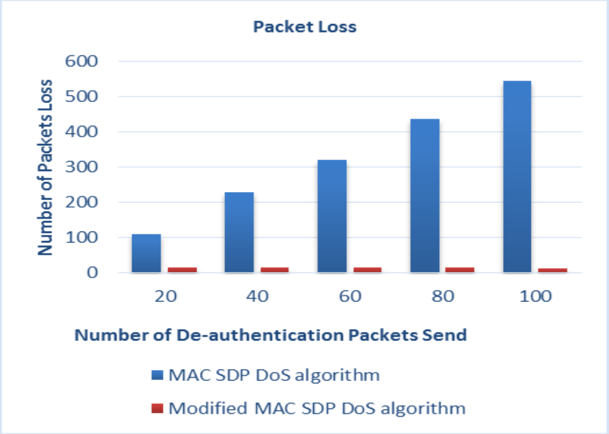


Fig. 9. Packets loss under de-authentication attack

Analysis:

- As shown in Fig. 9, Packet Loss are increasing as the de-authentication packets increase in MAC SDP DoS algorithm. This is because of the reason that AP is continuously sending packets to the victim even after detecting the attack. All the TCP packets under the attack were lost and not reached to the victim.
- In the proposed technique, Packet Loss is very less and constant because AP stops sending packets immediately after detecting the de-authentication attack. Under de-authentication attack, all packets are lost and don't reach to the victim, that's why AP stop sending packets to the victim after detecting the attack. But before detecting the attack, some packets sent by AP are lost. So there will be very less packet loss under the de-authentication attack. After recovery from the attack, AP will again resend packets from the last received sequence number by the victim.

E. Packet Resend

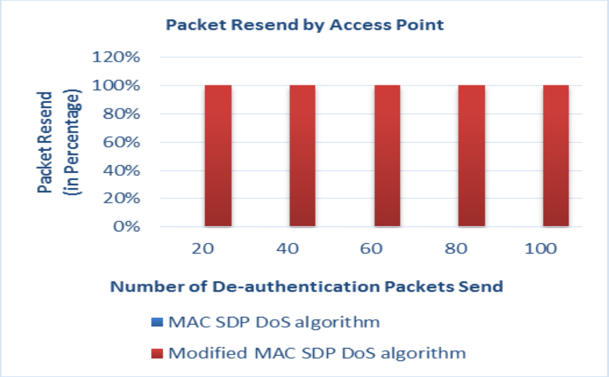


Fig. 10. Packets resend by AP after de-authentication attack

Analysis:

- As shown in Fig. 10, 0% packets are resend by AP in MAC SDP DoS algorithm. This is because packets send by AP to the victim don't stop before completion of attack. The detection algorithm will work but the prevention will not work in this case. Hence, during the attack, some packets are lost. After the attack, remaining packets will continuously reach to victim. The AP will not ask for legitimate sequence number from the victim and will not resend those packets which were lost during attack.
- In the proposed technique, AP immediately stops transmitting packets after detecting the attack and starts retransmitting packets to the victim from last received sequence number. All the packets which were lost during attack are resend by the AP. So it has 100% packet resend rate.

The overall result of calculated parameters are shown in Table- IV. These parameters are calculated by analyzing packets in Wireshark tool, for example, in order to calculate packet loss, we type "tcp.analysis.retransmission". In Table-IV, the number of packets received by legitimate user under proposed technique are 100%. The packet flow rate is increased by 20.36% in proposed technique as compared to MAC SDP DoS algorithm. The down time and recovery time is slightly reduced by 0.39 sec and 0.9 sec respectively in proposed technique as compared to MAC SDP DoS algorithm. The packet loss under MAC SDP DoS algorithm is 328 packets out of 2042 total number of packets, thus the percentage of packet loss becomes 16.08%. Similarly the packet loss under proposed technique is 14.08 packets out of 2042 total packets. Hence, the percentage of packet loss in proposed technique becomes 0.69%. Therefore the packet loss is decreased by 95.71% $\left[\left(\frac{16.08-0.69}{16.08}\right)*100\right]$ in proposed technique as compared to MAC SDP DoS algorithm. The packet resend rate is also 100% in proposed technique.

Table- IV: Overall Results

Parameters	MAC SDP DoS algorithm	Proposed technique
Number of TCP packets received by legitimate user	68.6%	100%
Packet flow rate	12.2792 packets per second	14.779 packets per second
Down time	33.31 sec	32.92 sec
Recovery time	32.11 sec	31.21 sec
Packet loss	16.08%	0.69%
Packet resend by AP	0%	100%

VII. CONCLUSIONS AND FUTURE WORK

Due to unencrypted management frames and lack of authentication mechanism, WLANs are susceptible to de-authentication DoS attack which can completely disconnect users from the network. In this paper, de-authentication attack is performed on WLAN and the proposed technique is implemented to improve the existing results to detect and prevent the de-authentication attack. The results show that the packet flow rate is increased and packet loss is reduced in the proposed technique. The down time and recovery time in proposed technique is reduced slightly as compared to MAC SDP DoS algorithm. The results show that proposed technique gives better result in terms of calculated parameters as compared to MAC SDP DoS algorithm.

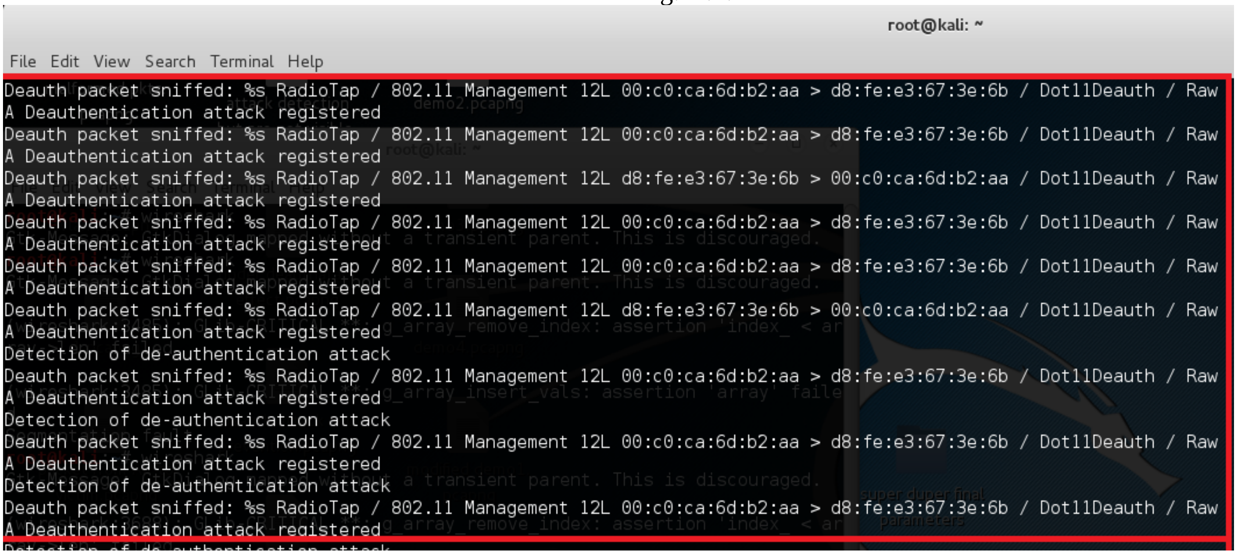
In this research work, the de-authentication attack is performed on single legitimate user. In future, the de-authentication attack can be performed on multiple legitimate users. In future, this technique can be modified to reduce the delay.

Retrieval Number: J12170881019/19©BEIESP
DOI: 10.35940/ijitee.J1217.0881019
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

APPENDIX

The detection of de-authentication attack at AP is shown in Fig. A.1.



```
Deauth packet sniffed: %s RadioTap / 802.11 Management 12L 00:c0:ca:6d:b2:aa > d8:fe:e3:67:3e:6b / Dot11Deauth / Raw
A Deauthentication attack registered
Deauth packet sniffed: %s RadioTap / 802.11 Management 12L 00:c0:ca:6d:b2:aa > d8:fe:e3:67:3e:6b / Dot11Deauth / Raw
A Deauthentication attack registered
Deauth packet sniffed: %s RadioTap / 802.11 Management 12L d8:fe:e3:67:3e:6b > 00:c0:ca:6d:b2:aa / Dot11Deauth / Raw
A Deauthentication attack registered
Deauth packet sniffed: %s RadioTap / 802.11 Management 12L 00:c0:ca:6d:b2:aa > d8:fe:e3:67:3e:6b / Dot11Deauth / Raw
A Deauthentication attack registered
Deauth packet sniffed: %s RadioTap / 802.11 Management 12L 00:c0:ca:6d:b2:aa > d8:fe:e3:67:3e:6b / Dot11Deauth / Raw
A Deauthentication attack registered
Deauth packet sniffed: %s RadioTap / 802.11 Management 12L d8:fe:e3:67:3e:6b > 00:c0:ca:6d:b2:aa / Dot11Deauth / Raw
A Deauthentication attack registered
Detection of de-authentication attack
Deauth packet sniffed: %s RadioTap / 802.11 Management 12L 00:c0:ca:6d:b2:aa > d8:fe:e3:67:3e:6b / Dot11Deauth / Raw
A Deauthentication attack registered
Detection of de-authentication attack
Deauth packet sniffed: %s RadioTap / 802.11 Management 12L 00:c0:ca:6d:b2:aa > d8:fe:e3:67:3e:6b / Dot11Deauth / Raw
A Deauthentication attack registered
Detection of de-authentication attack
Deauth packet sniffed: %s RadioTap / 802.11 Management 12L 00:c0:ca:6d:b2:aa > d8:fe:e3:67:3e:6b / Dot11Deauth / Raw
A Deauthentication attack registered
Detection of de-authentication attack
```

Fig. A.1. Detection of de-authentication attack at AP

REFERENCES

1. K. Pelechrinis, M. Iliofotou, and S. V Krishnamurthy, “Denial of Service Attacks in Wireless Networks : The Case of Jammers,” *IEEE Commun. Surv. TUTORIALS*, vol. 13, no. 2, pp. 245–257, 2011.
2. J. L. Feng and G. Gong, “WiFi-based Location Services Attack on Dual-band Hardware,” in *International Symposium on Electromagnetic Compatibility (EMC)*, 2014, pp. 155–158.
3. T. Farooq, D. Llewellyn-jones, and M. Merabti, “MAC Layer DoS Attacks in IEEE 802 . 11 Networks,” in *The 11th Annual Conference on the Convergence of Telecommunications, Networking and Broadcasting (PGNet 2010)*, 2010, pp. 1–7.
4. Netscout, “Eye on Networks - Wireless Security Series Part I: Deauthentication Attacks.” [Online]. Available: http://enterprise.netscout.com/content/eyeonnetworks-wlan-security-and-analysis. [Accessed: 10-Feb-2016].
5. J. Bellardo and S. Savage, “802.11 Denial-of-Service Attacks: Real Vulnerabilities and Practical Solutions,” in *Proceedings of the 12th conference on USENIX Security Symposium*, 2003, pp. 1–13.
6. J. Wright, “Detecting Wireless LAN MAC Address Spoofing,” *Cisco Certif. Netw. Assoc.*, pp. 1–20, 2003.
7. K. Bicakci and Y. Uzunay, “Pushing the Limits of Address Based Authentication : How to Avoid MAC Address Spoofing in Wireless LANs,” *Int. J. Electr. Comput. Energ. Electron. Commun. Eng.*, vol. 2, no. 6, pp. 1092–1101, 2008.
8. T. N. Nguyen, B. N. Tran, and D. H. M. Nguyen, “A Lightweight Solution for Wireless LAN: Letter-Envelop Protocol,” in *Communications and Networking*, 2008, pp. 1–5.
9. R. Cheema, D. Bansal, and S. Sofat, “Deauthentication / Disassociation Attack : Implementation and Security in Wireless Mesh Networks,” *Int. J. Comput. Appl.*, vol. 23, no. 7, pp. 7–15, 2011.
10. L. Wang and B. Srinivasan, “Analysis and Improvements over DoS Attacks against IEEE 802.11i Standard,” in *Second International Conference on Networks Security, Wireless Communications and Trusted Computing*, 2010, pp. 109–113.
11. L. Arockiam and B. Vani, “Medium Access Control Spoof Detection and Prevention Algorithm (MAC SDP DoS) for Spoofing Attacks in WLAN,” *Int. J. Comput. Sci. Inf. Technol. Secur.*, vol. 3, no. 2, pp. 165–171, 2013.
12. R. Agrawal, T. Imieliński, and A. Swami, “Mining association rules between sets of items in large databases,” in *International conference on Management of data*, 1993, pp. 207–216.
13. M. N. Shahidan and H. I. Mohammed, “An Automated Approach to Detect Deauthentication and Disassociation Dos Attacks on Wireless 802.11 Networks,” *Int. J. Comput. Sci. Issues*, vol. 12, no. 4, pp. 10–112, 2015.
14. M. A. C. Aung and K. P. Thant, “Detection and Mitigation of Wireless Link Layer Attacks,” in *International Conference on Software Engineering Research, Management and Applications*, 2017, pp. 173–178.
15. A. Arora, “Preventing wireless deauthentication attacks over 802.11 Networks,” *arXiv Prepr. arXiv1901.07301*, pp. 1–11, 2018.
16. V. Kumkar, A. Tiwari, P. Tiwari, A. Gupta, and S. Shrawne, “Vulnerabilities of Wireless Security protocols (WEP and WPA2),” *Int. J. Adv. Res. Comput. Eng. Technol.*, vol. 1, no. 2, pp. 34–38, 2012.
17. “Wireshark,” 2019. [Online]. Available: https://www.wireshark.org/. [Accessed: 24-Jun-2019].
18. P. Biondi, “About Scapy,” 2019. [Online]. Available: https://scapy.readthedocs.io/en/latest/introduction.html. [Accessed: 20-Jun-2019]

AUTHORS PROFILE

Shweta Sharma received B.Tech degree in Information Technology from Himachal Pradesh University, Shimla, India, in 2013 and M.Tech degree in Computer Science (specialization in Cyber Security) from Central University of Punjab, Bathinda, India, in 2016. She is pursuing PhD from the Department of Computer Science & Engineering at National Institute of Technical Teachers Training & Research (NITTTR) Chandigarh. Her research interests include Cyber Security, Malware Detection, and Machine Learning.

Meenakshi Mittal has done her Master of Engineering in Computer Science Engineering from Punjab Engineering College University of Technology, Chandigarh, India. She is currently working as an Assistant Professor in the Department of Computer Science and Technology, Central University of Punjab, Bathinda since 2011. Her main research work focuses on Information Security, Computer Networks, and IoT. She has more than 7 years of teaching experience.