IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

RECOGNITION OF ROGUE ACCESS POINTS USING A MACHINE LEARNING APPROACH

[1]Kashyap C. Patel, [2]Ajaykumar Patel

[1]Department of Computer Science, [2]A.M.Patel Institute of Computer Studies
[1]Ganpat University,
[1] Ganpat University, Gujarat, India

Abstract: The goal is to classify various types of wireless assaults, such as those that take advantage of vulnerable systems or rogue access points. There are a number of techniques that can be used to identify a rogue user at a specific point of entry. Such strategies and methods are quickly categorised into fundamental subfields, such as the client side, the server side, the wired side, the wireless side, the temporal aspects, etc. Every conceivable tactic has both advantages and disadvantages. The goal of this study is to discuss the difficulties and restrictions of existing strategies for detecting Rogue APs. Apply a machine learning (ML) technique to a real-time dataset constructed from these issues. With the use of ML-based methods, rogue APs were detected and analysed.

***Keywords* - Rogue AP, Rogue AP detection methods, ML based techniques, WLAN threat**

I. INTRODUCTION

A Rogue Access Point is an AP that has not been authorised by a network administrator and operates freely within a private or public network. This kind of rogue ap broadcast its activities to the public. This kind of Rogue AP poses a significant risk, and its default configuration mode is the primary cause. In this default configuration state, authentication methods and encryption techniques are often disabled. Because the wireless signals are able to penetrate building walls, glasses, and other obstacles, the Rogue AP (and its malicious wireless signals) is a dangerous threat for various industries and public places. Students and organizational employees deploying rogue access point for unconstrained internet connectivity and unlimited usage is known as soft access point (type of Rogue AP). Attackers can use Rogue AP to intercept data from any network using both active and passive methods of data interception. Data alteration is not possible using the passive manner of interception, but a malicious AP can read the data. For instance, it is feasible to intercept data passing through web applications (such as usernames, passwords, etc.), but it cannot be changed or updated. The process by which Rogue AP actively intercepts users' info of their live actions in cyberspace is referred as "internet footprinting." For instance, in the scenario of active interception, a rogue access point has the potential to reroute and transfer the funds into the scammer's account rather than a legal account of victim. [1] – [9]. Rogue ap connection architecture in WLAN is shown in Figure 1.

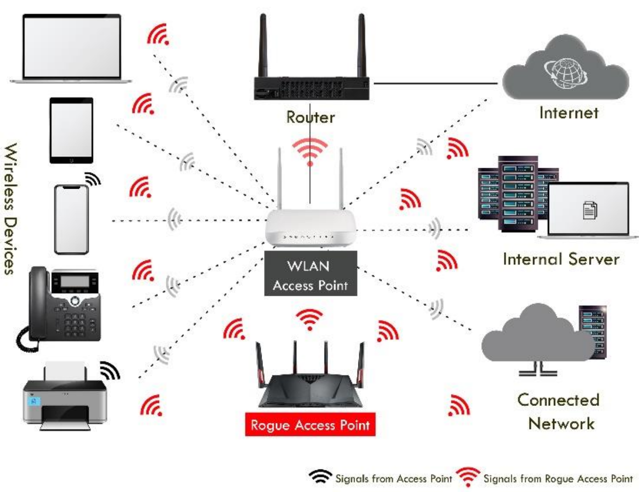

Figure 1: Core architecture of the Rogue Access Point [9]

A Rogue AP is a gadget that is not authorised by an administrator but is still functioning on the genuine network. This AP could have been set up by a legitimate staff member, or it could have been a malicious attempt at gaining unauthorised entry. It's also possible that a nearby business owns the AP.

Hacktivist doctrine, business animosities, boredom, extortion and blackmail, government-authorized cyberwarfare, cybervandalism, and other factors are key motivators for hackers. Through a Rogue AP in boulevard or commercial cable free networks, hackers can carry out Evil-Twin, MITM, Distributed Denial of service (DDoS), vehicular Rogue AP, IoT based Rogue ap, Wi-Fi Deauther, Wi-Fi signal interference, Rogue hotspot, duplicating MAC address, and WLAN spoofing sorts of attacks [2] [22] [14] [45].

II. ROGUE AP DETECTION APPROACH

2.1 Existing Approaches and Limitations of Rogue AP Detection Methodologies

An attacker or intruder in a Wi-Fi network can build up Rogue AP with the same BSSID, and SSID as the legitimate AP, and a wireless user in the system will think it is connecting to the network through the legitimate AP. Researchers, industry experts, and authors from a wide range of fields have all chipped in with strategies for finding and eliminating Rogue AP in WLANs. The most common detection methods used (whether it is based on time or snoop) in research and writing are those based on the following terms and keywords:

Traffic in the network, time interval (time-stamp, roundtrip), characteristics of radio signals, strength of signals, radio frequency, antennas, channels, delay, packet analysis (serial number of packets, time, sequencing etc.), radio signal, beacon, probe, fingerprinting, spoofing of physical address, sniffing techniques, SSL/TCP, gateway, etc [31]. Some methods for detecting Rogue APs are wireless, while others require cable connections, and still, others depend on hardware/software compatibility with various OS and infrastructure settings.

As a security and privacy concern, authors and researchers utilized various methodologies to detect Rogue AP in legitimate systems. Client-side approach (using Traceroute command, ICMP, DNS server), Server side approach [19] [11] [15] [12] [10], Hybrid approach [52] [52] [22] [10], Wired-side approach [22] [10][13], Tool-based approach [22] [10], Fingerprinting methods [19] [38] [14][13] [16] [15] [18] [11], Beacon-Framing [60], Radio Frequency [16], Admin side approach [17] [10], Positioning algorithm that relies on fingerprints [18], physical characteristics of AP like fingerprinting and clock-skew based [19] approach, Temporal features [17], TCP- round trip time [20], ACK pairs arrival time [20], signal strength [36] [24], Hidden Markov Model [30] [31], covert channel [21], RTT measurement [54], Tool-Kismet based detection approach [22], CSI (Channel State Information) [23] etc. are various detection techniques and parameters which are used by researchers, authors and technical experts for Rogue AP. Here I elaborate some challenges and limitations to detect the Rogue AP.

2.1.1 Client-Side Rogue AP detection approach:

Client-side wagering that an adversary will counterfeit gateway credentials to intercept consumer data in transit. The rogue wireless network's faster Internet access will prevent this detection. The intruder can cause the server and wireless user to reply at the same rate as packets travelling through the compromised AP. Due to wireless signal strength and AP network traffic load, the wireless customer and server response time may vary. If the network firewall discards traceroute packets for security, this detection method may fail. By monitoring the wireless data stream, an adversary can avoid traceroute evil-twin (Rogue AP) detection. Traceroute uses the insecure ICMP protocol to monitor the wireless device's travel to the remote server. Attackers can intercept traceroute results delivered to network devices using secure wireless networks. Then, the malicious actor can send the victim these findings via the false wireless network. Thus, both gateways will receive the same route data, enabling Rogue AP-based Evil-Twin detection without alarm [23] [39] [10] [18] [42] [43].

2.1.2 Server-side detection approach:

Wireless servers outperform desktop computers in memory and computing power. After cracking authentication, attackers can establish a Rogue AP or launch DoS and Man-in-the-middle attacks. The consumer need not update drivers, add-ons, or credentials. The AP, Gateway Router, or Switch instals updates and new software. This strategy's main issue is that consumers don't know which APs are trustworthy. The user unwittingly links to the AP it finds while wardriving [10] [11] [12] [15] [19].

2.1.3 Evil-Twin Rogue AP detection approach:

Client-based services find evil twins on users' devices, whereas admin-based solutions analyse RF signals. Researchers say their SSL/TCP-based Evil-Twin detection approach discovers several gateways. Hackers may avoid detection by sending client data through the same authorised gateway. SSL/TCP won't help authors find rogue APs. Client-side activities cannot discriminate between lawful and Rogue AP, which enable Internet connectivity. The client's detection method will fail if Rogue APs are discovered [31] [32] [33].

2.1.4 Delay-based approach:

Some WLAN experts focus on Rogue AP identification time. WLAN's medium susceptibility to interference and conflicts causes latency. This method is inefficient and unstable, especially in frequently trafficked WLANs. The WLAN medium's unpredictable and delay-prone nature, especially during high use, makes timestamped beacon frames, which are created at the AP and include the frame's inter-arrival timing at the client station, unreliable. Delay-oriented detection cannot detect evil-twin attacks. The attacker's gateway may make the Rogue AP's Internet connection speedier [60] [34] [10] [36] [36].

2.1.5 Air-Magnet tool based approach:

Air-Magnet uses wireless sniffing. Sensors round the network. In a distributed agent-server architecture, physical and data link sensors can detect Rogue APs. The Air-Magnetic analyser costs over $3,000, making this method prohibitively expensive [14].

2.1.6 Kismet tool approach:

To now, Kismet has only been able to recognise 802.11 wireless equipment. Since 802.11g is backwards-compatible with 802.11 b, Kismet may well be possible to perceive it, although if you chance to discover a Kismet-compatible 802.11a NIC, you can forget about the use of Kismet to discover some less widespread 802.11a networks [22] [10].

2.1.7 Covert-Channel approach:

Because the covert channel exclusively enables for one-way interaction, the AP can only send a beacon frame to the unit and not the other way around [10] [50] [51].

2.1.8 Distributed Detection Module approach

The Distributed Detection Module monitors and filters Gateway routers. Thus, any offender who sets up the malicious app behind the firewall and accesses the network, especially from the user end, can exploit the vulnerability and reach the server [10].

2.1.9 Channel-based techniques:

The channel-overlapping approach is strong at finding Rogue APs that use neighbouring channels, but it is less effective at recognising those that use the same channel. By fine-tuning throughput deterioration and interference degree, our technique may overcome this drawback [60] [38] [26] [23] [10].

2.1.10 Packet analysis method:

This packet analysis method cannot detect packets that bypass the core switch. A fake AP can use a 3G mobile internet connection. Packets can't cross any switch with port replication enabled and avoid packet analysis detection systems [37][10].

2.1.11 Radio Signal Strength terminology:

RSS-based indoor locating methods are mostly distance-based and fingerprint-based. The former is easier to build but requires the AP's transmit power. Uncertain transmit power complicates Rogue AP deployment. RSS is inaccurate for Rogue AP identification due to multipath and shadowing effects in a diversified wireless architecture. Multipath and shadowing affect RSS in complex indoor spaces. Thus, signal intensities may not necessarily indicate closeness to the malicious AP. Thus, RSS-based Rogue AP localization is imprecise, time-consuming, and maybe futile. AP placement alternatives employ the path loss model. Certain alternative AP localization algorithms use the RSS path loss model, which posits that the signal intensity will be greatest in the region nearest the AP when the LoS path is unobstructed. Multipath and darkness drastically impair RSS in the difficult enclosed region. The empirical studies show that a stable receiver's received signal intensity swings by 5dB in an indoor setting for around one minute, making RSS-based AP localization challenging to achieve optimum accuracy. [23] [39] [10] [18] [51] [52] [42] [43].

2.1.12 Some ML based approaches:

Many academics use ML-based techniques to detect Rogue APs using outdated network assault datasets. So, road map to discover optimal result in limited time with efficiency, I worked on real-time scenario to perform network assault through Rogue AP and generate dataset and apply some methodologies and algorithm of ML and achieve accuracy applying multiple approaches on collecting real-time dataset.

Liu et al. propose an AP confirmation method that uses channel condition data (CSI) to verify the target AP. At the start of online authentication, the AP authentication procedure is based on the CSI and the AP assessment model is trained using XGBoost [44]. The proposed AP authentication mechanism successfully identifies rogue APs in simulations. Amoordon et al. describe data link layer-based assaults such radio signal jamming, Rogue AP, and deauthentication frames utilising a machine learning technique for RSS value, sequence number gap, frame durations, and management subtypes. The Random Forest and KNN [45] model accurately detects deauthentication and spoofing attacks. SVM (Support Vector Machine), J48 (C4.5), KNN (K nearest neighbours), and MLP (Maximum-Likelihood Projection) are trained on a complete set of RTT information to distinguish allowed and illegitimate APs. This report contains an RTT dataset. The ML-based algorithm uses the data set to make a prediction, and the predictions from each method are compared to find the most accurate. Authors monitored and aggregated DNS server and AP RTT statistics using tracert.exe in Windows 10 [46].

I conducted experimental study to acquire the dataset, normalise it using ML data pre-processing techniques, and use several ML-based strategies. I tested ML-based algorithms on small and large datasets to detect Rogue APs.

III. Experimental Work

3.1 Data Collection Approach

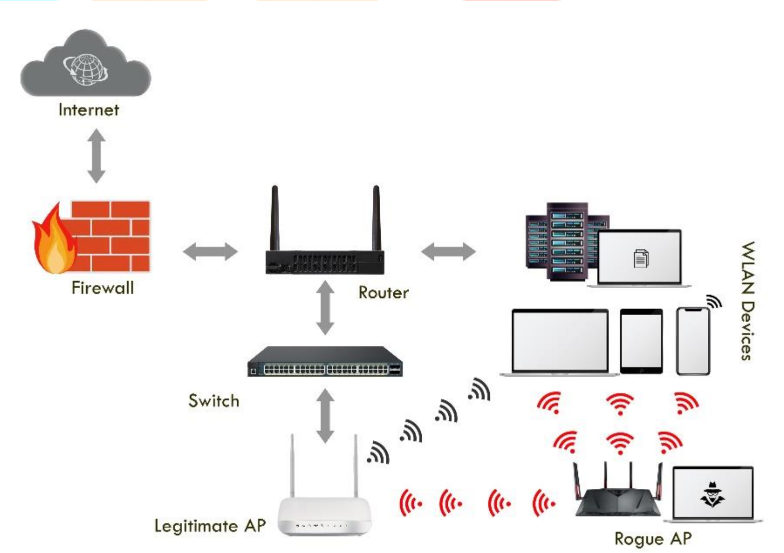

Figure 2: Network Scenario to collect logs or dataset

Attack Scenario 1: I created multiple Rogue APs using ESP8266, giving us an advantage over standard WLANs. In this section, I discussed attacker device specifications. We attacked the hardwood and glass cabinets and achieved signal strength up to 75 feet in a straight line. List the Rogue AP assault components in Table 3.1.

Table 3.1: Components details of Rogue AP based attack

Device Specifications	
Attacker's tools / devices	**Victim Networks**
Notebook: HP 240 Windows 10 Home Edition Intel core i3-3110M Breadboad, LED Male-Female Jumper Wire VMOS D1 Mini Tools: NodeMCU (ESP8266)	GNU_Staff Redmi 3S Prime Xiaomi 9 Pro

Attack Scenario 2: I exploited Wi-Fi Pumpkin to acquire users' email addresses, login credentials, and more via a Rogue AP MITM attack. Table 2 shows the Wi-Fi Pumpkin-based Rogue AP attack for MITM components.

Table 3.1: Components details of Rogue AP based attack

Device Specifications	
Attacker's Tools / Devices	**Victim Devices**
Notebook: Acer Travelmate P249, Intel core i3 Python, Kali Linux Tools:Wi-Fi Pumpkin, Tendaw 311	Samsung M31

Figure 3 depicts the real-time authenticate log-based dataset I created by performing a Rogue AP attack on a lawful network.

ime	ID	Priority	Ether Typ	Src. MAC	Src. Vend	Src. Int.	Src. Zone	Dst. MAC	Dst. Int.	Src. IP	Src. Port	Dst. IP	Dst. Port	IP Proto
ITC 01/25	267	Alert	2048	E8:65:49:F	CISCO SYS	X1	WLAN	18:B1:69:B3:C2:BD		59.49.146.	38911	120.72.90.	20038	tcp
ITC 01/25	82	Alert	2048	E8:65:49:F	CISCO SYS	X1	WLAN	18:B1:69:E	X1	118.193.10	7212	120.72.90.	1335	tcp
ITC 01/25	267	Alert	2048	E8:65:49:F	CISCO SYS	X1	WLAN	18:B1:69:B3:C2:BD		110.87.98.	8378	120.72.90.	55715	tcp
ITC 01/25	267	Alert	2048	E8:65:49:F	CISCO SYS	X1	WLAN	18:B1:69:B3:C2:BD		60.222.235	33127	120.72.90.	21828	tcp

Figure 2: Network Scenario to collect logs or dataset

3.2 Data cleansing and pre-processing techniques on collected dataset

3.2.1 Data Cleansing process:

Data cleaning is the first step in every machine learning project. It ensures the dataset is error-free. Data cleaning involves many steps that prepare data for analysis. Data is not "clean" due to human error in moderation and the inherent inadequacy of automated data collection and period. Data cleansing is crucial to the model's success since discrepancies and errors in training data might prevent algorithms from detecting patterns. The numbers and customer dataset let the model infer "dirty" goods. The authors started by constructing a model that can learn from both pure and corrupted data to forecast fault locations. The authors recommend retraining a model on a sample of actual statistics and testing it to see if it can identify user errors. After that, the authors tested a full dataset and found that the inferred model had above 90% correctness [47][48][51][11].

"Data preparation" involves several processes on raw data to make it machine-readable. A model's algorithm must understand training data to make accurate predictions. Logs and datasets need preprocessing. Most real-world datasets for machine learning have partial data, errors, and noise since they come from multiple sources. Data mining tools might struggle to identify trends in this skewed dataset. Thus, data must be interpreted to improve information. Reliable information is needed to make good decisions. Without data preparation, this high-quality data will be trashed [53] [55] [56].

Summary of the data pre-processing:

- ✓ Comprehend the data [55] [56].
- ✓ Checking at dataset can tell that what to priorities [55] [56].
- ✓ Utilize statistical techniques or pre-built frameworks to visualize dataset's class labels [55] [56].
- ✓ Summarize data repetitions, missing data, and abnormalities [55] [56]
- ✓ Eliminate fields that aren't needed for modelling or are relevant to other properties [55] [56].
- ✓ Data Pre-processing includes segmentation.
- ✓ Determine which features help significantly to overall model training.

3.3 ML-based method for Rogue AP

Pseudo Algorithm:

1. ***Start***
2. ***Input: Dataset Collection***
3. ***Output: Prediction of rogue ap and rogue ap based attack***
4. *If dataset of rogue ap attack is available*
5. ***Then** start data processing*
    a. *Do data cleaning*
        i. *Remove duplicate records*
6. *Do noise cleaning*
7. *Generate error free data*
8. ***Then** start data transformation*
    a. *Apply data abstraction and transformation*
        i. *Apply nominal data conversion*
        ii. *Apply categorical data conversion*
9. ***Then** apply Principal Component Analysis for dimensionally reduction of large dataset*
10. *Apply multiclass logistic regression for accuracy prediction of rogue ap*
11. ***Else***
12. *Restart the same flow for rogue ap and attack detection*

The above pseudo algorithm shows the data pre-processing required before using ML-based approaches to a log-based dataset for network Rogue AP identification. Another typical mistake is layout discrepancy. After cleaning, data abstraction and transformation include making qualitative qualities quantitative, adjusting input size to a given value, and applying normalised statistics to created datasets. After abstraction, Principal Component Analysis is applied. Principal Component Analysis simplifies data analysis by reducing dimensions. After dimensional reduction on larger datasets, multiclass logistic regression may accurately predict Rogue AP.

Here, I show a visual representation of the output of distinct ML-based methodology applied to a pre-processed dataset.

List of ML based methods [53]-[57]:

- ✓ MLR (Multiclass Logistic Regression)
- ✓ Random Forest Tree
- ✓ Random Tree
- ✓ SLR (Simple Logistic Regression)
- ✓ SMO (Sequential Minimal Optimization) with Polynomial kernel
- ✓ SMO with RBF (Radial basis function kernel kernel)

3.4 Output of performed various ML-based techniques to detect Rogue AP

```
=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1.000    0.012    0.992      1.000    0.996      0.990    1.000     1.000     TCP Xmas Tree Attack
                 1.000    0.001    0.995      1.000    0.997      0.997    1.000     1.000     Port Scan Possible
                 0.944    0.001    0.981      0.944    0.962      0.960    1.000     0.992     Possible TCP Flood
                 0.963    0.000    1.000      0.963    0.981      0.980    1.000     1.000     Possible TCP Flood Ceased
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     TCP Null Flag Attack
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     Port Scan Probable
                 0.000    0.000    ?          0.000    ?          ?        1.000     1.000     TCP FIN Scan
Weighted Avg.    0.993    0.008    ?          0.993    ?          ?        1.000     1.000

=== Confusion Matrix ===

   a   b   c   d   e   f   g   <-- classified as
 589   0   0   0   0   0   0 |   a = TCP Xmas Tree Attack
   0 195   0   0   0   0   0 |   b = Port Scan Possible
   3   0  51   0   0   0   0 |   c = Possible TCP Flood
   1   0   1  52   0   0   0 |   d = Possible TCP Flood Ceased
   0   0   0   0  50   0   0 |   e = TCP Null Flag Attack
   0   0   0   0   0  53   0 |   f = Port Scan Probable
   1   1   0   0   0   0   0 |   g = TCP FIN Scan
```

```
Time taken to build model: 0.57 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances       62193               99.2979 %
Incorrectly Classified Instances         6                0.7021 %
Kappa statistic                          0.9883
Mean absolute error                      0.0421
Root mean squared error                  0.105
Relative absolute error                 24.3925 %
Root relative squared error             35.8204 %
Total Number of Instances            62199
```

Figure 4: Result of Random Forest Tree

```
=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.997    0.297    0.829      0.997    0.905      0.759    0.992     0.994     TCP Xmas Tree Attack
                 0.579    0.019    0.883      0.579    0.700      0.665    0.950     0.821     Port Scan Possible
                 0.593    0.006    0.842      0.593    0.696      0.693    0.990     0.834     Possible TCP Flood
                 0.704    0.000    1.000      0.704    0.826      0.832    0.984     0.898     Possible TCP Flood Ceased
                 0.980    0.002    0.961      0.980    0.970      0.969    0.998     0.953     TCP Null Flag Attack
                 0.434    0.011    0.697      0.434    0.535      0.531    0.846     0.458     Port Scan Probable
                 0.500    0.000    1.000      0.500    0.667      0.707    0.924     0.506     TCP FIN Scan
Weighted Avg.    0.846    0.180    0.849      0.846    0.832      0.739    0.976     0.915

=== Confusion Matrix ===

   a   b   c   d   e   f   g   <-- classified as
 587   0   0   0   2   0   0 |   a = TCP Xmas Tree Attack
  72 113   0   0   0  10   0 |   b = Port Scan Possible
  22   0  32   0   0   0   0 |   c = Possible TCP Flood
  10   0   6  38   0   0   0 |   d = Possible TCP Flood Ceased
   1   0   0   0  49   0   0 |   e = TCP Null Flag Attack
  15  15   0   0   0  23   0 |   f = Port Scan Probable
   1   0   0   0   0   0   1 |   g = TCP FIN Scan
```

```
Time taken to build model: 0.03 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances       62045               84.5537 %
Incorrectly Classified Instances       154               15.4463 %
Kappa statistic                          0.7176
Mean absolute error                      0.0441
Root mean squared error                  0.1702
Relative absolute error                 25.5733 %
Root relative squared error             58.0728 %
Total Number of Instances            62199
```

Figure 5: Result of Random Tree

```
=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     TCP Xmas Tree Attack
                 0.995    0.000    1.000      0.995    0.997      0.997    1.000     1.000     Port Scan Possible
                 0.981    0.001    0.981      0.981    0.981      0.980    0.998     0.973     Possible TCP Flood
                 0.981    0.001    0.981      0.981    0.981      0.980    0.999     0.979     Possible TCP Flood Ceased
                 1.000    0.002    0.962      1.000    0.980      0.980    1.000     1.000     TCP Null Flag Attack
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     Port Scan Probable
                 0.000    0.001    0.000      0.000    0.000      -0.001   0.999     0.583     TCP FIN Scan
Weighted Avg.    0.995    0.000    0.994      0.995    0.995      0.994    1.000     0.997

=== Confusion Matrix ===

   a   b   c   d   e   f   g   <-- classified as
 589   0   0   0   0   0   0 |   a = TCP Xmas Tree Attack
   0 194   0   0   0   0   1 |   b = Port Scan Possible
   0   0  53   1   0   0   0 |   c = Possible TCP Flood
   0   0   1  53   0   0   0 |   d = Possible TCP Flood Ceased
   0   0   0   0  50   0   0 |   e = TCP Null Flag Attack
   0   0   0   0   0  53   0 |   f = Port Scan Probable
   0   0   0   0   2   0   0 |   g = TCP FIN Scan
```

```
Time taken to build model: 5.18 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances       62194               99.4985 %
Incorrectly Classified Instances         5                0.5015 %
Kappa statistic                          0.9917
Mean absolute error                      0.0515
Root mean squared error                  0.0891
Relative absolute error                 29.8894 %
Root relative squared error             30.3954 %
Total Number of Instances            62199
```

Figure 6: Result of Multiclass Logistic Regression

```
=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     TCP Xmas Tree Attack
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     Port Scan Possible
                 0.981    0.001    0.981      0.981    0.981      0.980    0.999     0.972     Possible TCP Flood
                 0.981    0.001    0.981      0.981    0.981      0.980    0.999     0.973     Possible TCP Flood Ceased
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     TCP Null Flag Attack
                 1.000    0.002    0.964      1.000    0.981      0.981    0.999     0.964     Port Scan Probable
                 0.000    0.000    ?          0.000    ?          ?        0.500     0.002     TCP FIN Scan
Weighted Avg.    0.996    0.000    ?          0.996    ?          ?        0.999     0.993

=== Confusion Matrix ===

   a   b   c   d   e   f   g   <-- classified as
 589   0   0   0   0   0   0 |   a = TCP Xmas Tree Attack
   0 195   0   0   0   0   0 |   b = Port Scan Possible
   0   0  53   1   0   0   0 |   c = Possible TCP Flood
   0   0   1  53   0   0   0 |   d = Possible TCP Flood Ceased
   0   0   0   0  50   0   0 |   e = TCP Null Flag Attack
   0   0   0   0   0  53   0 |   f = Port Scan Probable
   0   0   0   0   0   2   0 |   g = TCP FIN Scan
```

```
Time taken to build model: 2.07 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances       62195               99.5988 %
Incorrectly Classified Instances         4                0.4012 %
Kappa statistic                          0.9933
Mean absolute error                      0.2043
Root mean squared error                  0.3014
Relative absolute error                118.4653 %
Root relative squared error            102.8228 %
Total Number of Instances            62199
```

Figure 7: Result of SMO (Sequential Minimal Optimization) with RBFkernel

```
--- Detailed Accuracy By Class ---

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     TCP Xmas Tree Attack
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     Port Scan Possible
                 0.981    0.001    0.981      0.981    0.981      0.980    0.999     0.972     Possible TCP Flood
                 0.981    0.001    0.981      0.981    0.981      0.980    0.999     0.973     Possible TCP Flood Ceased
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     TCP Null Flag Attack
                 1.000    0.002    0.964      1.000    0.981      0.981    0.999     0.964     Port Scan Probable
                 0.000    0.000    ?          0.000    ?          ?        0.988     0.271     TCP FIN Scan
Weighted Avg.    0.996    0.000    ?          0.996    ?          ?        1.000     0.994

--- Confusion Matrix ---

   a   b   c   d   e   f   g   <-- classified as
 589   0   0   0   0   0   0 |   a = TCP Xmas Tree Attack
   0 195   0   0   0   0   0 |   b = Port Scan Possible
   0   0  53   1   0   0   0 |   c = Possible TCP Flood
   0   0   1  53   0   0   0 |   d = Possible TCP Flood Ceased
   0   0   0   0  50   0   0 |   e = TCP Null Flag Attack
   0   0   0   0   0  53   0 |   f = Port Scan Probable
   0   0   0   0   0   2   0 |   g = TCP FIN Scan
```

```
Time taken to build model: 1.58 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances       62195               99.5988 %
Incorrectly Classified Instances         4                0.4012 %
Kappa statistic                          0.9933
Mean absolute error                      0.2041
Root mean squared error                  0.3013
Relative absolute error                118.3941 %
Root relative squared error            102.768  %
Total Number of Instances            62199
```

Figure 8: Result of SMO (Sequential Minimal Optimization) with Polykernel

```
=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     TCP Xmas Tree Attack
                 0.969    0.012    0.950      0.969    0.959      0.949    0.991     0.954     Port Scan Possible
                 0.944    0.002    0.962      0.944    0.953      0.951    0.984     0.960     Possible TCP Flood
                 0.981    0.003    0.946      0.981    0.964      0.962    0.996     0.935     Possible TCP Flood Ceased
                 1.000    0.001    0.980      1.000    0.990      0.990    1.000     1.000     TCP Null Flag Attack
                 0.811    0.004    0.915      0.811    0.860      0.854    0.981     0.876     Port Scan Probable
                 0.500    0.001    0.500      0.500    0.500      0.499    0.999     0.750     TCP FIN Scan
Weighted Avg.    0.979    0.003    0.979      0.979    0.979      0.976    0.996     0.978

--- Confusion Matrix ---

   a   b   c   d   e   f   g   <-- classified as
 589   0   0   0   0   0   0 |   a = TCP Xmas Tree Attack
   0 189   1   0   0   4   1 |   b = Port Scan Possible
   0   0  51   3   0   0   0 |   c = Possible TCP Flood
   0   0   1  53   0   0   0 |   d = Possible TCP Flood Ceased
   0   0   0   0  50   0   0 |   e = TCP Null Flag Attack
   0  10   0   0   0  43   0 |   f = Port Scan Probable
   0   0   0   0   1   0   1 |   g = TCP FIN Scan
```

```
Time taken to build model: 21.89 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances       62178               97.8937 %
Incorrectly Classified Instances        21                2.1063 %
Kappa statistic                          0.965
Mean absolute error                      0.006
Root mean squared error                  0.077
Relative absolute error                  3.4853 %
Root relative squared error             26.2807 %
Total Number of Instances            62199
```

Figure 9: Simple Logistic Regression

IV. RESULT ANALYSIS

The following table mentioned the result summary of ML based Rogue AP detection methods.

Table 3.1: Components details of Rogue AP based attack

Algorithms	Time (Sec.)	Accuracy
Random Forest Tree	0.57	99.29%
Random Tree	0.03	84.55%

MLR	5.18	99.49%
SMO with RBFkernel	2.07	99.59%
SMO with polykernel	1.58	99.59%
SLR	21.89	97.89%

SOC analysts can control all system security functions. The Network Operations Center (NOC) analysts watch for system-wide risks and fix them before they propagate. False positives and negatives are the biggest threats to wireless network security today. False negatives occur when network security systems fail to detect network threats. This demonstration of results focuses on rogue ap detection method reliability and time restrictions. Most research have focused on RTT values, although other measurements can identify rogue access points in wireless networks. Based on the results, the ideal output scenario for Rogue AP localizations is SMO (Sequential Minimal Optimization) with Polynomial kernel generate accuracy in 1.58 seconds with 99.5988% data accuracy. The PolyKernel (Polynomial Kernel) and SMOregressor, a robust ML technique for SVM, integrate approximators and projections on time series. This method fills all blanks and converts nominal attributes to binary. Standardizes all parameters by default. Pairwise classification solves multi-class problems. Apply logistic regression models to SVM outputs for precise likelihood calculations.

V. CONCLUSION

In summary, as a part of my research, I fabricate a Rogue AP and employ multiple rogue AP-based attacks that compromised the network and its users into believing they are communicating with a legitimate service. ML-based SMO with a polynomial kernel may create accuracy in the shortest amount of time to detect rogue APs in WLAN, which benefits the admin and legitimate WLAN in the process of security. As a future scope authenticate user or administrator can manage whitelist of legitimate system (e.g. SSID, BSSID, channel details etc.) and performed DDoS or Wi-Fi DEauthentication attack as a counter strike on Rogue AP to compare with whitelist parameters.

REFERENCES

[1] Understanding Rogue Access Points - TechLibrary - Juniper Networks. (n.d.). Understanding Rogue Access Points - TechLibrary - Juniper Networks. Retrieved December 3, 2022, from https://www.juniper.net/documentation/en_US/junos-space-apps/network-director4.0/topics/concept/wireless-rogue-ap.html

[2] Vanjale, S. B., Dave, J., & Mane, P. B. (2012). Unapproved access point elimination in wlan using multiple agents and skew intervals. International Journal of engineering science and Technology, IJEST, 4(2).

[3] Wireless Access Point Protection: 5 Steps to Find Rogue Wi-Fi Networks and Comply with PCI DSS Requirement 11.1. (n.d.). SecurityMetrics. Retrieved December 3, 2022, from https://www.securitymetrics.com/blog/wireless-access-point-protection-finding-rogue-wi-fi-networks

[4] Most Common Wireless Network Attacks - WebTitan DNS Filter. (2021, June 19). WebTitan DNS Filter. Retrieved December 3, 2022, from https://www.webtitan.com/blog/most-common-wireless-network-attacks/

[5] How to Protect Against Rogue Access Points on Wi-Fi. (n.d.). How to Protect Against Rogue Access Points on Wi-Fi. Retrieved December 3, 2022, from https://www.byos.io/blog/how-to-protect-against-rogue-access-points-on-wi-fi

[6] Chow, A. S., & Bucknall, T. (2011, November 9). Library Technology and User Services. In Planning, Integration, and Usability Engineering. Chandos Publishing. (book for legitimate ap and rogue ap)

[7] Kurtz, J. A. (2016, December 8). Hacking Wireless Access Points. In Cracking, Tracking, and Signal Jacking. Syngress.

[8] What is a WiFi Access Point? | Linksys: US. (n.d.). What Is a WiFi Access Point? | Linksys: US. Retrieved December 3, 2022, from https://www.linksys.com/what-is-a-wifi-access-point.html

[9] K. C. Patel and A. Patel, "Taxonomy and Future Threat of Rogue Access Point for Wireless Network," 2022 9th International Conference on Computing for Sustainable Global Development (INDIACom), 2022, pp. 679-688, doi: 10.23919/INDIACom54597.2022.9763150.

[10] Anmulwar, S., Srivastava, S., Mahajan, S. P., Gupta, A. K., & Kumar, V. (2014, February). Rogue AP detection methods: A review. In International Conference on Information Communication and Embedded Systems (ICICES2014) (pp. 1-6). IEEE.

[11] F. Lanze et al., "Letting the puss in boots sweat: Detecting fake APs using dependency of clock skews on temperature", in Proc. 9th ACM Symposium on Information, Computer and Communications Security(ACM)., 2014, pp. 3-14

[12] Shivaraj, G., Song, M., & Shetty, S. (2008, November). A hidden Markov model based approach to detect Rogue APs. In MILCOM 2008-2008 IEEE Military Communications Conference (pp. 1-7). IEEE.

[13] Watkins, L., Beyah, R., & Corbett, C. (2007, November). A passive approach to Rogue AP detection. In IEEE GLOBECOM 2007-IEEE Global Telecommunications Conference (pp. 355-360). IEEE.

[14] Alotaibi, B., & Elleithy, K. (2015, May). An empirical fingerprint framework to detect Rogue APs. In 2015 Long Island Systems, Applications and Technology (pp. 1-7). IEEE.

[15] Swati Jadhav, and Sandeep Vanjale, "Wireless Rogue AP Detection Using Clock Skew Method", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 10, October 2013, pp.1344-1349

[16] Reising, D. R., Temple, M. A., & Jackson, J. A. (2015). Authorized and rogue device discrimination using dimensionally reduced RF-DNA fingerprints. IEEE Transactions on Information Forensics and Security, 10(6), 1180-1192.

[17] R. Beyah, S. Kangude, G. Yu, B. Strickland, and J. Copeland. Rogue AP detection using temporal traffic characteristics. In IEEE GLOBECOM, 2004]

[18] Wu, D., Guan, Y., Liu, K., Zhang, T., Xu, Z., & Liu, Y. (2018, December). A Robust RSS-Based Rogue AP Localization Algorithm with Unknown Transmit Power. In 2018 10th International Conference on Communications, Circuits and Systems (ICCCAS) (pp. 280-285). IEEE.

[19] Jana, S., & Kasera, S. K. (2008, September). On fast and accurate detection of unauthorized wireless APs using clock skews. In Proceedings of the 14th ACM international conference on Mobile computing and networking (pp. 104-115).

[20] Wei, W., Suh, K., Wang, B., Gu, Y., Kurose, J., & Towsley, D. (2007, October). Passive online Rogue AP detection using sequential hypothesis testing with TCP ACK-pairs. In Proceedings of the 7th ACM SIGCOMM conference on Internet measurement (pp. 365-378).

[21] Sawicki, K., & Piotrowski, Z. (2012, May). The proposal of IEEE 802.11 network AP authentication mechanism using a covert channel. In 2012 19th International Conference on Microwaves, Radar & Wireless Communications (Vol. 2, pp. 656-659). IEEE.

[22] Thejdeep, G., Sagar, B. S., Siddartha, L. K., & Chandavarkar, B. R. (2015, April). Detecting Rogue APs using Kismet. In 2015 International Conference on Communications and Signal Processing (ICCSP) (pp. 0172-0175). IEEE.

[23] C. Wang; X. Zheng; Y. Chen; J. Yang, "Locating Rogue AP using Fine-grained Channel Information," in IEEE Transactions on Mobile Computing , vol.PP, no.99, pp.1-1

[24] Kim, T., Park, H., Jung, H., & Lee, H. (2012, May). Online detection of fake APs using received signal strengths. In 2012 IEEE 75th vehicular technology conference (VTC Spring) (pp. 1-5). IEEE.

[25] Nakhila, O., Dondyk, E., Amjad, M. F., & Zou, C. (2015, January). User-side wi-fi evil twin attack detection using ssl/tcp protocols. In 2015 12th Annual IEEE Consumer Communications and Networking Conference (CCNC) (pp. 239-244). IEEE.

[26] Nakhila, O., & Zou, C. (2016, November). User-side wi-fi evil twin attack detection using random wireless channel monitoring. In MILCOM 2016-2016 IEEE Military Communications Conference (pp. 1243-1248). IEEE.

[27] Somayeh Nikbakhsh, Azizah Manaf, Mazdak Zamani, and Maziar Janbeglou, "A Novel Approach for Rogue AP Detection on the ClientSide", IEEE 26th International Conference on Advanced Information Networking and Applications Workshops, 684 - 87, 2012

[28] Sherwood, R. (2008). Discovering and securing shared resources on the Internet. University of Maryland, College Park.

[29] Nikbakhsh, S., Manaf, A. B. A., Zamani, M., & Janbeglou, M. (2012, March). A novel approach for Rogue AP detection on the client-side. In 2012 26th International Conference on Advanced Information Networking and Applications Workshops (pp. 684-687). IEEE.

[30] Burns, A., Wu, L., Du, X., & Zhu, L. (2017, December). A novel traceroute-based detection scheme for wi-fi evil twin attacks. In GLOBECOM 2017-2017 IEEE Global Communications Conference (pp. 1-6). IEEE.

[31] F. Barbhuiya, M. Agarwal, S. Purwar., S. Biswas, S. Roopa, R. Ratti, and S. Nandi, "Application of stochastic discrete event system framework for detection of induced low rate tcp attack," ISA Transactions, 2015

[32] Panch, A. and Sing, S. K. (2010). A Novel approach for Evil Twin or Rogue AP mitigation in wireless environment. International Journal of Security and Its Applications .Vol. 4, No. 4, October, 2010

[33] Hsu, F. H., Wang, C. S., Hsu, Y. L., Cheng, Y. P., & Hsneh, Y. H. (2017). A client-side detection mechanism for evil twins. Computers & Electrical Engineering, 59, 76-85.

[34] Awad, F., Al-Refai, M., & Al-Qerem, A. (2017, April). Rogue AP localization using particle swarm optimization. In 2017 8th International Conference on Information and Communication Systems (ICICS) (pp. 282-286). IEEE.

[35] Vanjale, S. B., Mane, P. B., & Patil, S. V. (2015, March). Wireless LAN intrusion detection and prevention system for malicious AP. In 2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom) (pp. 487-490). IEEE.

[36] Thite, S., Vanjale, S. A. N. D. E. E. P., & Mane, P. B. (2014). A novel approach for fake AP detection and prevention in wireless network. International Journal of Computer Science Engineering and Information Technology Reaserach, 35-42.

[37] Kao, K. F., Chen, W. C., Chang, J. C., & Te Chu, H. (2014, June). An accurate fake AP detection method based on deviation of beacon time interval. In 2014 IEEE Eighth International Conference on Software Security and Reliability-Companion (pp. 1-2). IEEE.

[38] D. Monica, and C. Ribeiro, "Wi-FiHop - Mitigating the Evil Twin Attack through Multi-hop Detection" , in Proc. 16th European Symposium on Research in Computer Security(ESORICS)., Leuven, Belgium, 2011, pp. 21-39.

[39] Chen, Yongle, et al. "Localizing AP through simple gesture." IEEE Access 6 (2018): 38870-38880.

[40] Wu, D., Guan, Y., Liu, K., Zhang, T., Xu, Z., & Liu, Y. (2018, December). A Robust RSS-Based Rogue AP Localization Algorithm with Unknown Transmit Power. In 2018 10th International Conference on Communications, Circuits and Systems (ICCCAS) (pp. 280-285). IEEE.

[41] S. Shah, S. Srirangarajan, and Tewfik. Implementation of a directional beacon-based position location algorithm in a signal processing framework. IEEE Transactions on Wireless Communications, 2010.

[42] Gustafsson F, Gunnarsson F. Mobile positioning using wireless networks: possibilities and fundamental limitations based on available wireless network measurements[J]. Signal Processing Magazine IEEE, 2005, 22(4):41-53

[43] K. Wu, J. Xiao, Y. Yi, M. Gao, and L. M. Ni, ''FILA: Fine-grained indoor localization,'' in Proc. IEEE INFOCOM, Orlando, FL, USA, Mar. 2012, pp. 2210–2218

[44] Liu, R., Zhang, Z., Wang, T., Wang, L., & Zhao, S. (2020, October). Machine Learning Based AP Verification Scheme for the Smart Grid. In 2020 9th International Conference on Power Science and Engineering (ICPSE) (pp. 6-11). IEEE.

[45] Amoordon, A., Deniau, V., Fleury, A., & Gransart, C. (2022). A single supervised learning model to detect fake APs, frequency sweeping jamming and deauthentication attacks in IEEE 802.11 networks. Machine Learning with Applications, 10, 100389.

[46] Kim, D., Shin, D., & Shin, D. (2017). Data Set Construction and Performance Comparison of Machine Learning Algorithm for Detection of Unauthorized AP. In Advances in Computer Science and Ubiquitous Computing (pp. 910-914). Springer, Singapore.

[47] V. Roth et al, “Simple and effective defense against evil twin APs,” in Proc. 1st ACM conference on Wireless network security(WiSec)., Alexandria, Virginia, USA, 2008, pp. 220-235.

[48] Han, H., Sheng, B., Tan, C. C., Li, Q., & Lu, S. (2011). A timing-based scheme for Rogue AP detection. IEEE Transactions on parallel and distributed Systems, 22(11), 1912-1925.

[49] M. K. Chirumamilla, and B. Ramamurthy, “Agent Based Intrusion Detection and Response System for Wireless LANs”, in Proc. IEEE International Conference on Communications(ICC)., Anchorage Alaska, USA, 2003, pp. 492 – 496

[50] Sawicki, K., & Piotrowski, Z. (2012, May). The proposal of IEEE 802.11 network AP authentication mechanism using a covert channel. In 2012 19th International Conference on Microwaves, Radar & Wireless Communications (Vol. 2, pp. 656-659). IEEE.

[51] Fan W H, Yu L, Wang Z, et al. The effect of wall reflection on indoor wireless location based on RSSI[C]// IEEE International Conference on Robotics and Biomimetics. IEEE, 2015:1380 - 1384.

[52] Ma, L., Teymorian, A. Y., & Cheng, X. (2008, April). A hybrid Rogue AP protection framework for commodity Wi-Fi networks. In IEEE INFOCOM 2008-The 27th Conference on Computer Communications (pp. 1220-1228). IEEE.

[53] Sagduyu, Y. E., Shi, Y., Erpek, T., Headley, W., Flowers, B., Stantchev, G., & Lu, Z. (2020). When wireless security meets machine learning: Motivation, challenges, and research directions. arXiv preprint arXiv:2001.08883.

[54] Kitisriworapan, S., Jansang, A., & Phonphoem, A. (2020). Client-side rogue access-point detection using a simple walking strategy and round-trip time analysis. EURASIP Journal on Wireless Communications and Networking, 2020(1), 1-24.

[55] Data Preprocessing in Machine Learning [Steps & Techniques]. (n.d.). Data Preprocessing in Machine Learning [Steps & Techniques]. Retrieved December 4, 2022, from https://www.v7labs.com/blog/data-preprocessing-guide

[56] Sarker, I. H. (2021). Machine learning: Algorithms, real-world applications and research directions. SN Computer Science, 2(3), 1-21.

[57] Moayedi, H., Tien Bui, D., Kalantar, B., & Kok Foong, L. (2019). Machine-learning-based classification approaches toward recognizing slope stability failure. Applied Sciences, 9(21), 4638.