Detection of wireless fake access points

Norbert Lovinger
The Faculty of Electrical Engineering
Brno University of Technology
Brno, Czech Republic
xlovin00@vutbr.cz

Tomas Gerlich
The Faculty of Electrical Engineering
Brno University of Technology
Brno, Czech Republic
tomas.gerlich@vutbr.cz

Zdenek Martinasek
The Faculty of Electrical Engineering
Brno University of Technology
Brno, Czech Republic
martinasek@feec.vutbr.cz

Lukas Malina
The Faculty of Electrical Engineering
Brno University of Technology
Brno, Czech Republic
malina@feec.vutbr.cz

***Abstract*—Wireless networks have become a common part of everyone's life who wants to connect to the Internet network. Despite the rapid development of wireless technologies, common users still do not pay much attention to the security questions. One of the most challenging security problems of Wi-Fi networks is to detect and prevent the fake access point attack. Existing solutions require the installation of proprietary hardware, protocol changes, measuring frame characteristics etc. Moreover, these solutions are strictly focused on an individual cyber attack detection. In this work, we propose our method that detects a fake access point and that allows the detection of multiple cyber attacks in local wireless networks. In our research, we focus on the detection of the fake access point attack, the KARMA attack, the frequency congestion attack and the de-authentication attack.**

***Index Terms*—Rogue Access Point, Fake Access Point, Intrusion Detection System, Evil Twin Attack**

I. Introduction

Nowadays, wireless networks allow users with various end-nodes to connect to the Internet and digital services. Wireless fidelity (Wi-Fi) is considered as one of the most popular wireless network technology. Wi-Fi has been adopted in many platforms such as smartphones, laptops, smart watches, and application used in the Internet of Things (IoT) [1]. Despite the rapid development of wireless technologies, common users still do not pay much attention to the security questions. Nowadays, users become accustomed to being online all the time with their smartphones, and they practically do not turn off the connection to a wireless network. Such behavior naturally increases the risk of a cyber-attack realization. This fact is based on a wireless communication channel that is used for transmitting and receiving the Wi-Fi signal between a users and an access point (AP) [2]. Therefore, an adversary does not require a physical access to the target network. The practical realization of the cyber attacks is much easier in Wi-Fi networks, and adversaries can eavesdrop, sniff, interrupt or re-transmit the frames.

Research described in this paper was financed by the Ministry of Interior under grant VI20192022149.

A. State of the art

The following text describes the most commonly used attacks in wireless Wi-Fi networks. During the **deauthentication attack**, an adversary is trying to break an existing connection between a user (victim) and AP by sending the deauthentication frame(s). Afterwards, an adversary can store the traffic of an authentication process because the victim is automatically re-authenticated to the access point, it presents the first result of the attack. An adversary can abuse the traffic stored in order to break security mechanisms that are based on shared secret.

The second impact of the deauthentication attack is coercion of the victim to connect to the fake access point that is created by an adversary. In this scenarios, the Man in the Middle (MitM) attack mostly follows. The Denial of Service (DoS) attack is the third impact of the deauthentication attack. In this case, an adversary is jamming the network through the deauthentication flood. The detection of the deauthentication attack is possible by the monitoring and subsequent analysis of network traffic, special tools must be utilized. Nevertheless, the attack is one of the most effective attacks in wireless networks. The article [3] describes the algorithm, that detects deauthentication attacks using own signatures created.

Generally, the goal of the **Denial of Service attacks** is to deplete the network or computing resources of a target device to denial of the target service for legitimate users. Methods for detecting and mitigating these attacks exist and work reliably, however their success depends on the real attack (strength, duration, type, combination etc.). A typical example of such attack in wireless network is a frequency congestion attack [4] or any kind of flooding attacks [5].

In wireless networks, one of the mitigation mechanism is to detect a malicious device that performing a DoS attack and disconnect the device from the network. Detection methods are based on monitoring several metrics at access points, often these parameters are signal strength, noise and the amount of data transmitted for an individual user. A signature-based

978-1-7281-9281-9/20/$31.00 ©2020 IEEE

detection that is the simplest way to mitigate DoS attacks is described in more details in [4], [6].

The best known active attack on network communication between two parties is **Man in the Middle attack**. The attacker's goal is to wedge between the user and AP communication without their knowledge. The attacker represents the transparent proxy in this scenario and can read or modify data transmitted. There are several variants of the MitM attack and they differ in the exploitation of application protocol's vulnerabilities (DNS spoof [7] or SSLstrip [8]). The detection of the MitM attack is not an easy task in wireless networks. The detection has to take the advantage of neighbouring access points information that are analyzed by the detection system in real time. In [9], [10], the detection of the man in the middle attack is based on comparing the strength of the transmitted signal and calculating the reliability of transmission of individual packets. The well-known implementation called WiFiHop is described in [11]. The detection mechanism is based on analyzing the watermarks of packets that a user sends to a legitimate access point.

In many cases, users store the Wi-Fi setting in their device and the device is automatically connected to the network if the network is within the range. The device does not know if the network is available, therefore, the device sends regularly probe frames to verify network availability. **KARMA attack** is based on this active scanning of the wireless network [12]. The attacker can capture these probe frames and replies the user a probe response that the desire network is within the range. As a result, the device is connected to the fake access point created by attacker. This attack is applicable only in networks with open authentication systems.

Currently, one of the most dangerous attack is called **Fake access point** [13]. The attacker's goal is to entice users to connect to the fake access point created by attacker. In fact, the fake access point is an unauthorized wireless device that masquerades as an legitimate access point. A typical feature is the same configuration with a nearby legitimate access point. The attacker broadcasts the same network SSID (Service Set Identifier) with stronger transmitting power. The detection of fake access points is not trivial. The authors of the article [14] modify the transmission of the access point by adding additional timestamps, which are monitored for detection. A method based on monitoring the DNS (Domain Name System) responses is described in [15]. In [13], the security threats for using the fake AP are analyzed and discussed. The experimental results show using the fake AP attack with DoS and MitM attack. No mitigation mechanism was proposed in the work. In [16], a model to detect fake APs is presented. In fact, this research proposes an algorithm that analyzes the interval, serial number, and timestamp of beacons frames. In the paper [17], a method for detecting the replacement of access points (APs) is described. The method is based on passive remote fingerprinting a physical device. Above described methods require the protocol modification because the special timestamps are utilized to mark Beacon frames. In these methods, the probability of false positive alarms is much higher due the time synchronization problems (restart AP). Moreover, the white list access control is applied that needs interaction of user. In the paper [18], an IoT-based approach is proposed to detect and prevent fake Access point. A single board computer and a wireless antenna are used as a detection system. To prevent fake AP, the detected the media access control address (MAC) of the fake AP has been assigned to an unauthorized Virtual Local Area Network (VLAN). The detection is based on simple comparison of Basic Service Set Identifiers (BSSID) for network with identical SSID, parameters are used. An attacker can easily obtain the BSSID and also can change it for the fake AP (tool `macchanger`). The work [19] proposes a simple method of locating the fake AP with a smartphone. Due to the prevalence of the smartphone, it is used to directly locate the fake AP.

B. Contribution

One of the most challenging security problems of Wi-Fi networks is to detect fake access points. The attack is called also as the rogue access point attack or the evil twin attack. In order to detect the presence of evil twin, existing solutions require the installation of proprietary hardware [1], protocol changes [14], [20], measuring frame characteristics [13], [16], [17] etc. Above described methods require the modification of the protocol because special timestamps are utilized or white list access control is applied that needs the interaction of a user. In these methods, the probability of false positive alarms is much higher due to time synchronization problems (restart AP). These methods are highly dependable on synchronized time among all nodes. Restart of AP cause de-synchronization of time that will increase false positive alarms. Furthermore, above mentioned works are strictly focused on one cyber attack detection. In this work, we have proposed a method that detects the fake access point and allows the detection of multiple cyber attacks in local wireless networks. In our research, we focus on the detection of the fake access point attack, the KARMA attack, the frequency congestion attack and the deauthentication attack. The final implementation of the detecting algorithm is realized in the Python language that allows easy portability and affordability. We utilize Raspberry Pi 4 as the hardware platform for our experimental tests.

II. Method proposal and implementation

Prior to proposing our method, we have analyzed the basic features of freely available Introduction Detection/Prevention Systems (IDS/IPS) such as Suricata [21] and Kismet [22]. Naturally, Suricata cannot be used in Wi-Fi networks because it does not work with intercepted communication at the link layer. Suricata provides a sophisticated logging system including the customizable detection mechanisms. On the other hand, Kismet offers a wide range of possibilities for monitoring wireless networks including the link layer. Nevertheless, advanced attack detection mechanisms are missing and a logging system is not so sophisticated. Based on the knowledge obtained during the IDS/IPS exploration, we propose a novel method that fulfill our two specific requirements. The

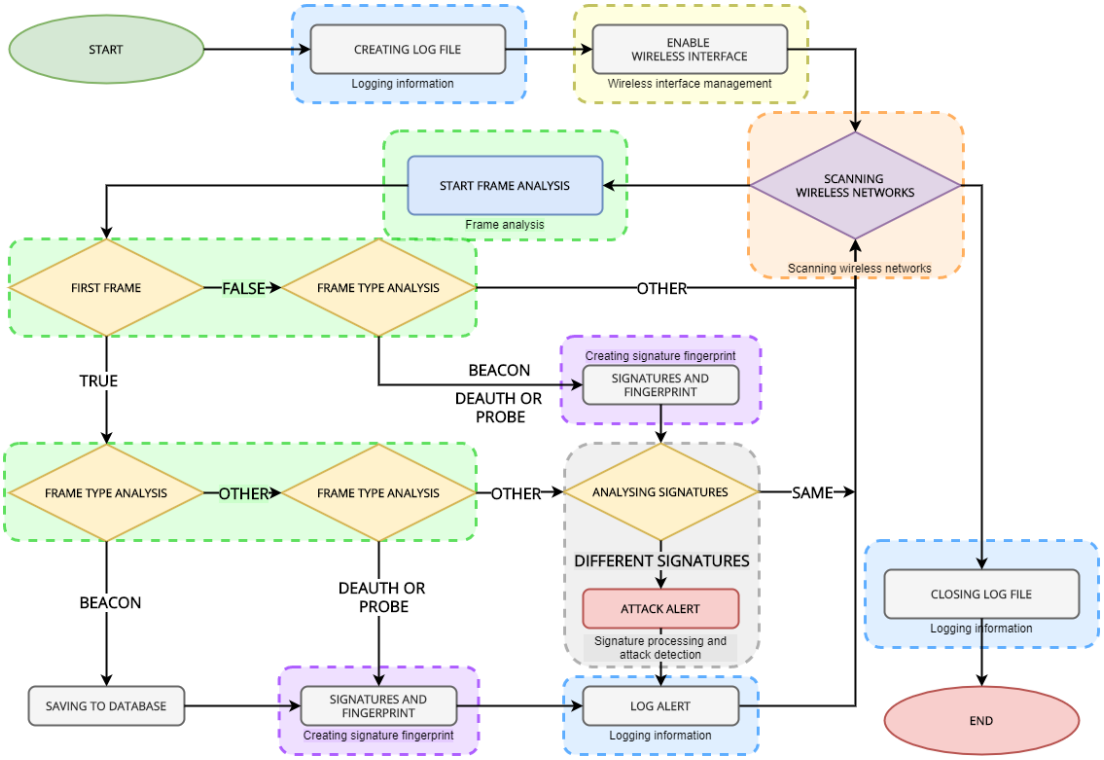

Fig. 1: Diagram of proposed detection method.

sophisticated logging system (Suricata) and analyzing wireless networks on the link layer (Kismet).

Our proposal is depicted in Fig. 1. The detection algorithm consists of several individuals modules that are independent and allow further development. In the following text, we describe implemented modules, the modules are labeled with colored frame in Fig. 1:

- **Wireless interface management** - operations for the wireless card to enable the monitoring mode and regular change of frequencies (yellow frame).
- **Scanning wireless networks** - passive capturing of transmitted frames from wireless networks within the range (orange frame).
- **Frame Analysis** - filtering captured frames in order to create signatures in the next step (green frame).
- **Create signature and fingerprints** - based on the signature created, the unique fingerprints are stored into the database (purple frame).
- **Signature processing and attack detection** - a detection algorithm for comparing the signatures of captured frames, based on the result obtained the alert is created (grey frame).
- **Logging information** - presenting important information to the console and writing it to a log file (blue frame).

The **Python 3.8** programming language was chosen for the final implementation of the detection system that should run on ARM (Advanced RISC Machine) device, e.g., Raspberry Pi 4. The chosen language has main advantages such as speed, portability between devices and low demands on computing requirements. **Individual Modules** were implemented sequentially. The first and most important module was the Wireless interface management that deals with the control of connected wireless interfaces. The main purpose of the module is to select network interface, switch to the monitoring mode and test interface functionality. The second goal is to control the channel changing during the traffic capturing. After the network interface inspection, the program starts the capture of network traffic at the link layer.

The Frame analysis module sorts the interesting intercepted communication according to the content into Beacon frames, Probe frames and Deauthentication frames. Selected signatures of Beacon frame are stored in local database and unique fingerprint is created utilizing a 256-bit hash function. The most important signatures are used as inputs to the hash function. The signatures are the following parameters **SSID, BSSID, Channel, Security, Country and basic bit rates**. The parameters are simply concatenated as string values and are inserted as the input of the hash function.

The second most important module of the program is **Signature processing and attack detection**. The module is analysing the traffic by comparing the current signatures with with signatures stored in the database. In other words, the algorithm is able to evaluate the captured signatures in real time and detected a potential cyber attack. In Fig. 1 if no previous signature is available than new one is saved to the database. As a result, the alert and log entry are created. Event logs have different priorities depending on the type of an attack. A user can terminate the program and the log file is

stored in the folder for future analysis.

III. Experimental Verification of Implementation

We created an experimental workplace in order to test our method implementation. The workplace represent a model of a ordinary wireless local area network (WLAN). Fig. 2 depicts the real form of the workplace created and the following list contains the main devices installed.

- **Legitimate access point Mikrotik hAP ac**[2] establishes a wireless network on the frequency of 2.4 GHz and default SSID and BSSID.
- **The Raspberry Pi 4** represents the detector with the Kali Linux operating system where our method was implemented. If an attack is detected, the detector creates an alert to the console and store the information into the log file.
- **Attacker** represents the notebook with the Kali Linux operation system, the external Wi-fi card is plug-in. The external network card supports monitoring mode and frame injection, therefore, the cyber attacks aimed at wireless networks could be realized.
- **Legitimate user** is a notebook running Windows 10 operating system, user is automatically connect to the AP wireless network utilizing the build in wireless network adapter.

Fig. 2: Real form of the workplace created.

We realized together four scenarios to test detection of our method implementation. The scenarios put into practice the following cyber attacks:

1) **the fake access point attack**,
2) **the KARMA attack**,
3) **the frequency congestion attack**,
4) **the deauthentication attack**.

A. Scenario: Fake Access Point Attack

The scenario with the fake access point attack is depicted in Fig. 3. A legitimate access point broadcasts standard Beacons frames to the legitimate user. These Beacons frames are also captured by the detector and the signatures with fingerprint is stored into the database. The attacker stored the signatures of AP utilizing the `airodump-ng` and `airbase-ng`. In the

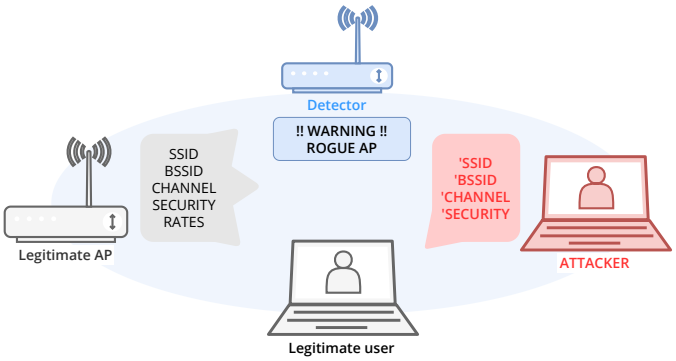

Fig. 3: Scenario with Fake access point attack.

next step, the attacker creates a fake access point with the same SSID, BSSID, broadcast channel and security setting. After the cyber attack realization, the detector system immediately detected the transmission of the same beacon frames with the signatures of the legitimate access point. The implemented algorithm detected that transmitted signatures do not match with the signatures stored in the database. The alert is created as the result that is depicted in Fig. 4.

```
WARNING: ROGUE AP DETECTION! [c4:ad:34:03:24:dd]
WARNING: ROGUE AP DETECTION! [c4:ad:34:03:24:dd] Different channel: 7≠4
WARNING: ROGUE AP DETECTION! [c4:ad:34:03:24:dd] Different rates: [2, 4, 11, 22, 140, 18, 24, 36]≠[2, 4, 11,22]
```

Fig. 4: Alert created to detect Fake access point attack.

B. Scenario: KARMA Attack

The scenario with the KARMA attack is depicted in Fig. 5. A legitimate user broadcasts standard Probe requests targeting the legitimate access point that is out of range in this case (user's home access point with SSID workWiFi). This scenario simulates the case when the user is at a company environment and forgets to turn off the Wi-Fi. These Probe requests are captured by the detector and the signatures are stored into the database. In fact, it is a non-existing network in the company environment. The attacker captured also the Probe requests and reply to the user with Probe response in that impersonates the user's access point. The detector captured and analyzes the Probe response frames and because the wireless network is not in the database, the KARMA attack alert is created. The alert created is depicted in Fig. 6.

C. Scenario: Frequency Congestion Attack

The scenario with the frequency congestion attack is depicted in Fig. 7.

The attack was created using `mdk3` tool that begins broadcasting a high number of fake Beacon frames on the same frequency as legitimate access point. The detector captures a large number of new Beacon frames in a short time period after the attack was activated. Moreover, the SSID of fake wireless networks also contained illegal characters generated by the attacker's tool. Based on these facts obtained from analysis, the detector alerts about the frequency congestion attack. The created alert is depicted in Fig. 8.

D. Scenario: Deauthentication Attack

The scenario with the deauthentication attack is depicted in Fig. 9. This scenario assumes that a legitimate access point broadcasts standard Beacons frames and the user is connected to the access point to browse Internet. These Beacons frames are also captured by the detector and the signatures with fingerprint is stored into the database. The attacker sniffs the communication and with the `aireplay-ng` tool launches the deauthentication attack that is aimed on all connected users. The detector captured and analyzes a large number of deauthentication frames transmitted from the same physical address in a short period of time. The alert is created as the result that is depicted in Fig. 10.

IV. Conclusion

In this paper, we proposed and implemented our novel method which enable us to detect together four cyber attacks in WiFi networks. Our method allows the detection of multiple cyber attacks in local wireless networks. The implementation was written in Python for a low-cost network probe based on Raspberry Pi 4. During our experiments, we chose one of the most challenging attack, i.e., the fake access point attack, and also the KARMA attack, the frequency congestion attack and the deauthentication attack. We verified the functionality of our method and our implementation at the experimental workplace. The detector successfully detected all attacks during all scenarios realized in our experiments. The final implementation offers easy portability and affordability.

References

[1] M. Agarwal, S. Biswas, and S. Nandi, "An efficient scheme to detect evil twin rogue access point attack in 802.11 wi-fi networks," *International Journal of Wireless Information Networks*, vol. 25, no. 2, pp. 130–145, 2018, ISSN: 1068-9605. DOI: 10.1007/s10776-018-0396-1. [Online]. Available: http://link.springer.com/10.1007/s10776-018-0396-1.

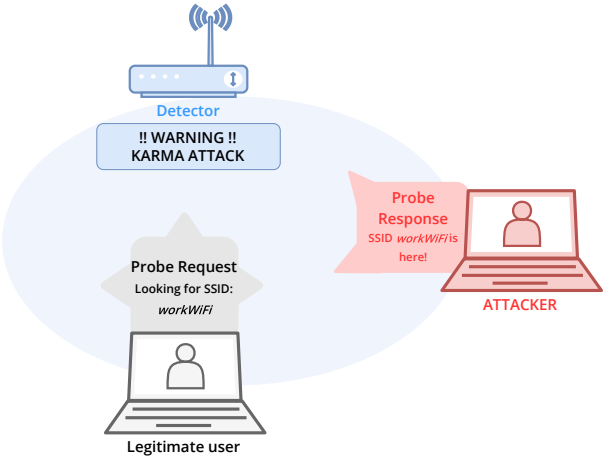

Fig. 5: Scenario with the KARMA attack.

```
Probe req STA: 04:d6:aa:11:ea:1e sending to AP: ff:ff:ff:ff:ff:ff SSID: default
Probe resp AP: 84:16:f9:19:81:6d sending to STA: 84:d6:aa:11:ea:1e SSID: default
WARNING: KARMA ATTACK DETECTION! [84:16:f9:19:81:6d]
WARNING: KARMA ATTACK DETECTION! [84:16:f9:19:81:6d] STA: 04:d6:aa:11:ea:1e SSID: default
```

Fig. 6: Alert created to detect the KARMA attack.

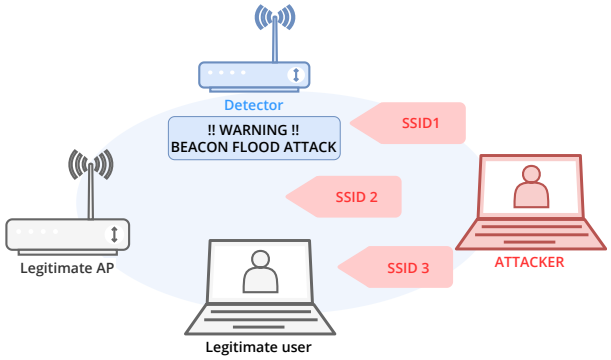

Fig. 7: Scenario with DoS frequency congestion attack.

```
WARNING: BEACON FLOODING ATTACK DETECTION! [a0:b9:22:47:7b:3e]
WARNING: BEACON FLOODING ATTACK DETECTION! [a0:b9:22:47:7b:3e] SSID: I(8?YZ$)0.h4]1w*H#GAz CHANNEL: 4 CRYPTO: OPN
```

Fig. 8: Alert created to detect the DoS frequency congestion attack.

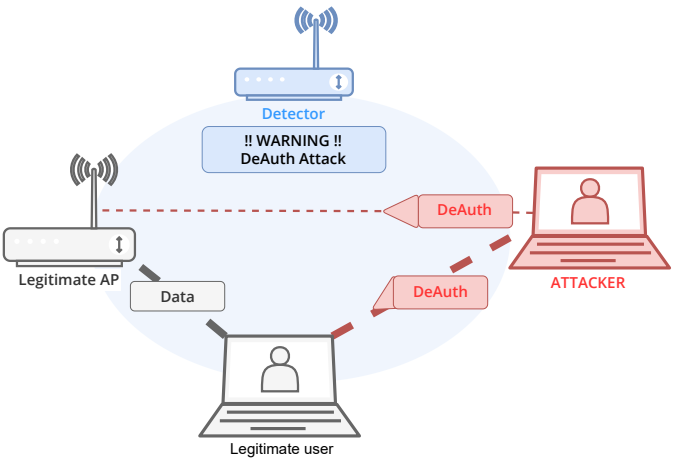

Fig. 9: Scenario with the deauthentication attack.

```
WARNING: DEAUTH ATTACK DETECTION! [c4:ad:34:03:24:dd]
WARNING: DEAUTH ATTACK DETECTION! [c4:ad:34:03:24:dd]: from: ff:ff:ff:ff:ff:ff reason: Class 3 Frame received
 from nonassociated STA
```

Fig. 10: Alert created to detect the deauthentication attack.

[2] S.-L. Wang, J. Wang, C. Feng, Z.-P. Pan, T. Gong, T. Yang, and J. Xu, "Wireless network penetration testing and security auditing," *ITM Web of Conferences*, vol. 7, 2016, ISSN: 2271-2097. DOI: 10.1051/itmconf/20160703001. [Online]. Available: http://www.itm-conferences.org/10.1051/itmconf/20160703001.

[3] M. A. C. Aung and K. P. Thant, "Detection and mitigation of wireless link layer attacks," *2017 IEEE 15th International Conference on Software Engineering Research, Management and Applications (SERA)*,

pp. 173–178, 2017. DOI: 10.1109/SERA.2017.7965725. [Online]. Available: http://ieeexplore.ieee.org/document/7965725/.

[4] M. Salem, A. Sarhan, and M. Abu-Bakr, "A dos attack intrusion detection and inhibition technique for wireless computer networks," in *The International Congress for global Science and Technology-CSIR*, vol. 7, 2007, pp. 17–24.

[5] S. M. Hussain and G. R. Beigh, "Impact of ddos attack (udp flooding) on queuing models," in *2013 4th International Conference on Computer and Communication Technology (ICCCT)*, 2013, pp. 210–216.

[6] C. Benzaïd, A. Boulgheraif, F. Z. Dahmane, A. Al-Nemrat, and K. Zeraoulia, "Intelligent detection of mac spoofing attack in 802.11 network," *Proceedings of the 17th International Conference on Distributed Computing and Networking - ICDCN '16*, pp. 1–5, 2016. DOI: 10.1145/2833312.2850446. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2833312.2850446.

[7] F. Callegati, W. Cerroni, and M. Ramilli, "Man-in-the-middle attack to the https protocol," *IEEE Security Privacy*, vol. 7, no. 1, pp. 78–81, 2009.

[8] Y. L. Zhang and G. S. Xia, "The ssl mimt attack with dns spoofing," in *Applied Scientific Research and Engineering Developments for Industry*, ser. Applied Mechanics and Materials, vol. 385, Trans Tech Publications Ltd, Nov. 2013, pp. 1647–1650. DOI: 10.4028/www.scientific.net/AMM.385-386.1647.

[9] L. Wang, "Detection of man-in-the-middle attacks using physical layer wireless security techniques," Thesis of Master's Degree, WORCESTER POLYTECHNIC INSTITUTE, WORCESTER POLYTECHNIC INSTITUTE, 2013. [Online]. Available: https://web.wpi.edu/Pubs/ETD/Available/etd-082713-125108/unrestricted/thesis.pdf (visited on 07/31/2020).

[10] X. Wei and C. Tang, "Location consistency-based mitm attack detection in 802.11 ad networks," in *International Symposium on Cyberspace Safety and Security*, Springer, 2019, pp. 18–29.

[11] D. Mónica and C. Ribeiro, "Wifihop - mitigating the evil twin attack through multi-hop detection," *Computer Security – ESORICS 2011*, pp. 21–39, 2011. DOI: 10.1007/978-3-642-23822-2_2. [Online]. Available: http://link.springer.com/10.1007/978-3-642-23822-2_2.

[12] L. Oliveira, D. Schneider, J. De Souza, and W. Shen, "Mobile device detection through wifi probe request analysis," *IEEE Access*, vol. 7, pp. 98 579–98 588, 2019.

[13] A. M. Alsahlany, A. R. Almusawy, and Z. H. Alfatlawy, "Risk analysis of a fake access point attack against wi-fi network," *International Journal of Scientific & Engineering Research*, vol. 9, pp. 322–326, 2018.

[14] S. Jadhav, S. Vanjale, and P. Mane, "Illegal access point detection using clock skews method in wireless lan," *2014 International Conference on Computing for Sustainable Global Development (INDIACom)*, pp. 724–729, 2014. DOI: 10.1109/IndiaCom.2014.6828057. [Online]. Available: http://ieeexplore.ieee.org/document/6828057/.

[15] H. Han, B. Sheng, C. C. Tan, Q. Li, and S. Lu, "A timing-based scheme for rogue ap detection," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 11, pp. 1912–1925, 2011, ISSN: 1045-9219. DOI: 10.1109/TPDS.2011.125. [Online]. Available: http://ieeexplore.ieee.org/document/6007016/.

[16] K. F. Kao, W. C. Chen, J. C. Chang, and H. T. Chu, "An accurate fake access point detection method based on deviation of beacon time interval," *2014 IEEE Eighth International Conference on Software Security and Reliability-Companion*, pp. 1–2, 2014. DOI: 10.1109/SERE-C.2014.13. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6901631.

[17] F. Lanze, A. Panchenko, B. Braatz, and T. Engel, "Letting the puss in boots sweat," *Proceedings of the 9th ACM symposium on Information, computer and communications security - ASIA CCS '14*, pp. 3–14, 2014. DOI: 10.1145/2590296.2590333. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2590296.2590333.

[18] İ. F. KILINÇER, F. ERTAM, and A. ŞENGÜR, "Automated fake access point attack detection and prevention system with iot devices," *Balkan Journal of Electrical and Computer Engineering*, ISSN: 2147-284X. DOI: 10.17694/bajece.634104. [Online]. Available: https://dergipark.org.tr/tr/doi/10.17694/bajece.634104.

[19] B. Xu, M. Peng, Q. F. Zhou, and X. Cheng, "Fake access point localization based on optimal reference points," *2018 IEEE 4th International Conference on Computer and Communications (ICCC)*, pp. 784–788, 2018. DOI: 10.1109/CompComm.2018.8780768. [Online]. Available: https://ieeexplore.ieee.org/document/8780768/.

[20] P. K. Dubey and J. N. Verma, *Method and apparatus for detecting a rogue access point in a communication network*, US Patent 8,549,634, Oct. 2013.

[21] D. Day and B. Burns, "A performance analysis of snort and suricata network intrusion detection and prevention engines," in *Fifth international conference on digital society, Gosier, Guadeloupe*, 2011, pp. 187–192.

[22] M. Korcák, J. Lámer, and F. Jakab, "Intrusion prevention/intrusion detection system (ips/ids) for wifi networks," *International Journal of Computer Networks & Communications*, vol. 6, no. 4, p. 77, 2014.