Detection Technique and Mitigation Against a Phishing Attack

Haytham Tarek Mohammed Fetooh[1]
Information Security Prog. Faculty of Computers and Information
Mansoura University, Mansoura, Egypt

M. M. EL-GAYAR[2]
Dept. of Information Technology, Faculty of Computers,
and Information, Mansoura University, Mansoura, Egypt

A. Aboelfetouh[3]
Dept. of Information Systems, Faculty of Computers and
Information, Mansoura University, Mansoura, Egypt

***Abstract*—Wireless networking is a main part of our daily life during these days, each one wants to be connected. Nevertheless, the massive progress in the Wi-Fi trends and technologies leads most people to give no attention to the security issues. Also detecting a fake access point is a hard security issue over the wireless network. All the currently used methods are either in need of hardware installation, changing the protocol or needs analyzing frames. Moreover, these solutions mainly focus on a single digital attack identification. In this paper, we proposed an admin side way of detection of a not real access point. That works on multiple cyber-attacks especially the phishing attack. We shed the light on detecting WI-phishing or Evil Twin, DE authentication attack, KARMA attack, advanced WI-phishing attack and differentiate them from the normal packets. By performing the frame type analysis in real time and analyzing different static and dynamic parameters as any change in the static features will be considered as an evil twin attack. Also, providing that the value of the dynamic parameters surpasses the threshold, it reflects Evil Twin. The detector has been tested experimentally and it reflects average accuracy of 94.40%, 87.08% average precision and an average specificity of 96.39% for the five types of attack.**

Keywords—Rogue access point; phishing attacks; KARMA attack; social engineering; hacking

I. Introduction and Background

Currently, the wireless techniques help users who are using terminals, phones, and tablets to use the internet services, in addition to being integrated in many interfaces and used implementation over the field of (IOT) Internet of Things. [1] Despite the growth of wi-fi technologies, users still do not care for security issues. As clients used to be online most of the time, this gives a higher availability of being victimized with many of the cyber security attacks. All these communications are done over the channel used for sending or receiving wireless waves in-between the access point (AP) and the user. Because of that, the attacker is in no need to physically access the victim's network. He can easily sniff, eavesdrop, resend frames using off the shelf tools [2]. While getting benefits from this technology, these vast numbers of non-smart connected cyber-physical devices have several properties that led to critical security issues, such as nodes mobility, wireless communications, lack of local security features, scale, and diversity. IoT botnets attack is considered a critical attack that affects IoT network infrastructure that launches a distributed denial of service (DDoS) [3].

Phishing is the attempt to acquire sensitive data or to inspire somebody to react in a desired method by simulating as a trusted one in the electronic atmosphere. As demanding a user to tap on a connection in an email or to give his Mastercard numbers or enter definite data as first, last name, address, age, and city. At that point, the hacker can access and use the data. Phishing assaults can be performed over different specialized strategies [2].

A. Impact on the Community and Motivation

Damage from cybercrime is expected to cost the world $6 trillion annually by 2021, raised from $3 trillion in 2015 according to Cybersecurity Ventures [3]. Phishing attacks are the most common type of cybersecurity breaches as stated by the official statistics from the cybersecurity breaches survey 2020 in the United Kingdom [4].

As Phishing attacks merge social psychology, technical systems, and security subjects. These attacks affect organizations and individuals alike, the loss for the organizations is significant as it includes the recovery cost, reputation loss, and productivity reduction [5].

According to a study named Proofpoint 2020, [6] around 90% of organizations suffered targeted phishing attacks during 2019. From which 88% experienced spear-phishing attacks, 83% faced voice phishing (Vishing), 86% dealt with social media attacks, 84% reported SMS/text phishing (SMishing), and 81% reported malicious USB drops.

According to the Phishing Activity Trends Report 1st Quarter 2021 [7] from the Anti-Phishing Working Group , APWG's records, with an unprecedented 245,771 attacks in one month which confirms that phishing attacks are on the rise.

This proofs that phishing attacks are in continuous raise in recent years and have become more sophisticated and gained more attention from cyber researchers. So that, this paper aims to contribute to solving a type of phishing attack which serves in solving the phishing problem and mitigate its impact over the community.

Therefore, the research problem is to address the limitation of the previous studies and security scheme that may offer attack detection but fails to offer it in real time over the network. The problem's solution arises whereas there is an increasing evolution of network devices as well as smart appliance for WI-FI services. Hence, this acts as motivating factor towards working for improving the security of networks and connections as addressed in this research.

The aim of this study is to improve the detection of the attack and contribute to solving the problem of the phishing attack by present a solution that is not costly and in real-time in addition to achieving the best performance with high accuracy and the decreasing the cost. The main aim is to reduce the spread of this attack, improving the detection rate, improving accuracy, decreasing the false alarms, and decreasing the cost of the proposed method.

Therefore, this study addresses the following research questions:

- Research Question 1 (RQ1) How to enable administrators to detect WI phishing in real time without using a special or expensive hardware?
- Research Question 2 (RQ2) How to find a reliable forensic way that visualize the different attacks underway?

It is worth noting that in this research WI-phishing is referred to as Evil Twin (ET) or Rouge Access Point (RAP). As WI-phishing [2] or Evil Twin, DE authentication attack [8] , and KARMA attack [9] are considered types of phishing, we focus on detecting these types. WI-phishing [2] or ET is a procedure of phishing that uses a wireless network where the phisher is in between the client and illegitimate wireless AP, using a Rouge access point as in Fig. 1.

WI phishing is one of the most dangerous and severing attacks [10] that deceives the user to join a rogue access point (RAP) instead of Legitimate Access Point (LAP), while RAP is a malicious device used by an adversary as if it is a real AP.

In which the intruder always copies the same configurations of one or more nearby LAPs to broadcast the same Service Set Identifier (SSID) and always with even stronger transmitting power.

The DE authentication attack is when the attacker tries to sniff or break the connection between the victim and an AP by flooding the network with DE authentication frames to force the client to reauthenticate. Then, the attacker can save traffic during the authentication process and this step is the base of attack's phase one. The attacker decrypts the pre-shared secret to have the secret key and bypass security encryption. The second step of the DE authentication attack is to force the client to connect to a RAP to sniff the whole communication which needs special tools to be detected. While RAP based on DE authentication is perhaps the most well-known assaults in Wi-Fi networks [11] as shown in Fig. 2.

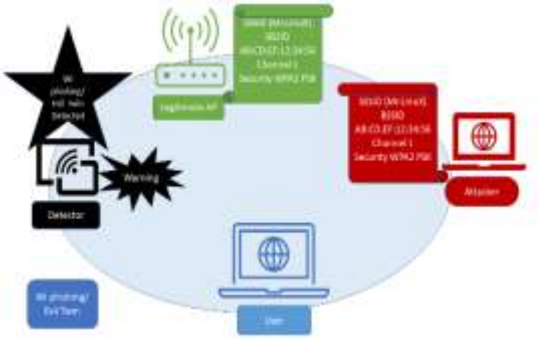


Fig. 1. Wi-phishing / Evil Twin Attack Detection.

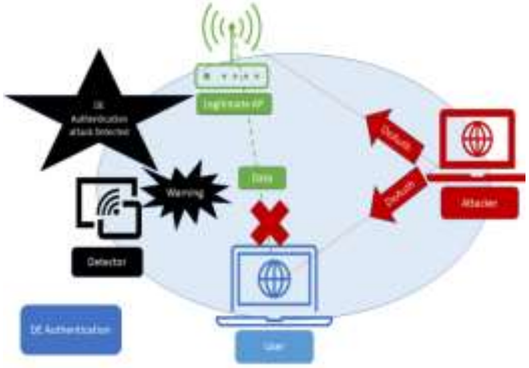


Fig. 2. De Authentication Attack Detection.

Karma attack depends on actively scanning the WLAN [12] to collect the probe frames requests from users' devices and then generate a corresponding probe response as if the required WLAN network is nearby. As the enormous growth of digital era more and more, many humans keep their own Wi-Fi settings in their device as it is for the device to automatically be connected to their known network if the network is in the nearby, devices send probe requests in probe frames to verify the existence of a network as the device does not know physically whether the network is in range or not. Resulting the targeted device is connected to the RAP which is made as a trap by the adversary.

Karma attacks can in any case influence customers that are utilizing active probing authentication. Also to perform it, the intruder can utilize a Pineapple AP 5 [13] that makes the assault a lot less difficult to achieve [9]. As shown in Fig. 3.

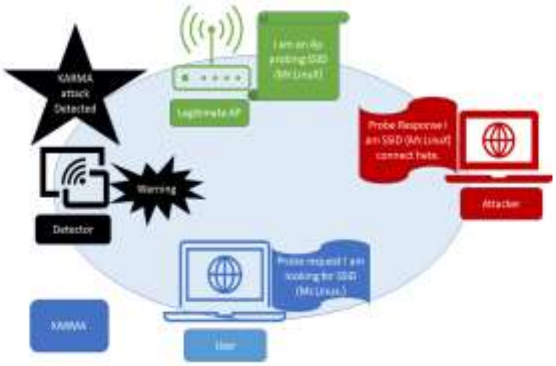

KARMA attack Detected
I am an Ap probing SSID (Mi Linux)
Legitimate AP
Probe Response I am SSID (Mi Linux) connect here.
Warning
Detector
Attacker
Probe request I am looking for SSID (Mi Linux)
KARMA
User

Fig. 3. KARMA Attack Detection.

B. Detecting Evil Twin Solutions

Detecting RAP solutions are categorized into two sets. The administrator-based set is a preliminary one that is based on observing the radio signals, and needs to be utilized on switches, servers, routers, or special devices. It may rely on the technique of whitelisting. From its name, it is deployed by the administrator of the network. But it has its downsides as most of it does not work in real time, needs protocol modification, or depends on a single point of failure as a server.

The second set of solutions is the client-based one. It is used by the user. Using the connection of TCP, Clock Skew, route option, IP packet header and data frame statistics are its main ways. But it has its con as it may needs a predefined data about the network [14].

For these reasons, such a study became more needed as it presents the following: Diverse detection logic to distinguish different kinds of attack signature. An empirical implementation of the proposed scheme for detecting WI-phishing [2]or ET, DE authentication attack , KARMA attack [9], advanced WI-phishing attack and the normal packets are prototyped under real attack. Recommending a better detection method by providing a real time admin side detection via specific parameters in the beacon frames as a signature of being under attack and then giving an alert. Furthermore, to send all beacon frames to a database for further processing and giving alert in case of any anomaly in the features of the previous packets. The final implementation is realized in the Python language using a Scapy [15] library to detect and classify frames.

Many studies have been published to address the problem of detecting phishing or WI Phishing attacks over the network, such as “Detecting fake access point or Evil Twin” as a relevant approach presented in [16] by Lovinger et al. , it depends on the network probing using Raspberry Pi 4. Capability, the author creates a logging system. Analyzing the wireless networks and scanning it, then filtering captured frames to create signatures and store it into the database. Upon the result, the alert is created after comparison. As for the challenges in this approach, it depends on a Raspberry Pi 4 as a device for detection which is a higher cost and limitations, while we developed a cross platform code with less cost. It depends on creating a unique fingerprint, he depends on a 256-bit hash function, which reflects more time, and he uses only static parameters while we depend on static and dynamic parameters for detecting the ET.

He depends on a log file which is not stable and harder to retrieve data rather than the visualized database used by our side. As a competitive advantage, we use all data from the database for further forensic purposes as it can store data for months which reflects stability and durability. Both of us work in real time detection. The other parts of the research are organized as follows: the related work is in Section 2; Section 3 describes the proposed method and the prototype. In Section 4, we present the experimental results. Then Section 5 contains the conclusion and the future work.

II. Related Work

A. Currently used Strategies for Detecting Evil Twin Attack are categorized in Five Groups as follows

1) Monitoring Wi-Fi traffic approach and limitation: Most wi-fi solutions attempt to spread sniffers over the network to accumulate fundamental data measurements such as MAC, SSID, working channel, RSS ID etc. The information accumulated deploying these sniffers enable the administrator to perceive the ET.

Kao et al. [17] detect the existence of ET by keeping a whitelist of the trusted MAC of Laps. Sniffers screen the remote traffic consistently, when an AP is discovered, whose MAC is not indexed in the whitelist, it is alerted as Evil Twin with accuracy of 96.4% average. As for the limitations of this approach, if the attacker sniffs the MAC address of the LAP, the approach has nothing to do in this case in addition of whitelisting only the MAC address disregarding other attributes that we have mentioned in our work.

Sachin et al. presented a way of identifying the malicious access point by setting up a whitelist of authentic APs' MAC address and IP. At that point send a broadcast packet over a central service to uncover the evil twin over the WLAN by receiving all replies from all access points and contrast them against the pre prepared whitelist. Yet, its downside is that it works over wireless networks when all terminals are in a similar range. It does not support the detection in real time as well as relying upon the whitelisting techniques [18] [19].

Sriram et al. [20]. and Chirumamilla et al [21] presented an agent-based intrusion detection system (IDS) to reveal RAP by screening the networks for the existence of new Aps and if these access points are not recorded in the pre-approved records, they will be flagged as RAP. As a downside in these methods, if an evil twin AP has a similar SSID and MAC, both these procedures are not as powerful as an insidious twin's MAC similar to the MAC of the approved AP. It depends on a server which is a single point of failure, and this method is useful only if the RAP is connected directly to the LAN and if the attacker has its own internet, the approach has nothing to do with it.

2) Timing based methods and feature extraction approach and limitation: In the frame analysis mechanism, the system aggregates all the frames using the mirror port of a core switch or by analyzing the frames obtained from the remote sensors distributed over the network. Utilizing the data from the gathered frames, many features are extracted to gain vital information regarding the existence of evil twin. The evil twin access point structures a scaffold between the real one and the customer to give Internet features. Due to crossing over an extra bounce, timing is put together and works with respect to the additional deferral happening because of the extra hop. This additional deferral gives proof to discovery of the evil twin.

Diogo M´onica et al. introduced an approach to distinguish a multi-hop evil twin via a real time detection device utilized by the client. It is not exposed to idleness or bandwidth with no need for a pre-prepared list. Using this way, channels are being scanned in less than 500ms. But it has constraints as a real access point can be distinguished as a fake one.it distinguishes evil twin in 30 seconds which is a major issue.[22].

Burns et al. in his paper [23] founded out the method of traceroute bidirectionally . To make a traceroute between a terminal and a remote server from terminal-to-server and then from server-back-to-terminal then to compare former both sides traceroute results. By comparing the number of hops both ways, if deference is found between the number of hops both ways that would be considered as an indicator of Evil Twin attack. The drawback of this method is that it has a single point of failure as it depends on an external server and this method is completely useless if the intruder uses his/her own internet connection like 4G.

Han et al. [24] have presented a technique that calculate the Round Trip Time (RTT) of a DNS query, while Mano et al. [25] utilizes a local RTT metric in addition to a frame payload slicing method to detect RAP too. As for the limitations of these solutions is that the RTT is affected by the congestion of the network. If there is a repeater in the network, this method is useless. In addition of depending on a server as it is an extra cost and considered as a single point of failure.

Yimin et al. [26] used the inter arrival time (IAT) to figure out the extra delay resulted by ET. Yet, the mentioned schemes fail when ET gives its own private connection causing the delay resulted because of the extra hop. It depends on training data to run the main algorithm of the detector and it is not easy for each admin to extract.

Fingerprinting a physical device remotely and passively in [27] is presented by F.Lanze et al. to mitigate the RAP but, it required a white listing with user interaction and a protocol modification for a spatial timestamp are utilized to mark beacon frames, resulting, the increase of false positive alarms probability as a result of the time synchronization problems, using only 50 observations for training, it detects RAP in 90% of all cases but it still depends on training data.

Based on the work of Kohno et al. in [27], Jana et al. [28] have presented the clock skew method to distinguish unapproved APs existence over the network. By calculating the clock skews of different APs using the IEEE 802.11-time synchronization function (TSF), as timestamps is a part of beacon frames. If the clock skew for a device does not equal the one kept previously, it is flagged as evil twin AP.

These solutions have many limitations as:

- It causes a higher load on the core switch because of the additional burden of feature processing.
- In case, an intruder uses his own Internet connectivity, the traffic doesn't arrive at the Centre switch, leaving the assault not detected.
- a spoofed response can be sent to the user to keep away from the time difference that may result from the extra hop.
- The approach results in a lot of false positives in case the frames are queued by a busy router which causes an additional delay.

3) Proprietary hardware approach and limitation: Pradip et al. [29] used a probing device that sends a pre-detection message to all connected users advising them to ignore the probe request. Afterwards, it sends another probe request and mark all responding APs as ET.

Eman et al. [30] used a chipset to detect the evil twin deauthentication attack depending on analyzing the packet frames especially the management frame named DE authentication frames.

These solutions have many limitations as:

- Ignoring the 802.11 probe request is a violation of the 802.11 standards [31].
- If attacker ignores reacting to the probe request to stay covered up and make, the method not useful.
- Special hardware means higher cost.

4) Signature-based and anomaly-based IDS's approach and limitation: It means using a database that contains the known intrusion patterns or signatures to be used for detection but in case of a new pattern or signature, it will never be detected. While an anomaly-based IDS creates profile for a host or network in the normal situation depending on statistics. It can recognize both known and unknown attacks. Both mentioned types lead to a large number of false positives. A survey of many anomaly based IDSs is mentioned in [32], It uses honeypot and anomaly analysis for making an IDS [2], it consists of filtering, IDS, and honeypot as the traffic after passing filtering and IDS. They rerouted all attacks to a honeypot for in-depth investigation, with False positive rate using anomaly detection system with Specificity of 0,62 and False Positive rate of 38% and it is a limitation.

5) RAP-based DE authentication/disassociation attack's approaches and limitations: **N**o single method recognizes all ET types. The practical technique is the one that identifies a wide range of RAP, needs no adjustment in protocol nor a determined equipment. All current methods have at least one of these highlights, yet none of them has all the features. As it is quite hard to detect Evil Twin, S. Jadhav et al. in [33] have modified the transmission protocol by additional timestamps, which are being observed for detection but protocol modification in itself is a limitation.

A. M. Alsahlany et al. in [34] have presented a good discussion and analysis for the security threats of RAP and its results shows that the RAP always comes in conjunction with DoS and MitM attacks in an experimental way but the author didn't provide any mitigation mechanism.

İ.F.KILINÇER et al. in [35] have presented a RAP mitigation method, an IoT-based approach depending on. A single board computer and a wireless antenna make a RAP detection system by detecting their media access control address (MAC) of the RAP which is assigned to an unauthorized (VLAN) Virtual Local Area Network. The detection mechanism depends on making a comparison between MAC and Basic Service Set Identifiers (BSSID) with identical SSID lists.

These solutions have many limitations as:

- The attacker can overcome the mitigation using an open-source tool (like mac changer) to obtain the LAP BSSID. Benefiting from the propagation of smartphones they used a simple approach to locate the RAP.
- The detection is based on simple comparison of BSSID for networks with identical SSID parameters. An attacker can easily obtain the BSSID and change it for the fake AP (tool mac changer).

B. *Existing Solutions Summary and Limitations*

So, the current techniques have many downsides: (1) Deployment is with high cost. (2) Protocol adjustment is needed. (3) In need of specified equipment or mainly works on a server or a special hardware. (4) Patching customer software. (5) Based on whitelisting, Result in a large number of false positives. These mentioned points are the reasons for choosing the proposed method which forms a need for addressing such kind of problems.

Therefore, this paper proposes an admin-side solution that defeats the issues related with the current strategies and distinguishes the RAP with higher accuracy and detection rate. It detects WI-phishing [2]or Evil Twin, DE authentication attack, KARMA attack [9], advanced WI-phishing attack and differentiate them from the normal packets. As a kind of phishing attack, based on performing the frame type analysis. In contrast with the previously mentioned solutions, our method of detection has many pros: (a) It needs no whitelisting technique of access points. (b) It provides real-time detection. (c) It does not need any training data of the targeted wireless network. (d) It does not depend on a remote server nor any hardware (e) No need for protocol modification. (f) Determine the attacker's MAC. (g) It needs no prior connection with any AP for detection.

III. Proposed Method

This section clarifies the utilized indicators for the proposed module via analyzing beacon frames and extracting features that are considered as a sign of attack. It helps in detecting WI-phishing [2] or Evil Twin, DE authentication attack , KARMA attack [9], advanced WI-phishing attack and differentiate them from the normal packets in real time and making a long-timed database that is used for forensics for detecting more sophisticated beacon-based attacks via a python language and SCAPY library. The research has a significant impact on the community of network system administrators as it will ease the process of detection in real time and forensics of the evil twin attack.

We have previously mentioned that there is a drawback in IEEE 802.11 protocol as the beacon is sent unencrypted, it helps in the occurrence of many attacks such as RAP. While methods given by researchers as some assisted in securing from the evil twin attack however had their own downsides. These methods range from the installation of special hardware [1],protocol modification [33] and measuring frame characteristics [34], [36], etc.

Limitations in the available Intrusion Detection/Prevention Systems such as Suricata [37] which works only on LAN, and Kismet [38] which has no sophisticated logging method as its pcap file cannot be analyzed in real-time in addition to its massive size.

In our approach, the proposed method countered these drawbacks as it does not need any change protocol, does not depend on learning data or expensive monitoring devices. It depends on a native, better, and real time detection method, depending on analyzing, storing, and visualizing sub types under beacon frames in real-time to figure out the anomaly which reflects the existence of RAP. As well as the long-term real time logging database analysis and visualization which gives more capability and elasticity for further forensics and threat analysis. This database based on Elasticsearch [39] and MongoDB [40], it provides a real time chart, detects anomalies, and generate an alert. It can even send it to the administrator by email. Our method implementation is realized using the language of Python -which enables cross platform implementation- that allows affordability and portability.

A. *Detecting Beacon Frames*

In the initial step, Scapy library is used for packet capturing phase. So, we can divide our detection algorithm into two different sub algorithms.

B. *Real Time Detection Algorithm*

1) Depends on IEEE 802.11 management frame -static parameters- that the attacker can sniff, any change in one or more static parameter would be considered as Evil Twin; These static parameters include BSSID, SSID, Channel, Encryption type, Country code, Supported channels and First Channel. We assume that the administrator knows all attributes of his network which should be assigned by the administrator at the first time.

a) Our algorithm can defend one or more Wi-Fi networks as the administrator can provide one or more network properties. These attributes are always being compared against all properties which are being captured in real time from all surrounding networks.

2) Also, there are dynamic Wi-Fi network's parameters, which are very hard for the attacker to imitate as the timestamp and signal strength of the network.

a) The algorithm depends on the fact that the timestamp increases regularly over time in the Wi-Fi access points that we are defending. This means, if we find another access point with the same static attributes but differ in timestamp as if it is less than or equals the last received timestamp from the access point, which means that the last access point is an evil twin.

b) Also for signal strength (RSSIs), we depend on the paper in [41] which is presented by Vanjale et al. as they stated that if the signal differs by 10 dB greater or less, that indicates an evil twin coexistence. Because the attacker may place his RAP nearby the LAP with stronger signal strength or closer to the target to lure them to use his access point. We considered 10 dB< or > it is considered as evil twin.

3) If there is a coexistence of two BSSIDs, in case of a DE authentication attack that exceeds the threshold, it reflects DE authentication attack which in this case considered as an indicator of evil twin. We have set our threshold in this phase as 10 DE authentication packets as mentioned in [42] by Chibiao Liu et al.

4) For detecting KARMA attack, which is only can be pursued in open WIFI networks. So, in case of new open WI-FI networks existence in the surrounding, it is considered as a KARMA Attack. Nevertheless, it will lead to high false positive rate, but it achieves higher recall.

C. Long-time Database for Logging and Forensics

1) We used a strong database that depends on MongoDB [40] Elasticsearch [39] to overcome the weakness in Kismet , like pcap. captured file analysis. As capturing a large file for many days will be extremely hard to be analyzed by using a normal computer.

2) By using this combination of MongoDB with Elasticsearch, we analyze and visualize the captured data for weeks; and in case of anomaly an alert is generated.

3) This database has many advantages for the administrator as knowing if the network is always receiving DE authentication attack from a particular MAC address, or if there is someone probing the users by using a famous Wi-Fi name which is open like the names for airports or cafes to lure the user to use it which is of course a KARMA attack. It enables the administrator to know about his physical location and if this open Wi-Fi is in the surroundings or not.

4) Using the mentioned database, the administrator can know how long the Wi phishing attack was underway. Whether the network targeted by a script kiddie, with a raspberry-pi, who floods the network with DE authentication frames and whether the DE authentication frames was targeting specific device or department.

D. Classes of the Proposed Algorithm

The proposed algorithm has seven correlated classes as follows:

1) Wireless interface management - enabling the monitoring mode in the wireless card and starting channel hopping over frequencies.

2) Scanning wireless networks - capturing transmitted frames of all the surrounding networks.

3) Frame Analysis and filtering - captured frames to find data frames out of the beacon ones, and to separate beacon frames into DE authentication frames beacons, to be compared against the threshold value, and other types of beacons.

4) Compare other types of beacon frames against the predefined parameters - IF difference found between real time captured frame attributes and the predefined attributes, an alert is generated. These attributes are BSSID, SSID, Channel, Encryption Type, Country Code, Supported Channels, First Channel.

5) Compare beacon frames' timestamp and signal strength- generate alert if the timestamp is not incremental or the difference in signal strength is > or < 10 dB than the previously recorded ones.

6) Listing all surrounding open WI-FI, if any, in case of new open Wi-Fi loomed, it is considered as a KARMA attack.

7) Store in database and start visualization - send all frames to a long-time database for forensics purposes and visualize it for further analysis. In case of anomaly, an alert is generated and then returned to the network scanning step.

For final implementation, Python 3.8 programming language was chosen, Fig. 4 figures out the seven phases of the algorithm of detection as follows.

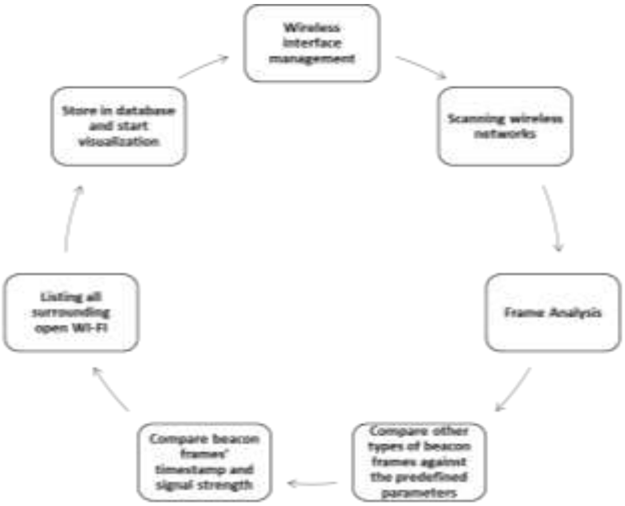

Wireless interface management
Scanning wireless networks
Frame Analysis
Compare other types of beacon frames against the predefined parameters
Compare beacon frames' timestamp and signal strength
Listing all surrounding open WI-FI
Store in database and start visualization

Fig. 4. Phases of the Detection Algorithm.

E. Detector's Pseudo Code

This section shows the proposed algorithm module. As shown in Fig. 5, we start the detection by setting the interface into monitoring mode, then start a continuous channel hopping that goes to each channel and scan it from channel 1 to channel 11. And define static parameters and calculate the dynamic parameters to detect attack in case of a difference occurrence. It also detects De authentication and KARMA attack.

```
➢ Start.
➢ Put interface into monitoring mode.
➢ Channel hop and scan wireless networks.
➢ Parse all captured beacon frames and send to logging database
  for further analysis.
➢ Start real time analyses.
  • Define the static attributes for network includes $MAC_d$, $SSID_d$,
    $Channel_d$, Encryption $Type_d$, Country $Code_d$, Supported
    $Channels_d$ and First $Channel_d$.
  • Define dynamic attributes for network $Timestamp_t$, Signal-
    $strength_t$.
  • Define all surrounding open Wi-Fi.
  • Start Frame analyses.
  ▪ IF more than {SSID [coexists] AND DE authentication packets
    threshold detected}.
    ▪ [Generate alert] "Warning.… Evil Twin detected."
    ▪ Else IF deference found between real time captured frame
      attributes and the pre-defined attributes ({ $MAC_d$ !=MAC} OR
      {$SSID_d$ != SSID} OR {$Channel_d$ != Channel} OR {Encryption
      $Type_d$ != Encryption Type} OR {Country $Code_d$ != Country
      Code} OR {Supported $Channels_d$ != Supported Channels} OR
      {First $Channel_d$ != First Channel})
    ▪ [Generate alert] "Warning.… Evil Twin detected."
    ▪ Else IF difference found in timestamp ({$Timestamp_t$ <=
      $Timestamp_{now}$} OR {Signal-$Strength_{t\,[}$difference >10 dB]
      Signal-$Strength_{NOW\,\}}$.
    ▪ [Generate alert] "Warning.… Advanced Evil Twin detected.
    ▪ Else IF found new Open Wi-Fi network.
    ▪ [Generate alert] "Warning.… KARMA attack detected.
    • AND start database visualization.
    ▪ If found anomaly
    ▪ [Generate alert] "Warning.… Evil Twin detected.".
    ▪ Else return to network scanning
```

Fig. 5. Pseudo Code of the Proposed Method.

F. Flow Chart of the Proposed Method All Captured Beacon

In the following flow chart, Fig. 6, the program starts by putting interface in monitoring mode to scan all the nearby networks by making a channel hopping between channel 1 and 14. Then monitoring scanning and analyzing all beacon frames properties which captured by our network in promiscuous/monitoring mode; Firstly send all captured beacon frames and make a real time comparison between all features that is hard coded from the admin for the needed to be a defended network or networks against the fetched properties from captured beacon frames and in case of matched it generates an alert in real time. Furthermore, the long-term database which can handle, analyze, and visualize features of captured beacon frames to generate valuable statistics to forecast attacks.

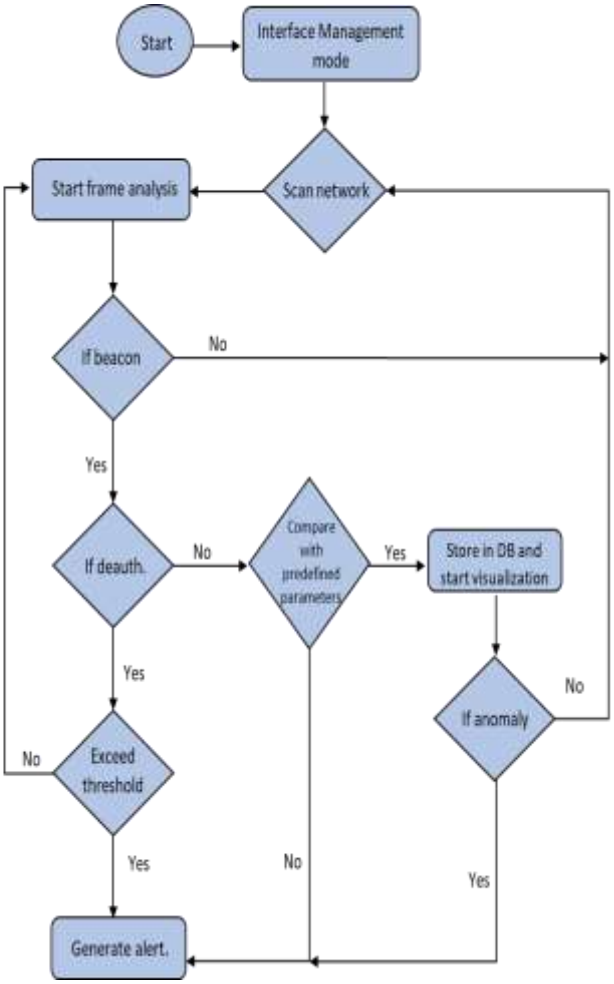


Fig. 6. The Flowchart of the Proposed Method.

IV. RESULTS

This section is determined for the proposed technique's evaluation. In which we describe the laboratory, the design of the detector, the efficiency of the detector, evaluation measures and lastly the results are mapped over a confusion matrix as a predictive analysis method.

A. Laboratory Description

The experiment has been implemented over a network named "MR. Linux" for evaluating the proposed methodology. Python 3.8 programming language, Scapy library and an ALFA model AWUS036H [43] in the promiscuous mode have been used for monitoring and analyzing the packets sent and received. We used airgeddon [44] and Wi-Fi pumpkin [45] to launch the attack, using a Lenovo G4080 Core I5 , 4th generation with 4 gigabyte RAM laptop loaded by Wifislax 2.4 64 bits [46] which is a distribution GNU / Linux. The attacker is launched using a wireless interface card ALFA model AWUS036H. [43] The device that is used for detection is hp EliteBook 745 G3 with AMD64 A10, 8 gigabyte RAM loaded with a Ubuntu 20.4 OS [46] having a Pre-installed Python 3.8, We also use Elasticsearch [39] and MongoDB [40].

B. Proposed Detector's Design

Our proposed algorithm overcomes admin side vulnerabilities in solutions as mentioned in section II as the efficiency of the algorithm does not depend on protocol modification, data sampling, machine learning algorithms, dedicated server, or RTT parameters. As well as it is real-time that does not depend upon training knowledge or Wi-Fi network's fingerprint.

- (KARMA) attack. We simulate a KARMA attack by simulating an open Wi-Fi of a well-known place that is not on the range, if detected in real time.
- (Common Wi-Fi phishing) attack/ replacement WI phishing: We simulate the attack by cloning the BSSID address to be similar to the legitimate AP and flood DE authentication / disassociation frames over all the Wi-Fi network and create another similar fake Wi-Fi network with a different one or more attribute (Encryption type - Channel - First Supported channel or Country Code), it is detected in real time.
- (Common Wi-Fi phishing) attack/ WI phishing coexistence: We simulate the attack by cloning the BSSID to be similar to the legitimate AP and create a coexisting fake Wi-Fi with different one or more attribute (Encryption type - Channel - First Supported channel or Country Code), it is detected in real time.
- (Advanced Wi-Fi phishing) attack with higher signal strength. We simulate the attack by cloning all parameters of the real AP, but with higher signal strength to lure the users to connect to the fake one, it is detected in real time.
- (Advanced Wi-Fi phishing) attack with time difference: We simulate the attack by cloning all parameters of the real AP, but with less timestamp value and based on time difference, it is detected in real time.
- (Real time database is made for more analysis and forensic purposes).
    - To know which of our clients was connected to the Wi-phishing AP, as all beacon frames are logged.
    - Using the mentioned database, we know how long the different attacks were underway.
    - By analysing the Realtime database, we can answer the question, was our network targeted or were DE authentication frames targeting specific devices or departments.

C. Efficiency of Proposed Algorithm

This section is dedicated for evaluating the proposed method. The section is separated into two main parts. The first part evaluates the performance of the proposed detector for classifying the types of attacks on detecting (1) KARMA attack (2) DE authentication attack [9] (3) WI-phishing [2] or Evil Twin, (4) advanced WI-phishing, (5) and differentiating them from the normal packets in real time, furthermore to database all beacon frame for visualization, forensics, and further anomaly inspection. The second part compares the results against the method in [16] proposed by Lovinger et al. , Zeeshan Afzal et al. in [47] and Mayank Agarwal et al. in [1].

D. Evaluation Measures

The performance of the proposed method is analyzed via the estimation of different evaluation metrics like TN rate, TP rate, Specificity Accuracy, false negative rate, and false positive rate, Precision, Recall and F-Measure which are detailed in the subsequent descriptions:

These measures are calculated over a confusion matrix classification as a predictive analysis method based on equation number. 1, 2, 3, 4 and 5. In which TP, FN and FP represent numbers of true positives, false negatives, and false positives, respectively.

1) Specificity: The parameter of specificity is defined as the ratio of total true negatives to the summation of total true negative and false positive value. True negative rate is called specificity.

$$Specificity = \frac{TNs}{TNs + FPs} \qquad (1)$$

2) Accuracy: The accuracy metrics are estimated by the parameters value of specificity and sensitivity, which are expressed by equation number 1. Also accuracy refers to how accurate the proposed method can classify frame types in a correct way, and this is expressed by equation number 2, which is applied to return the accuracy value. The accuracy value expresses a comparison between frames that are correctly classified with the whole frames.

$$Accuracy = \frac{(TP + TN)}{TP + TN + FP + FN} \qquad (2)$$

3) Precision: The value of precision refers to the number of frames or a category frame that is classified correctly divided by the total frames classified of the same type. Precision is calculated by equation 3. And precision is also referred to as positive predictive value; other related measures used in classification include true negative rate and accuracy.

$$Precision = \frac{TP}{TP+FP} \qquad (3)$$

4) Recall: Nevertheless, recall shows how many percent of mentioned attacks are correctly classified by the classification. Equation 4 is used for resulting the value of recall. Recall in this context is also referred to as the true positive rate or sensitivity, and it is defined as the ratio of total true positives to the summation of total false negative and false positive value.

$$Recall = \frac{TP}{TP+FN} \qquad (4)$$

5) F1: is the harmonic average value of precision and recall and calculated by equation number 5.

$$F - score = \frac{(2*Recall*Precision)}{Precision+Recall} \qquad (5)$$

E. Evaluation of the Proposed Detector

For evaluation, a comprehensive analysis is conducted. Results showed the FP, TP, FN, and TN presented and analyzed via a confusion matrix as a predictive analysis method. The used set of data for evaluation is outlined in Table I to make sure that the predicted attacks are the actual ones that were sent by the attacking tool. The calculations done via the confusion matrix, Table II shows results with average accuracy of 94.40%, 87.08% average precision and an average specificity of 96.39%.

Table II shows the classifications performance based on the number of each frame type by classifying each type of attack. While Fig. 7 shows the high value of TN with higher value than the TP which increases the overall algorithm detection accuracy which is reflected in Fig. 8.

F. Testing

We have tested our solution against airgeddon [44] and wifipumpkin3 [45] for launching the attack to run the previously mentioned attacks against a network that we have permission to, and calculate the response. We used the OS Wifislax 2.4 64 bits [46] which is a distribution GNU / Linux. The alert reflects the kind of attack underway as shown in Fig. 9 that shows the real time detection. Fig. 10, 11, 12 and 13 show the detection of different types of attack, Fig. 14 shows a sample of anomaly detection through the database. While Fig. 15 represents the database visualization in real time.

TABLE I. CONFUSION MATRIX OF THE PROPOSED METHOD

	KARMA	DE auth.	WI-phishing	advanced WI-phishing	Normal
KARMA	100	0	0	0	0
DE auth.	0	89	3	5	3
WI-phishing	0	2	90	4	4
advanced WI-phishing	0	2	8	87	3
Normal	16	6	0	9	69

TABLE II. CLASSIFICATION AND RESULTS OF THE PROPOSED DETECTOR

	KARMA	DE auth.	WI-phishing	advanced WI-phishing	Normal
TP	100	89	90	87	69
TN	335	346	345	348	366
FP	16	10	11	18	10
FN	0	11	10	13	31
Accuracy	96.45%	95.39%	95.39%	93.35%	91.39%
precision	86.21%	89.90%	89.11%	82.86%	87.34%
Recall	100.00%	89.00%	90.00%	87.00%	69.00%
specificity	95.44%	97.19%	96.91%	95.08%	97.34%
F-score	92.59%	89.45%	89.55%	84.88%	77.09%

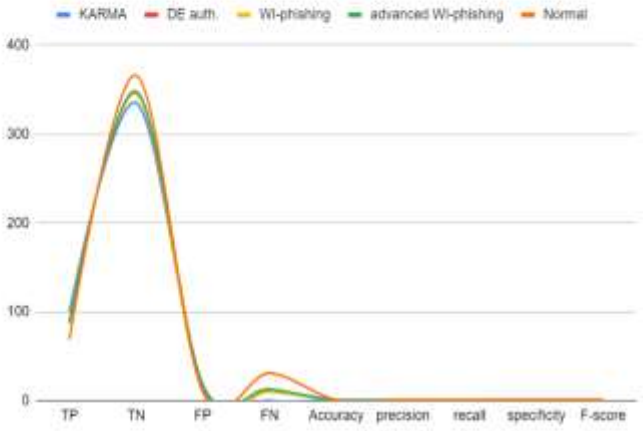


Fig. 7. Matrix Representation Graph.

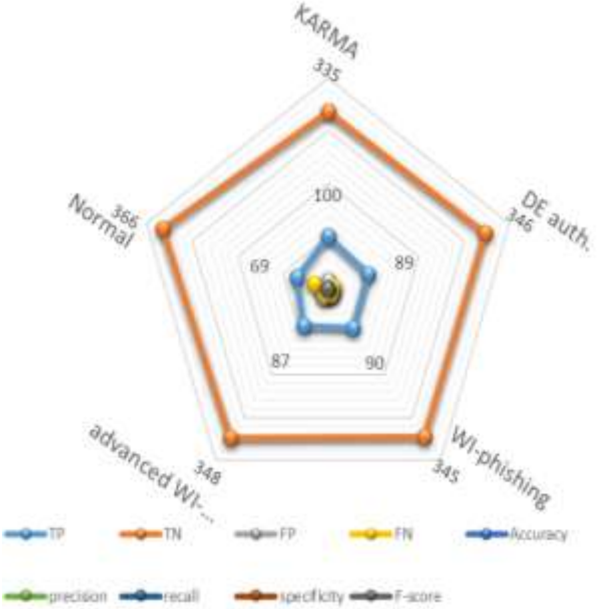


Fig. 8. Algorithm Detection Accuracy.

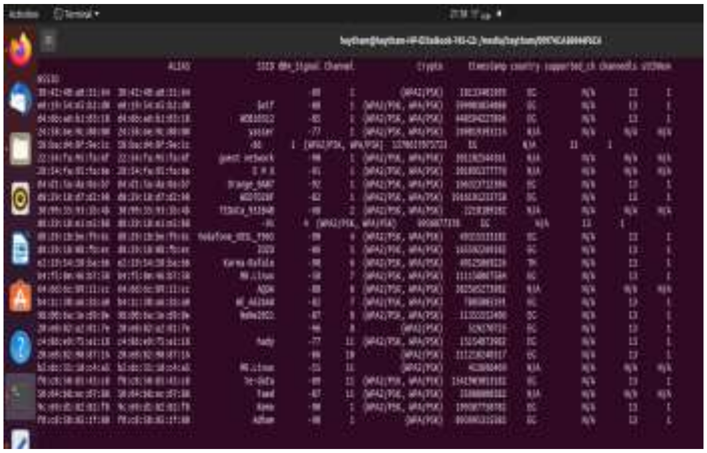
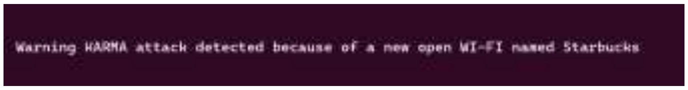
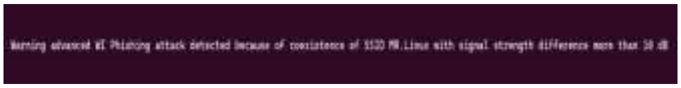


Fig. 9. Real Time Detection.



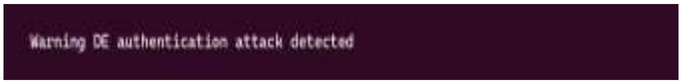
```
Warning KARMA attack detected because of a new open WI-FI named Starbucks
```

Fig. 10. KARMA Attack Detected.



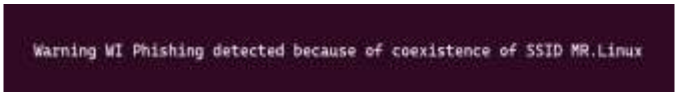
```
Warning advanced WI Phishing attack detected because of coexistence of SSID MR.Linux with signal strength difference more than 10 dB
```

Fig. 11. Advanced WI Phishing Attack Detection.



```
Warning DE authentication attack detected
```

Fig. 12. De Authentication Attack Detected.



```
Warning WI Phishing detected because of coexistence of SSID MR.Linux
```

Fig. 13. WI Phishing Attack Detected.

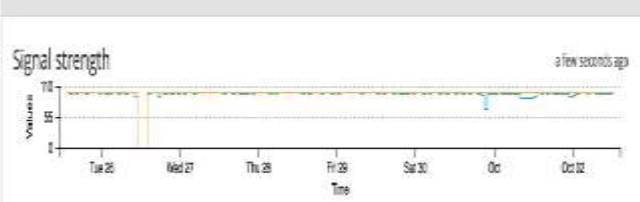


Fig. 14. Anomaly Detection in Signal Strength.

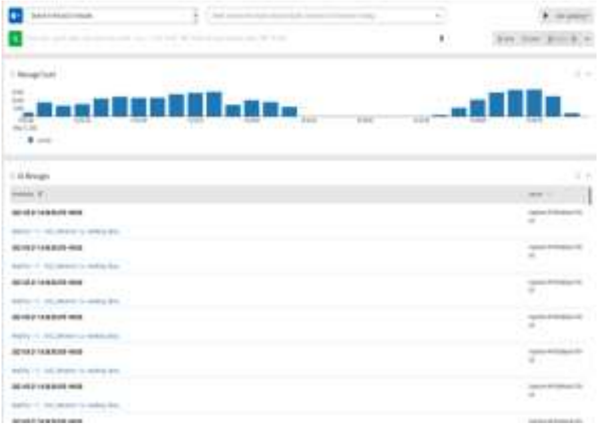


Fig. 15. Real Time Database Visualization.

G. Comparison and Evaluation

Here is a comparison between the proposed method and the method presented in [16] by Lovinger et al. As this approach depends on the network probe using the Raspberry Pi 4. Capability, and creates a logging system. Analyzing the wireless networks and scanning it, then filtering captured frames to create signatures and store it into the database, based on the result obtained the alert is created after comparison. As for the challenges in this approach, it depends on a Raspberry Pi 4 as a device for detection which means more cost and higher limitations, while we developed a cross platform code with less cost. It depends on the log file which is not stable and harder to retrieve data. It depends on creating a unique fingerprint, he depends on a 256-bit hash function, which reflects more time, he uses only static parameters.

While our proposed method depends on static and dynamic parameters for detecting the evil twin, it provides real time detection, it is not passive, but it is active, it is an admin solution, it is cheap, it does not store SSID in DB nor perform bookkeeping of all the APs in the neighborhood. It detects more than one type of attack as WI-phishing or Evil Twin, DE authentication attack, KARMA attack and differentiate them from the normal packets as a kind of phishing attack by depending on Sniffing and analyzing the wireless frames. It has an average accuracy of 94.40%, 87.08% average precision and an average specificity of 96.39% for the five types of attack. The average of the false positive rate is 13 % for all tested types, and it also detects the attacker's MAC address. The results prove that the detector's accuracy is quite high and provide most of the expected features. It also shows that the proposed system can be used for forensic purposes as it can store data for a long time which reflects stability and durability, for data that is stored in the database and start visualization. Table III represents a comparison between the proposed method and three methods for different authors, Levinger et al.[16], Zeeshan Afzal [47] and Mayank Agarwal [1].

To wrap up, we conclude that we have achieved best results after comparing with the previously proposed methods in addition to performing the detection in real time.

TABLE III. A COMPARISON BETWEEN THE PROPOSED METHOD AND THREE OTHER METHODS FOR DIFFERENT AUTHORS

	Lovinger et al.[16]	Zeeshan Afzal et al. [47]	Mayank Agarwal et al. [1]	proposed method
real time detection	y	n	n	y
perform channel hopping	y	n	n	y
using a special device	y	n	n	n
creates a logging system	y	y	y	n
database for forensics	n	n	n	y
analyzing the wireless networks	y	y	y	y
creating network signature	y	y	n	n
based on whitelisting	y	n	n	y
using log file	y	y	y	n
using static parameters	y	-	y	y
using dynamic parameters	n	-	n	y
detecting evil twin	y	y	y	y
detecting KARMA attack	y	n	n	y
detecting DE authentication	y	y	y	y
advanced WI-phishing	n	y	n	y
perform bookkeeping of APs	y	n	y	n
accuracy	-	89%	100%	94.4% av.
false positive rate	-	14.6%.	-	13% av.

V. Conclusion

The wireless network is a primary portion in our world, on account of being used in many life aspects. In this paper, a real time attack detection method has been proposed and helped in detecting different types of wireless attacks as detecting WI-phishing or Evil Twin, DE authentication attack, KARMA attack, advanced WI-phishing attack and differentiate them from the normal packets. While the previously mentioned algorithms of other researchers are either outdated, limited in their detection methods, architecture and/or scope of detection. The implementation was written in Python using the Scapy library by analyzing beacon frames properties in real time and extracting features to be compared against the pre-stored features of LAPs beacon properties and consider any change or a threshold exceeding as a sign of attack. The proposed detector has the advantages of being stable, working in real time, low cost, it does not need extra hardware. It is also powered by a database that can store frames for a long time, which by analyzing them the detector has an added value of forensics, forecasting and detecting anomaly. The detector's efficiency was modelled in a mathematical way and implemented in real life scenarios, returning average accuracy of 94.40%, a value of 87.08% average precision and an average specificity of 96.39% for the different attack scenarios.

VI. Future Work

In the future, we need to analyze our collected data using AI, machine learning and deep learning to generate attack vectors that will help for faster and better detection of the different types of attack. In addition to being willing to deploy the mentioned algorithm for detecting other types of attack rather than the previously mentioned. Also, we can use semantic analysis and ranking technique evaluated in [48] for detecting and ranking other types of attack. We need to make a real time probe request analysis to reduce the value of false positive for the real time detection of the KARMA. Trying not only to detect the attack, but also make a counterattack.

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," Int. J. Wirel. Inf. Networks, vol. 25, no. 2, pp. 130–145, 2018, doi: 10.1007/s10776-018-0396-1.

[2] K. L. Chiew, K. S. C. Yong, and C. L. Tan, "A survey of phishing attacks: Their types, vectors and technical approaches," Expert Syst. Appl., vol. 106, pp. 1–20, 2018, doi: 10.1016/j.eswa.2018.03.050.

[3] S. Morgan, "2019 Official Annual Cybercrime Report," Cybersecurity Ventur., p. 12, 2019.

[4] "Cyber Security Breaches Survey 2020 - GOV.UK." https://www.gov.uk/government/statistics/cyber-security-breaches-survey-2020/cyber-security-breaches-survey-2020 (accessed Sep. 18, 2021).

[5] E. Medvet, E. Kirda, and C. Kruegel, "Visual-similarity-based phishing detection," Proc. 4th Int. Conf. Secur. Priv. Commun. Networks, Secur., 2008, doi: 10.1145/1460877.1460905.

[6] Proofpoint, "2020 State of the Phish," Proofpoint, pp. 1–48, 2020, [Online]. Available: https://www.proofpoint.com/sites/default/files/gtd-pfpt-us-tr-state-of-the-phish-2020.pdf.

[7] Anti-Phishing WoPhishing Activity Trends Report 1 Quarterrking Group, "Phishing Activity Trends Report 1 Quarter," Most, no. March, pp. 1–12, 2021.

[8] A. Arora, "Preventing wireless deauthentication attacks over 802.11 Networks." 2018, [Online]. Available: http://arxiv.org/abs/1901.07301.

[9] R. Gonçalves, M. E. Correia, and P. Brandão, "A flexible framework for rogue access point detection," ICETE 2018 - Proc. 15th Int. Jt. Conf. E-bus. Telecommun., vol. 2, 2018.

[10] S. M. Hussain, "Impact of DDoS Attack (UDP Flooding) on Queuing Models," pp. 210–216, 2013.

[11] M. A. C. Aung and K. P. Thant, "Detection and mitigation of wireless link layer attacks," Proc. - 2017 15th IEEE/ACIS Int. Conf. Softw. Eng. Res. Manag. Appl. SERA 2017, pp. 173–178, 2017, doi: 10.1109/SERA.2017.7965725.

[12] A. A. Al-zubi, "The International Congress for global Science and Technology ICGST International Journal on Computer Network and Special Issue on Network Security Techniques," 2015.

[13] "WiFi Pineapple - Hak5." https://shop.hak5.org/products/wifi-pineapple (accessed Jul. 18, 2021).

[14] C. Benzaïd, A. Boulgheraif, F. Z. Dahmane, A. Al-Nemrat, and K. Zeraoulia, "Intelligent detection of MAC spoofing attack in 802.11 network," ACM Int. Conf. Proceeding Ser., vol. 04-07-Janu, 2016, doi: 10.1145/2833312.2850446.

[15] "Scapy." https://scapy.net/ (accessed Jul. 20, 2021).

[16] N. Lovinger, T. Gerlich, Z. Martinasek, and L. Malina, "Detection of wireless fake access points," Int. Congr. Ultra Mod. Telecommun. Control Syst. Work., vol. 2020-Octob, pp. 113–118, 2020, doi: 10.1109/ICUMT51630.2020.9222455.

[17] K. Kao, T. Yeo, W. Yong, H. C. the 2011 A. S. on, and undefined 2011, "A location-aware rogue AP detection system based on wireless packet sniffing of sensor APs," dl.acm.org, Accessed: Nov. 07, 2019. [Online]. Available: https://dl.acm.org/citation.cfm?id=1982195.

[18] S. R. Sonawane and S. Vanjale, "Wireless LAN Intrusion Prevention System (WLIPS) for Evil Twin Access Points," IJCST - Int. J. Comput. Sci. Technol., vol. 4–8491, no. 2, pp. 2–5, 2013.

[19] Sachin R. Sonawane, Sandeep P. Chavan, and Ajeet A. Ghodeswar, "Study of Different Rogue Access Point Detection and Prevention Techniques in WLAN," Int. J. Adv. Res. Comput. Sci. Softw. Eng., vol. 3, no. 10, pp. 1232–1237, 2013.

[20] V. Sriram, G. S.-2010 I. 2nd, and undefined 2010, "Detecting and eliminating Rogue Access Points in IEEE-802.11 WLAN-a multi-agent sourcing Methodology," ieeexplore.ieee.org, Accessed: Nov. 07, 2019. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/5422999/.

[21] M. K. Chirumamilla and B. Ramamurthy, "Agent based intrusion detection and response system for wireless LANs," IEEE Int. Conf. Commun., vol. 1, pp. 492–496, 2003, doi: 10.1109/icc.2003.1204225.

[22] D. Mónica and C. Ribeiro, "WiFiHop - Mitigating the evil twin attack through multi-hop detection," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2011, vol. 6879 LNCS, pp. 21–39, doi: 10.1007/978-3-642-23822-2_2.

[23] A. Burns, L. Wu, X. Du, and L. Zhu, "A novel traceroute-based detection scheme for Wi-Fi Evil twin attacks," 2017 IEEE Glob. Commun. Conf. GLOBECOM 2017 - Proc., vol. 2018-Janua, pp. 1–6, 2018, doi: 10.1109/GLOCOM.2017.8253957.

[24] H. Han, B. Sheng, C. Tan, Q. Li, and S. Lu, "HAN ET AL.: A TIMING-BASED SCHEME FOR ROGUE AP DETECTION A Timing-Based Scheme for Rogue AP Detection," MobiCom, vol. 11, pp. 104–115, 2018, [Online]. Available: https://pdfs.semanticscholar.org/1dd9/786e51dd4fbe5df185f4a6ae3e1d70113207.pdf.

[25] C. Mano, A. Blaich, Q. Liao, Y. J.-A. T. on, and undefined 2008, "RIPPS: Rogue identifying packet payload slicer detecting unauthorized wireless hosts through network traffic conditioning," dl.acm.org, Accessed: Nov. 07, 2019. [Online]. Available: https://dl.acm.org/citation.cfm?id=1330334.

[26] Y. Song, C. Yang, and G. Gu, "Who is peeping at your passwords at starbucks? - To catch an evil twin access point," 2010, doi: 10.1109/DSN.2010.5544302.

[27] F. Lanze, A. Panchenko, B. Braatz, and T. Engel, "Letting the puss in boots sweat: Detecting fake access points using dependency of clock

skews on temperature," ASIA CCS 2014 - Proc. 9th ACM Symp. Information, Comput. Commun. Secur., pp. 3–14, 2014, doi: 10.1145/2590296.2590333.

[28] S. Jana and S. K. Kasera, "On fast and accurate detection of unauthorized wireless access points using clock skews," IEEE Trans. Mob. Comput., 2010, doi: 10.1109/TMC.2009.145.

[29] "US20120124665A1 - Method and apparatus for detecting a rogue access point in a communication network - Google Patents." https://patents.google.com/patent/US20120124665 (accessed Nov. 04, 2019).

[30] E. A. Metwally, N. A. Haikal, and H. H. Soliman, "Detecting Semantic Social Engineering Attack in the Context of Information Security," pp. 43–65, 2022, doi: 10.1007/978-981-16-2275-5_3.

[31] IEEE Standard for Information Technology—Telecommunications and information exchange between systems—Local and metropolitan area networks—Specific requirements - ANSI/IEEE Std 802.11, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications - p.447 Table 18-9., vol. 1999, no. June. 2007.

[32] N. Agrawal and S. Tapaswi, "The Performance Analysis of Honeypot Based Intrusion Detection System for Wireless Network," Int. J. Wirel. Inf. Networks, vol. 24, no. 1, pp. 14–26, Mar. 2017, doi: 10.1007/s10776-016-0330-3.

[33] S. Jadhav, S. B. Vanjale, and P. B. Mane, "Illegal access point detection using clock skews method in wireless LAN," in 2014 International Conference on Computing for Sustainable Global Development, INDIACom 2014, 2014, pp. 724–729, doi: 10.1109/IndiaCom.2014.6828057.

[34] A. M. Alsahlany, A. R. Almusawy, and Z. H. Alfatlawy, "Risk analysis of a fake access point attack against Wi-Fi network," vol. 9, no. 5, pp. 322–326, 2018.

[35] İ. F. KILINÇER, F. ERTAM, and A. ŞENGÜR, "Automated Fake Access Point Attack Detection and Prevention System with IoT Devices," Balk. J. Electr. Comput. Eng., no. January, 2020, doi: 10.17694/bajece.634104.

[36] K. F. Kao, W. C. Chen, J. C. Chang, and H. Te Chu, "An accurate fake access point detection method based on deviation of beacon time interval," Proc. - 8th Int. Conf. Softw. Secur. Reliab. - Companion, SERE-C 2014, pp. 1–2, 2014, doi: 10.1109/SERE-C.2014.13.

[37] D. Day and B. Burns, "A Performance Analysis of Snort and Suricata Network Intrusion Detection and Prevention Engines," ICDS 2011, Fifth Int. Conf. Digit. Soc., no. c, pp. 187–192, 2011, [Online]. Available: http://www.thinkmind.org/index.php?view=article&articleid=icds_2011_7_40_90007.

[38] M. Kor, J. Lámer, and F. Jakab, "Intrusion Prevention / I Ntrusion Detection System (IPS / IDS) F OR W I F I N ETWORKS," vol. 6, no. 4, pp. 83–95, 2014.

[39] "GitHub - elastic/elasticsearch: Free and Open, Distributed, RESTful Search Engine.".

[40] "The most popular database for modern apps | MongoDB.".

[41] S. Vanjale and P. B. Mane, "A novel approach for elimination of rogue access point in wireless network," 2015, doi: 10.1109/INDICON.2014.7030418.

[42] C. Liu and J. Yu, "Rogue access point based DoS attacks against 802.11 WLANs," Proc. - 4th Adv. Int. Conf. Telecommun. AICT 2008, pp. 271–276, 2008, doi: 10.1109/AICT.2008.54.

[43] "AWUS036NH (EOL) – ALFA Network Inc." https://www.alfa.com.tw/products/awus036nh?variant=36481029374024 (accessed Jan. 23, 2021).

[44] "GitHub - v1s1t0r1sh3r3/airgeddon: This is a multi-use bash script for Linux systems to audit wireless networks." https://github.com/v1s1t0r1sh3r3/airgeddon (accessed Jan. 23, 2021).

[45] "GitHub - P0cL4bs/wifipumpkin3: Powerful framework for rogue access point attack." https://github.com/P0cL4bs/wifipumpkin3 (accessed Jan. 23, 2021).

[46] "Live Wifislax." https://www.wifislax.com/ (accessed Jan. 23, 2021).

[47] Z. Afzal, J. Rossebo, B. Talha, and M. Chowdhury, "A Wireless Intrusion Detection System for 802.11 networks," Proc. 2016 IEEE Int. Conf. Wirel. Commun. Signal Process. Networking, WiSPNET 2016, pp. 828–834, 2016, doi: 10.1109/WiSPNET.2016.7566249.

[48] M. M. El-Gayar, N. E. Mekky, A. Atwan, and H. Soliman, "Enhanced search engine using proposed framework and ranking algorithm based on semantic relations," IEEE Access, vol. 7, pp. 139337–139349, 2019, doi: 10.1109/ACCESS.2019.2941937.