Position-based Rogue Access Point Detection

Wenjie Liu
Networked Systems Security Group
KTH Royal Institute of Technology
Stockholm, Sweden
wenjieli@kth.se

Panos Papadimitratos

Networked Systems Security Group

KTH Royal Institute of Technology

Stockholm, Sweden

papadim@kth.se

Abstract-Rogue Wi-Fi access point (AP) attacks can lead to data breaches and unauthorized access. Existing rogue AP detection methods and tools often rely on channel state information (CSI) or received signal strength indicator (RSSI), but they require specific hardware or achieve low detection accuracy. On the other hand, AP positions are typically fixed, and Wi-Fi can support indoor positioning of user devices. Based on this position information, the mobile platform can check if one (or more) AP in range is rogue. The inclusion of a rogue AP would in principle result in a wrong estimated position. Thus, the idea to use different subsets of APs: the positions computed based on subsets that include a rogue AP will be significantly different from those that do not. Our scheme contains two components: subset generation and position validation. First, we generate subsets of RSSIs from APs, which are then utilized for positioning, similar to receiver autonomous integrity monitoring (RAIM). Second, the position estimates, along with uncertainties, are combined into a Gaussian mixture, to check for inconsistencies by evaluating the overlap of the Gaussian components. Our comparative analysis, conducted on a real-world dataset with three types of attacks and synthetic RSSIs integrated, demonstrates a substantial improvement in rogue AP detection accuracy.

1. Introduction

Wi-Fi APs enable devices to connect to local area networks and the Internet wirelessly. However, rogue Wi-Fi APs, unauthorized APs installed in the area of a network without approval, pose significant cybersecurity risks. Rogue APs facilitate man-in-the-middle attacks (intercepting and altering communication of devices, attracted to connect to them instead of legitimate APs); or they are used in phishing campaigns (unsuspecting users connect to the rogue AP, unknowingly reveal login credentials and other sensitive information) [1]. Hence, considering the ease of deploying rogue APs and given most of the public APs are open networks, it is vital to have effective and robust rogue AP detection.

Existing solutions for rogue AP detection, [2]–[4], simply use a whitelist of Wi-Fi APs. Other detection methods leverage techniques based on CSI, RSSI, or their combination. They identify unauthorized APs by analyzing signal characteristics or monitoring changes in the wireless environment. [5] uses RSSI and piggybacks information in IEEE 802.11 beacon frames to localize the Wi-Fi AP and support location verification, but it modifies

Wi-Fi. [6] establishes feature vectors using RSSIs and then performs a clustering analysis on them, albeit with low detection accuracy. Location, hardware, and environment-related fingerprints based on CSI are explored in [7]–[9]. However, these solutions require specialized hardware, complex configuration, or extensive computational resources, which limit practicality [1]. Furthermore, dynamic network environments, as CSI is sensitive to environmental changes [10], lead to false positives or missed detections.

Detecting a rogue AP that mimics a legitimate AP (e.g., Evil-twin [1], [11], [12]) is challenging and an active detection scheme, accessible to network operators and clients/users, is needed. Attackers can replicate hardware information from legitimate signals, allowing them to evade detection by hardware or signal fingerprinting methods. On the other hand, Wi-Fi RSSI-based indoor positioning is already very mature for daily use, with meter-level accuracy [13]. This motivates the following question: can a position-based method, leveraging the deployment of multiple APs, typically so in many environments nowadays, detect rogue APs? This paper answers positively, inspired by the RAIM method, developed for global navigation satellite system (GNSS) spoofing detection [14], detecting rogue Wi-Fi APs based on positioning results and their cross-validation. Based on that, rogue AP can be detected and excluded, relying solely on RSSIs.

The scheme we propose can work with contemporary mobile devices and mainstream Wi-Fi cards. The approach is intuitive: with access to a positioning algorithm and AP-specific data, the mobile localizes itself. If it does so while taking into account a rogue AP, the computed device position will be different from the one(s) computed with benign APs only. The mobile device uses RSSI subsets of all available APs and detects the attack through the positions that deviate when one (or more) rogue APs are part of a subset.

Our rogue AP detection is compatible with many Wi-Fi positioning algorithms as long as they provide position and uncertainty, e.g., RSSI fingerprint-based positioning (based on a database [13], not hardware fingerprints), or distance-based positioning [15], [16] (based on the positions of the deployed APs). Given our method is built on state-of-the-art positioning algorithms [13], [16], it inherits advantages, such as insensitivity to environmental change. By analyzing the positioning results obtained from different AP RSSI subsets, the method identifies positions consistent across multiple subsets. Inconsistency implies that a rogue APs is part of one or more subsets.

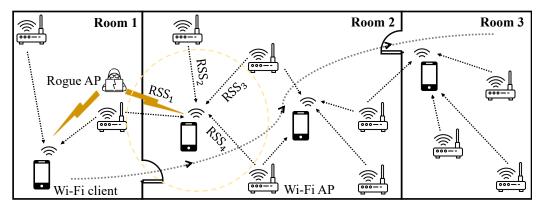


Figure 1: System and adversary model. For example, the mobile Wi-Fi client in the dashed yellow circle can locate itself using the subsets of four RSSIs from APs in range; the subset containing the rogue AP would be inconsistent.

For example, in Fig. 1, four APs are within range of the Wi-Fi client entering (on the left of) Room 2: for a subset of two legitimate and one rogue AP, the client position estimation differs from that involving any three benign APs in Room 2. The inconsistency reveals an attack and an examination of a sufficient number of subsets (positions) can reveal which is the rogue AP, and then have it excluded (or even localized).

Our key contribution is a scheme that can work either as a tool used by wireless network operators for actively monitoring APs or by mobile users that can independently detect rogue APs. We just need RSSIs to make the scheme compatible with other rogue AP detection schemes. We adopt and adapt the RAIM used in GNSS to achieve rogue Wi-Fi AP detection, and we term this Gaussian mixture RAIM. We exploit position fusion from Wi-Fi positioning algorithms and rogue AP exclusion strategies. Our numerical experiments and a comparative analysis on a partially real-world Wi-Fi RSSI dataset show significant improvement over baseline rogue AP detection schemes.

2. Related Work

2.1. Rogue Wi-Fi AP Detection

Rogue Wi-Fi AP detection has received significant attention. Some industrial solutions [2]-[4] ignore unknown APs or use a whitelist of AP medium access controls (MACs) and service set identifiers (SSIDs) to find unauthorized ones; however, attackers can relatively easily forge this information. For example, many commercial Wi-Fi routers can set any SSID, while opensource router systems, such as OpenWrt, can even modify their MAC address. The method in [5] uses RSSI measurements to identify rogue APs, but its clustering and two-step algorithm is affected by signal variations due to environmental factors (interference and multipath effects). Another approach employs wireless fingerprinting techniques [8], independent of client devices; however fingerprinting scalability and robustness in dynamic network environments (rainfall, high pedestrian traffic, etc.) is challenging. PRAPD [6] introduces a method based on RSSI for practical rogue AP detection. Real-time identification of rogue Wi-Fi connections in operational networks was explored in [9]. Additionally, CSI based on environment-related semantics for Internet-of-Things (IoT) [7], while offering potential accuracy advantages, requires specialized hardware and large-scale scanning of frequency bands.

2.2. RAIM for GNSS Spoofing

Receiver autonomous integrity monitoring (RAIM) leverages redundant data and performs consistency checks based on subsets of visible satellites [17]. Over the past few decades, there are mainly two types of RAIMs: residual-based and solution separation [18]. Residualbased RAIM [19], [20] uses statistical hypothesis testing on residual errors, identifying potential faulty measurements. The residual can be from least-squares or filters: extended Kalman filter (EKF)-combined RAIM utilizes rolling window filters to identify and eliminate outliers using global positioning system (GPS) and inertial sensors [19], [20]. Solution separation RAIM [14], [21] recursively assumes faulty satellites, generates subsets of the remaining satellites to derive solutions, and then identifies which subsets contain the faults. For example, [21] integrates RANSAC clustering to classify position solutions. Additionally, advanced RAIM [22] extends fault exclusion to include multiple constellations like GPS and Galileo, providing enhanced security beyond GPS. Recent signal of opportunity (SOP) techniques [14], [23] leverage wireless network signals for RAIM, integrating kinematics models, cellular pseudoranges, or Wi-Fi measurements to improve RAIM performance.

3. System Model and Adversary

3.1. System Model

We consider a mobile Wi-Fi device shown in Fig. 1, with unknown position $\mathbf{p}_{\mathrm{c}}(t) \in \mathbb{R}^3$ (coordinates and height)—the device can be held by a security patrol person from the network provider or be a user device (e.g., a smartphone). The client actively explores Wi-Fi APs and gathers beacon information from them. Then, $\mathrm{RSS}_j(t), j \in \mathcal{J}(t), t=1,2,...,T$, is RSSI information at time t, where $\mathcal{J}(t)$ is the set of APs identifiers at time t providing RSSIs, and T is the last time index.

Although the client is indoors, walking and without GNSS reception, we assume pre-surveyed data, depending

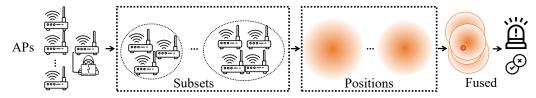


Figure 2: System overview of position-based rogue AP detection.

on the requirements of the Wi-Fi positioning algorithm. The data can be: (i) a fingerprint database of RSSIs, $\mathcal{T} = \{\{\mathrm{RSS}_j(t) \mid j \in \mathcal{J}(t)\}, \mathbf{p_c}(t)\}_{t=T_1}^{T_2}$, where $t = T_1, ..., T_2$ are the time indexes for the database, or (ii) a list of AP positions, $\mathbf{p}_{\mathrm{AP}}^{(j)}(t) \in \mathbb{R}^3$ for $\mathrm{RSS}_j(t)$.

3.2. Adversary

The rogue APs, often Evil-twin APs of legitimate ones [11], [12], emulate the hardware model, SSID, basic service set identifier (BSSID), and MAC address of legitimate APs, deceiving users (Wi-Fi clients) to connect. The attacker may employ a "coexistence strategy": if the RSSI of the legitimate AP is relatively weak, the rogue AP can boost its strength to compel clients to connect as well as relay packets to/from the legitimate AP.¹

However, the attacker can not remove legitimate signals or completely mimic them. Hence, we assume that while rogue APs join the environment of in the same nearby legitimate ones to lure unsuspecting users, they inevitably change signal properties (RSSI) and differ in positions, thus leading to deviations of indoor Wi-Fi positioning results.

4. Proposed Scheme

4.1. Subset Generation

Subsets are generated as illustrated in Fig. 2, without presuming a known number of rogue APs. We systematically generate subsets considering all possible combinations of APs, varying in size—from the minimum to the maximum allowable for the given positioning algorithm. These subsets of RSSI indexes, j, are denoted as $\mathcal{S}_l(t)$, where l=1,2,...,L(t) and L(t) is the number of all subsets. An analysis of RAIM resilience in [14] (single infrastructure case) shows that position estimates from benign subsets are around the actual position, and the number of them is most likely larger than the number of the attack subsets.

4.1.1. Sampling Strategy for Subsets. The generation of subsets is a critical step for rogue AP detection. Given the number of subsets can be large, we employ a straightforward sampling strategy: a random selection of subsets, with each subset chosen based on a predetermined probability distribution. The rationale is to select a diverse range of subsets without introducing significant bias or skewness. Despite its simplicity, this approach is theoretically robust with negligible impact on the cross-validation

process, ensuring adaptability to various network configurations and attack scenarios. Most importantly, it enhances computation efficiency of our method.

4.1.2. Fingerprint-based Positioning. It is necessary to construct a database of fingerprint vectors of RSSIs associated with the client positions in advance. Then, the positioning algorithm compares the current RSSI vector with the database and returns the estimated latitude and longitude coordinates and height of the receiver.

Given a subset that needs to be used for positioning, $\{RSS_j(t) \mid j \in \mathcal{S}_l(t)\}$, and the fingerprint database $\mathcal{T} = \{\{RSS_j(t) \mid j \in \mathcal{J}(t)\}, \mathbf{p}_c(t)\}_{t=T_1}^{T_2}$, which is presurveyed and without adversarial data, we look for the most K similar fingerprints in the database \mathcal{T} . The function for scoring similarity to the fingerprint at t' is defined as:

$$f\left(\{\text{RSS}_{j}(t) \mid j \in \mathcal{S}_{l}(t)\}, \{\text{RSS}_{j}(t') \mid j \in \mathcal{J}(t')\}\right)$$

$$= \sum_{j \in \mathcal{J}(t)} \frac{\mathbb{I}\left\{j \in \mathcal{S}_{l}(t)\right\}}{\max(|\text{RSS}_{j}(t) - \text{RSS}_{j}(t')|, d_{\min})}$$
(1)

where $\mathbb{I}\{A\}$ is 1 when the condition A is satisfied, and $d_{\min} > 0$ is the minimal difference of two RSSIs. Suppose the client position associated with the most K similar fingerprints are $\mathbf{p}_{\mathrm{c}}^{(k)}, k = 1, 2, ..., K$, with similarity scores $f^{(k)}, k = 1, 2, ..., K$. Then, the positioning result of $\hat{\mathbf{p}}_l(t)$ is the weighted average:

$$\hat{\mathbf{p}}_{l}(t) = \frac{\sum_{k=1}^{K} f^{(k)} \mathbf{p}_{c}^{(k)}}{\sum_{k=1}^{K} f^{(k)}}$$
(2)

with the reciprocal of the average of $\{f^{(k)}\}_{k=1}^K$ as the uncertainty of positioning, $\hat{\sigma}_l(t)$.

4.1.3. Distance-based Positioning. Geolocation localization [16] proposes a weighted nonlinear least squares problem to minimize the weighted sum of squared distances between APs and the estimated location. These weights are determined based on the inverse square of the signal strengths:

$$\min_{\hat{\mathbf{p}}_{l}(t)} \quad \sum_{j} \left(\frac{||\mathbf{p}_{AP}^{(j)}(t) - \hat{\mathbf{p}}_{l}(t)||_{2}}{RSS_{j}(t)} \right)^{2}. \tag{3}$$

Then, the positioning result and the least squares residual (position uncertainty) from lth subset at time t are $\hat{\mathbf{p}}_l(t)$ and $\hat{\boldsymbol{\sigma}}_l(t)$, where l(t)=1,2,...,L(t).

Apart from the fingerprint and distance-based methods, our subset generation is compatible with other Wi-Fi positioning algorithms providing position and uncertainty.

^{1.} If the rogue AP replaces or is very close to the legitimate ones, the security patrol person can easily find it without relying on our method [1].

4.2. Position Validation

We see the positioning results as random variables following $\mathcal{N}(\hat{\mathbf{p}}_l(t), \hat{\boldsymbol{\sigma}}_l(t))$. Then, the Gaussian mixture at time t is fused into a temporary position, $\tilde{\mathbf{p}}(t)$. Based on $\tilde{\mathbf{p}}(t)$, rogue AP detection and exclusion are carried out. Furthermore, an ultimate position $\hat{\mathbf{p}}(t)$ can be estimated based on the APs after exclusion.

The temporary position fused from the Gaussian mixture is:

$$\tilde{\mathbf{p}}(t) = \frac{\sum_{l=1}^{L(t)} \hat{\mathbf{p}}_l(t) \circ \hat{\boldsymbol{\sigma}}_l(t)}{\sum_{l=1}^{L(t)} \hat{\boldsymbol{\sigma}}_l(t)}$$
(4)

where \circ is the Hadamard product, i.e., $[\hat{\mathbf{p}}_l(t) \circ \hat{\boldsymbol{\sigma}}_l(t)]_i = [\hat{\mathbf{p}}_l(t)]_i[\hat{\boldsymbol{\sigma}}_l(t)]_i$. Then, the deviation between the lth positioning result and the fused position at time t is:

$$d_l(t) = \|\hat{\mathbf{p}}_l(t) - \tilde{\mathbf{p}}(t)\|. \tag{5}$$

If the deviation is larger than a threshold Λ , the subset for this positioning result is identified as a set containing rogue AP. The threshold is:

$$\Lambda = \mathbb{E}[d_l(t)] + n_{\Lambda} \cdot \sqrt{\mathbb{V}[d_l(t)]}$$
 (6)

where n_{Λ} is a factor controlling coverage, usually taking a value of 3, according to the three-sigma rule of thumb. In most cases, larger n_{Λ} comes with a higher false positive rate. Hence, by adjusting n_{Λ} , we can get the desired false positive and true positive rate.

4.2.1. Rogue AP Exclusion. After validating all subsets $S_l(t), l = 1, 2, ..., L(t)$, the detected non-benign subsets can find the intersection or vote for the rogue APs.

Denote the set of the index l of non-benign subsets as $\mathcal{L}(t)$. Then, the identifiers of rogue APs at t are:

$$\bigcap_{l \in \mathcal{L}(t)} \mathcal{S}_l(t). \tag{7}$$

However, considering an environment with positioning noise, we use the following voting scheme for classifying the jth AP in $\mathcal{J}(t)$ as rogue or not:

$$\begin{cases} \mathbf{A} = \sum_{l \in \mathcal{L}(t)} \mathbb{I}\{j \in \mathcal{S}_l(t)\} \\ \mathbf{B} = \sum_{l \notin \mathcal{L}(t)} \mathbb{I}\{j \in \mathcal{S}_l(t)\} \end{cases}$$
 (8)

If A > B, the *j*th AP is rogue at time *t*. Then, $\hat{\mathbf{p}}(t)$ is calculated based on the remaining APs.

5. Experiments

5.1. Dataset

The dataset from [13] was collected on the fourth floor of a shopping mall, covering an area of 170 by 90 m. 8 APs are visible at each position, while the landmarks and AP positions were used to generate trajectories. The device WHUWearTrack was employed for trajectory acquisition as ground truth, incorporating a low-cost inertial measurement unit (IMU), a Bluetooth module, and a multiprotocol system-on-chip. The reference trajectories have decimeter-level positioning accuracy. The Wi-Fi positioning algorithm from [13] exhibits meter-level accuracy.

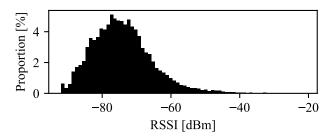


Figure 3: The distribution of the RSSI dataset.

5.1.1. Attack Simulation. We generate rogue AP signals by modifying the RSSI data: (i) An AP is randomly chosen to have $\mathcal{N}(\mu_{\mathrm{adv}}, \sigma_{\mathrm{adv}}^2)$ dB additive signal strength in a random duration (around one-third of the entire trace), where $\mu_{\mathrm{adv}} = 10, \sigma_{\mathrm{adv}} \in \{2, 4\}$, (ii) RSSIs of an AP are replaced by values following $\mathcal{U}_{[-70, -55]}$ in dBm, or (iii) A rogue AP located at (30.52868, 114.35086) and 12 meters in height, with its RSSIs simulated based on an indoor path loss model [24]. 16 traces containing rogue AP signals for each attack are generated, so 48 in total. The distribution of RSSIs containing attacks is depicted in Fig. 3.

5.2. Baseline Methods

The first baseline method is a clustering-based method inspired by [6], which uses a 2-class clustering algorithm to process RSSI of individual AP. The *k*-medoid clustering algorithm has a distance measurement method that can deal with missing values and noise. If the distance between clusters is larger than a threshold, the corresponding AP is classified as rogue.

Another baseline is an anomaly detection scheme, ECOD [25], calculating the differences of RSSIs over t per AP. It estimates the underlying distribution of the input data through empirical cumulative distributions for each AP, without relying on predetermined parameters. These estimates serve as the basis for determining tail probabilities associated with each data vector across all APs. Last, ECOD derives an outlier score by aggregating the estimated tail probabilities across APs at t.

We use true positive rate, $P_{\rm TP}$, as the metric for comparison, defined as the number of true positive detections (or exclusions) of rogue APs over the total number of positive (rogue AP launching attacks) timestamps. To compare the methods fairly, we fix the false positive rate, $P_{\rm FP}$, to 0.01, 0.02, ..., 0.1. A false positive is a detection result that wrongly believes a rogue AP attack takes place while it does not.

5.3. Detection Accuracy

Detection accuracy measures the ability of a system to correctly identify the presence of rogue APs while maintaining a predefined maximum $P_{\rm FP}$. It quantifies the proportion of detection outcomes that correctly raised an alarm that rogue APs were present. In other words, detection accuracy evaluates how effectively the system identifies potential threats without generating false positives. The comparison results are shown in Fig. 4. Solid

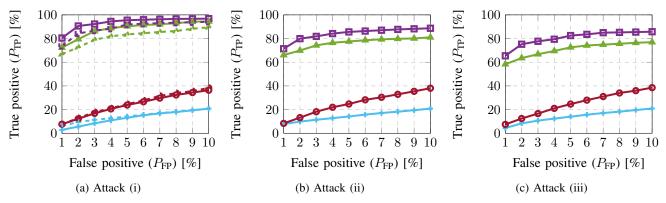


Figure 4: Detection and exclusion P_{TP} of the proposed and baseline methods. \blacksquare is detection and \blacksquare is exclusion performance for our scheme; \blacksquare for the clustering-based detection and \blacksquare for ECOD-based detection.

lines are for $\sigma_{\rm adv}=2$ and dashed lines are for $\sigma_{\rm adv}=4$ in attack (i).

In attack (i) scenarios (recall: attacks (i)–(iii) defined in Sec. 5.1.1), our detection consistently achieves higher P_{TP} (67% to 97%) compared to the clustering-based and ECOD-based detection methods. Clustering-based detection exhibits low rates (below 21%) across all P_{FP} and σ_{adv} . While ECOD-based detection shows some improvement over clustering-based detection, it still falls short of our proposed approach variants, particularly at lower P_{FP} . In attack (ii) scenarios, our detection demonstrates increasing P_{TP} (71% to 89%) as P_{FP} grows, outperforming clustering-based and ECOD-based detection consistently. Similarly, in attack (iii), our detection achieves higher P_{TP} (65% to 85%) compared to clustering-based and ECOD-based detection, across P_{FP} .

The signals of randomly chosen AP in attack (i) are manipulated with varying levels of additive noise characterized by $\sigma_{\rm adv}$: the detection accuracy tends to decrease as $\sigma_{\rm adv}$ increases. This trend suggests that higher levels of noise significantly affect the ability to accurately identify rogue APs, resulting in lower P_{TP} and increased false positives. In attack (ii), where RSSIs of some APs are replaced by random values within a specified range, the detection accuracy is lower compared to attack (ii). This suggests that attack (ii) poses more of a challenge to detection algorithms compared to the noise-induced variations in attack (i). In attack (iii), P_{TP} for our detection method is comparable to those for attack data (i) and (ii) at higher P_{FP} . A lower P_{FP} , the performance of detecting attack (iii) tends to be slightly lower compared to (i) and (ii). Despite this, the proposed detection still outperforms clustering-based detection and ECOD-based detection across all attacks.

5.4. Rogue AP Exclusion

Rogue AP exclusion requires correctly identifying those in the subsets deemed affected. Exclusion accuracy measures the ability to exclude rogue APs after detection of an attack (rogue AP in range). The results are also shown in Fig. 4. For attack (i), the proposed exclusion achieves $P_{\rm TP}$ ranging from 67% to 94%. While its $P_{\rm TP}$ is slightly lower than the proposed detection naturally, as it is challenging to correctly exclude rogue APs, it still has

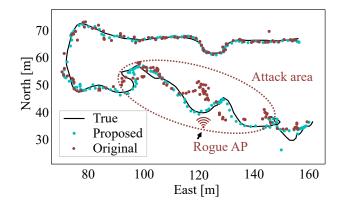


Figure 5: An illustration of the positioning results before (i.e., original) and after (i.e., proposed) exclusion.

a high accuracy. Similarly, in attack (ii) scenarios, the exclusion method achieves $P_{\rm TP}$ ranging from approximately 66% to 81% for $P_{\rm FP}$ of 1% to 10%, showing improvement with increasing $P_{\rm FP}$. However, for attack (iii) scenarios, the exclusion accuracy initially drops to around 60% at lower $P_{\rm FP}$ due to the difficulty in distinguishing rogue and legitimate APs in close proximity. As $P_{\rm FP}$ increases, $P_{\rm TP}$ improves to 77%.

5.5. Position Recovery

Position recovery refers to the process of determining the position of a Wi-Fi client based on RSSIs from the deemed benign APs after the estimated rogue APs were excluded. Then, the indoor Wi-Fi positioning algorithm relying solely on benign RSSI measurements estimates the client position accurately. The effectiveness of position recovery can be evaluated by comparing the reference positions (ground truth) with the positioning results obtained before and after the exclusion of rogue APs. This comparison allows to assess the accuracy of the client's position estimation and, naturally, how rogue AP detection and exclusion improve the accuracy of the Wi-Fi positioning system. The positioning results before and after rogue APs exclusion are shown in Fig. 5. The red solid dots are the original Wi-Fi positioning results, showing the attack area deviation from the actual path, drawn as a black line. In

TABLE 1: P_{TP} versus sampling ratio.

| Sampling Ratio | 0.25 | 0.4 | 0.7 | 1.0 |
|--------------------|------|-----|-----|-----|
| Detection Accuracy | 83% | 89% | 95% | 96% |

contrast, the cyan-blue solid dots are the results of the proposed method with much lower deviation.

5.6. Effect of Sampling

Based on the sampling strategy for subsets proposed in Sec. 4.1.1, we evaluate the detection P_{TP} for different ratios of sampling subsets, under attack (i) with $\sigma_{\rm adv} = 2$. $P_{\rm FP}$ is fixed to 0.05. $P_{\rm TP}$ versus sampling ratio is shown in Tab. 1. The results demonstrate a clear trend of increasing P_{TP} with higher sampling ratios. As the sampling ratio increases from 0.25 to 1.0, there is an improvement in $P_{\rm TP}$ from 83% to 96%. This shows that a higher sampling ratio enables capturing a wider range of RSSI subsets; while with a lower sampling ratio, the algorithm may miss crucial subsets, leading to decreased accuracy. However, accuracy is not very sensitive to the sampling ratio. When the sampling rate decreases from 1 to 0.25 (4 times computation reduction), P_{TP} only increases by 13%. While higher sampling ratios generally lead to better performance, there may be practical considerations, computational resources and time constraints to take into account.

6. Limitations

A limitation of this work emerges when the attacker (rogue APs) can adjust the transmission power so that it can appear to be close to the victim APs. If the position computed including rogue AP measurements is very close to that based on benign APs, they are clustered together and consequently the rogue AP is not excluded. In addition, we have not considered our approach in conjunction with other schemes that reveal inconsistencies or spurious traffic.

7. Conclusions

This paper proposes a scheme that effectively identifies rogue APs without the need for specialized hardware. We rely on redundant position information obtained from Wi-Fi positioning, through a RAIM-style method for cross-validation. The two key components of our approach, subset generation and position validation using Gaussian mixture RAIM, show higher rogue AP detection accuracy compared to baseline methods. In future work, we will consider integrating our position-based detection with other cross-validation approaches. We will combine the sampling strategy with pruning algorithms to reduce computations. We will compare to other related solutions (as baselines), consider more realistic attacks, and work on locating rogue APs.

Acknowledgment

This work was supported in parts by the Knut and Alice Wallenberg Foundation and the China Scholarship Council.

References

- B. Alotaibi and K. Elleithy, "Rogue access point detection: Taxonomy, challenges, and future directions," Wirel. Pers. Commun., vol. 90, pp. 1261–1290, 2016.
- [2] ArubaOS, "Detecting rogue APs," Hewlett Packard Enterprise Development LP, 2024. [Online]. Available: https://www.arubanetworks.com/techdocs/ArubaOS_64_Web_Help/Content/ArubaFrameStyles/New_WIP/Rogue_AP_Detection.htm
- [3] I. S. N. P. Management, "Wifi monitoring for modern networks," IBM, 2024. [Online]. Available: https://www.ibm.com/products/ sevone-network-performance-management/wifi-monitoring
- [4] D. Gantenbein, "Finding and remediating rogue access points on the Microsoft corporate network," *Microsoft*, 2024. [Online]. Available: https://www.microsoft.com/insidetrack/blog/finding-rogue-access-points-on-the-microsoft-corporate-network/
- [5] N. M. Ahmad, A. H. M. Amin, S. Kannan, M. F. Abdollah, and R. Yusof, "A RSSI-based rogue access point detection framework for Wi-Fi hotspots," in *Proc. 2nd IEEE ISTT*, Langkawi, Malaysia, Nov. 2014.
- [6] W. Wu, X. Gu, K. Dong, X. Shi, and M. Yang, "PRAPD: A novel received signal strength-based approach for practical rogue access point detection," *Int. J. Distrib. Sens. Netw.*, vol. 14, no. 8, pp. 1–11, 2018.
- [7] I. E. Bagci, U. Roedig, I. Martinovic, M. Schulz, and M. Hollick, "Using channel state information for tamper detection in the internet of things," in *Proc. 31st ACSAC*, Los Angeles, CA, USA, Dec. 2015
- [8] Y. Lin, Y. Gao, B. Li, and W. Dong, "Accurate and robust rogue access point detection with client-agnostic wireless fingerprinting," in *Proc. IEEE PerCom*, Austin, TX, USA, Mar. 2020.
- [9] D. Yan, Y. Yan, P. Yang, W.-Z. Song, X.-Y. Li, and P. Liu, "Real-time identification of rogue WiFi connections in the wild," *IEEE Internet Things J.*, vol. 10, no. 7, pp. 6042–6058, 2022.
- [10] Y. Ma, G. Zhou, and S. Wang, "WiFi sensing with channel state information: A survey," ACM Comput. Surv., vol. 52, no. 3, pp. 1–36, 2019.
- [11] M. Vanhoef and F. Piessens, "Advanced Wi-Fi attacks using commodity hardware," in *Proc. 30th ACSAC*, New Orleans, LA, USA, Dec. 2014
- [12] M. Vanhoef, N. Bhandaru, T. Derham, I. Ouzieli, and F. Piessens, "Operating channel validation: Preventing multi-channel man-inthe-middle attacks against protected Wi-Fi networks," in *Proc. 11th* ACM WiSec, Stockholm, Sweden, Jun. 2018.
- [13] Z. Zheng, Y. Li, Z. Liao, Y. Xue, J. Kuang, Y. Zhuang, and P. Zhang, "The necessity of modeling location uncertainty of fingerprints for ubiquitous positioning," *IEEE Sens. J.*, vol. 23, no. 16, pp. 18413–18422, 2023.
- [14] W. Liu and P. Papadimitratos, "Extending RAIM with a Gaussian mixture of opportunistic information," in *Proc. ION ITM*, Long Beach, CA, USA, Jan. 2024.
- [15] W. Liu and J. Chen, "Geography-aware radio map reconstruction for UAV-aided communications and localization," in *Proc. IEEE ICC*, Montreal, QC, Canada, Jun. 2021.
- [16] Mozilla, "Ichnaea," https://github.com/mozilla/ichnaea/blob/main/ ichnaea/api/locate/mac.py, 2023.
- [17] R. G. Brown, "A baseline GPS RAIM scheme and a note on the equivalence of three RAIM methods," *J. Inst. Navigation*, vol. 39, no. 3, pp. 301–316, 1992.
- [18] M. Joerger, F.-C. Chan, and B. Pervan, "Solution separation versus residual-based RAIM," *J. Inst. Navigation*, vol. 61, no. 4, pp. 273– 291, 2014.
- [19] S. Khanafseh, N. Roshan, S. Langel, F.-C. Chan, M. Joerger, and B. Pervan, "GPS spoofing detection using RAIM with INS coupling," in *Proc. IEEE/ION PLANS*, Monterey, CA, USA, May 2014.
- [20] P. F. Roysdon and J. A. Farrell, "GPS-INS outlier detection & elimination using a sliding window filter," in *Proc. Am. Control Conf.*, Seattle, WA, USA, May 2017.

- [21] K. Zhang and P. Papadimitratos, "Fast multiple fault detection and exclusion (FM-FDE) algorithm for standalone GNSS receivers," *IEEE Open J. Commun. Soc.*, vol. 2, pp. 217–234, 2021.
- [22] J. Blanch, T. Walker, P. Enge, Y. Lee, B. Pervan, M. Rippl, A. Spletter, and V. Kropp, "Baseline advanced RAIM user algorithm and possible improvements," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 51, no. 1, pp. 713–732, 2015.
- [23] M. Maaref and Z. M. Kassas, "Autonomous integrity monitoring for vehicular navigation with cellular signals of opportunity and an IMU," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 6, pp. 5586–5601, 2021.
- [24] W. Liu and P. Papadimitratos, "Probabilistic detection of GNSS spoofing using opportunistic information," in *Proc. IEEE/ION PLANS*, Monterey, CA, USA, Apr. 2023.
- [25] Z. Li, Y. Zhao, X. Hu, N. Botta, C. Ionescu, and G. Chen, "ECOD: Unsupervised outlier detection using empirical cumulative distribution functions," *IEEE Trans. Knowl. Data Eng.*, 2022.