Hindawi
Mobile Information Systems
Volume 2017, Article ID 1248578, 14 pages
https://doi.org/10.1155/2017/1248578

Research Article

Exploiting Wireless Received Signal Strength Indicators to Detect Evil-Twin Attacks in Smart Homes

Zhanyong Tang,[1] Yujie Zhao,[1] Lei Yang,[1] Shengde Qi,[1] Dingyi Fang,[1] Xiaojiang Chen,[1] Xiaoqing Gong,[1] and Zheng Wang[2]

[1]*School of Information Science and Technology, Northwest University, Xi'an, China*
[2]*School of Computing and Communications, Lancaster University, Lancaster, UK*

Correspondence should be addressed to Dingyi Fang; dyf@nwu.edu.cn and Zheng Wang; z.wang@lancaster.ac.uk

Received 20 September 2016; Accepted 21 November 2016; Published 17 January 2017

Academic Editor: Qingchen Zhang

Evil-Twin is becoming a common attack in smart home environments where an attacker can set up a fake AP to compromise the security of the connected devices. To identify the fake APs, The current approaches of detecting Evil-Twin attacks all rely on information such as SSIDs, the MAC address of the genuine AP, or network traffic patterns. However, such information can be faked by the attacker, often leading to low detection rates and weak protection. This paper presents a novel Evil-Twin attack detection method based on the received signal strength indicator (RSSI). Our approach considers the RSSI as a fingerprint of APs and uses the fingerprint of the genuine AP to identify fake ones. We provide two schemes to detect a fake AP in two different scenarios where the genuine AP can be located at either a single or multiple locations in the property, by exploiting the multipath effect of the Wi-Fi signal. As a departure from prior work, our approach does not rely on any professional measurement devices. Experimental results show that our approach can successfully detect 90% of the fake APs, at the cost of a one-off, modest connection delay.

1. Introduction

Smart homes consist of many intelligent, automation systems which are often connected to each other and the Internet through Wi-Fi to provide the inhabitants with sophisticated monitoring and control over the property's functions. Smart homes are increasingly becoming a target for cyber attackers [1–4]. Many of smart home targeting attacks exploit a technique called Evil-Twin where an adversary makes a rogue (i.e., Evil-Twin) access point (AP) with the same identity (or SSID) as an authorized AP, hoping that many of the wireless clients will connect to the rogue AP due to the commonly used automatic access point selection option [5]. An adversary can use an Evil-Twin AP as a platform to launch a variety of attacks, including privacy and data theft. Privacy concerns become evident because there are a large number of private data by various applications in the smart city, such as sensitive data of governments or proprietary information of enterprises [6].

How to detect Evil-Twin AP has recently received much attention [7, 8]. Generally speaking, there are two widely used approaches in this domain. The first approach uses traffic characteristics from the network flow [9, 10] to detect rogue APs. By analyzing information such as the packet arrival time, the request/response time of TCP ACKs, one can distinguish authorized APs from fake ones. Such approaches, however, depend on many environmental factors, such as the type and bandwidth of the network and traffic congestion (which can change from time to time). Therefore, such an approach is only applicable to a limited set of environments where the network traffic pattern is known ahead of time and is stable. The second approach, namely, fingerprint identification detection, uses hardware features [11–18], to identify rogue APs. This requires collecting fingerprint information from the hardware and systems software components (e.g., the firmware, the chip, and the driver) of the authentic APs. This approach is based on an assumption that it is difficult for the attacker to set up an AP with identical hardware information. However, building a fingerprint library is nontrivial and extracting the fingerprints from the APs could be time-consuming. These drawbacks make such approaches infeasible when real-time is an essential requirement.

This paper introduces a novel method for detecting Evil-Twin APs. Our approach targets smart homes. Our approach exploits the following observations: (1) the position of an AP is often fixed in a smart home environment; (2) the received signal strength indicator (RSSI) of a fixed AP is relatively stable. We consider the RSSI signal as the fingerprint of a genuine AP and use this information to identify rogue APs. One of the advantages of our approach is that we do not require any additional sensor/actuator infrastructure. Instead, we first use the stable RSSI to estimate the distance between the signal point and the receiving point [19–27] and then use the distance to detect rogue Evil-Twin APs. We show that our approach achieves on average a successful rate of over 90% with a one-off connection delay of less than 20 seconds.

The main contribution of this paper is a novel Evil-Twin attack detection system based on RSSI. We have shown that RSSI is a viable means for detecting rogue APs in smart home environments. Although our approach is evaluated in a smart home environment, similar ideas can be expanded to other Wi-Fi environments.

2. Background

SSID and BSSID are always used to identify Wi-Fi hot point since the protocol 802.11 does not define a strong sign to do it. In fact, both of them could be easily got by attacker, because the wireless network not only shares the media but also cannot control the signal range. Although the access point is protected by password and sophisticated encryption, for an experienced attacker, it is not difficult to crack it during a short time. The original 802.11 security organization that try to solve these problems was the Wired Equivalent Privacy (WEP). In spite of having mechanisms to provide authentication, confidentiality, and data integrity, WEP was found to be unsafe and trivially cracked after an attacker has gathered enough frames with the same initialization vector [28]. By actively accelerating the gather of frames, the latest WEP attack is able to complete breaking of WEP in under a minute [29]. WEP is increasingly being replaced by the Wi-Fi Protected Access (WPA). Nevertheless, to hold backward compatibility, WPA has not totally solved some security problems. Because control and management frames can be tricked and faked even with WPA enabled, wireless Local Area Networks (LANS) reserve impressionable to identity attacks and denial of service attacks [12]. Once the attacker got the password, they will soon forge the same one called the Evil-Twin AP (i.e., the rogue or fake AP), which is not easily recognized by user. Over the past few years, this kind of attack mainly exists in some public environments such as airports and cafes. However, as the development of the IoT, nowadays gigantic crowd-sourced data from mobile devices have become widely available in social networks [30], the attack value of private Wi-Fi rises rapidly, and the attack develops towards the private Wi-Fi in the smart home and other environments, such as privacy concerns that become evident on the cloud because there are a lot of private data in multimedia data sets [31]. Once the user connects the network to the fake AP, the intruder can control the network environment of the user, and further, privacy sniffing, malicious data tampering, and other advanced attacks can be realized. The behavior of the intelligent device even can be controlled, for instance, opening or closing an intelligent lock.

According to the IEEE 802.11 standard, when there are multiple APs nearby, the one with the strongest signal is to be chosen [16]. So the fake AP is always putting at the nearest of attack target in order to be chosen. This kind of attack can be called Fishing, which contains active Fishing and passive Fishing. Passive Fishing is named because the fake AP is just waiting for the connection from the terminal. This kind of attack cannot easily be found since it does not affect the Real AP; at the same time, the attack successful rate is not high. Active Fishing means that to connect with the terminal, fake AP cut the connection between Real AP and the terminal by Evil-Twin attack. Such attack can be carried out to precise attacks without affecting the other equipment except the target.

3. Attacking Scenarios

Attacking Scenarios. Figure 1 illustrates the scenarios where the Evil-Twin attack can be applied. Evil-Twin is designed to look like real Wi-Fi hotspots. In those scenarios, the adversary is able to set a fake AP to launch an Evil-Twin attack from a laptop. Its signal might be stronger to the victim than the Real AP. Once disconnected from the legitimate Real AP, the tool then forces offline computers and devices to automatically reconnect to the Evil-Twin, allowing the hacker to intercept all the traffic to that device. When people in smart homes are using the Internet through an Evil-Twin, they can unknowingly expose their passwords and other sensitive online data to hackers. According to the Wi-Fi Alliance, a sophisticated Evil-Twin can even control what websites appear when users access the Internet. That allows hackers to capture their passwords.

Our Assumptions. Our attacks require the adversary to set up the Evil-Twin at different locations. We believe that the adversary may not set the fake AP very close to the smart homes in order to avoid being caught. If a profile for the legitimate AP exists, the client device will automatically connect to the faked AP.

4. DRET Overview

Figure 2 is shown as the overview of DRET System. DRET is a system that helps wireless home owner to discover and prevent evil access points (AP) from attacking wireless users. The application can be run in regular intervals to protect your wireless network from Evil-Twin attacks. By configuring the tool you can get notifications sent to your alarm signal whenever an evil access point is discovered. Additionally you can configure DRET to perform DoS on the legitimate wireless users to prevent them from connecting to the discovered evil AP in order to give the administrator more time to react. However, notice that the DoS will only be performed for evil APs which have the same SSID but different BSSID (AP's MAC address) or running on a different

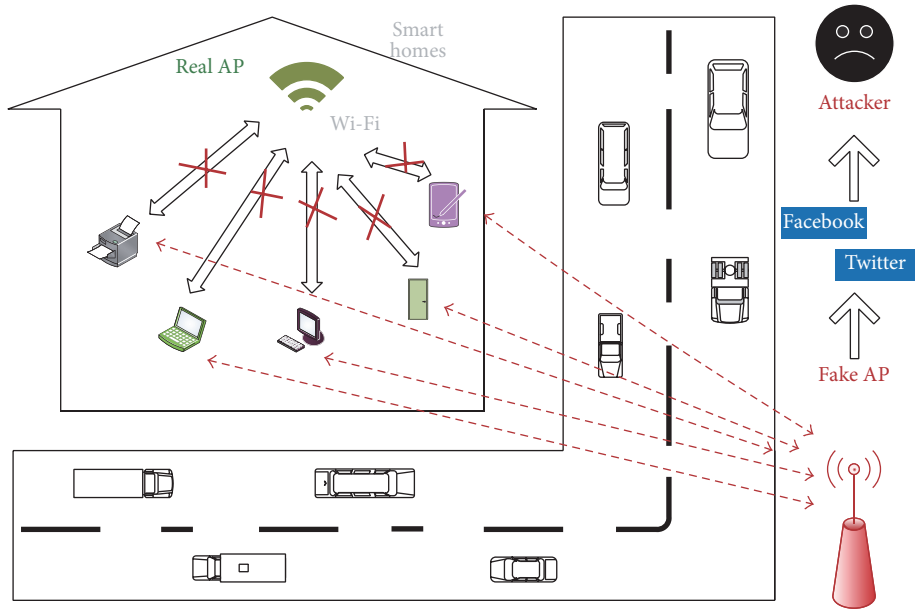

Figure 1: Example scenarios in which the attacker can easily launch an Evil-Twin attack to steal information using a fake AP. This kind of attack typically happens when a hacker constructs a mock (but still functional) Wi-Fi access point (AP) right at the place where there ought to be an original and legitimate access point. The reason this works so well is that, for a well-orchestrated attack, the illegitimate AP has stronger signals than the legitimate one and hence the unsuspecting users might log on to this mock-up connection and then use the Internet while sharing all their precious data, all the way from their user's IDs and passwords to credit/debit card information.

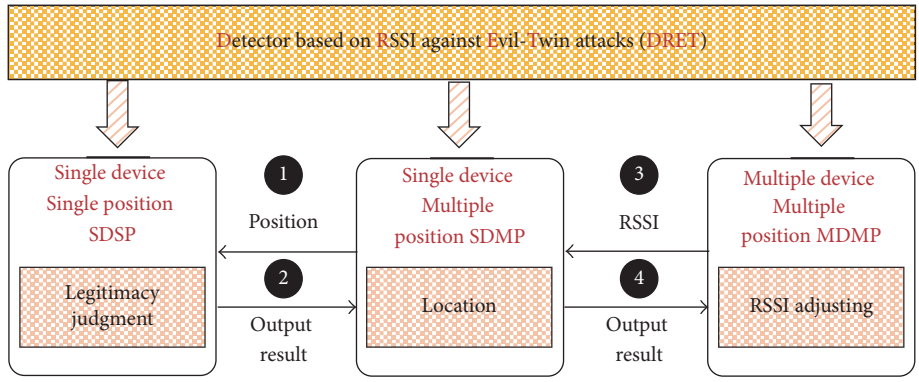

Figure 2: The overview of DRET System. DRET mainly consists of three parts (SDSP, SDMP, and MDMP).

channel. This method can prevent DoS from attacking your legitimate network.

Following a common practice in fake AP detection, DRET will choose different modules depending on different circumstances. SDSP meet the simple scenario such as during night and when nobody is at home. However, SDSP is limited and the success rate is closely related to the detector location. To address this limitation, SDMP is proposed, which locates the mobile phone firstly; the RSS fingerprint value is drawn to SDSP (❶), so the SDSP can determine the location of legitimacy (❷); the result returns to SDMP. Sometimes in many devices working in multiplaces, these devices need to use only one set of fingerprint data to check at the same time. MDMP will start; the RSSI is adjusted and then sent to SDMP (❸); the result done by SDMP returns to MDMP (❹).

5. Preliminaries

In order to construct a real environment, the attacker will do everything to improve the fake AP so that it has the same features of a Real AP, including traffic characteristics and hardware fingerprint characteristics. In real-world applications, the environment may have some negative effects on the identification of the target [32]. However, the attacker cannot forge the position of the Real AP. Recent literature advances Wi-Fi signals to "see" people's motions and locations. By detecting and analyzing signal reflection, they enable Wi-Fi to "see" target objects [33]. In smart homes, the intuition underlying our design is that each Real AP has its fixed position, and the attacker cannot put the fake AP exactly in the right place. Therefore, a new smart home fake AP detected method based on RSSI is proposed in this paper.

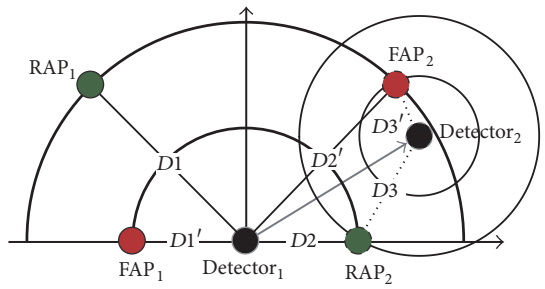

Figure 3: The figure shows two Real APs (in green) and two Fake APs (in red). The figure illustrates how the detector (in black) recognizes the FAP by using the differences of the RSSI that the APs locate differently.

Figure 3 is shown as the principle of fake AP detection based on RSSI. RAP and FAP are, respectively, represented Real AP and fake AP. Detector receives the signal from each AP. $D1$ is the distance between the $Detector_1$ and the Real AP, and $D1'$ is the distance between the $Detector_1$ and the fake AP. If $D1$ is greater than $D1'$, it means that the intensity of $Detector_1$ received from the fake AP is stronger than the real one. In general, when there exists multipath effect, detector always chooses the strongest signal in the homologous signals. So, undoubtedly, when the attacker turns on FAP_1, $Detector_1$ will choose it rather than the real RAP_1. But when the attacker turns off the FAP_1, $Detector_1$ will choose RAP. According to the upper analysis, we can easily identify the fake AP from the real one by comparing the RSSI of them. In this scene, If $RSSI_1'$ is greater than $RSSI_1$, it means that FAP_1 is fake AP.

However, there is another scene where the distance between the Real AP and detector is less than the fake ones. In this condition, no matter how open or shut down the fake AP is, the detector would always choose the Real AP. So, we should try to build a scene like the previous one, namely, moving the detection's position to $Derector_2$, making $D3'$ greater than $D3$; then we can detect the fake AP.

In free space, the path loss of signal propagation expresses signal attenuation, which is defined as the difference value between the effective radiated power and the received power. So the path loss in free space can be computed by the following formula. G_t and G_r separately express the antenna gain of the sender and the receiver. λ indicates the signal wave length; d is the distance between the sender and receiver.

$$\mathrm{PL\,(dB)} = 10\log\frac{P_t}{P_r} = -10\log\left[\frac{G_t G_r \lambda^2}{(4\pi)^2 d^2}\right]. \quad (1)$$

Frequency of Wi-Fi channel 1~13 is from $2.412 * 10^9$~$2.472 * 10^9$. And there exists $\lambda = c/f$ and $c \approx 3 * 10^8$ m/s, so the value range of λ is 0.1214~0.1244. We did some experiment to study factors effecting the attenuation and the attenuation curve is shown in Figure 4. In Figure 4(a), both of the sender and receiver have unity-gain, and the channel is 1. In Figure 4(b), both of the sender and receiver have unity-gain, and the channel is 13. In Figure 4(c), the antenna gain product of the sender and receiver is 100, and the channel is 13. From Figure 4, we can find the following rules. (1) From (a) and (b), we can find that the effect of channel on attenuation is very small. (2) From (b) and (c), we can find that antenna gain has a great influence on attenuation. (3) From (a), (b), and (c), we can find that the distance is the main factor to affect the attenuation, and the attenuation is less sensitive to the distance with the increase of distance.

RSSI (Signal Strength Indicator Received) is the intensity of the received signal; its value can be calculated by the following formula: RSSI = Transmit Power + antenna gain − path loss.

For a fixed transmitter and receiver, the Transmit Power and antenna gain are both constant, and the path loss is a function of the distance D, so RSSI can be expressed as $RSSI = f(d)$. Then d will be $d = f'(\mathrm{RSSI})$. Therefore, RSSI can be used directly to replace the distance for positioning.

In order to be simplify the calculation, we proposed signal space and signal distance. Signal distance can be abbreviated as sd; then $sd = |RSSI|$. In Figure 5, the left is the physical space, and the right is the signal space. Both of them take AP as the reference point. Points a, b, c, and d are the position of four mobile phones. In the physical space, the distance separately between a, c, and d is equal, less than the distance between b and AP. But there are obstacles at the points a and d, where the attenuation of the black obstacle is higher than the gray obstacle, so $sd_a > sd_d > sd_c$ and $sd_b > sd_c$. In general, the signal strength of straight line is the best when there is no obstacle, and wireless devices always give priority to the best signal when dealing with multipath effects. So, from the physical space to the signal space, the distance of their signal has some slight changes, which is shown as the right figure.

In order to verify that the RSSI can be used as the defection factor, we did an experiment. In normal circumstances, we build a fingerprint library by using the signal distance. Terminal MX3 is used as director to collect RSSI signal and the TL-WR882N is used as AP. The distance between them is 5 m, and data collection rate is 2 times per second. We collected about 14000 of the total data, keeping surrounding environment not changed during the process of collecting data, except when someone walked across. Its probability distribution histogram is shown in Figure 6.

By analyzing the experiment data, it is found that the measured value of the actual measurement is near to a stable value, and the probability distribution is approximately normal distribution. That means the RSSI can be used as the defection factor.

Actually, it seems that both of the fake and Real AP is similar to the detector, which are difficult to be distinguished. According to multipath effect, the detector will select the one with the strongest signal to associate and compute the distance between the selected AP and it, which will be compared with the distance recorded in signal distance fingerprint database. If they are different, that means the AP should be forged. The mobile phone will be used as the detector. Depending on whether the mobile phone used as a detector in smart home is moving or not, two different kinds of solution have been proposed in this paper; they are a single fixed position detection and the multiposition collaborative detection.

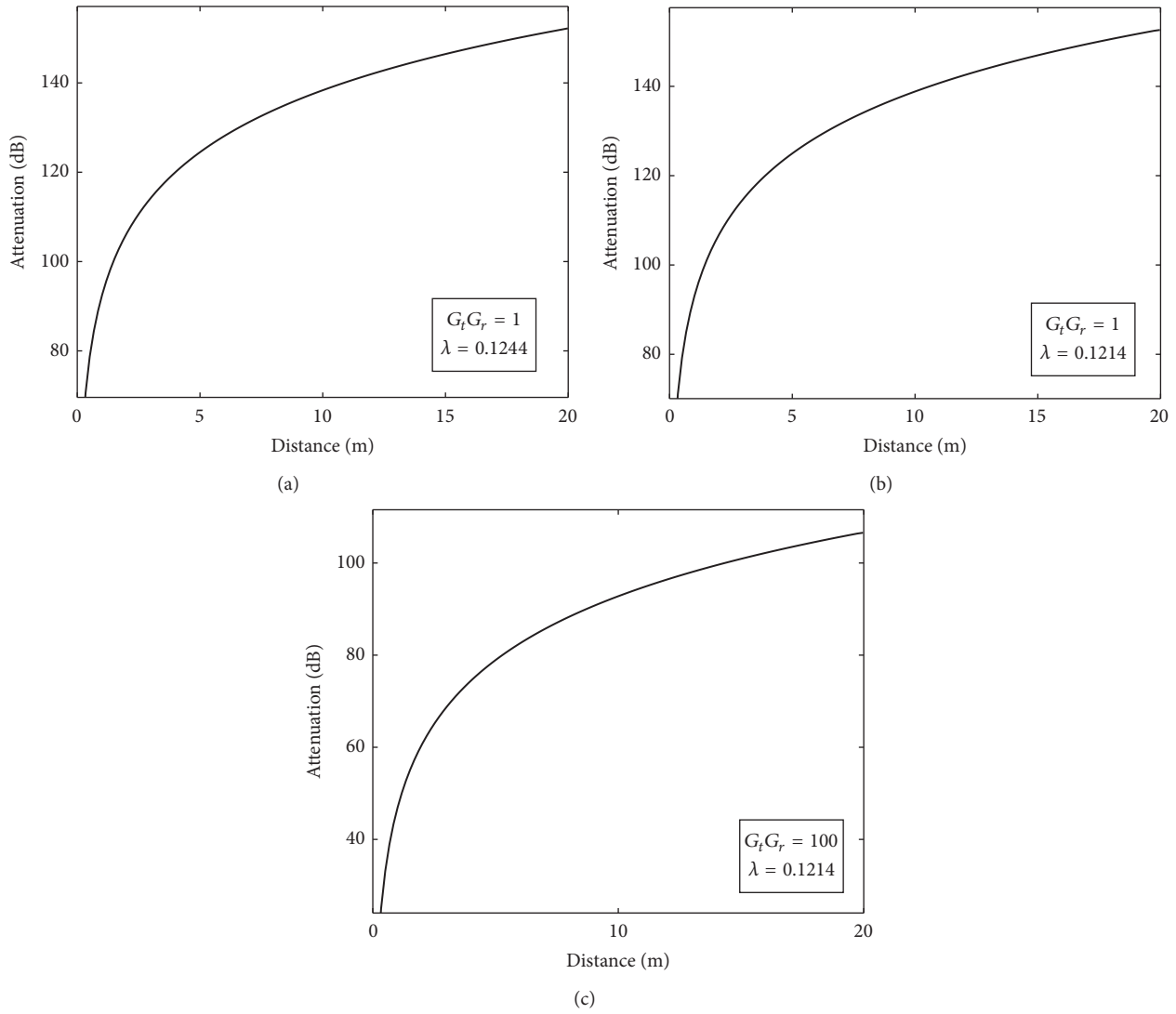

Figure 4: Signal attenuation curve.

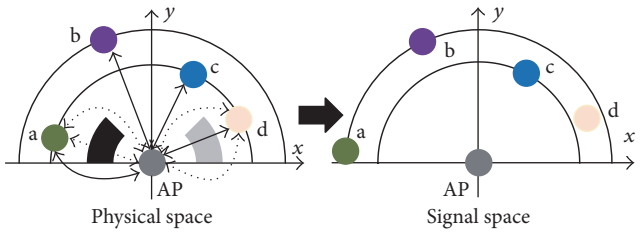

Figure 5: Physical space convert to signal space.

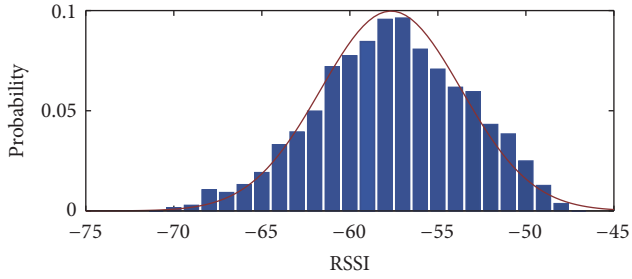

Figure 6: Probability distribution histogram.

6. Automated Detection Analysis

6.1. A Single Fixed Position Detection. Smart homes devices still need work under networking even if there is nobody at home, so the detector can also finish the detecting of false AP. Therefore, we install the detector in a fixed position, and let it work 24 hours. Detector establishes target AP RSSI fingerprint library in normal sense, which would be used as sample when detecting. Only the detected distance is within the error range of distances recorded in fingerprint database; it is considered as the fake AP; otherwise, it is true AP.

It is assumed that the deployment of hot spot and detector is shown in Figure 7. The position of fake AP and true AP is different, but the other features are the same, such as network

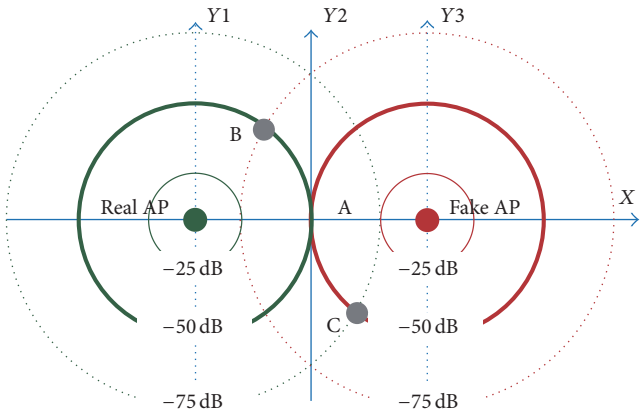

Figure 7: A single fixed position detection.

Table 1: FSSI and variance in the security state.

Location	Average	Variance
A	$\mu_A = -50$	σ_A
B	$\mu_B = -50$	σ_B
C	$\mu_C = -75$	σ_C

card hardware features, antenna gain, and stability. A, B, and C are the positions of three detectors. The signal intensity of true AP and fake AP is the same in the position A (shown as $Y2$ axis). The signal intensity of true AP is stronger than ones of fake AP in the position B and the opposite in position C.

In the security state, that is, where the fake AP does not exist, the RSSI and variance of signal intensity separately received by three detectors at positions A, B, and C are shown in Table 1.

Fake APs working will lead to multipath effect. Therefore, it is assumed that P_A, P_B, and P_C are the probability of selecting true AP signal in A, B, and C. Under ideal conditions, $0 \le P_C < P_A = 0.5 < P_B \le 1$, and the new average and variance are shown in Table 2. Both of them wave in a certain range of fluctuation due to kinds of factors like the multipath effect, the external interference, and so forth. It is assumed that the average an variance meet the following conditions: $\mu - M \le \mu \le \mu + M$, $\sigma \le \Sigma$.

From Figure 7, we can see that when the detector is in region C, it will select fake AP whose signal intensity is stronger than the Real APs, which can be described with a formula like $\mu' > \mu$. When $\mu' > \mu + M$, we can say that there exists a fake AP in the network. When the detector is in region A, $\mu' = \mu$; that means we cannot distinguish the Real AP and the fake one. In region B, although the signal intensity of Real AP is higher than fake AP, but the detector considers both of them are the same signal; the latter still cannot be detected.

As analysis shows detector and Real AP cannot be too close that will lead to high misdetection rate, so the best deployment location of detector is in region C where the signal is weak, far away from the Real AP and near the fake AP.

Table 2: FSSI and variance when fake AP is working.

Location	Average	Variance
A	$\mu'_A = \mu_A = -50$	$\sigma'_A = \sigma_A$
B	$-75 < \mu'_B < -50$	$\sigma'_B > \sigma_B$
C	$75 < \mu'_C < -50$	$\sigma'_C > \sigma_C$

6.2. Multiposition Detection. Obviously, a single fixed position detection method can only solve part of the problem. In this part, multiposition detection is proposed. Multiposition detection relies on mobile phones; with it we can convert multiposition to single fixed position detection. So, first what we need to do is determine the position of the mobile phone. The most well-known and highly accurate positioning method is GPS, while GPS devices have been known to not work very well indoors. In this paper, we use the Wi-Fi signal for locating the position of mobile phone by three-point positioning method. With the popularity of Wi-Fi, there are almost always more than three Wi-Fi hotspots that will be found when we are indoors.

As shown in Figure 8, AP_1, AP_2, and AP_3 are three different APs, assuming their positions are known. O is the mobile phone's position. The original distance can be defined as sd which represents the distance between AP and mobile phone. $sd_i = |OO_i|$, $i = 1, 2, 3, 4, 5$. So AP_1, AP_2, and AP_3 can locate the position of the mobile phone in the signal space. Then we can convert the multiposition detection to a single fixed position detection.

There are two stages in multiposition cooperative detection: fingerprint gathering stage and detection stage. The first stage should be done in a safe state; we collect the RSSI information both of reference AP and target AP in many different positions, to build a fingerprint library. In the detection stage, using reference AP to locate the phone and the fingerprint data in a single fixed position detection, the program framework is shown in Figure 8; we can locate the mobile phone's position by using reference AP and then using the method mentioned in the previous chapter to detect.

In Figure 9, AP_0 is the target AP, AP_2~AP_n are the candidate's reference AP, and the whole process can be divided into the following 5 steps:

Step ❶: RSSI acquisition.

Step ❷: effective data selection.

Step ❸: establishment of fingerprint database.

Step ❹: mobile position determination.

Step ❺: validity judgment.

6.2.1. RSSI Acquisition. In the experiment, the value of RSSI is collected by mobile phone; the detection program can import corresponding management package and call relevant interface (Android: android.net.wifi.*; IOS: SystemConfiguration/CaptiveNetwork.h) so that it can make mobile phone acquire enough RISS value in daily routines.

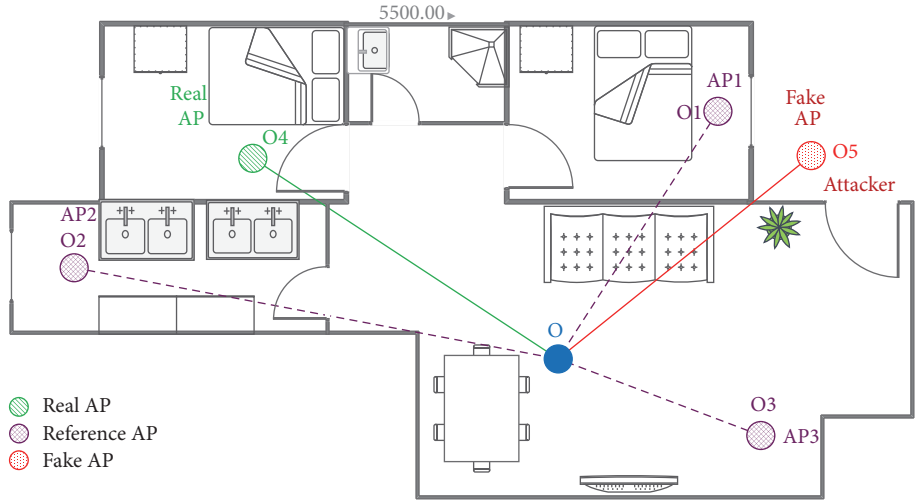

Figure 8: Multiposition detection transformation. The figure shows that any three APs could be chosen as reference in the signal space. They are used to locate the positions of the mobile phone which is a detector in smart homes.

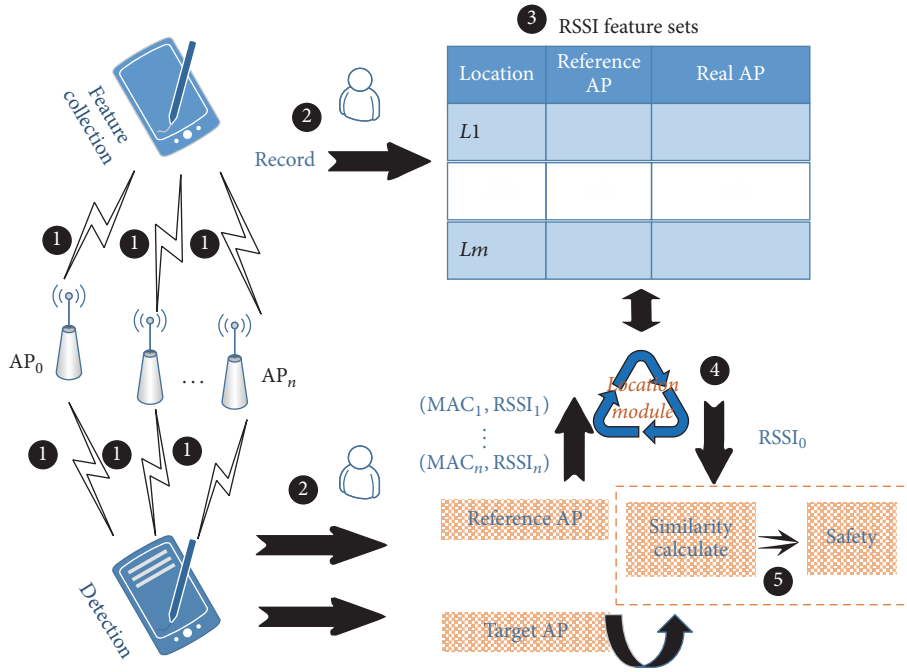

Figure 9: Multiposition detection framework.

6.2.2. Effective Data Selection

Effective RSSI Values Selection. It is a challenging job to choose the right RSSI values since the mobile phones are always moving. However the RSSI value we need should be waved in a small range, which is shown in Figure 10. The data in two boxes are what we want; the others are generated by mobile phone when it is moving. When the distance between mobile phone and AP is 1 m and there is no interference, it can generate the data in the first box. Data in the second box is generated in the condition that the distance between mobile phone and AP is 4 m and there are two sources of interference. The other data is generated in the condition that someone takes the mobile phone and go around the house with the speed of 1.5 m/s.

In the first experiment, variance increment method is used to judge whether the mobile phone is moving. It is assumed that the size of sliding window is 120. When the amount of data is less than the window, it is invalid data.

$$W_i = \{r_{i-\text{ws}+1}, r_{i-\text{ws}+2}, \ldots, r_{i-1}, r_i\} \quad i \geq \text{ws},\ r_i \in R. \tag{2}$$

R is the whole RSSI sequence, r_i is the value of RSSI, and ws is the window size.

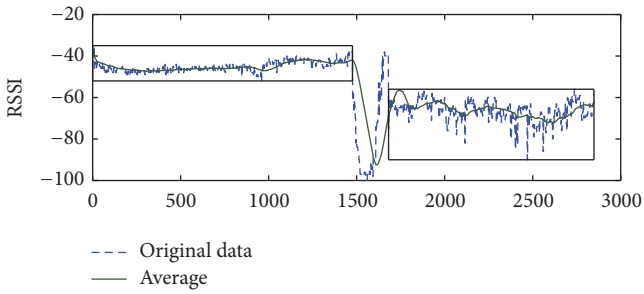

Figure 10: The RSSI sequence.

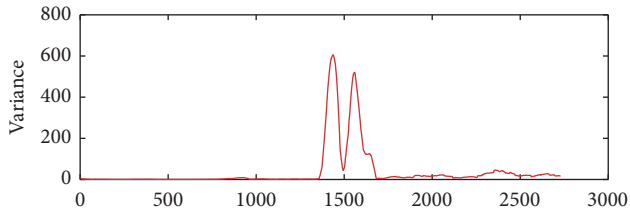

Figure 11: The RSSI sequence variance.

The variance can be used to measure the deviation between the RSSI data and the mean value of the window. The variance of W_i is σ_i which expresses the data fluctuation of W_i. The greater the data fluctuation, the greater the variance.

As shown in Figure 11, the window size is 120, with two peaks in the middle corresponding to the moving process; that is, it corresponds to the parts not in those two boxes in Figure 10. However, the cause of the big variance is not necessarily a person's movement; the stability of the signal will also affect it. Therefore, the slope of the variance curve is used to determine whether the current is moving. The variance increment

$$k(i) = \frac{d_{\sigma_i}}{d_i} = \frac{\sigma_i - \sigma_{i-1}}{i - (i-1)} = \sigma_i - \sigma_{i-1}. \quad (3)$$

In Formula (3), σ_i is the variance of W_i and σ_{i-1} is the variance of W_{i-1}.

The improved results are shown in Figure 12. When $k(i)$ is near to 0, it means that the original variance is stable in a certain range; that also means the mobile phone is not moving or moving in a small range. We set a threshold to detect whether the mobile phone is moving. If $|k(i)| \le K$, the mobile phone is considered to be stable; otherwise it means the position of mobile phone has changed.

Those sequences with a stable position have the following characteristics:

Start point: $[|k(i)| \le K] - \mathrm{ws}(+1)$.

End point: $[|k(i)| > K] - \mathrm{ws}/2$.

Effective Reference AP Selection. In order to improve the accuracy of multiposition detection, it is needed to improve the accuracy of the location. Because of the complexity of the wireless signal transmission in the indoor environment,

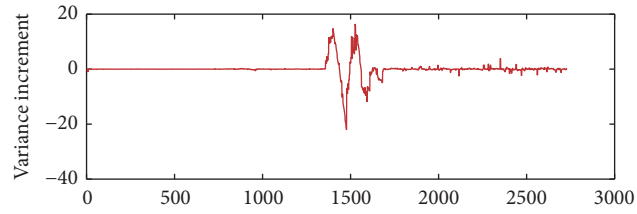

Figure 12: RSSI sequence variance.

the AP signal is not stable. In the network environment, a position can be detected by more than one AP. Therefore, signal stability and the relevance with target AP are the two factors in choosing AP. Relevance here means that both the target AP and the reference AP moving with the mobile phone; that is why the fluctuations of the variance between the target AP and the reference AP should be consistent.

We use dynamic (dynamic time warping, DTW [34]) algorithm to calculate the distance and determine the validity of the reference AP. DTW is a method that calculates an optimal match between two given sequences (e.g., time series) with certain restrictions. The sequences are "warped" nonlinearly in the time dimension to determine a measure of their similarity independent of certain nonlinear variations in the time dimension. This sequence alignment method is often used in time series classification.

As is shown in Figure 13, (a) calculates distance without using dynamic time but (b) uses it; by using dynamic time, (b) can reach the minimum distortion when it comes to calculate the distance.

When selecting the effective reference AP, each AP is considered as the candidate reference AP. The large number of its variance increment is stored as well as the distance between its variance increment sequence and the target's. After getting the distance of all candidate reference APs and target APs, all candidate reference APs will be ordered by the distance. The smaller the distance, the better the effectiveness. Therefore, four candidate reference APs with the minimum distance will be chosen as the reference APs to locate the mobile phone's position. In general, three points are enough for location. In order to prevent that one of the three reference APs from failure, so we choose four reference APs from the candidate lists.

6.2.3. Establishment of Fingerprint Database. The RSSI fingerprint library (RSSI-MAP) is built by the RSSI sequence generated in previous section. RSSI-MAP is shown in Table 3. $R_J = (r_{1,J}, r_{2,J}, \ldots, r_{L,J})$ represent the fingerprint information in RSSI-MAP. J is the position where the mobile phone is stayed for detecting. L is the number of candidate reference APs. r is the fingerprint information of AP, which can be described by triple like $r(\overline{\mathrm{rssi}}, \mathrm{var}, \mathrm{len})$. Items in triple represent the average, variance, and length of RSSI sequence.

6.2.4. Mobile Position Determination. $R_T = (r_{1,T}, r_{2,T}, \ldots, r_{L,T})$ represents RSSI fingerprinting information of the reference APs detected at the position T. $R'_T = r'_{0,T}$ represents the

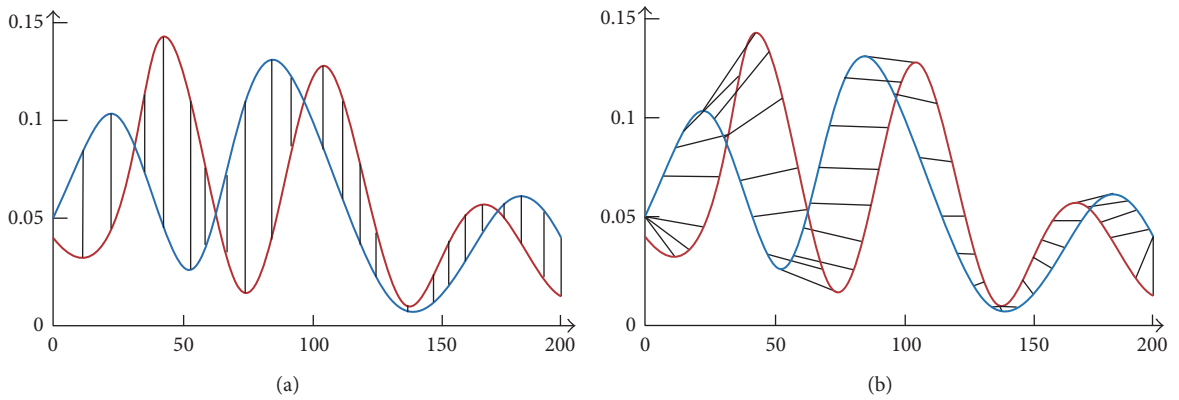

Figure 13: Dynamic time warp (DTW).

Table 3: Structure of RSSI-MAP.

Location	Reference AP	Target AP
1	$R_1 = (r_{1,1}, r_{2,1}, \ldots, r_{L,1})$	$R'_1 = r_{0,1}$
2	$R_2 = (r_{1,2}, r_{2,2}, \ldots, r_{L,2})$	$R'_2 = r_{0,2}$
⋮	⋮	⋮
J	$R_J = (r_{1,J}, r_{2,J}, \ldots, r_{L,J})$	$R'_J = r_{0,J}$

RSSI fingerprinting information of the target AP detected by the position T. $\mathrm{Dist}(R_T, R_J)$ is the distance between R_T and R_J. $\overline{\mathrm{rssi}_{i,T}}$ is the average value of RSSI for reference AP; $\overline{\mathrm{rssi}_{i,J}}$ is the average of the RSSI sequence for reference AP. J is the position where the distance between T and one in RSSI-MAP is the shortest. When there are more than three reference APs, we can locate the mobile phone.

$$\mathrm{Dist}\left(R_T, R_J\right) = \sqrt{\sum_{i=1}^{L} \left(\overline{\mathrm{rssi}_{i,T}} - \overline{\mathrm{rssi}_{i,J}}\right)^2}. \tag{4}$$

$\mathrm{Dist}(R_T, R_J)$ in Formula (4) depend on the number of L, in order to reduce the effect on Dist_T that the number of reference AP is different in different position. The formula is improved as the following.

$$\mathrm{Dist}_T = \min\left[\frac{\mathrm{Dist}\left(R_T, R_J\right)}{L}\right]. \tag{5}$$

When L is greater than or equal to 3, the fingerprint of the first three APs can be used for location by using Formulas (4) and (5). When L is equal to 2, there will be more than one position and all of them have the same distance. Then we should choose the one who is the nearest one with the target AP. When L is equal to 1, in order to increase the accuracy of the positioning, the variance is used to measuring the similarity between position T and position J. From the previous section, the RSSI form one AP at the same position which is approximate normal distribution; that is, the RSSI sequence is represented as follows:

$$P\left(\mathrm{rssi}\right) = \frac{1}{\sqrt{2\pi}\sigma} e^{(\mathrm{rssi}-\mu)^2/2\sigma^2}. \tag{6}$$

In Formula (6), $\sigma = \mathrm{var}$; $\mu = \overline{\mathrm{rssi}}$.

In the information theory, KL [35, 36] (Kullback-Leibler, divergence) can be used to describe the difference between two probability distributions of Q and P; $D_{\mathrm{KL}}(P \parallel Q)$ is the information loss caused by that Q which is used to fit the true distribution P. So the distance between the T and the RSSI probability distribution can be calculated using the KL divergence. KL divergence is defined in

$$D_{\mathrm{KL}}\left(P \parallel Q\right) = \sum P\left(i\right) \ln \frac{P\left(i\right)}{Q\left(i\right)}. \tag{7}$$

So, we can get formula (8) from formula (6) and formula (7).

$$\begin{aligned}\mathrm{Dist}\left(R_T, R_J\right) &= D_{\mathrm{KL}}\left(R_T \parallel R_J\right) \\ &= \sum_{\mathrm{rssi}=-100}^{0} \frac{P\left(\mathrm{rssi}\right)}{2}\left[\frac{\left(\mathrm{rssi}-\mu_1\right)^2}{\sigma_1^2} - \frac{\left(\mathrm{rssi}-\mu_2\right)^2}{\sigma_2^2}\right].\end{aligned} \tag{8}$$

In the formula (8),

$$\begin{aligned}\sigma_1 &= \mathrm{var}_{L,T}, \\ \mu_1 &= \overline{\mathrm{rssi}_{L,T}}, \\ \sigma_2 &= \mathrm{var}_{L,J}, \\ \mu_2 &= \overline{\mathrm{rssi}_{L,J}}, \\ P\left(\mathrm{rssi}\right) &= \frac{1}{\sqrt{2\pi}\sigma_1} e^{(\mathrm{rssi}-\mu_1)^2/2\sigma_1^2}.\end{aligned} \tag{9}$$

Then, according to the distance got by formula (8), the nearest neighbor algorithm is used to find the corresponding position in the J RSSI-MAP.

6.2.5. Legitimacy Judgment. $\max(\overline{\mathrm{rssi}})$ represents the maximum mean of target RSSI at position J. It can be easily query in RSSI-MAP when we find the position J. $\overline{\mathrm{rssi}}$ is the mean value being detected. Then, there is $\mathrm{Diff}_T = \overline{\mathrm{rssi}} - \max(\overline{\mathrm{rssi}})$.

If $\mathrm{Dist}_T \le M$ and $\mathrm{Diff}_T \le 0$, it is safe and there is no fake AP.

If $\mathrm{Dist}_T \le M$ and $\mathrm{Diff}_T > 0$, it is unsafe and there exits fake AP.

If $\mathrm{Dist}_T > M$, fingerprint database should be updated. You can find the details in next section.

6.2.6. Dynamic Update of Fingerprint Database. The dynamic update of RSSI fingerprint database consists of two parts: one is the addition of the new fingerprint, and the other is the update of the existing fingerprint.

The new fingerprint should be added because of various reasons in the training phase of the RSSI fingerprint database. It cannot cover all the spatial subregions of M, so it is necessary to improve the fingerprint database in the later stage.

The update of the existing fingerprint is caused by environmental changes such as survival status of reference AP, the correlation between the candidate reference AP and the target AP, and the change of the reference AP's position. At this point, we need to update the fingerprint information which already exists in the fingerprint database in detection stage.

$$\left[R_J\left(r_{1,J}, r_{2,J}, \ldots, r_{L,J}\right), R'_J\left(r_{0,J}\right)\right]. \tag{10}$$

Assume there are four valid candidate reference APs; they are $\mathrm{AP}_1, \mathrm{AP}_2, \mathrm{AP}_3, \mathrm{AP}_4$, and the relationship or their effectiveness is as the following: $E1 > E2 > E3 > E4$; then there is $\mathrm{Dist}_T = \mathrm{Dist}_T(\mathrm{AP}_1, \mathrm{AP}_2, \mathrm{AP}_3)$. The corresponding position is J.

When there is $\mathrm{Dist}_T > M$

$$\begin{aligned}
\mathrm{Dist}_{T3} &= \mathrm{Dist}_T\left(\mathrm{AP}_1, \mathrm{AP}_2, \mathrm{AP}_4\right),\\
\mathrm{Dist}_{T2} &= \mathrm{Dist}_T\left(\mathrm{AP}_1, \mathrm{AP}_3, \mathrm{AP}_4\right),\\
\mathrm{Dist}_{T1} &= \mathrm{Dist}_T\left(\mathrm{AP}_2, \mathrm{AP}_3, \mathrm{AP}_4\right).
\end{aligned} \tag{11}$$

If $\mathrm{Dist}_{Ti} \le M$, then we can use $r_{i,T}$ instead of $r_{i,J}$ in the RSSI-MAP to update the existing fingerprint. If $\mathrm{Dist}_{Ti} > M$, then put (R_T, R'_T) into the RSSI-MAP. If $\mathrm{Dist}_{Ti} \le M$ and $r_{i,t}\mathrm{len} \ge r_{i,j}\mathrm{len}$, then we can use $r_{i,T}$ instead $r_{i,J}$ in the RSSI-MAP.

7. Evaluation in SPD and MPD

In order to verify the feasibility and effectiveness of the AP Evil-Twin detection method based on RSSI, we implement a number of experiments.

We use the Terminal MX3 to collect RSSI signal. The TL-WR882N is used as the true AP. A fake AP has been simulated by hostapd in a notebook. The experiment is done in a room with 100 square meters. In the detection phase, we set the different $F - R$ ($F - R$ is defined as the mean difference, resp., between the fake AP and the true AP's RSSI. The mean difference is equal to the distance between two APs.).

7.1. Experiment and Assessment for Single Position Detection

Discussion of Sliding Window Size. The previous section shows the size of the sliding window affects the delay rate and false negative rate of detection. That means the bigger the window, the higher the delay rate, and the higher the false negative rate. In order to find a suitable value for the size of sliding window, we design an experiment like the following.

In order to verify the effect of window size on the delay, we set the mean difference, respectively, between the fake AP and the true RSSI as 25 and 10; that is, $F - R = 25$ and $F - R = 10$. The window size in turn is 1, 40, 80, 120, 160, 200, and 240. The safety threshold value for each round of detection is the maximum mean of RSSI in 30 minutes. There are 14 sets of experiment; each set of experiment will be done 30 times, and the result is as shown in Figure 14. From (a) we can see that when the difference of mean between true AP and fake AP is bigger, the delay rate is smaller. When the window size is 120, the average delay time is less than 20 s.

To verify the effect of window size on accuracy, when it is in the condition that $F - R = 10$, we set the windows size in turn: 1, 40, 80, 120, 160, 200, and 240. After the test program running 10 minutes, open the fake AP and let it run for 3 minutes then close it for 3 minutes, because it needs a certain delay that the mean value is changed from abnormal status to normal status.

The mean from abnormal status returning to normal needs a certain delay, so if there occurs wrong or missed detection in every 3 minutes after the delay time, it will be assumed as a wrong one. If there is wrong or missed detecting after delayed time, it is considered as the error status. This experiment is done 50 times, and the result is shown on the right in Figure 14. According to the experiment results, when the window size is 80, 120, and 160, the accuracy is more than 98%. If the windows size is too small or too big, the accuracy is lower since the false positive rate is higher.

Discussion of Threshold Value. In this experiment, we set the window size as 120 and the $F - R$ as 25 or 10. Assume that the threshold value is Rmax, Rmax $-$ 2, Rmax + 2, Rmax + 4, and Rmax + 8. So there are 10 sets of experiment. In each experiment the following step is done 50 times. After the test program running 10 minutes, open the fake AP and let it run for 3 minutes and then close it for 3 minutes. We can get the result of this experiment from Figure 15, when the security threshold value is Rmax and the accuracy is up to 96%. When the security threshold value is Rmax + 2, the accuracy of the condition is $F - R = 25$ up to 100% and $F - R = 25$ is 99%.

Discussion of Distance. In this experiment, we set $F - R =$ 0, 5, 10, 15, and 20, and the threshold value is Rmax. Each experiment is to be done as the following step 50 times. After the test program is running for 10 minutes, open the fake AP and let it run for 3 minutes and then close it for 3 minutes. We can get the result of this experiment from Figure 16. When $F - R = 10$, the accuracy is more than 96%; the missing rate is less than 3%.

7.2. Experiment and Evaluation of Multiposition Cooperative Detection

Validation of Variance Increment Method. In this experiment, the window size is 120, and K is 4; then split the RSSI sequence using Variance increment method. The result is shown in Table 4. Dropping out the fragment whose length is shorter than 120, then we can get two effective RSSI sequence fragments (S_1 and S_10), the total length is 2598,

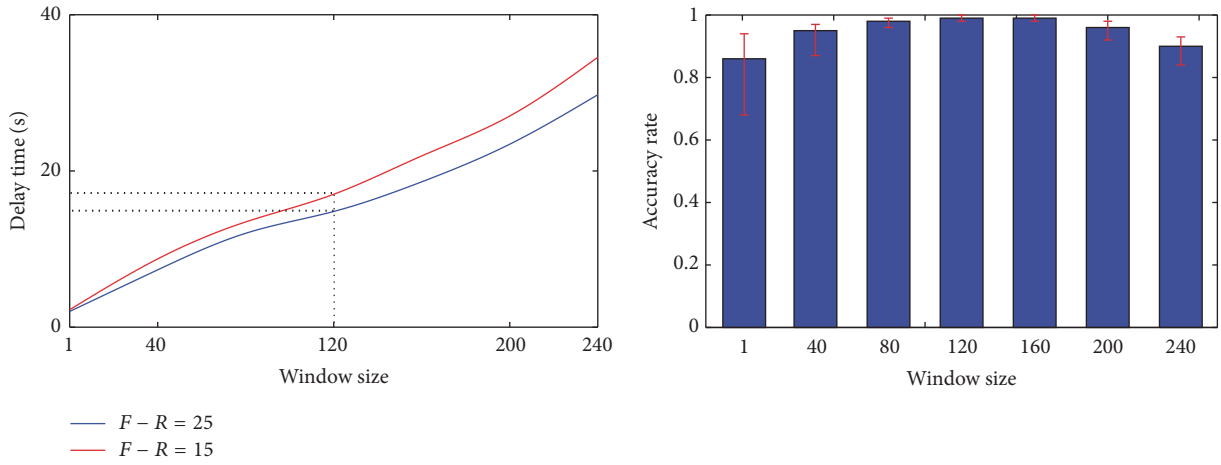

Figure 14: Effect of window size on delay and accuracy.

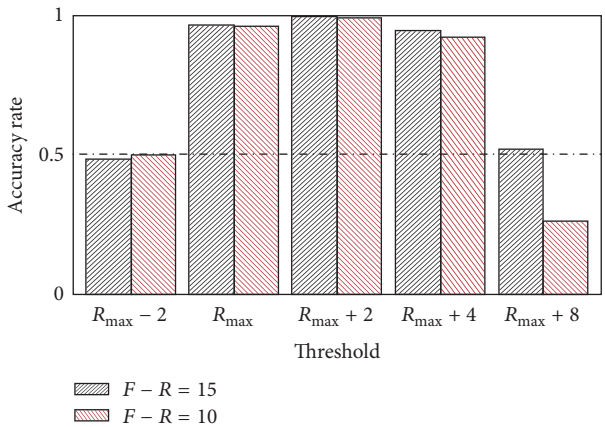

Figure 15: Effect of safety threshold on the accuracy of detection.

Table 4: First time to split the RSSI sequence.

Flag	Range	Length	Range	Mean
S_1	1–1422	1422	[−52, −35]	−45.15
S_2	1366–1431	66	[−44, −39]	−42.5
S_3	1424–1502	79	[−84, −38]	−50.04
S_4	1489–1560	72	[−100, −64]	−91.17
S_5	1507–1569	63	[−100, −87]	−95.95
S_6	1552–1620	69	[−100, −72]	−90.91
S_7	1609–1718	110	[−76, −38]	−56.54
S_8	1660–1726	67	[−75, −40]	−56.68
S_9	1669–1731	63	[−75, −40]	−59.95
S_10	1861–2848	1168	[−90, −56]	−66.37

and the effective fragment length was 2605 in the original data sequence. So the accuracy is 99.7%.

The Validity of DTW Algorithm. To verify that the DTW algorithm could be used to choose the valid AP, we open the detecting software which could find all the AP and get their RSSI. Then we let the detecting software move with the speed of 1.5 m, staying at three different locations and staying at each place for 15 minutes. At the end, there are 28 APs being found, including 1 target AP and 27 candidate reference APs. For each of 27 candidate reference APs, we use DTW algorithm to calculate the distance of variance increment sequence between target AP and it. Finally, we are successful to find four suitable reference APs.

The Validity of Localization Algorithm. In a room with 100 square meters, we collect a set of data per 4 square meters. So there are 25 sets of data. In detecting stage, we stayed at every position for 5 minutes, then moving to another position with the speed of 1.5 m/s. For the four suitable reference AP found in previous section, there are three kinds of conditions; that is, the first 4 AP should be considered as the reference AP, and the first 3 and the first 2, respectively, calculate their Euclidean distance. When there is only one reference AP, the accuracy of location is 62%. When there are two reference APs, the accuracy of location is 85%. When there are three reference APs, the accuracy of location is 90%.

The Validity of Multiposition Cooperative Detection. We play a role of an attack, simulating a fake AP in a notebook. And the experiment is done still in a room with 100 square meters, dividing it into 25 regions. In each region, we collect data for every 30 minutes and use the maximum mean of this region as the safety threshold. In detecting stage, we stayed at every position for 5 minutes, then moving to another position with the speed of 1.5 m/s. Experiments were carried out for 200 times, 100 times is to open the fake AP, and the other 100 times is to turn off the fake AP. When the fake AP is turned on, if there is any position detected by the fake AP, then the detection is successful, if all the positions are not detected by the fake AP, then the detection fails. Close the fake AP; if there is any position to detect the false AP, then the detection fails; if all the positions are not detected in the fake AP, then the

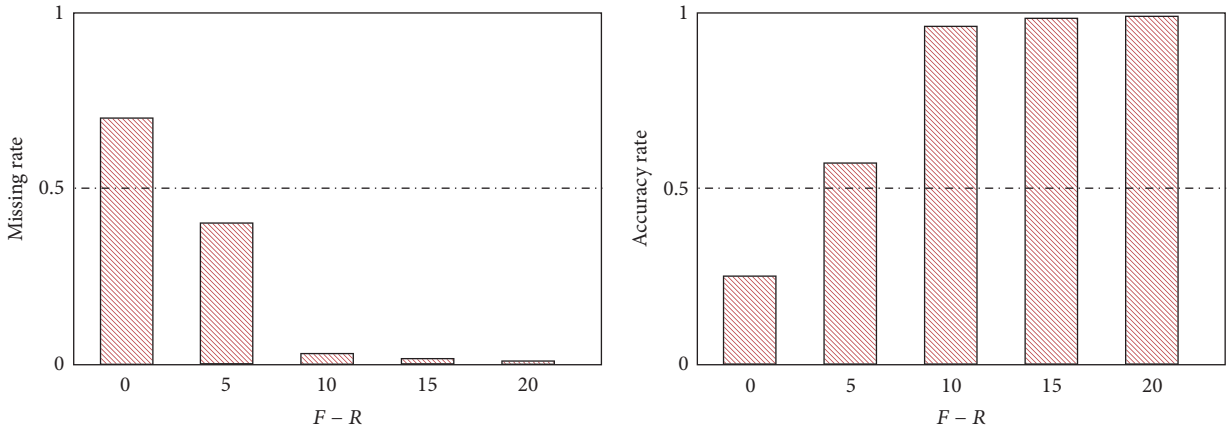

Figure 16: Effect of distance on the detection results.

detection is successful. When there is only one reference AP, the accuracy of location is 58%. When there are two reference APs, the accuracy of location is 80%. When there are three reference APs, the accuracy of location is 90%.

8. Related Work

At present, most Evil-Twin detection methods work for the public Wi-Fi environment. There are two key approaches in this domain. One is based on hardware feature; the other is flow feature.

The hardware feature testing method utilizes the characteristic that different network card chips and different drives possess different fingerprint features to set up a fingerprint feature library and decide whether the fake AP existed or not through matching fingerprint data in the fingerprint feature library during testing. Bratus et al. [9] send some SIMULATING frames which possess false formats but are not prohibited by a standard protocol. Although different network card chips or drives have different responses to various SIMULATING frames, the testing method is easy to be found by an intruder. McCoy et al. [11] characterize the drivers during the "active scanning period." This method is undefined in the IEEE 802.11 standard on the frequency and order of sending probe requests. Therefore, each manufacturer employs its own algorithm. This technique cannot distinguish between two devices using the same network card and driver. So this technique may not be used for identifying individual devices. However, the attacker cannot forge the position of the Real AP. In smart homes, the intuition underlying our design is that each Real AP has its fixed position, and the attacker cannot put the fake AP exactly in the right place. Desmond et al. [12] used fingerprint client station, which sends probe requests in light of periodic characteristic by surveying probe requests. The period itself is attached to slight variations. Far from being consistent, these variations can be clustered. With enough detection time, each cluster slowly derives, with a slope proportional to the time skew. This work is able to particularly identify client station; however, this requires more than one hour of traffic and is only applicable to client stations. In a word, McCoy et al. [11] and Desmond et al. [12] utilize the characteristic that different wireless network cards send different probe request frames with different periods during scanning to set up the fingerprint library. As the equipment only sends a small number of probe request during joining the network and the method can be valid when passive scanning is used, the expensive time overhead and the relatively bad real-time property are involved; Neumann et al. [13] utilize the arrival time of interframe space to identify the wireless equipment, but the characteristic can be faked by the intruder and the testing method based on the characteristic can be bypassed. The testing methods for the hardware fingerprint feature of the equipment above-mentioned cut both ways: various fake AP can be tested effectively and the cost of faking the hardware feature of the intruder is relatively high; the fingerprint database can be built in many ways [37], but the cost of building the hardware feature fingerprint library is high, the time for extracting the hardware fingerprint is long, the testing real-time property is worse, and the expansibility is bad. However, Our approach builds the feature fingerprint library without collecting deliberately. You will achieve the feature fingerprint library as soon as you open the phone.

According to the flow feature testing method, the network flow feature is different when the fake AP is existent or nonexistent; so, whether Evil-Twin AP is existent or not can be tested. The method is excellent in extendibility but also has some disadvantages. Beyah et al. [14] utilize the arrival time space of each data packet to build a flow feature library; as the method is influenced by flow shaping greatly, the practical operation and the applicability are not good; Wei et al. [15] propose that the arrival time of the ACK data packet in a TCP protocol can be used to set up the flow feature library; as the arrival time is influenced by TCP, the testing efficiency is limited; Sheng et al. [16–18] propose that data round trip time can be used to test whether the fake AP is existent or not, but the data round trip time is influenced by the network type, the bandwidth, and the state of congestion at the same time.

Besides, Han et al. [38] put forward the wireless fake AP attack in an in-vehicle network and, meanwhile, give the testing method based on RSSI. The method requires that all

of the APs are equipped with GPS modules to report their own positions; a user judges whether the fake AP is existent or not through whether the measured RSSI is matched with the position or not. The method can effectively test the fake AP attack in the in-vehicle network but is not suitable for indoor environment because the GPS signal is weakened, even shielded, indoors.

9. Conclusions

This paper has presented a novel approach to detect fake APs in a smart home environment. Our approach uses RSSI as the fingerprint of the authentic AP to detect fake APs. We have proposed two methods to identity fake APs in two different scenarios where the genuine AP locates on a single, fixed, or multiple positions. Our experimental results show that our approach can detect 90% of the fake APs with little extra overhead to the communication delay time.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This work was partly supported by the National Natural Science Foundation of China under Grant Agreement no. 61672427, no. 61672428, and no. 61272461, the Key Project of Chinese Ministry of Education under Grant Agreement no. 211181, the International Cooperation Foundation of Shaanxi Province, China, under Grant Agreement no. 2013KW01-02 and no. 2015 KW-003, the Research Project of Shaanxi Province Department of Education under Grant no. 15JK1734, the Research Project of NWU, China (no. 14NW28), and the UK Engineering and Physical Sciences Research Council (EPSRC) under Grants EP/M01567X/1 (SANDeRs) and EP/M015793/1 (DIVIDEND).

References

[1] M. Chan, D. Estève, C. Escriba, and E. Campo, "A review of smart homes—present state and future challenges," *Computer Methods and Programs in Biomedicine*, vol. 91, no. 1, pp. 55–81, 2008.

[2] J. Schulz-Zander, L. Suresh, N. Sarrar, A. Feldmann, T. Hühn, and R. Merz, "Programmatic orchestration of wifi networks," in *Proceedings of the USENIX Annual Technical Conference (USENIX ATC '14)*, pp. 347–358, USENIX Association, Philadelphia, Pa, USA, June 2014.

[3] F. Lanze, A. Panchenko, I. Ponce-Alcaide, and T. Engel, "Undesired relatives: protection mechanisms against the evil twin attack in IEEE 802.11," in *Proceedings of the 10th ACM Symposium on QoS and Security for Wireless and Mobile Networks (Q2SWinet '14)*, pp. 87–94, ACM, Québec, Canada, September 2014.

[4] J. Zhang, X. Zheng, Z. Tang et al., "Privacy leakage in mobile sensing: your unlock passwords can be leaked through wireless hotspot functionality," *Mobile Information Systems*, vol. 2016, Article ID 8793025, 14 pages, 2016.

[5] D. A. D. Zovi and S. A. Macaulay, "Attacking automatic wireless network selection," in *Proceedings of the 6th Annual IEEE System, Man and Cybernetics Information Assurance Workshop (SMC '05)*, pp. 365–372, West Point, NY, USA, June 2005.

[6] Q. Zhang, L. T. Yang, and Z. Chen, "Privacy preserving deep computation model on cloud for big data feature learning," *IEEE Transactions on Computers*, vol. 65, no. 5, pp. 1351–1362, 2016.

[7] J. Herzen, R. Merz, and P. Thiran, "Distributed spectrum assignment for home WLANs," in *Proceedings of the 32nd IEEE Conference on Computer Communications (INFOCOM '13)*, pp. 1573–1581, Turin, Italy, April 2013.

[8] O. Nakhila, E. Dondyk, M. F. Amjad, and C. Zou, "User-side Wi-Fi evil twin attack detection using SSL/TCP protocols," in *Proceedings of the 12th Annual IEEE Consumer Communications and Networking Conference (CCNC '15)*, pp. 239–244, IEEE, January 2015.

[9] S. Bratus, C. Cornelius, D. Kotz, and D. Peebles, "Active behavioral fingerprinting of wireless devices," in *Proceedings of the 1st ACM Conference on Wireless Network Security (WiSec '08)*, pp. 56–61, New York, NY, USA, 2008.

[10] J. Cache, "Fingerprinting 802.11 implementations via statistical analysis of the duration field," *Uninformed.org*, vol. 5, 2006.

[11] D. McCoy, J. Franklin, J. Van Randwyk, D. Sicker, and P. Tabriz, Passive data-link layer 802.11 wireless device driver fingerprinting, January 2006.

[12] L. C. C. Desmond, C. C. Yuan, T. C. Pheng, and R. S. Lee, "Identifying unique devices through wireless fingerprinting," in *Proceedings of the ACM Conference on Wireless Network Security (WISEC '08)*, pp. 46–55, Alexandria, Va, USA, April 2008.

[13] C. Neumann, O. Heen, and S. Onno, "An empirical study of passive 802.11 device fingerprinting," in *Proceedings of the 32nd IEEE International Conference on Distributed Computing Systems Workshops (ICDCSW '12)*, pp. 593–602, Macau, China, June 2012.

[14] R. Beyah, S. Kangude, G. Yu, B. Strickland, and J. Copeland, "Rogue access point detection using temporal traffic characteristics," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '04)*, vol. 4, pp. 2271–2275, Dallas, Tex, USA, November 2004.

[15] W. Wei, K. Suh, B. Wang, Y. Gu, J. Kurose, and D. Towsley, "Passive online rogue access point detection using sequential hypothesis testing with TCP ACK-pairs," in *Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement (IMC '07)*, pp. 365–378, ACM, San Diego, Calif, USA, October 2007.

[16] H. Han, B. Sheng, C. C. Tan, Q. Li, and S. Lu, "A timing-based scheme for rogue AP detection," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 11, pp. 1912–1925, 2011.

[17] C. D. Mano, A. Blaich, Q. Liao et al., "Ripps: rogue identifying packet payload slicer detecting unauthorized wireless hosts through network traffic conditioning," *ACM Transactions on Information and System Security*, vol. 11, no. 2, article no. 2, 2008.

[18] G. Qu and M. M. Nefcy, "RAPiD: an indirect rogue access points detection system," in *Proceedings of the IEEE 29th International Performance Computing and Communications Conference (IPCCC '10)*, pp. 9–16, IEEE, December 2010.

[19] K. S. A. P. Levis, "Rssi is under appreciated," in *Proceedings of the 3rd Workshop on Embedded Networked Sensors*, vol. 3031, p. 239242, Cambridge, Mass, USA, 2006.

[20] D. Kotz, C. Newport, R. S. Gray, J. Liu, Y. Yuan, and C. Elliott, "Experimental evaluation of wireless simulation assumptions," in *Proceedings of the 7th ACM Symposium on Modeling, Analysis*

and Simulation of Wireless and Mobile Systems (ACM MSWiM '04), pp. 78–82, ACM, Venice, Italy, October 2004.

[21] N. Patwari, A. O. Hero III, M. Perkins, N. S. Correal, and R. J. O'Dea, "Relative location estimation in wireless sensor networks," *IEEE Transactions on Signal Processing*, vol. 51, no. 8, pp. 2137–2148, 2003.

[22] M. Kotaru, K. Joshi, D. Bharadia, and S. Katti, "Spotfi: decimeter level localization using wifi," *SIGCOMM Computer Communication Review*, vol. 45, no. 4, pp. 269–282, 2015.

[23] K. Wu, J. Xiao, Y. Yi, M. Gao, and L. M. Ni, "FILA: fine-grained indoor localization," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM '12)*, pp. 2210–2218, Orlando, Fla, USA, March 2012.

[24] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, "Zee: zero-effort crowdsourcing for indoor localization," in *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking (MobiCom '12)*, pp. 293–304, ACM, Istanbul, Turkey, August 2012.

[25] H. Liu, Y. Gan, J. Yang et al., "Push the limit of WiFi based localization for smartphones," in *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking (MobiCom '12)*, pp. 305–316, ACM, Istanbul, Turkey, August 2012.

[26] S. Sen, J. Lee, K.-H. Kim, and P. Congdon, "Avoiding multipath to revive inbuilding WiFi localization," in *Proceedings of the 11th Annual International Conference on Mobile Systems, Applications, and Services (MobiSys '13)*, pp. 249–262, ACM, Taipei, Taiwan, June 2013.

[27] J. Schulz-Zander, C. Mayer, B. Ciobotaru, S. Schmid, A. Feldmann, and R. Riggio, "Programming the home and enterprise WiFi with OpenSDWN," in *Proceedings of the ACM Conference on Special Interest Group on Data Communication (SIGCOMM '15)*, pp. 117–118, ACM, London, UK, August 2015.

[28] N. Borisov, I. Goldberg, and D. Wagner, "Intercepting mobile communications: the insecurity of 802.11," in *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking (MobiCom '01)*, pp. 180–189, Rome, Italy, 2001.

[29] E. Tews, R. P. Weinmann, and A. Pyshkin, "Breaking 104 bit wep in less than 60 seconds," in *Proceedings of the Information Security Applications, International Workshop (Wisa '07)*, pp. 188–202, Jeju Island, Korea, August 2007.

[30] Q. Wang, Y. Zhang, X. Lu, Z. Wang, Z. Qin, and K. Ren, "Real-time and spatio-temporal crowd-sourced social network data publishing with differential privacy," *IEEE Transactions on Dependable and Secure Computing*, 2016.

[31] Q. Zhang, H. Zhong, L. T. Yang, Z. Chen, and F. Bu, "Privacy preserving highorder cfs algorithm on the cloud for clustering multimedia data," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 12, no. 4s, pp. 66:1–66:15, 2016.

[32] G. Wang, Y. Zou, Z. Zhou, K. Wu, and L. M. Ni, "We can hear you with Wi-Fi!," in *Proceedings of the ACM International Conference on Mobile Computing and Networking (MobiCom '14)*, pp. 593–604, Maui, Hawaii, USA, September 2014.

[33] J. Wang, X. Chen, D. Fang, C. Q. Wu, Z. Yang, and T. Xing, "Transferring compressive-sensing-based device-free localization across target diversity," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 4, pp. 2397–2409, 2015.

[34] J. Wang and D. Katabi, "Dude, where's my card? RFID positioning that works with multipath and non-line of sight," in *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication (SIGCOMM '13)*, pp. 51–62, ACM, August 2013.

[35] S. Kullback and R. A. Leibler, "On information and sufficiency," *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[36] S. Kullback, "Letter to the editor: the kullback-leibler distance," *The American Statistician*, vol. 41, no. 4, pp. 340–341, 1987.

[37] Y. Wen, X. Tian, X. Wang, and S. Lu, "Fundamental limits of RSS fingerprinting based indoor localization," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM '15)*, pp. 2479–2487, Hong Kong, May 2015.

[38] H. Han, F. Xu, C. C. Tan, Y. Zhang, and Q. Li, "Defending against vehicular rogue aps," in *Proceedings of the IEEE INFOCOM*, pp. 1665–1673, April 2011.