Virtual Multipath Attack and Defense for Location Distinction in Wireless Networks

Song Fang, Yao Liu, Wenbo Shen, Haojin Zhu and Tao Wang

Abstract—In wireless networks, location distinction aims to detect location changes or facilitate authentication of wireless users. To achieve location distinction, recent research has focused on investigating the spatial uncorrelation property of wireless channels. Specifically, differences in wireless channel characteristics are used to distinguish locations or identify location changes. However, we discover a new attack against all existing location distinction approaches that are built on the spatial uncorrelation property of wireless channels. In such an attack, the adversary can easily hide her location changes or impersonate movements by injecting fake wireless channel characteristics into a target receiver. To defend against this attack, we propose a detection technique that utilizes an auxiliary receiver or antenna to identify these fake channel characteristics. We also discuss such attacks and corresponding defenses in OFDM systems. Experimental results on our USRP-based prototype show that the discovered attack can craft any desired channel characteristic with a successful probability of 95.0% to defeat spatial uncorrelation based location distinction schemes and our novel detection method achieves a detection rate higher than 91.2% while maintaining a very low false alarm rate.

Index Terms—Channel impulse response, multipath, security, MIMO, OFDM.

1 Introduction

Location distinction in wireless networks aims to detect a wireless user's location change, movement or facilitate location-based authentication. Enforcing location distinction is important for many wireless applications [1], [2]. For example,

- Wireless sensor networks are usually utilized to monitor a target area by sensing the physical or environmental conditions (e.g., temperature, sound, and pressure). Administrators of the sensor networks would like to enforce location distinction to prevent an unauthorized person from moving the sensors away from the area of interest.
- Wireless networks are vulnerable to sybil attacks due to the broadcast nature of the wireless medium [3]. Here, an adversary forges a significant amount of fake user identities to fool a networked system. Location distinction can tell whether or not all identities are originated from the same location, and thus detect such attacks.
- Active radio frequency identification (RFID) tags are often used in warehouses for tracking inventory and maintaining the physical security. It has been assumed that "location distinction is critical to provide a warning and to be able to focus resources (e.g., security, cameras, and personnel) on moving objects" [1].

Location distinction using wireless physical layer information has been extensively studied during the past several years (e.g., [1]–[6]). Scientists have discovered that wireless channel characteristics become uncorrelated every half carrier wavelength over distance (spatial uncorrelation property) [7]. This property has been widely explored and adopted to enforce location distinction of wireless devices (e.g., [1]–[6]). Specifically, changes of wireless channel characteristics have been utilized to identify location changes of a wireless transmitter.

In our study, however, we discover a new attack against all existing location distinction approaches built on the spatial uncorrelation property of wireless channels. By launching such an attack, the adversary can generate any chosen wireless channel characteristics at a target receiver to deteriorate the location distinction capability of the receiver. The key idea of the discovered attack is to create a *virtual multipath channel* as undetectable camouflage to make the receiver believe a specified channel characteristic chosen by the attacker.

To demonstrate the virtual multipath channel, we first explain the multipath effect, which is the fundamental reason for the spatial uncorrelation property. Wireless signals normally propagate in the air through multiple paths due to obstacle reflection, diffraction, and scattering [1]. Therefore, for wireless signals sent from different locations, the receiver can observe different channel characteristics from these signals, because they experience different multipaths and accordingly undergo different channel effects (e.g, power attenuation, phase shifting, and delay). To fool a receiver, the attacker needs to create an "artificial channel" that can exhibit a multipath propagation feature similar to the real-world multipath.

We give an example to illustrate how the attacker can create such a channel. Figure 1(a) shows a simple real multipath scenario, where a signal sent by the transmitter travels on two paths, i.e., the reflection path and the direct path. At time t_0, the receiver starts to receive the signal copy that travels on the direct path. The reflection path is longer than the direct path, and thus at a later time $t_0 + \Delta_t$, the receivers receives the aggregation of the signal copy from the direct path and the one from the reflection path.

- *Song Fang, Yao Liu and Tao Wang are with the Department of Computer Science and Engineering, University of South Florida, Tampa, FL. E-mail: songf@mail.usf.edu, yliu@cse.usf.edu and taow@mail.usf.edu.*
- *Wenbo Shen is is with the Department of Computer Science, North Carolina State University, Raleigh, NC. E-mail: wshen3@ncsu.edu.*
- *Haojin Zhu is is with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, China. E-mail: zhu-hj@cs.sjtu.edu.cn.*

An earlier version of the work was published in MobiCom'14.

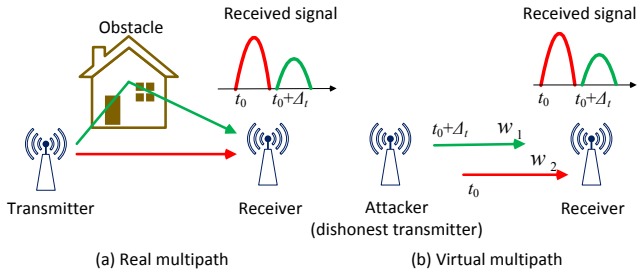


Fig. 1. Creating a virtual multipath.

Now consider the scenario in Figure 1(b): there is only one direct path between the attacker (i.e., a dishonest transmitter) and the receiver, but the attacker wants to make the receiver believe that two paths exist similar to the real multipath propagation shown in Figure 1(a). To this end, the attacker sends the signal alone first. After duration Δ_t, she superimposes a fresh signal copy onto the one already in transmission. The attacker scales both the original signal and the time-delayed copy by attenuation factors w_1 and w_2 to mimic the signal amplitude attenuation caused by real paths. Consequently, the receiver observes an aggregation of one signal plus a time-delayed copy, with each undergoing a certain amplitude attenuation, and thus thinks that they are caused by the multipath effect.

The example in Figure 1(b) assumes that there exists only one direct path between the attacker and the receiver (i.e., no multipath effect is considered). In practice, the attacker's crafted multipath signal is affected by the real multipath effect as well, and she should have a way to deal with the impact of this real multipath. Our research reveals that the attacker can easily achieve this goal by reverse-engineering existing wireless channel estimation algorithms and performing linear transformations on the original signal.

To defend against this attack, we propose a detection technique utilizing an auxiliary receiver (or antenna) at a different location to identify the virtual multipath channels and the fake channel characteristics. Specifically, the attacker must craft its transmitting signal to make the target receiver believe a particular channel characteristic. However, we show that this crafted signal exhibits inconsistent channel characteristics to the auxiliary receiver. Based on this result, we create a defense scheme that does not require the receivers to have any prior knowledge about the real channel characteristics between themselves and the transmitter.

We perform real-world experimental evaluation on the Universal Software Radio Peripherals (USRPs). Experimental results show that an attacker, by using the virtual multipath channel as camouflage, can fool a target to believe any desired channel characteristic with successful probability of 95.0%. However, our defense can discover this attack with probability more than 91.2% and the false alarm rate can be reduced to 0 with a carefully chosen detection threshold. The experimental results suggest the discovered attack is a real threat to existing location distinction schemes using the spatial uncorrelation property, and demonstrate the success of the defense approach. Our contributions are summarized as follows.

- We discover that multipath propagation can be artificially made in a lab environment, and create a technique that can successfully generate virtual multipath channels.
- Based on the virtual multipath channel, we identify a new type of attack that can defeat all existing location distinction algorithms using the spatial uncorrelated property of wireless channels.
- We create a defense technique to detect such attacks and protect location distinction systems. We specifically explore such attacks in OFDM systems and craft corresponding defenses according to the objective of attackers.
- We implement real-world prototypes to examine the practical impact of the attacks and the effectiveness of the proposed defense method.

2 Preliminaries

In this section, we show how location distinction is usually enforced and introduce the prevalent algorithms that are used to estimate wireless channel characteristics.

2.1 Channel Impulse Response

As discussed, a wireless signal usually propagates in the air along multiple paths due to reflection, diffraction, and scattering. A receiver then receives multiple copies of the signal from different paths, each of which has a different delay due to the path it traverses. The received signal is the sum of these time delayed copies. Each path imposes a *response* (e.g., delay and attenuation) on the signal traveling along it [1], and the superposition of all responses between two nodes is referred to as a *channel impulse response* [8]. Wireless channels can be characterized by channel impulse responses.

The multipath effects of different wireless links are different, and so are the channel impulse responses [1]. Due to this reason, a channel impulse response has been utilized to provide location distinction [1], [2]. Specifically, to determine if the transmitter has changed its location, the receiver estimates the channel impulse response of a newly received signal and compares it with the previous estimation result. The location change is detected if the difference between the newly estimated channel impulse response and the previous one exceeds a certain threshold.

2.2 Estimating Channel Impulse Responses

Estimating channel impulse responses is a must-have function for most modern wireless systems [8], [9]. Note that the signal propagation paths are unresolvable (i.e., each multipath component signal can not be extracted from the composite signal) if the differences between the arrival times of the signals traveling on these paths are much smaller than the symbol duration, which is the transmission time of a wireless physical-layer unit [8]. Hence, existing channel estimation algorithms assume a resolvable multipath, i.e., the arrival times of signal copies traveling on different paths are larger than the symbol duration.

Channel impulse responses are usually estimated using training sequences [10]. Specifically, the transmitter sends a training sequence (i.e., a sequence of bits) over the wireless

channel, while the receiver uses the same training sequence and the corresponding received signal samples to estimate the channel impulse response. The training sequence can be pre-shared [10] or reconstructed from the received signal [1].

The physical layer channel estimation can be processed in either frequency (e.g. [1], [2]) or time domain (e.g., [10]), which are inter-convertible due to the linear relation between the two domains. In the following, we describe the channel estimation method in the time domain.

Mathematical Formulation: Channel impulse response estimation exploits the (known) training sequence and corresponding received samples. The transmitter converts the training sequence into M physical layer symbols (i.e., complex numbers that are transmission units at the physical layer [8]). The transmitter then sends the M symbols to the wireless channel. Let $\mathbf{x} = [x_1, x_2, ..., x_M]$ denote the transmitted symbols in the training sequence. Assume that there exist at most L resolvable paths (L can be computed based on practice wireless system configurations [8]). Thus, the receiver can receive L copies of $\mathbf{x}$, each traveling on one path and undergoing a response caused by the corresponding path. The vector $\mathbf{y}$ of received symbols is the convolution sum of the L copies of $\mathbf{x}$. Let $\mathbf{h} = [h_1, h_2, ..., h_L]^T$ be the channel impulse response, where h_i is the response of the i-th path. The received symbols $\mathbf{y}$ can be represented by [10]

$$\mathbf{y} = \mathbf{h} * \mathbf{x} + \mathbf{n}, \tag{1}$$

where $\mathbf{n}$ is the noise and $*$ is the convolution operator. The matrix form of Equation (1) is

$$\mathbf{y} = \begin{bmatrix} x_1 & 0 & \cdot & 0 \\ x_2 & x_1 & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot \\ x_L & \cdot & \cdot & x_1 \\ \cdot & \cdot & \cdot & \cdot \\ x_M & \cdot & \cdot & x_{M-L+1} \\ 0 & x_M & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & \cdot & x_M \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ \cdot \\ \cdot \\ h_L \end{bmatrix} + \mathbf{n} \tag{2}$$

Rewriting Equation (2) in a compact matrix form yields

$$\mathbf{y} = \mathbf{X}\mathbf{h} + \mathbf{n}, \tag{3}$$

where $\mathbf{X}$ is a $(L + M - 1) \times L$ Toeplitz matrix, containing L delayed versions of the transmitted symbols $\mathbf{x}$, and $\mathbf{y}$ is a vector consisting of $(L + M - 1)$ received symbols.

Estimation: Two estimators are generally used to estimate $\mathbf{h}$ from Equation (3): least-square (LS) and linear minimum mean squared error (LMMSE) [11]. LS is given by $\hat{\mathbf{h}}_{\text{LS}} = (\mathbf{X}^H\mathbf{X})^{-1}\mathbf{X}^H\mathbf{y}$, where $(\cdot)^H$ and $(\cdot)^{-1}$ are the conjugate transpose and matrix inverse operators [12]. LMMSE is written as $\hat{\mathbf{h}}_{\text{LMMSE}} = \mathbf{R}_h(\mathbf{R}_h + \sigma_n^2(\mathbf{X}\mathbf{X}^H)^{-1})^{-1}\hat{\mathbf{h}}_{\text{LS}}$, where $\mathbf{R}_h$ is the multipath channel correlation matrix (i.e., the statistical expectation of $\mathbf{h}\mathbf{h}^H$) and σ_n^2 is the variance of the noise [13], both assumed prior knowledge. If the correlation matrix $\mathbf{R}_h$ and noise variance σ_n^2 are both known, LMMSE is used; otherwise, LS is used. We here focus on the LS estimator, because for location distinction schemes in a realistic environment, precise channel correlation statistics and noise knowledge are difficult to obtain due to the time-variant property of wireless channels and potential movements of wireless nodes.

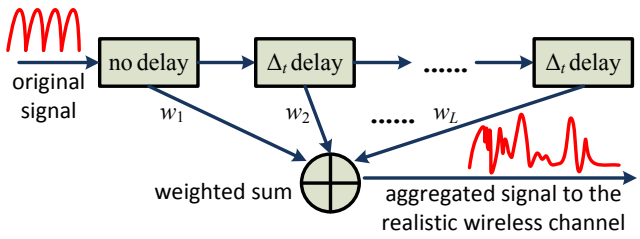


Fig. 2. The delay-and-sum process

3 Assumptions and Attack Model

The location distinction system consists of a transmitter and a receiver. Both are equipped with radio interfaces that can transmit and receive wireless signals. The receiver aims to verify whether or not the transmitter has changed location. Towards this goal, the receiver estimates the channel impulse response from a wireless signal received from the transmitter, and then compares it with the previous estimation results to generate a decision. To constantly enforce the location distinction, the receiver periodically sends an inquiry to the transmitter, and the transmitter responds to the inquiry by sending wireless signals back to the receiver.

We assume that the transmitter is malicious and aims to hide her location change or impersonate movements while she is actually static. To achieve this objective, the transmitter attempts to mislead the receiver through creating a virtual multipath channel, which can fool the receiver to estimate a fake wireless channel impulse response chosen by the transmitter. We assume that the malicious transmitter knows the training sequence used for the channel estimation.

We assume that the channel impulse response is stable in a short period of time (e.g., a packet duration), which is a common assumption for designing wireless communications. We further assume that the malicious transmitter knows the actual channel impulse response between herself and the receiver. This can be achieved by estimating the channel impulse response from the wireless signals (e.g., location distinction inquiries) emitted by the receiver.

4 Virtual Multipath Attack

In this section, we describe how to create a virtual multipath channel to defeat location distinction algorithms. The attacker can launch two types of attacks. In a basic attack, the attacker can use any weights to craft a virtual multipath signal. This will fool the receiver to obtain random, incorrect estimates of the channel impulse response. In an advanced attack, with the knowledge of the real channel impulse response between herself and the receiver, the attacker is able to compute exact weights that make the receiver estimate the chosen channel impulse responses specified by the attacker. In the following discussion, we focus on the advanced attack due to the more misleading nature of such attacks.

4.1 Overview of The Attack

To launch the attack, the attacker needs to know when to add a delayed copy into the transmitting signal. According to Equation 2, the channel estimator models each path by delaying it for one symbol duration. Specifically, the i-th arrived signal copy arrives at time $t_0 + (i-1) \cdot 1/R$, where t_0 is the arrival time of the first arrived signal copy and R is the transmission symbol rate. Thus, the attacker can superimpose a copy into the transmitting signal at time t'_0, $t'_0 + 1/R$, $\cdots$, $t'_0 + (L-1) \cdot 1/R$ to emulate L paths, where t'_0 is the start time of the attacker's first transmission. Accordingly, the time delay for a signal copy is $\Delta_t = 1/R$. Figure 2 illustrates the attacker's signal manipulation and transmission process. For the i-th delayed signal copy $\mathbf{s}_i$, she multiplies it with a weight of w_i. Hence, the attacker's transmitting signal $\mathbf{x}_a$ can be represented as $\sum_{i=1}^{L} w_i \mathbf{s}_i$. These weights ensure that when the transmitting signal $\mathbf{x}_a$ propagates to the receiver through the real multipath environment, it can result in the attacker's desired channel impulse response observed at the receiver.

As a high-level overview for obtaining these weights, let $\mathbf{h}$ denote the channel impulse response between the attacker and the receiver. The signal $\mathbf{y}_a$ received from the attacker can be represented as $\mathbf{y}_a = \mathbf{h} * \mathbf{x}_a + \mathbf{n}$, where $\mathbf{x}_a$ and $\mathbf{n}$ are the transmitting signal and the channel noise, respectively. The receiver uses $\mathbf{y}_a$ to estimate the channel impulse response, and the estimation result is given by $(\mathbf{X}^H\mathbf{X})^{-1}\mathbf{X}^H\mathbf{y}_a$, where $\mathbf{X}$ is a Toeplitz matrix constructed from the training sequence. Let $\mathbf{h}_a$ denote the channel impulse response chosen by the attacker. The attacker aims to make this estimation result equal to $\mathbf{h}_a$, i.e., $(\mathbf{X}^H\mathbf{X})^{-1}\mathbf{X}^H\mathbf{y}_a = \mathbf{h}_a$. By substituting $\mathbf{y}_a = \mathbf{h} * \mathbf{x}_a + \mathbf{n}$ and $\mathbf{x}_a = \Sigma_{i=1}^{L} w_i \mathbf{s}_i$ into this equation, the attacker can solve the weights and we show the detailed calculation process in Section 4.2.

4.2 Obtaining the Weights

A technical challenge for the attacker is that she needs to obtain the weights used in the virtual multipath channel to make the receiver believe a particular channel impulse response. In the following, we show how the attacker can obtain such weights. When training sequence $[x_1, x_2, \cdots, x_M]$ first goes through the virtual channel with weights w_1, w_2, $\cdots$, w_L, the resulting transmitting signal $\mathbf{x}_a$ can be represented in the following matrix form.

$$\mathbf{x}_a = \begin{bmatrix} x_1 & 0 & \cdot & 0 \\ x_2 & x_1 & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot \\ x_L & \cdot & \cdot & x_1 \\ \cdot & \cdot & \cdot & \cdot \\ x_M & \cdot & \cdot & x_{M-L+1} \\ 0 & x_M & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & \cdot & x_M \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \cdot \\ \cdot \\ w_L \end{bmatrix} = \mathbf{X}\mathbf{w}.$$

The length of $\mathbf{x}_a$ is $L+M-1$, and we let $\mathbf{x}_a = [x_{a_1}, x_{a_2}, ..., x_{a_{L+M-1}}]$. The transmitting symbols x_a will go through the real multipath channel and the corresponding received symbols $\mathbf{y}_a$ is (we omit the noise term for the sake of simplicity)

$$\mathbf{y}_a = \mathbf{h} * \mathbf{x}_a = \mathbf{X}_a\mathbf{h} = \begin{bmatrix} x_{a_1} & 0 & \cdot & 0 \\ x_{a_2} & x_{a_1} & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot \\ x_{a_L} & \cdot & \cdot & x_{a_1} \\ \cdot & \cdot & \cdot & \cdot \\ x_{a_{M+L-1}} & \cdot & \cdot & x_{a_M} \\ 0 & x_{a_{M+L-1}} & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & \cdot & x_{a_{M+L-1}} \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ \cdot \\ \cdot \\ h_L \end{bmatrix}.$$

The length of $\mathbf{y}_a$ is $L + (L+M-1) - 1 = 2L + M - 2$. Assume that the receiver is not aware that the original training sequence has been manipulated by the attacker. He thinks that the length of the training sequence is M, the number of paths is L, and hence the number of corresponding received symbols should be $M+L-1$. The receiver then uses the first received $M+L-1$ symbols to calculate the channel impulse response. Let $\mathbf{y}'_a$ denote the vector formed by these symbols and we can represent $\mathbf{y}'_a$ as $\mathbf{y}'_\mathbf{a} = \mathbf{I}\mathbf{y}_a = \mathbf{I}(\mathbf{X_a}\mathbf{h})$, where $\mathbf{I}_{L+M-1}$ is an $(L+M-1)\times(2L+M-2)$ matrix whose diagonal elements are all 1's. The receiver estimates the channel impulse response based on the equation $\mathbf{y}'_a = \mathbf{X}\hat{\mathbf{h}}$. The attacker must make $\hat{\mathbf{h}} = \mathbf{h}_a$ hold. Thus, using matrix operations, we have

$$\begin{aligned} \mathbf{y}'_a &= \mathbf{X}\hat{\mathbf{h}} = \mathbf{X}\mathbf{h}_a = \mathbf{I}(\mathbf{X_a}\mathbf{h}) \\ &= \begin{bmatrix} h_1 & 0 & . & 0 & 0 & . & 0 \\ h_2 & h_1 & . & \cdot & 0 & . & 0 \\ . & . & . & . & . & . & . \\ . & . & . & h_1 & 0 & . & 0 \\ h_L & . & . & h_2 & h_1 & . & 0 \\ 0 & h_L & . & . & h_2 & . & 0 \\ . & . & . & . & . & . & . \\ 0 & 0 & . & h_L & h_{L-1} & . & h_1 \end{bmatrix} \begin{bmatrix} x_{a_1} \\ x_{a_2} \\ . \\ . \\ x_{a_M} \\ x_{a_{M+1}} \\ . \\ x_{a_{M+L-1}} \end{bmatrix} \\ &= \mathbf{H}\mathbf{x_a}, \end{aligned}$$

where $\mathbf{H}$ is a Toeplitz matrix of $\mathbf{h}$. We can then solve $\mathbf{x}_a$ from the above equation, and $\mathbf{x}_a = (\mathbf{H^H H})^{-1}\mathbf{H}^H\mathbf{y}'_a = (\mathbf{H^H H})^{-1}\mathbf{H}^H(\mathbf{X}\mathbf{h_a})$. Note that $\mathbf{x}_a = \mathbf{X}\mathbf{w}$. Thus, we can solve the weights $\mathbf{w}$ from the above equations, and obtain

$$\mathbf{w} = (\mathbf{X^H X})^{-1}\mathbf{X}^H[(\mathbf{H^H H})^{-1}\mathbf{H}^H(\mathbf{X}\mathbf{h_a})].$$

4.3 Initial Simulation

As an initial validation, we simulate the virtual multipath attack using the CRAWDAD data set [14], which contains over 9300 real channel impulse responses measured in an indoor environment with obstacles (e.g., offices and furniture) and scatters (e.g., windows and doors).

4.3.1 Simulation Process

We pick two nodes (i.e., nodes 31 and 40) from the data set as the attacker and the receiver, and obtain the channel impulse response $\mathbf{h}$ between them. We randomly choose another channel impulse response $\mathbf{h}_a$ (i.e., the one between nodes 34 and 40) from the data set, and the attacker aims to fool the receiver to get a channel estimation result of

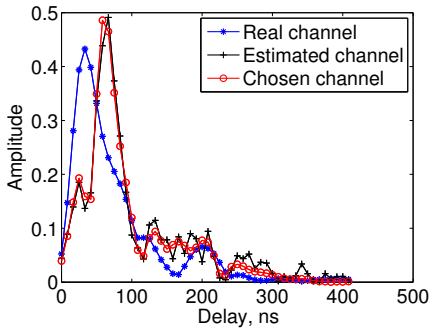


Fig. 3. The channel impulse responses.

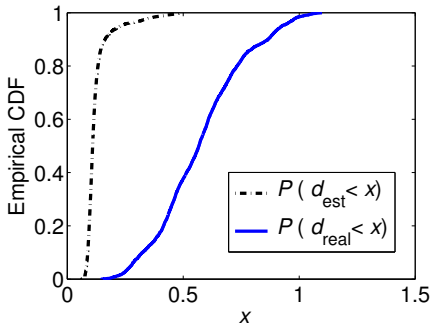


Fig. 4. The empirical cumulative distribution functions of d_{real} and d_{est} using the CRAWDAD data set.

$\mathbf{h}_a$ rather than $\mathbf{h}$. We generate a training sequence $\mathbf{x}$ of 64 bits using a pseudorandom number generator. The attacker computes the weights based on $\mathbf{h}$, $\mathbf{h}_a$, and $\mathbf{x}$, and then creates a virtual multipath channel by aggregating the weighted delayed copies of the training sequence $\mathbf{x}$ as shown in Figure 2. Thus, the corresponding received symbols $\mathbf{y}'_a$ can be computed via $\mathbf{y}'_a = \mathbf{I}(\mathbf{X_a h})) + \mathbf{n}$, where $\mathbf{n}$ is the gaussian noise and we set the signal-to-noise (SNR) 20dB in the simulation. Finally, the receiver estimates the corresponding channel impulse response from the virtual channel.

4.3.2 Simulation Result

Figure 3 plots the real channel impulse response $\mathbf{h}$ between the attacker and the receiver, the chosen channel impulse response $\mathbf{h}_a$ that the attacker wants to emulate, and the channel impulse response $\mathbf{h}_r$ estimated by the receiver. We can observe that $\mathbf{h}_a$ is very close to $\mathbf{h}_r$ under the virtual multipath attack.

The CRAWDAD data set stores five measurements of the channel impulse response for every pair of nodes. In the simulation, for the real channel impulse response $\mathbf{h}$, we randomly pick one as the comparison base. The Euclidean distance between the other four real channel impulse responses and $\mathbf{h}$ ranges between 0.0490 and 0.2297. The Euclidean distance between the estimated channel impulse response $\mathbf{h}_r$ and $\mathbf{h}$ is 0.5782, which is out of the above range. However, the Euclidean distance between $\mathbf{h}_r$ and $\mathbf{h}_a$ is 0.1054, which falls into the normal range of variation of the channel impulse responses. This means that once the attacker establishes a virtual multipath channel, the attacker can hide her real locations since $\mathbf{h}_r \neq \mathbf{h}$, or impersonate a node at a different location since $\mathbf{h}_r \approx \mathbf{h}_a$.

We repeated the simulation using all data in the CRAWDAD data set. Figure 4 plots the empirical empirical cumulative distribution functions (CDFs) of the Euclidean distance d_{real} between the the chosen channel and the real channel response, as well as that of the Euclidean distance d_{est} between the chosen one and the channel impulse response estimated under the attack. We can see that the probability that d_{est} is smaller than d_{real} is high. In particular, 95.13% of d_{est} is less than 0.2295, whereas only 1.59% of d_{real} is less than this value. Thus, if the receiver uses 0.2295 as the detection threshold to verify channel impulse responses, the receiver will get a mis-detection rate of 0.9513 and a false alarm rate of 0.9841 (i.e., 1 - 0.0159).

The simulation result demonstrates the theoretical feasibility of the virtual multipath attack. In Section 7, we reveal the practical impact of such attacks with real world experiments.

4.4 Discussion

4.4.1 Complexity at the Attacker

To launch virtual multipath attacks, the attacker requires to sum all delayed signal components with weights, as shown in Figure 2. This delay-and-sum process can be easily implemented using software (e.g. designing a delay-and-sum C++ module in GNU radio for USRP) or hardware (e.g. using flip-flop components to delay signals and using accumulators to sum all signal components in FPGA). Such an architecture does not significantly incur software or hardware complexity.

4.4.2 Message Demodulation at the Receiver

By adding delayed signals together, a virtual multipath attacker introduces inter-symbol interference to its transmission signals. We note that such signals are decodable at the receiver. It is common for a receiver to receive signals with inter-symbol interference due to the wireless multipath effect. A receiver normally uses channel estimation results to learn multipath channel conditions [8]. The estimated channel impulse response is then used in the demodulation process to compensate the multipath effect and convert the self-interference signal into a meaningful message.

As long as the attacker passes the training and the information payload through the same virtual channel as shown in Figure 2, the received signal at the receiver will go through the same combined channel effect of virtual and realistic channels. In this regard, although the receiver obtains the estimation of a fake channel impulse response, such an estimation result still represents the combined channel effect that the data goes through. Therefore, the receiver will successfully decode the original message using this estimation result. The only impact of virtual multipath attacks is that the receiver is fooled by fake channel impulse responses.

4.4.3 Impact of the Time Delay

Theoretically, the attacker can set an arbitrarily small delay (e.g, 1 nanosecond) to create a much richer virtual multipath effect at the receiver. However, modern channel estimation

algorithms estimate only resolvable paths whose inter-arrival durations are no less than one symbol duration, and it has been shown that using the estimation of resolvable paths is sufficient to compensate the channel effect for signal demodulation. Thus, at the receiver's point of view, the channel consists of multiple resolvable paths. This means that it is sufficient to set the delay in virtual channel generation to be one symbol duration (e.g., just generate resolvable paths) to fool the receiver's view on the channel. Even if the attacker reduces the delay to generate a more fine-grained virtual multipath channel, the receiver can still observe the resolvable paths and the corresponding channel impulse response. Thus, decreasing the delay can only add implementation complexity to the attacker, but will not cause more impact of the attack at the receiver. On the other hand, if attacker utilize a larger delay (e.g., larger than the symbol duration), the receiver may not observe enough multipath effect under the virtual multipath attacks and thus the attack impact is limited. Therefore, it is reasonable to set the delay to be one symbol duration to balance the attack effect and complexity.

4.4.4 Example Attack Scenarios

The example scenarios where virtual multipath attacks may exist include: (1) movement detection: an attacker may hide its movement by creating a static virtual channel impulse response at the receiver, e.g., a wireless sensor can be moved from the monitoring area but the movement is not detected; (2) detection of sybil attacks: an attacker may bypass the detection of sybil attack by pretending identities that are originated from different locations; (3) authentication: the attacker may impersonate another wireless transmitter. This attack scenario requires the attacker to know the channel impulse response between the target transmitter and the receiver, and thus imposes some limitations to the attacker. However, since the virtual multipath channel attacks can produce any channel estimation results at the receiver, such attacks are still a threat to existing channel fingerprinting based authentication schemes; (4) In addition to the attack scenarios, on the other hand, the attacks can be further utilized to enhance the wireless security. For example, the virtual channels can be used to provide a rich set of shared keys between two wireless devices, or enable anonymous communications by protecting location privacy of wireless users via virtual channel camouflage.

5 Defending against The Virtual Multipath Attack

Virtual multipath attackers are able to make the receiver believe any channel characteristic the attacker chooses. At the receiver, it seems that there is no way to tell whether the signal goes through real or virtual multipath scenario. Hence, existing location distinction methods built upon distinguishing locations from channel characteristics (e.g., [1]–[3], [6]) will be easily defeated by virtual multipath attacks.

The intuition behind our defense strategy is that *nobody can craft one key to open two different doors*. In other words, if a receiver cannot tell whether there is an attack or not, maybe a second receiver can. As a result, the proposed approach makes

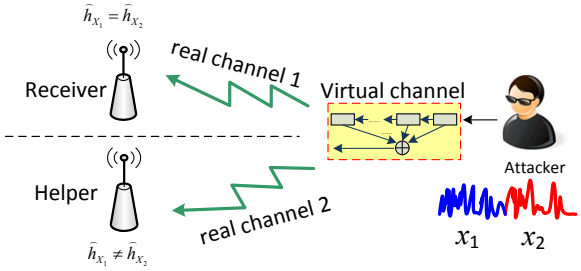


Fig. 5. Defense against virtual multipath attacks

use of an auxiliary receiver or antenna, which we refer to as a *helper*. The helper is placed more than half a wavelength away from the receiver to ensure a distinct channel characteristic. We let the receiver use two different training sequences $\mathbf{x}_1$ and $\mathbf{x}_2$ to estimate the channel impulse response alternatively. Without loss of generality, we assume that the receiver uses $\mathbf{x}_1$ to estimate the channel from the first transmission, and uses $\mathbf{x}_2$ to estimate the channel from the second transmission.

We discover that for both transmissions, at the receiver, the virtual channel created by a malicious transmitter (i.e., the attacker) can result in the same estimated channel impulse responses (equal to the one chosen by the attacker). However, at the helper, the virtual channel leads to different estimated channel impulse responses. We summarize the defense approach in Figure 5. The reason that the attacker cannot fool both the receiver and the helper is detailed next.

5.1 Defense Analysis

Let $\mathbf{h}$ denote the real channel impulse response between the attacker and the receiver. For the first transmission, the attacker must solve the weights, so that the equation $\mathbf{h} * \mathbf{x}_{a1} = \mathbf{h}_a * \mathbf{x}_1$ hold and the receiver will obtain $\mathbf{h}_a$ as the channel impulse response, where $\mathbf{x}_{a1}$ is the aggregated signal with weighted time-delayed copies of the training sequence $\mathbf{x}_1$. Let $\mathbf{h}_{\text{help}}$ denote the real channel impulse response between the attacker and the helper. The corresponding signal received by the helper can be represented as $\mathbf{h}_{\text{help}} * \mathbf{x}_{a1}$. Thus, the channel impulse response $\hat{\mathbf{h}}_{\text{help}_1}$ estimated by the helper can be solved from the equation that $\hat{\mathbf{h}}_{\text{help}_1} * \mathbf{x}_1 = \mathbf{h}_{\text{help}} * \mathbf{x}_{a1}$, and we have

$$\hat{\mathbf{h}}_{\text{help}_1} = (\mathbf{X_1}^H \mathbf{X_1})^{-1} \mathbf{X_1}^H (\mathbf{h}_{\text{help}} * \mathbf{x}_{a1}), \tag{4}$$

where $\mathbf{X}_1$ is a Toeplitz matrix of $\mathbf{x}_1$.

For the second transmission, both the receiver and the helper use the training sequence $\mathbf{x}_2$ to estimate the channel. Similarly, to fool the receiver, the attacker must generate another weights $\mathbf{w}_2$, so that the corresponding aggregated signal $\mathbf{x}_{a2}$ makes the equation $\mathbf{h} * \mathbf{x}_{a2} = \mathbf{h}_a * \mathbf{x}_2$ hold. The corresponding channel impulse response $\hat{\mathbf{h}}_{\text{help}_2}$ estimated by the helper is

$$\hat{\mathbf{h}}_{\text{help}_2} = (\mathbf{X_2}^H \mathbf{X_2})^{-1} \mathbf{X_2}^H (\mathbf{h}_{\text{help}} * \mathbf{x}_{a2}), \tag{5}$$

where $\mathbf{X}_2$ is a Toeplitz matrix of $\mathbf{x}_2$.

Note that for both transmissions, the channel impulse response estimated by the receiver are always the same, because the weights are "customized" so that the receiver will obtain the attacker's chosen channel impulse response after the

channel estimation. However, from Equations 4 and 5, we can see that the first estimated channel impulse response $\hat{\mathbf{h}}_{\text{help}_1}$ is not necessarily equal to the second estimated channel impulse response $\hat{\mathbf{h}}_{\text{help}_2}$, because $\mathbf{X}_1 \neq \mathbf{X}_2$. This means the attacker cannot fool the receiver and the helper at the same time.

Thus, if the successive estimated channel impulse responses show dramatic changes in a short time at the helper, the helper then triggers an alert at the receiver regarding the existence of potential virtual multipath attacks. In practice, the helper may use a threshold to enforce the detection. If $||\hat{\mathbf{h}}_{\text{help}_1} - \hat{\mathbf{h}}_{\text{help}_2}||$ is larger than the threshold, then the attack is assumed. The threshold can be selected based on the empirical studies to achieve an optimized detection accuracy. In Section 7.4, we show an example of the threshold selection. Note that in the defense system, the helper and the receiver can switch their roles, i.e., if the attacker attempts to fool the helper instead of the receiver, the receiver will estimate two different channel impulse responses and therefore detect such an attack.

5.1.1 Attackers with Helper

The attacker may also bring a second transmitter to confuse the receiver. Figure 6 shows such a scenario. We refer to the attacker's second transmitter as the *attacker's helper*. Let $\mathbf{h}_{11}$, $\mathbf{h}_{12}$, $\mathbf{h}_{21}$, $\mathbf{h}_{22}$ denote the channel impulse responses between the attacker and the receiver, the attacker and the receiver's helper, the attacker's helper and the receiver, and the attacker's helper and the receiver's helper, respectively. To successfully launch the virtual channel attacks without being detected, the attacker must generate the same channel impulse response at the receiver's helper for both transmissions. Let $\mathbf{h}_{\text{help}}$ denote such a channel impulse response. Further let $\mathbf{h}_a$ denote the one that the attacker expects to generate at the receiver for both transmissions. The attacker needs to make the following equation hold:

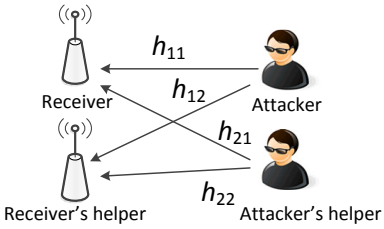


Fig. 6. The attacker also brings a second transmitter to confuse the receiver.

$$\begin{cases} \mathbf{h}_{11} * \mathbf{x}_{a1} + \mathbf{h}_{21} * \mathbf{x}_{h1} = \mathbf{h}_a * \mathbf{x}_1 \\ \mathbf{h}_{12} * \mathbf{x}_{a1} + \mathbf{h}_{22} * \mathbf{x}_{h1} = \mathbf{h}_{\text{help}} * \mathbf{x}_1 \\ \mathbf{h}_{11} * \mathbf{x}_{a2} + \mathbf{h}_{21} * \mathbf{x}_{h2} = \mathbf{h}_a * \mathbf{x}_2 \\ \mathbf{h}_{12} * \mathbf{x}_{a2} + \mathbf{h}_{22} * \mathbf{x}_{h2} = \mathbf{h}_{\text{help}} * \mathbf{x}_2 \end{cases}, \tag{6}$$

where $\mathbf{x}_{a1}$, $\mathbf{x}_{h1}$, $\mathbf{x}_{a2}$, and $\mathbf{x}_{h2}$ are the actual signals to be transmitted by the attacker and her helper for the first and second transmissions. To break the proposed defense, the attacker must solve them from Equation 6. This implies that $\mathbf{h}_{11}$, $\mathbf{h}_{12}$, $\mathbf{h}_{21}$, $\mathbf{h}_{22}$ should be all available to the attacker. Otherwise, the linear system lacks necessary coefficients to generate solutions. However, the acquisition of $\mathbf{h}_{12}$ and $\mathbf{h}_{22}$ will impose difficulty for the attacker, because the receiver's helper can be designed passive, i.e., it receives wireless signals but doesn't actively send out wireless signals to the channel. Due to the close proximity, the receiver can communicate with its helper through the cable connection or internal circuit. A passive helper of the receiver eliminates the chance for the attacker to extract the channel impulse responses based on heard wireless signals.

5.1.2 Extending to MIMO systems

In case of a very powerful attacker, who is able to set up a collaborator transmitter that is co-located with the receiver's helper (i.e., at the exact physical location of the receiver's helper), $\mathbf{h}_{12}$ and $\mathbf{h}_{22}$ may be obtained from the wireless signals sent by the collaborator transmitter. Nevertheless, the defense methods can be easily extended to deal with these attacks by increasing the number of helpers at the receiver.

To facilitate the reader's understanding, we consider a multiple-input and multiple-output (MIMO) scenario, where the receiver and the attacker have M and N antennas respectively. Assume the fake channel impulse responses that the attacker aims to generate at the receiver's antennas are $\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_M$, and the real channel impulse responses between each of the attacker's antenna and each of the receiver's antennas is denote as $\mathbf{h}_{ij}$, where $i = 1, 2, ..., N$ and $j = 1, 2, ..., M$. We assume $\mathbf{h}_{ij}$ are all available to the attacker due to the existence of the collaborator transmitters placed at the same locations as the receiver's antennas. Let $\mathbf{x}_{a1_i}$ and $\mathbf{x}_{a2_i} (i = 1, 2, \cdots, N)$ denote the signals to be transmitted by the attacker's i-th antenna for the first and second transmissions. Similar to the previous discussion, the attacker must solve them from $\sum_{i=1}^{N} \mathbf{h}_{ij} * \mathbf{x}_{a1_i} = \mathbf{h}_j * \mathbf{x}_1$ and $\sum_{i=1}^{N} \mathbf{h}_{ij} * \mathbf{x}_{a2_i} = \mathbf{h}_j * \mathbf{x}_2$ for $\forall j \in \{1, 2, ..., M\}$.

If $N \geq M$, the attacker can find a unique solution or infinite solutions of $\mathbf{x}_{a1_i}$ and $\mathbf{x}_{a2_i}$. However, if $N < M$, this linear system is overdetermined, which yields no feasible solution. This means that the attacker cannot find appropriate values of transmitted signals (or weights), so that the receiver will observe the same channel impulse responses at all antennas for two transmissions. Therefore, if the number of the receiver's helper nodes is greater than that of the attacker's helper nodes, the virtual multipath channel attacks can be detected.

5.1.3 Defense Discussion

The receiver can normally use one passive helper, i.e., a secret wireless tap, to detect the attacks. The exception happens when the attacker knows all channel information from her and her helpers to the receiver's passive helper (by placing a spy node co-located with or extremely close to the receiver's helper), which is in fact a very harsh requirement for the attacker.

We point out that under this circumstance it is still feasible to detect virtual multipath attacks as long as the receiver has more helpers than the attacker. A significant advantage of the receiver over the attacker is that the receiver just needs to find contradiction to detect the attack; while the attacker has to know all channel information for signal manipulation to make sure no contradiction is found. In particular, when the

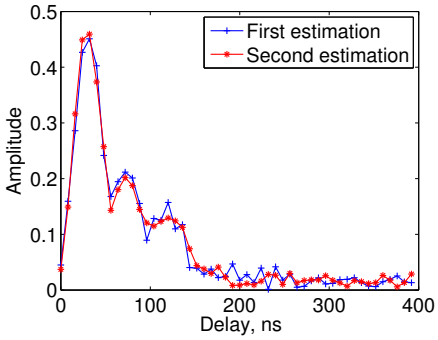


Fig. 7. Both estimates are consistent with each other.

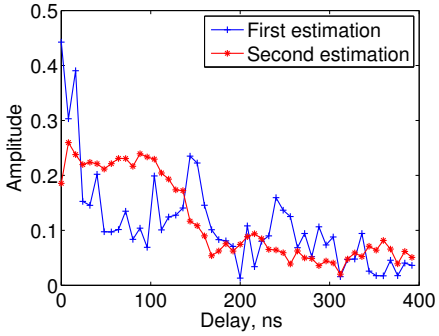


Fig. 8. Both estimates significantly differ from each other.

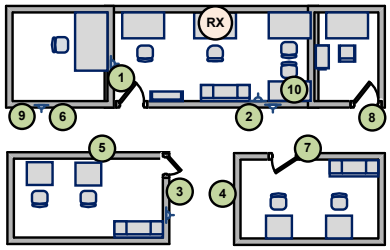


Fig. 9. Floorplan of the building where the experiment is conducted.

receiver adds one more passive helper, it actually reduces the attack situation to the normal case. In order to beat the defense, the attacker must meet all the following requirements at the same time to beat the receiver: (1) add one helper, (2) add one spy node at the exact location of the receiver's new helper to know the channel information, (3) synchronize herself and all her helpers to transmit the manipulated signal at the physical-layer symbol level. Hence, the attacker has much more costs to beat the receiver with more passive helpers.

5.2 A Case Study

We show an example of the defense approach using the real measured channel data from the CRAWDAD data set. We randomly pick three nodes from the data set, and they are used as the attacker (node 14), the receiver (node 3), and the helper (node 32), respectively. We also randomly pick one channel impulse response (between nodes 4 and 9) from the data set, and it is used as the fake channel impulse response that the attacker would like to fool the receiver. Let $\mathbf{h}$, $\mathbf{h}_{\text{help}}$, and $\mathbf{h}_a$ denote the channel impulse responses between the attacker and the receiver, the attacker and the helper, and the fake one chosen by the attacker.

We generate two 64-bit training sequences $\mathbf{x}_1$ and $\mathbf{x}_2$. For the first and the second transmissions, we compute the weight vectors $\mathbf{w}_1$ and $\mathbf{w}_2$, so that the corresponding virtual channels will result in estimated channel impulse responses that are equal to $\mathbf{h}_a$ at the receiver. As discussed earlier, these weight vectors should be computed based on $\mathbf{h}$, $\mathbf{h}_a$, $\mathbf{x}_1$, and $\mathbf{x}_2$.

Figure 7 shows the channel estimation outcomes at the receiver for the first and the second transmissions, respectively. We can see that both estimated channel impulse responses are consistent with each other. The Euclidean distance between them is 0.1127. We also calculate the channel estimation results at the helper. As shown in Figure 8, these channel estimates significantly differ from each other. The Euclidean distance between them is as high as 0.5701, which is out of the normal range of variation of the channel impulse responses. Thus, the virtual multipath attack is detected.

6 Virtual Multipath Attacks and Defenses in OFDM Systems

Orthogonal frequency-division multiplexing (OFDM) is a popular wireless communication scheme that encodes the digital signal using multiple sub-carrier frequencies. These sub-carriers are normally narrow-band (e.g., 802.11 a/g physical layer advocates an OFDM sub-carrier bandwidth less than 0.5 MHz). Thus, OFDM systems are robust against channel fading caused by the multipath effect. For an OFDM system, the channel estimation is done by estimating the channel impulse response of each sub-carrier. Due to the lack of the multipath fading, the channel estimation result of each sub-carrier is a complex number rather than a vector, and the final channel estimation output of an OFDM system is formed by these complex numbers. In this section, we explore virtual multipath attacks and corresponding defenses in OFDM systems.

6.1 Attacks against OFDM Systems

The virtual multipath attacks can be easily extended to OFDM systems, because the mapping from the time-domain to frequency-domain is linear. The delay-and-sum process can be replaced by a much simpler procedure, in which the attacker multiplies chosen weights to sub-carriers. Specifically, let $[h_1, h_2, ..., h_n]$ denote the actual channel characteristic between the attacker and the receiver, where h_i is the channel characteristic of the i-th sub-carrier and n is the number of sub-carriers. Further let $[x_1, x_2, ..., x_n]$ denote the training sequence encoded by the OFDM modulator, where x_i is the i-th element of the encoded training sequence. The symbol received at the i-th carrier can be represented by $y_i = h_i x_i$. To fool the receiver to obtain a fake channel estimation result of $[h_{a_1}, h_{a_2}, ..., h_{a_n}]$, the attacker needs to make the equation $h_i x_{a_i} = h_{a_i} x_i$ hold, where x_{a_i} is the symbol to be transmitted by the attacker at the i-th sub-carrier. Thus, $x_{a_i} = \frac{h_{a_i} x_i}{h_i}$, and the weights that the attacker needs to multiply to sub-carriers are $\frac{h_{a_1}}{h_1}, \frac{h_{a_2}}{h_2}, ..., \frac{h_{a_n}}{h_n}$.

6.2 Defenses in OFDM systems

Despite the ease for an attacker to extend virtual multipath attacks to OFDM systems, as described above, there are no straightforward ways to extend the previously discussed detection approach to these systems, because the channel estimation of an OFDM system is significantly different from that of a traditional communication system.

Let h_i^r and h_i^h denote the actual channel characteristic between the attacker and the receiver and between the attacker

and the helper, respectively. Let x_{i_1} and x_{i_2} denote the i-th element of the first and second training sequences. Let $x_{a_{i1}}$ and $x_{a_{i2}}$ denote the symbol to be transmitted by the attacker at the i-th sub-carrier in the first and second transmissions. Further let $h^r_{a_i}$ and $h^h_{a_i}$ denote the fake channel estimation results that the attacker would like to generate at the i-th sub-carrier of the receiver and the helper. The conditions for the attacker to launch the attack without being detected are summarized as

$$\begin{cases} h^r_i x_{a_{i1}} = h^r_{a_i} x_{i_1} \\ h^h_i x_{a_{i1}} = h^h_{a_i} x_{i_1} \\ h^r_i x_{a_{i2}} = h^r_{a_i} x_{i_2} \\ h^h_i x_{a_{i2}} = h^h_{a_i} x_{i_2} \end{cases}.$$

We can see that there exists a solution for $h^h_{a_i}$ which is

$$h^h_{a_i} = h^r_{a_i} \cdot \frac{h^h_i}{h^r_i}. \tag{7}$$

Thus, when the attacker causes the receiver to observe the same channel estimation results for the first and second transmissions, the two channel estimation results at the helper side are also the same. Therefore, the virtual multipath attack in OFDM systems cannot be detected by the previously proposed regular defense, which just observes the difference of two channel estimates at the helper side for two transmissions with different training sequences.

However, we identify alternative ways to close the loophole of the regular defense and defend against virtual multipath attacks in OFDM systems. We first categorize two typical objectives of attackers to confuse the location distinction:

1. *Motion camouflage*: The attacker is moving but she aims to deceive the receiver about the moving activities. Towards this end, the attacker makes the receiver believe that she is stationary by causing the estimated channel at the receiver to appear unchanged.
2. *Immobility camouflage*: When the attacker is stationary, she wants to make the receiver believe that she moves to a new location by changing the estimated channel at the receiver. The typical example targeting this objective is the Sybil attack, in which the attacker pretends to change her location and therefore identity while she indeed just changes the channel between herself and the receiver, as the receiver will observe differing channels between transmitters in different locations.

In practice, the two objectives may happen alternatively. For attacks against OFDM systems, we propose a corresponding defense strategy for each attack goal.

6.2.1 Motion camouflage

To detect motion camouflage, we propose to utilize a passive helper at the receiver side and observe the difference of the two channel estimates at this helper side.

To illustrate the defense against motion camouflage, we use Figure 10 as an example, where the attacker is previously at location 1 and then moves to location 2, and she wants to make the receiver believe that she is stationary. Let $h^r_{L_i}$

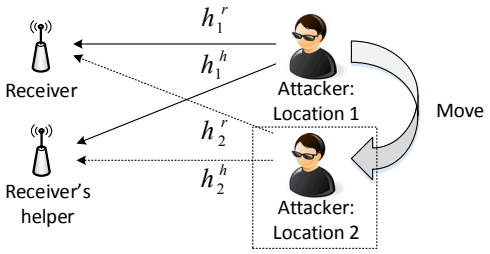


Fig. 10. Motion camouflage.

and $h^h_{L_i}$ denote the actual channel characteristic between the attacker and the receiver and that between the attacker and the helper when the attacker is at location i, respectively. Let h^r_a denote the fake channel estimation results that the attacker would like to generate at the receiver. Based on Equation 7, we can obtain the estimated channel $h^h_{a_1}$ at the helper when the attacker is at location 1 as $h^h_{a_1} = h^r_a \cdot \frac{h^h_{L_1}}{h^r_{L_1}}$. Similarly, when the attacker moves from location 1 to location 2, we can obtain the estimated channel $h^h_{a_2}$ at the helper under the attack as $h^h_{a_2} = h^r_a \cdot \frac{h^h_{L_2}}{h^r_{L_2}}$. Note that though the attacker is actually at the new location (i.e., location 2), the channel estimation result that the attacker would like to generate at the receiver is still h^r_a so that the receiver believes that the attacker is stationary (i.e., remaining at location 1).

In the normal case when no virtual multipath attack occurs, when the estimated channel at the receiver is unchanged (i.e., the receiver is actually stationary), the estimated channel at the receiver's helper should maintain the same for both channel estimations, i.e., $h^h_{a_1} = h^h_{a_2}$ should hold. However, from their calculation formulas above, we can see that $h^h_{a_1} = h^h_{a_2}$ does not necessarily hold since the actual channels $h^h_{L_1}$ and $h^h_{L_2}$ are unknown to the attacker when the receiver's helper is passive, except when the following equation holds

$$\frac{h^r_{L_1}}{h^h_{L_1}} = \frac{h^r_{L_2}}{h^h_{L_2}}. \tag{8}$$

However, Equation 8 rarely holds in practice as the real channel impulse response is uncontrollable and unpredictable. A real world experiment is presented to demonstrate this in Section 6.2.2.

Therefore, when the receiver realizes that its two successive estimated channels are the same, it should discern one of two possible reasons: either the attacker's location is not changed, or the attacker wants to achieve motion camouflage. Meanwhile, if a difference between the two channel estimates can be observed at the receiver's helper side, the virtual multipath attack aiming to achieve motion camoufage is discovered.

6.2.2 Immobility camouflage

For this case, since the attacker changes the channel estimation result generated at the receiver, based on Equation 7, the estimated channel at the receiver's helper changes correspondingly. Thus, merely observing the difference of two channel estimates at the receiver's helper side is not feasible to distinguish immobility camouflage. Instead, we still propose to use a passive helper at the receiver side but observe the

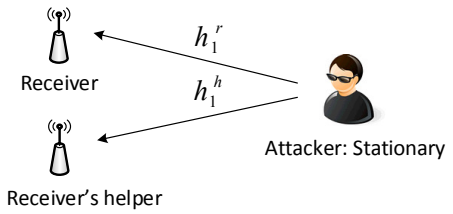


Fig. 11. Immobility camouflage.

ratio between the estimated channels at the receiver and at the receiver's helper to identify immobility camouflage.

Similar to the discussion of motion camouflage, we illustrate the defense against immobility camouflage using Figure 11, where the attacker is stationary while she aims to make the receiver believe that she moves. Suppose that the attacker is stationary at location 1. Let h_a^r and h_a^h denote the estimated channel at the receiver and that at the receiver's helper respectively when the attacker launches the attack. Then, the attacker manipulates the transmitted symbol x_{a1} so that the following equation holds

$$\begin{cases} x_{a_1} h_{L_1}^r = x h_a^r \\ x_{a_1} h_{L_1}^h = x h_a^h \end{cases}. \tag{9}$$

where x is the training symbol for channel estimation. Based on Equation 9, we have $h_a^r/h_a^h = h_{L_1}^r/h_{L_1}^h$. Similarly, if the attacker actually moves to a new location (e.g., location 2), the estimated channel estimates h_a^r and h_a^h should satisfy the equation $h_a^r/h_a^h = h_{L_2}^r/h_{L_2}^h$. Thus, to make the receiver believe that she is at location 2 while she is actually at location 1, the attacker needs to make Equation 8 hold. Otherwise, the receiver can utilize the ratio of the estimated channel at the receiver to that at the receiver's helper to detect the immobility camouflage, as the ratio remains the same when the attacker is stationary (i.e., normal case) and changes when the attacker moves (i.e., immobility camouflage case). We now explore how this ratio differs in the normal case and an attack case through a real world experiment.

We collect channel data at the receiver and its helper, and then calculate the ratios $h_{a_i}^r/h_{a_i}^h$ of the estimated channel at the receiver to that at the corresponding receiver's helper. In the normal case, we put the attacker at two different locations (e.g., location 1 and 2) without launching attacks. Therefore, the estimated channel should be the real channel, i.e., $h_{a_1}^r = h_{L_1}^r$, and $h_{a_2}^r = h_{L_2}^r$. If an attack based on immobility camouflage occurs, the attacker aims to make the receiver believe that she is at two different locations. We introduce a new metric, called ratio proximity and denoted with η, to demonstrate how close the two ratios are. In order to make η range between 0 and 1, we divide the minimum valued ratio by the maximum valued ratio. Mathematically, we have

$$\eta = \frac{min(h_{a_1}^r/h_{a_1}^h, h_{a_2}^r/h_{a_2}^h)}{max(h_{a_1}^r/h_{a_1}^h, h_{a_2}^r/h_{a_2}^h)}. \tag{10}$$

Thus, when η is close to 1, it indicates that the two ratios are close.

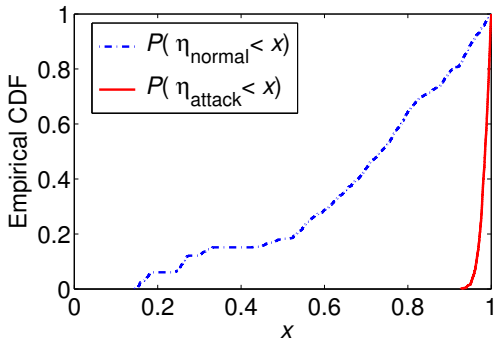


Fig. 12. The empirical CDFs of η_{normal} and η_{attack}.

We obtain two ratio proximity values η_{normal} and η_{attack} in the normal and attack cases, respectively. Figure 12 shows the empirical CDFs $P(\eta_{\text{normal}} < x)$ and $P(\eta_{\text{attack}} < x)$. We can see that η_{normal} varies from 0.14 to 1, and is less than 0.95 with the probability of 0.90, while η_{attack} is greater than 0.95 with the probability of 0.98. This means, in the normal case, the ratio proximity most likely deviates from 1, and consequently, Equation 8 rarely holds in practice. On the other hand, under immobility camouflage, ratio proximity is always near 1, and thus we can use this metric to successfully distinguish immobility camouflage.

Therefore, when the receiver finds that its two successive estimated channels are not the same, it should be aware that either it suffers from immobility camouflage or the attacker indeed changes her location. Furthermore, if the receiver realizes that the two corresponding values of the ratio h_a^r/h_a^h of the estimated channel at the receiver to that at the receiver's helper are the same, the attack is detected and the possibility that the attacker changes her location is excluded.

7 Experimental Evaluation

We build a prototype channel measurement system to demonstrate the impact of the identified attack and the effectiveness of the proposed defense. Our prototype is implemented on top of USRPs [15]. The software toolkit is GNURadio [16].

7.1 Evaluation Setup

We perform the experiment in a campus building with small offices, wooden doors, windows, metal and wooden furniture, and computers. Our prototype system consists of a malicious transmitter and a receiver. Each node is a USRP connected to a commodity PC, and each USRP uses a XCVR2400 daughter boards operating in the 2.4 GHz range as transceivers. The receiver estimates the channel impulse responses from received signals, and verifies whether or not there is a location change by comparing a newly estimated channel impulse response with an old one. The transmitter runs the attacker program, which computes the weight vector to form the virtual channel, passes the original signal through the virtual channel, and then feeds the virtual channel output to the real wireless channel. Note that the maximum number of resolvable multipaths L is usually configured to an empirical constant value depending

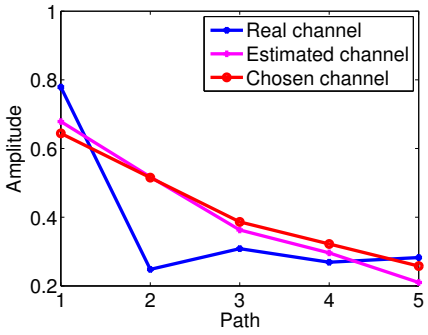


Fig. 13. The Euclidean distance of the real and estimated channels.

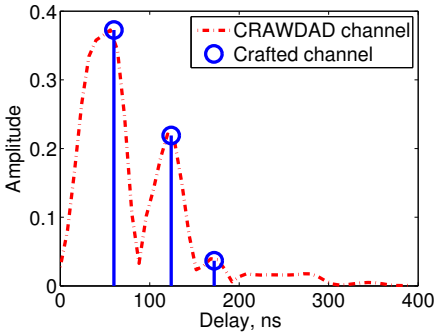


Fig. 14. A replica of the CRAWDAD channel impulse response.

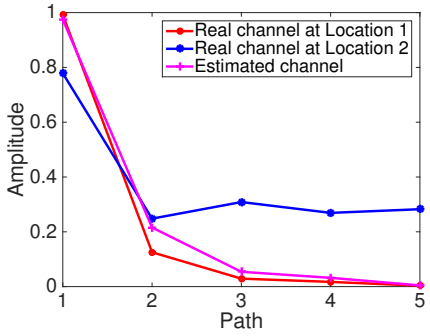


Fig. 15. Real location mimicking.

on wireless system setups [8]. In this experiment, we set $L = 5$ for our proof-of-concept implementation.

Figure 9 shows the positions of the receiver and the transmitter. We place the transmitter at 10 different locations to launch the attack, and the receiver periodically estimates the channel impulse responses.

7.2 Example Attacks

We examine three example attacks: (1) injecting a randomly chosen channel impulse response into the receiver, (2) reproducing a same channel impulse response in the CRAWDAD data set; and (3) mimicking another location while hiding the true location. For all three attacks, we place the transmitter at location 2 shown in Figure 9.

7.2.1 Generating a Random Channel Response

First we show an attack with intent to generate a random channel impulse response. Figure 13 plots the real channel impulse response between the transmitter and the receiver, the channel impulse response chosen by the attacker, and the estimated channel impulse response at the receiver. The y-axis and the x-axis indicate the power gain and the relevant path respectively. We can see that the chosen channel impulse response and the estimated one are very similar to each other, but both of them significantly deviate from the real channel. The Euclidean distance between the chosen channel and the real channel is 0.3025, whereas that between the chosen channel and the estimated channel is as small as 0.0686.

7.2.2 Replicating a Same Channel Response in a Different Building

In the second example, an attacker aims to generate a channel impulse response in our office building such that the generated channel impulse response is exactly the same as one in the CRAWDAD data set, which was collected in an office building in the University of Utah. We note our USRP system is different from the CRAWDAD measurement system, Sigtek model ST-515, which has a much higher bandwidth (40 MHz) than the USRP (10 MHz). Therefore, the CRAWDAD measurement system can observe richer multipaths. Nevertheless, even with a relatively low-end USRP, we can still duplicate the resolvable paths in a channel impulse response measured in the CRAWDAD data set.

Specifically, we select one channel impulse response (between nodes 14 and 43) from the CRAWDAD data set and we plot it as "CRAWDAD channel" in Figure 14. We can see that this channel impulse response carries three peaks and thus exhibits three resolvable multipaths. We launch the virtual multipath attack to make a replica of the same three resolvable multipaths observed at the receiver in our experiment, which is shown as "Crafted channel" in Figure 14. The attack's crafted channel impulse response of the resolvable multipaths closely matches the CRAWDAD channel response and their Euclidean distance is as small as 0.0036.

7.2.3 Actual Location Mimicking

In the third example, the attacker performs actual location mimicking, mimicking location 1 from location 2 shown in Figure 9. The attacker first records the real channel impulse response between herself and the receiver when she is at location 1, and then mimics this obtained channel impulse response when it moves to location 2. Figure 15 plots the real channel impulse responses between the transmitter and the receiver when the transmitter is at location 1 and 2 respectively, as well as the estimated channel impulse response at the receiver when the attacker performs the attack.

We can see that in normal situation, the real channels between the attack and the receiver when the attacker is at location 1 and location 2 are quite different, and the Euclidean distance between them is 0.5290. However, when the attacker launches the virtual multipath attack at location 2, the estimated channel at the receiver is quite close to the real channel between the attacker and the receiver when the attacker is at location 1, and the Euclidean distance between the two channels turns to as small as 0.0964. Therefore, the attacker is able to effectively make the receiver believe that she is at location 1 while she is actually at location 2.

7.3 Overall Attack Impact

To examine the overall attack impact, we perform the following experiment. For each location in Figure 9, we estimate the channel impulse responses during a short time window (around 10 – 30 seconds). For each estimates, we perform 100 trials, and in each trial we randomly generate a length-5 vector whose elements range between 0 and 1. This vector is used

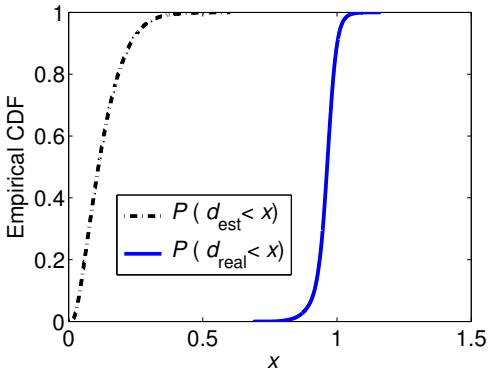


Fig. 16. The empirical CDFs of d_{real} and d_{est}.

as the attacker's chosen channel impulse response. We then launch the virtual multipath attack and record the Euclidean distance d_{real} between the chosen channel impulse response and the pervious channel impulse response estimated in the absence of the attacks (i.e., the real channel response), and also record the Euclidean distance d_{est} between the chosen one and the channel impulse response estimated under the attacks. We repeat the same experiment for the other 9 locations.

Ideally, a successful attacker should have a large value of d_{real} (indicating that the attacker's chosen channel significantly differs from the real channel) and a small value of d_{est} (indicating that the attacker's chosen channel is close to the receiver's estimated channel).

Denoted by $P(d_{\text{real}} < x)$ and $P(d_{\text{est}} < x)$ the empirical CDFs of d_{real} and d_{est}, respectively. Figure 16 shows $P(d_{\text{real}} < x)$ and $P(d_{\text{est}} < x)$ for $0 \leq x \leq 1.5$. We can see that d_{est} is less than 0.25 with probability 95.0%, d_{real} is larger than 0.9 with probability 95.0%. This means that d_{real} is much larger than d_{est} with high probability, therefore the attacker can drag the estimated value of channel impulse response far away from its true value, and make it very close to her specified one.

Existing schemes in general compare the difference between the receiver's current estimated channel and previous reference channel with a threshold to check a location change [1], [2]. Since our attacker can inject any random channel impulse response into the receiver with a very high accuracy, the performance of existing location distinction schemes can be significantly degraded by the virtual multipath attack. For example, given a threshold set less than 0.5 for location change detection in our system, when the attack is launched, the receiver will think that the transmitter moves because all the differences between the estimated channel in the presence of the attack and the reference channel (attack-free channel) exceed the threshold of 0.5. However, the estimated channel and the real channel are actually measured at the same location, and thus the location distinction false alarm rate is raised to 100% under the virtual multipath attack.

Similarly, the virtual multipath attack can also easily defeat any method verifying that nodes are from different locations based on examining the difference of their channel impulse responses (e.g., [3], [6]).

7.4 Evaluation of the Defense Method

We first show the practical feasibility of our defense method, then evaluate the performance.

7.4.1 Feasibility Evaluation

The defense approach functions based on a critical observation that the attacker cannot fool both the receiver and the helper at the same time. Thus, in our feasibility evaluation, we would like to examine how the channel estimation results of the receiver and the helper differ from each other, so that such an inconsistency can reveal the existence of the virtual multipath attack. Towards this goal, we perform the following experiment.

We place the attacker and the helper at each pair of the 10 locations, and we have $10 \times 9 = 90$ pairs of locations in total. Throughout the experiment, the receiver maintains its original position as marked in Figure 9. The attacker launches the virtual multipath attack, and both the receiver and the helper continuously do the channel estimation. Two 16-bit training sequences $\mathbf{x}_1$ (0xacdd) and $\mathbf{x}_2$ (0xa4e2) are alternatively used for estimating the channel impulse responses.

The helper and the receiver estimate the channel impulse responses from two successive transmissions, then calculate the Euclidean distance between both estimates. Let d_{helper} and d_{rec} denote the distances computed by the helper and the receiver, respectively. As analyzed in Section 5.1, d_{helper} should be much larger than d_{rec}.

Figures 17 and 18 show the channel impulse responses estimated using $\mathbf{x}_1$ and $\mathbf{x}_2$ at the receiver and the helper, when the attacker and the helper are placed at locations 2 and 8, respectively. We can see that the virtual multipath attack leads to a much larger distance at the helper than the receiver, i.e., $d_{\text{helper}} \gg d_{\text{rec}}$. Specifically, $d_{\text{rec}} = 0.0093$ and $d_{\text{helper}} = 0.1199$.

7.4.2 Performance Evaluation

As mentioned earlier, the helper may use a threshold to enforce the detection. If d_{helper} is larger than the threshold, then the attack is assumed. In general, detection and false positive rate are two performance metrics associated with a detection method. The detection rate is the probability that d_{helper} is larger than the threshold when there is indeed an attack. The false positive rate is the probability that d_{helper} is larger than the threshold when there is no attack.

In this experiment, we evaluate the performance of the proposed defense approach in terms of detection and false positive rates. We have 90 pairs of locations to place the attacker and the helper. From each pair of the locations, we can obtain the corresponding distances d_{helper} and d_{rec}.

We show the empirical CDFs $P(d_{\text{helper}} < x)$ and $P(d_{\text{rec}} < x)$ in Figure 19. We can see that in all experiments, d_{rec} is always less than 0.0151 (i.e., $P(d_{\text{rec}} < 0.0151)$=1), whereas d_{helper} is always greater than 0.0156. This means that if the helper uses 0.0151 as the detection threshold, the defense system can achieve a detection rate of 1 as well as a false positive rate of 0. In general, any threshold ranging between 0.0151 and 0.0156 can lead to the detection of all attacks, and meanwhile maintain the usability of the receiver.

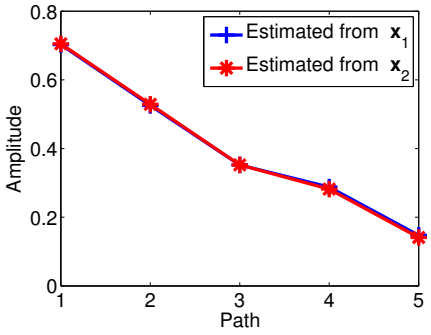


Fig. 17. The channel impulse responses estimated at the receiver.

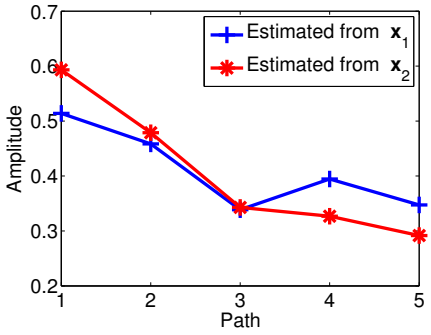


Fig. 18. The channel impulse responses estimated at the helper.

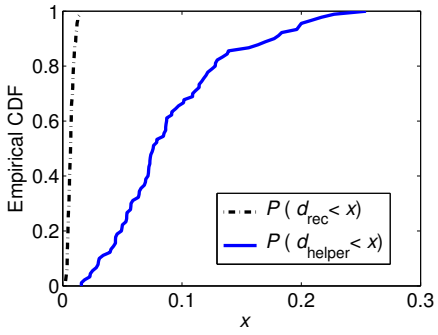


Fig. 19. The empirical CDFs of d_{helper} and d_{rec}.

The helper may select imperfect thresholds that do not fall in this range. However, it is still possible to achieve a high detection accuracy. For example, if the threshold is set to 0.02, the detection rate is as high as 91.2%, the false positive rate is still 0, which are obtained from Figure 19.

8 Related Work

Existing location distinction approaches have been focused on exploiting the spatial uncorrelation property of wireless channels (e.g., [1]–[3], [5], [6]). These approaches demonstrated their success in various wireless scenarios, especially for the high-frequency systems (e.g., WiFi networks) that feature a very short electromagnetic wavelength. However, two recent studies identified a vulnerability of these approaches [7], [17], and discovered that the wireless spatial uncorrelation property may be violated in a poor multipath environment (e.g., strong line-of-sight path). The work in [4], [18] made a further attempt to attack location distinction systems using channel impulse responses. The authors found that a third-party attacker may impersonate Alice to Bob by mimicking the channel impulse response of the wireless link between them, and the authors named such attacks as *mimicry attacks*. Although both mimicry attacks and the virtual multipath attacks are against the security measures based on the wireless channel characteristics, they differ from each other in the following aspects:

First, a pre-condition to launch mimicry attacks is the knowledge of the real channel impulse response between Alice and Bob (thus they assume the existence of a spy node). However, a virtual multipath attacker can still launch attacks without this knowledge. Moreover, if the attacker knows the real channel impulse response, she can make the receiver believe a specific channel impulse response. Therefore, virtual multipath attacks have a broader attack impact and less pre-requisites. In addition, we extend the virtual multipath attacks and the defense to MIMO and OFDM systems. It should be possible to extend mimicry attacks to these systems as well, because the attacker can directly manipulate the training signals for OFDM and MIMO systems with the knowledge of all channel information. However, mimicry attacks require to place a spy node close to the receiver. Thus, it becomes much more difficult to launch mimicry attacks when the receiver is equipped with a MIMO system, because the attacker has to place one spy node for each antenna to know the channel.

Second, both attacks differ in technical design methodology. The essential way of mimicry attacks is to manipulate the training signal such that the receiver believes an impersonated channel impulse response. Such a manipulation at the training signal level fools the receiver to accept an incorrect channel estimate, but the data payload after the training signal still goes through the real channel. As a result, the receiver will use an incorrect channel estimate to compensate the real channel effect, leading to incorrect packet decoding. In contrast, the virtual multipath attack uses a delay-and-sum process (with chosen weights) to create a virtual channel and pass all the data (e.g., training sequence and data payload) to be transmitted through this virtual channel. The receiver then not only gets a faked channel impulse response, but also uses it to successfully decode the entire data payload. Hence, the design methodology of virtual channel attacks ensures more stealthiness and consistency to fool the receiver.

Third, the proposed defense against the virtual multipath attack does not require any shared key between the transmitter and the receiver, whereas the defense proposed in [4] requires that the communicators to share a key. Such a requirement indicates that a key distribution and management system should be deployed prior to the enforcement of the defense, reducing the scalability and feasibility of the relevant approach.

Finally, because of the simplicity of the delay-and-sum process, as discussed earlier, the virtual multipath attacks can be interestingly extended to enhance the wireless security. For example, researchers have proposed to establish a key between two wireless devices using the channel impulse responses between them. Such a key is totally determined by the wireless physical layer feature and cannot be easily manipulated by the users. The idea of virtual channel attacks can be utilized here to enable the transmitter to control and update the shared key periodically and provide a rich set of shared keys among wireless users. Such attacks can also enable anonymous communications by protecting location privacy of wireless users via virtual channel camouflage.

Another recent work that is closely relevant with the proposed defense approach is SecureArray [19]. This work utilizes the physical angle-of-arrive (AOA) of a multi-antenna access point to enforce user authentication. Our proposed

defense technique uses channel impulse responses observed by multiple antennas to protect location distinction systems. However, our defense targets attacks against location distinction systems built on the spatial uncorrelation property of wireless channels, whereas SecureArray is designed to combat spoofing attacks that attempt to impersonate legitimate WiFi clients. Both approaches apply to different application domains.

We point out that the virtual multipath attack discovered in this paper doesn't target traditional localization systems using AOA, TOA, RSS, etc. Thus, complementary analysis and measures are necessary to protect these systems. Besides, in our future work, we will consider extending existing location distinction algorithms so that they can be adaptive to a more dynamic environment.

Acknowledgements

This work is supported by the National Science Foundation under grants 1527144 and 1553304, and the Army Research Office under grant W911NF-14-1-0324.

9 Conclusion

We identified a new attack against existing location distinction approaches built on the spatial uncorrelation property of wireless channels. By launching such attacks, the attacker can create virtual multipath channels to deteriorate the location distinction capability of a target receiver. To defend against this attack, we proposed a detection technique that utilizes a helper receiver to identify the existence of virtual channels. We also explored virtual multipath attacks and corresponding defenses in OFDM systems. We performed real-world evaluation on the USRP platform running GNURadio. The experimental results demonstrated both the feasibility of the virtual multipath attack and the effectiveness of the defense approach.

References

[1] N. Patwari and S. K. Kasera, "Robust location distinction using temporal link signatures," in *Proc. of ACM MobiCom '07*, September 2007, pp. 111–122.

[2] J. Zhang, M. H. Firooz, N. Patwari, and S. K. Kasera, "Advancing wireless link signatures for location distinction," in *Proc. of ACM MobiCom '08*, September 2008, pp. 26–37.

[3] L. Xiao, L. J. Greenstein, N. B. Mandayam, and W. Trappe, "Channel-based detection of sybil attacks in wireless networks," *IEEE Trans. Information Forensics and Security*, vol. 4, no. 3, pp. 492 – 503, 2009.

[4] Y. Liu and P. Ning, "Enhanced wireless channel authentication using time-synched link signature," in *Proc. of IEEE INFOCOM '12*, March 2012.

[5] Z. Li, W. Xu, R. Miller, and W. Trappe, "Securing wireless systems via lower layer enforcements," in *Proc. of ACM WiSe '06*, September 2006, pp. 33–42.

[6] Y. Liu, P. Ning, and H. Dai, "Authenticating primary users' signals in cognitive radio networks via integrated cryptographic and wireless link signatures," in *Proc. of IEEE S&P '10*, May 2010, pp. 286–301.

[7] X. He, H. Dai, W. Shen, and P. Ning, "Is link signature dependable for wireless security?" in *Proc. of IEEE INFOCOM '13*, April 2013.

[8] A. Goldsmith, *Wireless Communications*. Cambridge University Press, 2005.

[9] A. F. Molisch, *Wireless Communications, 2nd Edition*. Wiley India Pvt. Limited, 2007.

[10] R. Safaya, "A multipath channel estimation algorithm using a kalman filter," Thesis, University of Kansas, 2000.

[11] M. Biguesh and A. B. Gershman, "Training-based mimo channel estimation: A study of estimator tradeoffs and optimal training signals," *IEEE Trans. Signal Processing*, vol. 54, no. 3, pp. 884–893, March 2006.

[12] K. S. Shanmugan and A. M. Breipohl, *Random signals: detection, estimation, and data analysis*. Wiley, May 1988.

[13] O. Edfors, M. Sandell, J. J. V. de Beek, S. K. Wilson, and P. O. Borjesson, "OFDM channel estimation by singular value decomposition," *IEEE Trans. Communications*, vol. 46, no. 7, pp. 931 – 939, 1998.

[14] SPAN, "Measured channel impulse response data set," http://span.ece.utah.edu/pmwiki/pmwiki.php?n=Main.MeasuredCIRDataSet.

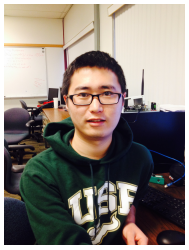
[15] ETTUS, "USRP-Universal Software Radio Peripheral," http://www.ettus.com.

[16] "GNU Radio Software," http://gnuradio.org/trac.

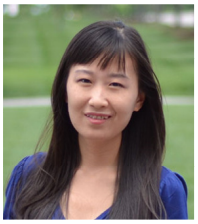
[17] M. Edman, A. Kiayias, and B. Yener, "On passive inference attacks against physical-layer key extraction," in *Proceedings of the Fourth European Workshop on System Security*, 2011.

[18] Y. Liu and P. Ning, "Poster: Mimicry attacks against wireless link signature," in *Proc. of ACM CCS'11*, 2011.

[19] J. Xiong and K. Jamieson, "Securearray: Improving wifi security with fine-grained physical-layer information," in *Proc. of ACM MobiCom '13*, 2013, pp. 441–452.



Song Fang is currently a PhD candidate in Computer Science, Univ. of South Florida, Tampa, FL. His research interests are in the area of network security and system security. He received the B.S. degree from South China Univ. of Technology, Guangzhou, China, and the M.S. degree from Beijing Univ. of Posts and Telecommunications, Beijing, China.



Yao Liu received the Ph.D. degree in computer science from North Carolina State Univ. in 2012. She is now an assistant professor at the Dept. of Computer Science and Engineering, Univ. of South Florida, Tampa, FL. Dr. Liu's research is related to computer and network security, with an emphasis on designing and implementing defense approaches that protect emerging wireless technologies. She was the recipient of Best Paper Award for the 7th IEEE International Conference on Mobile Ad-hoc and Sensor Systems.

Wenbo Shen received his Ph.D. degree in computer science from North Carolina State University, Raleigh, in 2015, the B.S. degree from Harbin Institute of Technology, Harbin, China, in 2010. His research area is the wireless network and system security, mainly focusing on leveraging physical layer signal properties to preserve the wireless security.

Haojin Zhu received the Ph.D. degree in electrical and computer engineering from the University of Waterloo, Canada, in 2009. He is now an associate professor with the Dept. of Computer Science and Engineering, Shanghai Jiao Tong Univ., China. His current research interests include wireless network security and distributed system security. He is a corecipient of best paper awards of IEEE ICC 2007 - Computer and Communications Security Symposium and Chinacom 2008- Wireless Communication Symposium.

Tao Wang received the BS degree in electrical engineering from Jilin University, China. Currently, he is a Ph.D. candidate in the Department of Computer Science and Engineering, University of South Florida, Tampa, FL. His research interests include wireless network, mobile security, and cyber-physical system security.