A Session Hijacking Attack Against a Device-Assisted Physical-Layer Key Agreement

Qiao Hu, Bianxia Du, Konstantinos Markantonakis, and Gerhard P. Hancke, *Senior Member, IEEE*

***Abstract*—Physical-layer key agreement is used to generate a shared key between devices on demand. Such schemes utilize the characteristics of the wireless channel to generate the shared key from the device-to-device channel. As all characteristics are time-dependent and location-dependent, it is hard for eavesdroppers to get the key. However, most research works in this area use passive attack models whereas active attacks that aim at manipulating the channel and key are also possible. Physical-layer key agreement with User Introduced Randomness (PHY-UIR) is a solution similar to the Diffie–Hellman protocol against such a kind of active attack. The users (devices) introduce their own randomness to help to prevent active attacks. In this paper, we analyze the possibility of launching a session hijacking attack on PHY-UIR to allow an attacker to control the shared key established. The session hijacking attack manipulates the key agreement through a man-in-the-middle interaction and forces legitimate devices to run the PHY-UIR protocol with the attacker. Our simulation and experiment results validate our attack and show the high performance of our attack on manipulating the generated key. We also propose PHY-UIR$^+$ where devices simultaneously exchange information about the established shared keys, which allows them to detect whether they have agreed to different keys with a third party.**

***Index Terms*—Key management, physical-layer security.**

I. Introduction

THE Internet of Things (IoT) is rapidly growing and will continue to do so with the help of other emerging technologies, such as fifth-generation wireless systems. For the IoT to really reach its envisaged goal of interconnecting all "things," we must be able to ensure secure device-to-device communication. To provide basic security services between devices, these devices must share cryptographic keys. Without effective key management within a system, other security services related to confidentiality, integrity, and authentication cannot function correctly as these all rely on devices holding suitable cryptographic keys. Key management is not easy, and becomes increasingly difficult as systems scale, for example, each device needs to share a unique secret symmetric key with every other device it might wish to communicate with. This leads to the storing of a large number of keys to ensure that two devices that do meet already have a shared key. Key management, especially key establishment (the process whereby two devices end up with a shared key) is, therefore, important yet challenging [1]–[5].

Physical-layer key generation schemes [6], [7] are a kind of key establishment method that generates the shared key based on the physical properties of channel measurements of two devices and some schemes have been proven to be information-theoretic secure [8]. If considering current key management standards, like ISO/IEC 11770, models herein always have the assumption that the parties to the establishment already have a trusted relationship, i.e., a shared key, and that they can then establish a new key. The low complexity of generating physical-layer keys on demand and the benefit that it needs no prior shared secrets make this type of key establishment a promising alternative to classical cryptographic methods of key establishment [9]. Studying the vulnerabilities of physical layer key generation schemes to construct more robust and secure schemes is one of the important future research scopes [10], while the approach has also been proposed in industrial applications [11].

While passive attacks on physical-layer key generation schemes attract much attention, only a few authors focus on analyzing active attacks. There are two kinds of active attacks. One is the jamming attack [7], [12], which aims at decreasing the speed of key generation. The other is the key manipulation attack [13], [14], which aims at manipulating parts of or the whole generated key. To mitigate the second kind of active attack, researchers mainly focus on authenticating transmitters by leveraging the fingerprints of previous signals that have not been attacked. There are two kinds of fingerprints: hardware fingerprints [15], [16] and channel fingerprints [17], [18]. However, hardware fingerprints have been broken by estimating and reproducing attacks [19], while channel fingerprints are vulnerable to an analog man-in-the-middle (MITM) attack by contaminating the signals used to generate fingerprints [20].

Physical-layer key agreement with User Introduced Randomness (PHY-UIR) [14] is a solution mitigating key manipulation attacks, working on similar principles to the Diffie–Hellman

Manuscript received May 10, 2019; accepted May 23, 2019. Date of publication June 19, 2019; date of current version January 4, 2020. This work was supported in part by the Research Grants Council of Hong Kong Project CityU 21204716 and in part by the Fundamental Research Funds for the Central Universities under Grant 531118010328. Paper no. TII-19-1830. *(Corresponding author: Bianxia Du.)*

Q. Hu is with the College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China (e-mail: huqiao@hnu.edu.cn).

B. Du is with the Institute of Software, Chinese Academy of Sciences, Beijing 100190, China (e-mail: dubianxia1992@gmail.com).

K. Markantonakis is with the Information Security Group Smart Card Centre and the IoT Security Centre (SCC), Royal Holloway, University of London, TW20 0EX Egham, U.K. (e-mail: k.markantonakis@rhul.ac.uk).

G. P. Hancke is with the Department of Computer Science, City University of Hong Kong, Hong Kong (e-mail: gp.hancke@cityu.edu.hk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TII.2019.2923662

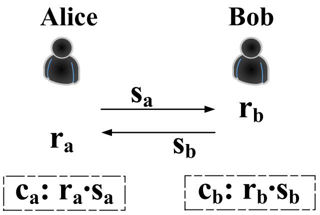


Fig. 1. RSS quantization procedure in PHY-UIR.

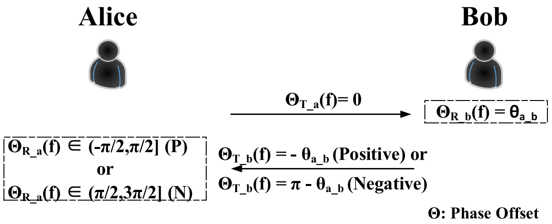

Alice
Bob
$\Theta_{T\_a}(f) = 0$
$\Theta_{R\_b}(f) = \theta_{a\_b}$
$\Theta_{R\_a}(f) \in (-\pi/2, \pi/2]$ (P)
or
$\Theta_{R\_a}(f) \in (\pi/2, 3\pi/2]$ (N)
$\Theta_{T\_b}(f) = -\theta_{a\_b}$ (Positive) or
$\Theta_{T\_b}(f) = \pi - \theta_{a\_b}$ (Negative)
Θ: Phase Offset

Fig. 2. Phase quantization procedure in PHY-UIR.

key exchange protocol. The question is whether this protocol is also vulnerable to the MITM attack, which the Diffie–Hellman protocol suffers from. In our previous paper [21], we proposed the possibility of launching a kind of MITM attack called the session hijacking attack, which utilizes a fast reactive jamming technology to force legitimate devices generating the shared key with signals from the attacker when these devices are executing the PHY-UIR protocol. This paper adds theoretical analysis on the factors that impact on the success rate of the session hijacking attack and proposes a countermeasure against the session hijacking attack.

The main contributions of this paper are as follows.

1) We analyze the feasibility of the session hijacking attack in more detail and identify the main factors that impact on the success rate of the attack.
2) We investigate the impact of these factors by simulation, and then conduct experiments with software-defined radios to validate the simulation result.
3) An enhanced scheme PHY-UIR$^+$ is proposed and tested to mitigate the attack in question.

II. PHY-UIR Key Generation Method

In this section, we depict the workflow of the PHY-UIR key generation method. As with other generation schemes, both users exchange orthogonal frequency-division multiplexing (OFDM) signals and generate the key from received OFDM signals, but PHY-UIR extracts the key from two aspects: received signal strength (RSS) and phase.

PHY-UIR extracts two bits from each round of signal exchange. We show the process in Fig. 1. Unlike with other key agreement schemes, devices in PHY-UIR transmit different signals in each round and multiply what they receive with what they transmit to generate the shared signals and apply a multilevel quantization method proposed in [22]. PHY-UIR measures the phase of the subcarrier frequency of the signal to determine the sign of the corresponding two bits. We illustrate the process in Fig. 2. Bob receives the signal with a phase offset θ_{a_b}. Then, he compensates for this phase offset by adjusting the phase offset of its signal. Adding an additional π phase change indicates a negative sign, otherwise, the value has a positive sign. Alice can also know the sign from the phase offset she calculates from what she receives.

III. Analysis of Session Hijacking Attack

In this section, we depict our session hijacking attack in detail. Our attacker hijacks the communication during the channel probing period and establishes two different shared keys with each of the two devices separately. Our attacker can act as a legitimate participant to communicate with both devices. The main element of the attack is to eliminate the randomness introduced by the user and replace the randomness of channel state information (CSI). To achieve this, the attacker transmits powerful fake signals to overshadow legitimate signals. In this way, the randomness of the fake signals and the CSI between the attacker and users are known to the attacker and the attacker can generate shared keys with Alice or Bob. The shared keys consist of two parts, which are generated by RSS and phase quantization schemes. We analyze how our attack affects these two schemes separately. Table I summarizes the notations we use in this paper.

A. OFDM Model

First, we discuss background information of OFDM. In a discrete-time system, an OFDM symbol consists of many sample points that can be expressed as

$$s(k) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} S(n) e^{j2\pi nk/N} \tag{1}$$

where $s(k)$ is the value of the kth point of the OFDM symbol, N is the number of subcarriers, and $S(n)$ is the data that has been modulated onto the nth subcarrier in this symbol. Here, we assume that the original phase offset is 0.

Parts of an OFDM symbol that corresponds to the nth subcarrier can be expressed as

$$s(k,n) = \frac{1}{\sqrt{N}} S(n) e^{j2\pi nk/N}. \tag{2}$$

After transmission, the OFDM symbol will experience various reflections, backscattering, and refraction in the multipath channel, which we represent by adopting the multipath model in [23]. In each channel tap, the channel impulse response (CIR) can be expressed as

$$h(l,t) = \mu(\tau_l, t) e^{j\phi(\tau_l, t)} \tag{3}$$

where l represents the lth channel path. $\mu_l(t)$ and $\phi_l(t)$ are the amplitude attenuation and phase offset caused by this channel path at time t. The total CIR of this channel can be written as

$$h_{\text{whole}}(\tau,t) = \sum_{l=0}^{L-1} h(l,t) \delta(\tau - \tau_l) \tag{4}$$

where L is the total number of the channel taps, τ_l is the delay time of the lth channel tap, and δ is the Dirac delta function. As

TABLE I
SYMBOLS FOR EQUATIONS USED IN THIS PAPER

Symbol	Notation
$s(k)$	The value of the k^{th} point of an OFDM symbol
$s(k,n)$	The value of the k^{th} point of an OFDM symbol at the n^{th} subcarrier
$r(k,n)$	The value of the k^{th} point of the signal received at the receiver corresponding to the n^{th} carrier
$s_a(k,n)$ $(s_b(k,n))$	Signals transmitted by Alice (Bob)
$s_{att\_a}(k,n)$ $(s_{att\_b}(k,n))$	Signals transmitted by the attacker to Alice (Bob)
$s_{att\_att}(k,n)$	Signals transmitted by the attacker to the attacker
$r_a(k,n)$ $(r_b(k,n))$	Signals received by Alice (Bob)
$r_{att\_a}(k,n)$ $(r_{att\_b}(k,n))$	Signals received by the attacker from Alice (Bob)
$c_a(k,n)$ $(c_b(k,n))$	Final signals used to generate the shared key by Alice (Bob)
$c_{a\_att}(k,n)$ $(c_{b\_att}(k,n))$	Final signals used to generate the shared key by the attacker corresponding to Alice (Bob)
$K\_S_a(k,n)$	Main parts of $c_{a\_att}(k,n)$ to generate the shared key
$w(k,n)$	White Gaussian Noise
$h_{whole}(\tau,t)$	The total CIR of a channel
$h(l,t)$	CIR of the l^{th} channel tap
$h_{a\_b}(l_{a\_b})$	CIR of the $l_{a\_b}^{th}$ channel tap in the channel between Alice and Bob
$h_{a\_att}(l_{a\_att})$	CIR of the $l_{a\_att}^{th}$ channel tap in the channel between Alice and the attacker
$h_{b\_att}(l_{b\_att})$	CIR of the $l_{b\_att}^{th}$ channel tap in the channel between Bob and the attacker
h_{att_att}	CIR of the channel between two antennas on the attacker
τ_l	The delay time of l^{th} channel tap
ε	The time offset between two devices
$\Theta_{R\_b}(n)$ $(\Theta_{R\_att}(n))$	Phase offset measured by Bob (the attacker)
$rand_b(n)$ $(rand_{att}(n))$	A random number generated by Bob (the attacker). It has only two possible values, 0 and 1
k_{b_a} (k_{att_a})	Total offset measured by Alice from the received signals coming from Bob (the attacker)
k_{a_b} (k_{att_b})	Total offset measured by Bob from the received signals coming from Alice (the attacker)
k_{a_att} (k_{b_att})	Total offset measured by the attacker from the received signals coming from Alice (Bob)
$\tau_{l_{a\_b}}$	The delay time of l^{th} channel tap in the channel between Alice and Bob
$\tau_{l_{a\_att}}$	The delay time of l^{th} channel tap in the channel between Alice and the attacker
$\tau_{l_{b\_att}}$	The delay time of l^{th} channel tap in the channel between Bob and the attacker
ε_{b_a} (ε_{a_b})	Time offset between Alice and Bob which is measured by Alice (Bob)
ε_{att_a} (ε_{a_att})	Time offset between Alice and the attacker which is measured by Alice (the attacker)
ε_{att_b} (ε_{b_att})	Time offset between Bob and the attacker which is measured by Bob (the attacker)
τ_{ahead_a} (τ_{ahead_b})	Differences of arrival times between the attack signals and the signals from Alice (Bob).
R,S,H,Z	Equivalent frequency domain value of r,s,h,w
η_{att},η_b	Amplitude attenuation

in the model, the tap delays are assumed to be sample-spaced, which means that $\tau_l = lT$. T is the sampling period of the receiving device.

According to [24], a channel in a rich scattering environment is divided into small time frames. Each time frame can be treated as a wide-sense stationary (WSS) random process, which means each time frame is uncorrelated with the other one. Each attacking unit in this paper can be treated as a WSS random process. As an example, we analyze our attack in one attack unit. For coherence time T_{co}, the CIRs of the channels are constant and the length of one attack unit is less than one coherence time, for simplicity we omit t in (3) and (4).

In this paper, we also assume that the transmitter and the receiver both have a perfect sampling clock [23]. Also as in a slow fading environment, h can be treated as constant for one OFDM symbol. Then, the signal received at the receiver corresponding to the nth carrier can be represented by the following equation [23]:

$$r(k,n) = \sum_{l=0}^{L-1} h(l)s(k-\tau_l-\varepsilon,n) + w(k,n) \tag{5}$$

where ε is the time offset that is caused by the imperfect synchronization. The synchronization is used to find the start point of the signal to extract the data perfectly. Due to the impact of the channel, the noise and the synchronization protocol, there is always imperfect synchronization. w is white Gaussian noise (WGN).

B. Attack Model

There are three entities in our attack model: an attacker and two legitimate devices, Alice and Bob. Devices generate the shared key with the help of PHY-UIR. The attacker can sniff legitimate signals and transmit attack signals during the channel probing stage to hijack the communication. What the attacker does not know are the amplitudes of the signals before they are transmitted and the channel state between two legitimate devices. The attacker has knowledge of the procedures of PHY-UIR and some settings only used for communication, such as carrier frequency and modulation scheme. We adopt the same wireless network settings used in [25] and during the period of coherence time, all channels between the three entities have constant states.

We analyze the impact of our session hijacking attack on these two schemes as follows.

C. Hijacking Attack on RSS Quantization Scheme

We illustrate the workflow of our attack on the RSS quantization step in Fig. 3. We choose one bidirectional signal exchange as an example. In this attack, s_a and s_b are normal signals with random amplitude. $s_{\text{att\_b}}$ and $s_{\text{att\_a}}$ are the attack signals transmitted by the attacker with constant amplitude to Bob and Alice separately. According to Fig. 2, Bob will compensate the offset in s_b. Also, the attacker will copy this action to enable the attack that we will depict later in this section. The RSS quantization scheme chooses a certain subcarrier to generate the shared key. The signals at this subcarrier can be expressed as

$$s_a(k,n) = A_a s(k,n) \tag{6}$$

$$s_b(k,n) = A_b \frac{1}{\sqrt{N}} S(n) e^{j(2\pi nk/N - \Theta_{R\_b}(n) + \text{rand}_b * \pi)} \tag{7}$$

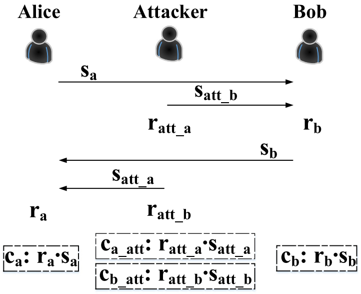

Alice
Attacker
Bob
$\mathbf{s_a}$
$\mathbf{s_{att\_b}}$
$\mathbf{r_{att\_a}}$
$\mathbf{r_b}$
$\mathbf{s_b}$
$\mathbf{s_{att\_a}}$
$\mathbf{r_a}$
$\mathbf{r_{att\_b}}$
$\mathbf{c_a: r_a \cdot s_a}$
$\mathbf{c_{a\_att}: r_{att\_a} \cdot s_{att\_a}}$
$\mathbf{c_{b\_att}: r_{att\_b} \cdot s_{att\_b}}$
$\mathbf{c_b: r_b \cdot s_b}$

Fig. 3. Workflow of a session hijacking attack on the RSS quantization scheme.

$$s_{\mathrm{att}\_b}(k,n) = A_{\mathrm{att}} s(k,n) \tag{8}$$

$$s_{\mathrm{att}\_a}(k,n) = A_{\mathrm{att}} \frac{1}{\sqrt{N}} S(n) e^{j(2\pi nk/N - \Theta_{R\_\mathrm{att}}(n) + \mathrm{rand}_{\mathrm{att}} * \pi)}. \tag{9}$$

In (6) and (7), A_a and A_b are coefficients used by Alice and Bob to change the amplitude of the signals. A_{att} is the coefficients used by the attacker. Each signal has a constant coefficient but from the view of the whole channel probing procedure, A_a and A_b vary randomly from one signal to another to form the randomness introduced by the user while A_{att} is a constant value. Θ_{R_b} is the phase offset measured by Bob. Θ_{R_att} is the phase offset measured by the attacker from the signal transmitted by Alice. Bob and the attacker make a decision on the sign of current signal by rand, 0 means positive, and 1 means negative. Then, the received signals r_a and r_b at the certain subcarrier are

$$\begin{aligned} r_a(k,n) = & \sum_{l_{a\_b}=0}^{L_{a\_b}-1} h_{a\_b}(l_{a\_b}) s_b(k_{b\_a}, n) \\ & + \sum_{l_{a\_\mathrm{att}}=0}^{L_{a\_\mathrm{att}}-1} h_{a\_\mathrm{att}}(l_{a\_\mathrm{att}}) s_{\mathrm{att}\_a}(k_{\mathrm{att}\_a}, n) + w_a(k,n) \end{aligned} \tag{10}$$

where

$$\begin{aligned} k_{b\_a} &= k - \tau_{l_{a\_b}} - \varepsilon_{b\_a} + \tau_{\mathrm{ahead}\_a} \\ k_{\mathrm{att}\_a} &= k - \tau_{l_{a\_\mathrm{att}}} - \varepsilon_{\mathrm{att}\_a} \end{aligned}$$

and

$$\begin{aligned} r_b(k,n) = & \sum_{l_{a\_b}=0}^{L_{a\_b}-1} h_{a\_b}(l_{a\_b}) s_a(k_{a\_b}, n) \\ & + \sum_{l_{b\_\mathrm{att}}=0}^{L_{b\_\mathrm{att}}-1} h_{b\_\mathrm{att}}(l_{b\_\mathrm{att}}) s_{\mathrm{att}\_b}(k_{\mathrm{att}\_b}, n) + w_b(k,n) \end{aligned} \tag{11}$$

where

$$\begin{aligned} k_{a\_b} &= k - \tau_{l_{a\_b}} - \varepsilon_{a\_b} + \tau_{\mathrm{ahead}\_b} \\ k_{\mathrm{att}\_b} &= k - \tau_{l_{b\_\mathrm{att}}} - \varepsilon_{\mathrm{att}\_b}. \end{aligned}$$

In the abovementioned two equations, L_{a_b}, L_{a_att}, and L_{b_att} are the total number of the channel taps in channels Alice–Bob, Alice–Attacker, and Bob–Attacker, respectively. The received s_a and s_b also have the same time offset with their relayed versions, because the receiver has synchronized with their relay versions. τ_{ahead_a} and τ_{ahead_b} are the time differences that the signals arrives at the receiver ahead of the attack signals.

According to PHY-UIR, the final signals used to generate parts of the shared key by the RSS quantization scheme at legitimate devices are

$$\begin{aligned} c_a(k,n) = & \sum_{l_{a\_b}=0}^{L_{a\_b}-1} h_{a\_b}(l_{a\_b}) s_a(k,n) s_b(k_{b\_a}, n) \\ & + s_a(k,n) w_a(k,n) \\ & + \sum_{l_{a\_\mathrm{att}}=0}^{L_{a\_\mathrm{att}}-1} h_{a\_\mathrm{att}}(l_{a\_\mathrm{att}}) s_a(k,n) s_{\mathrm{att}\_a}(k_{\mathrm{att}\_a}, n) \end{aligned} \tag{12}$$

and

$$\begin{aligned} c_b(k,n) = & \sum_{l_{a\_b}=0}^{L_{a\_b}-1} h_{a\_b}(l_{a\_b}) s_b(k,n) s_a(k_{a\_b}, n) \\ & + s_b(k,n) w_b(k,n) \\ & + \sum_{l_{b\_\mathrm{att}}=0}^{L_{b\_\mathrm{att}}-1} h_{b\_\mathrm{att}}(l_{b\_\mathrm{att}}) s_b(k,n) s_{\mathrm{att}\_b}(k_{\mathrm{att}\_b}, n). \end{aligned} \tag{13}$$

From the view of the attacker, received signals at corresponding subcarrier during the signal transmission time of Alice and Bob are r_{att_a} and r_{att_b} shown as

$$\begin{aligned} r_{\mathrm{att}\_a}(k,n) = & \sum_{l_{a\_\mathrm{att}}=0}^{L_{a\_\mathrm{att}}-1} h_{a\_\mathrm{att}}(l_{a\_\mathrm{att}}) s_a(k_{a\_\mathrm{att}}, n) \\ & + h_{\mathrm{att}\_\mathrm{att}} s_{\mathrm{att}\_\mathrm{att}}(k,n) + w_{a\_\mathrm{att}}(k,n) \end{aligned} \tag{14}$$

where

$$k_{a\_\mathrm{att}} = k - \tau_{l_{a\_\mathrm{att}}} - \varepsilon_{a\_\mathrm{att}}$$

and

$$\begin{aligned} r_{\mathrm{att}\_b}(k,n) = & \sum_{l_{b\_\mathrm{att}}=0}^{L_{b\_\mathrm{att}}-1} h_{b\_\mathrm{att}}(l_{b\_\mathrm{att}}) s_b(k_{b\_\mathrm{att}}, n) \\ & + h_{\mathrm{att}\_\mathrm{att}} s_{\mathrm{att}\_\mathrm{att}}(k,n) + w_{b\_\mathrm{att}}(k,n) \end{aligned} \tag{15}$$

where

$$k_{b\_\mathrm{att}} = k - \tau_{l_{b\_\mathrm{att}}} - \varepsilon_{b\_\mathrm{att}}.$$

In the abovementioned two equations, r_{att_a} and r_{att_b} represent the signal corresponding to the certain subcarrier received by the attacker. These two equations are a little different than (10) and (11). As the attack signals are transmitted and received all by the attacker and the receiving antenna and the transmitting antenna are very close at the attacker there is almost no multipath effect. We use $h_{\mathrm{att}\_\mathrm{att}}$ to represent the CIR of the channel between the receiving antenna and the transmitting antenna of the attacker.

As self-interference cancelation technology can be used to eliminate the interference caused by $h_{\mathrm{att}\_\mathrm{att}} s_{\mathrm{att}\_\mathrm{att}}$, we adopt the technology in [26], which can reduce the self-interference to the noise floor. This means $h_{\text{att\_att}}s_{\text{att\_att}}$ in (14) and (15) can be merged into $w_{a\text{\_att}}$ and $w_{b\text{\_att}}$.

The final signals $c_{a\text{\_att}}$ and $c_{b\text{\_att}}$ that are used for RSS quantization with devices Alice and Bob are

$$c_{a\text{\_att}}(k,n) = \sum_{l_{a\text{\_att}}=0}^{L_{a\text{\_att}}-1} h_{a\text{\_att}}(l_{a\text{\_att}})s_a(k_{a\text{\_att}},n)s_{\text{att\_}a}(k,n) + w_{a\text{\_att}}(k,n)s_{\text{att\_}a}(k,n) \tag{16}$$

and

$$c_{b\text{\_att}}(k,n) = \sum_{l_{b\text{\_att}}=0}^{L_{b\text{\_att}}-1} h_{b\text{\_att}}(l_{b\text{\_att}})s_b(k_{b\text{\_att}},n)s_{\text{att\_}b}(k,n) + w_{b\text{\_att}}(k,n)s_{\text{att\_}b}(k,n). \tag{17}$$

Now legitimate devices and the attack device all get the signals used for key generation through the RSS quantization scheme. This scheme is performed at a certain subcarrier. Parts of c_a and $c_{a\text{\_att}}$ that belongs to the certain subcarrier are used to generate the key between Alice and the attacker while corresponding parts of c_b and $c_{b\text{\_att}}$ are used to generate the key between Bob and the attacker. As Alice and Bob generate the key with the attacker in the same way, we analyze Alice and the attacker only. As the noise can be neglected, the key generated by the attacker to communicate with Alice comes from the rest of $c_{a\text{\_att}}$, which we call it key source K_S_a

$$\begin{aligned} K\_S_a(k,n) &= \sum_{l_{a\text{\_att}}=0}^{L_{a\text{\_att}}-1} h_{a\text{\_att}}(l_{a\text{\_att}})s_a(k_{a\text{\_att}},n)s_{\text{att\_}a}(k,n) \\ &= \sum_{l_{a\text{\_att}}=0}^{L_{a\text{\_att}}-1} h_{a\text{\_att}}(l_{a\text{\_att}})A_a\frac{1}{\sqrt{N}}S(n)e^{j2\pi nk_{a\text{\_att}}/N} \\ &\quad\cdot\left(A_{\text{att}}\frac{1}{\sqrt{N}}S(n)e^{j(2\pi nk/N-\Theta_{R\text{\_att}}+\text{rand}_b*\pi)}\right) \\ &= \sum_{l_{a\text{\_att}}=0}^{L_{a\text{\_att}}-1} h_{a\text{\_att}}(l_{a\text{\_att}})s_a(k,n)s_{\text{att\_}a}(k_{\text{att\_}a}+k_d,n) \end{aligned} \tag{18}$$

where

$$k_d = \varepsilon_{\text{att\_}a} - \varepsilon_{a\text{\_att}}.$$

Alice generates her key based on the power of the signal in (12) and the attacker generates corresponding key based on the power of the signal in (18). Lets compare these two equations. We can see that if k_d is equal to zero, K_S_a is equal to parts of c_a. In (12), except the parts which is equal to K_S_a and the negligible noise, only $\sum_{l_{a\text{\_}b}=0}^{L_{a\text{\_}b}-1} h_{a\text{\_}b}(l_{a\text{\_}b})s_a(k,n)$ $s_b(k_{b\text{\_}a},n)$ remains, which has negative impact on our attack as it may change the generated key. Comparing this remaining part in (12) with (18), we can see that if the power of the signal $\sum_{l_{a\text{\_att}}=0}^{L_{a\text{\_att}}-1} h_{a\text{\_att}}(l_{a\text{\_att}})s_{\text{att\_}a}(k_{\text{att\_}a}+k_d,n)$ has more power than $\sum_{l_{a\text{\_}b}=0}^{L_{a\text{\_}b}-1} h_{a\text{\_}b}(l_{a\text{\_}b})s_b(k_{b\text{\_}a},n)$, the generated key at Alice should be more similar to the key generated at the attacker. $\sum_{l_{a\text{\_att}}=0}^{L_{a\text{\_att}}-1} h_{a\text{\_att}}(l_{a\text{\_att}})s_{\text{att\_}a}(k_{\text{att\_}a}+k_d,n)$ is the signal received by Alice, which is transmitted by the attacker. $\sum_{l_{a\text{\_}b}=0}^{L_{a\text{\_}b}-1} h_{a\text{\_}b}(l_{a\text{\_}b})s_b(k_{b\text{\_}a},n)$ is the signal received by Alice, which is transmitted by the Bob. So there are three factors that have an impact on the phase of the received signals: $\varepsilon_{\text{att\_}a}(\varepsilon_{\text{att\_}b})$, $\varepsilon_{a\text{\_att}}(\varepsilon_{b\text{\_att}})$, and the power ratio of the attack signal to the legitimate signal at nth subcarrier at Alice (Bob).

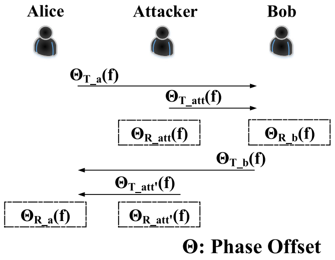


Fig. 4. Workflow of session hijacking attack on phase quantization.

D. Session Hijacking Attack on Phase Quantization Scheme

After RSS quantization, we depict how the session attack affects the phase quantization. The workflow is shown in Fig. 4. At first, Alice transmits a normal signal with no phase offset ($\Theta_{T\text{\_}a}(f)=0$). When the signal arrives at the attacker, it can be written as

$$\begin{aligned} R_{\text{att\_}a}(n) &= \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1} r_{\text{att\_}a}(k,n)e^{-j2\pi nk/N} \\ &= S_a(n)H_{a\text{\_att}}(n)e^{-j2\pi k\varepsilon_{a\text{\_att}}/N} + Z_{a\text{\_att}}(n) \end{aligned} \tag{19}$$

where

$$S_a(n) = \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1} s_a(k,n)e^{-j2\pi nk/N}$$

$$Z_{a\text{\_att}}(n) = \sum_{k=0}^{N-1} w_{a\text{\_att}}(k)e^{-j2\pi nk/N}$$

$$H_{a\text{\_att}}(n) = \sum_{l_{a\text{\_att}}=0}^{L_{a\text{\_att}}-1} h_{a\text{\_att}}(l_{a\text{\_att}})e^{-j2\pi nl_{a\text{\_att}}/N}.$$

The attacker transforms $R_{\text{att\_}a}(n)$ into a signal $s_{R\text{\_att}}$ depending on (20). Then, it extracts the phase offset $\Theta_{R\text{\_att}}$ from $s_{R\text{\_att}}$ as the attacker knows s. $R_{\text{att\_}a}(n)$ can be written as (21). η_{att} is the amplitude attenuation

$$S_{R_{\text{att\_}a}}(k,n) = \frac{1}{\sqrt{N}}R_{\text{att\_}a}(n)e^{j2\pi kn/N} \tag{20}$$

$$\begin{aligned} R_{\text{att\_}a}(n) &= \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1} \eta_{\text{att}}s(k,n)e^{(\Theta_{R\text{\_att}}-j2\pi kn/N)} \\ &= \eta_{\text{att}}S(n)e^{j\Theta_{R\text{\_att}}} \end{aligned} \tag{21}$$

where

$$S(n) = \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1} s(k,n)e^{-j2\pi nk/N}.$$

The signal received by Bob is written as

$$\begin{aligned} R_b(n) &= \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1} r_b(k,n)e^{-j2\pi nk/N} \\ &= S_a(n)H_{a\_b}(n)e^{-j2\pi k(\varepsilon_{a\_b}-\tau_{\text{ahead\_}b})/N} \\ &\quad + S_{\text{att\_}b}(n)H_{b\_\text{att}}(n)e^{-j2\pi k\varepsilon_{\text{att\_}b}/N} + Z_b(n) \end{aligned} \tag{22}$$

where

$$\begin{aligned} S_{\text{att\_}b}(n) &= \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1} s_{\text{att\_}b}(k,n)e^{-j2\pi nk/N} \\ H_{a\_b}(n) &= \sum_{l_{a\_b}=0}^{L_{a\_b}-1} h_{a\_b}(l_{a\_b})e^{-j2\pi nl_{a\_b}/N} \\ H_{b\_\text{att}}(n) &= \sum_{l_{b\_\text{att}}=0}^{L_{b\_\text{att}}-1} h_{b\_\text{att}}(l_{b\_\text{att}})e^{-j2\pi nl_{b\_\text{att}}/N} \\ Z_b(n) &= \sum_{k=0}^{N-1} w_b(k,n)e^{-j2\pi nk/N}. \end{aligned}$$

After Bob has received the signals, he calculates the phase offset $\Theta_{R\_b}(n)$ of the nth subcarrier from $R_b(n)$ by (23) and (24). η_b is the amplitude attenuation

$$s_{R_b}(k,n) = \frac{1}{\sqrt{N}}R_b(n)e^{j2\pi kn/N} \tag{23}$$

$$R_b(n) = \eta_b S(n)e^{j\Theta_b}. \tag{24}$$

As we depict previously, Bob will compensate the calculated phase offset in the signal he transmits. This signal is written as (25) when it arrives at the attacker

$$\begin{aligned} R_{\text{att\_}b}(n) &= \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1} r_{\text{att\_}b}(k,n)e^{-j2\pi nk/N} \\ &= S_b(n)H_{b\_\text{att}}(n)e^{-j2\pi k\varepsilon_{b\_\text{att}}/N} + Z_{b\_\text{att}}(n) \end{aligned} \tag{25}$$

where

$$\begin{aligned} S_b(n) &= \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1} s_b(k,n)e^{-j2\pi nk/N} \\ Z_{b\_\text{att}}(n) &= \sum_{k=0}^{N-1} w_{b\_\text{att}}(k,n)e^{-j2\pi nk/N}. \end{aligned}$$

We can observe from this equation that the phase offset $\Theta_{R\_\text{att}'}$ consists of two kinds of phase offsets caused by the channel impulse, $H_{b\_\text{att}}(n)$, and the time offset, $e^{-j2\pi k\varepsilon_{b\_\text{att}}/N}$. Next, the attacker transmits the attack signal to Alice. The signal received by Alice is written as

$$\begin{aligned} R_a(n) &= \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1} r_a(k,n)e^{-j2\pi nk/N} \\ &= S_b(n)H_{b\_a}(n)e^{-j2\pi k(\varepsilon_{a\_b}-\tau_{\text{ahead\_}a})/N} \\ &\quad + S_{\text{att\_}a}(n)H_{a\_\text{att}}(n)e^{-j2\pi k\varepsilon_{\text{att\_}a}/N} + Z_b(n) \end{aligned} \tag{26}$$

where

$$\begin{aligned} S_{\text{att\_}a}(n) &= \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1} s_{\text{att\_}a}(k,n)e^{-j2\pi nk/N} \\ H_{b\_a} &= H_{a\_b} \\ Z_a(n) &= \sum_{k=0}^{N-1} w_a(k,n)e^{-j2\pi nk/N}. \end{aligned}$$

Because of channel reciprocity, H_{b_a} is equal to H_{a_b}. Equation (26) tells us that there are five factors that have an impact on the phase offset Θ_{R_a} measured by the attacker from the signal transmitted by Bob: the channel impulses H_{a_b} and H_{a_att}, the time offsets ε_{a_b} and $\varepsilon_{\text{att\_}a}$, and the time difference between the legitimate signal and the attack signal, $\tau_{\text{ahead\_}a}$.

Now, it is time to extract bits from the phase offset. Also, we take a look at Alice and the attacker. Alice extracts bits of the shared key determined by rand_{att}, which is determined by the attacker from (26). If the power of the attack signal is much higher than the legitimate signal, $\varepsilon_{\text{att\_}a}$ is equal to $\varepsilon_{a\_\text{att}}$ and we ignore the negligible noise. By incorporating (6), (9), (19), and (21) into (26), $R_a(n)$ can be expressed as (27), which shows that Alice can extract the correct phase offset from $\frac{A_{\text{att}}}{A_a}e^{j\,\text{rand}_{\text{att}}*\pi}\eta_{\text{att}}S(n)$

$$\begin{aligned} R_a(n) &= S_{\text{att\_}a}(n)H_{a\_\text{att}}(n)e^{-j2\pi k\varepsilon_{\text{att\_}a}/N} \\ &= \frac{A_{\text{att}}}{A_a}e^{j(-\Theta_{R\_\text{att}}(n)+\text{rand}_{\text{att}}*\pi)} \\ &\quad \cdot S_a(n)H_{a\_\text{att}}(n)e^{-j2\pi k(\varepsilon_{\text{att\_}a}-\varepsilon_{a\_\text{att}})/N} \\ &= \frac{A_{\text{att}}}{A_a}e^{j\,\text{rand}_{\text{att}}*\pi}\eta_{\text{att}}S(n). \end{aligned} \tag{27}$$

The attacker extracts bits of the shared key determined by Bob from (25). If the power of the attack signal is much higher than the legitimate signal and $\varepsilon_{\text{att\_}b}$ is equals $\varepsilon_{b\_\text{att}}$, by incorporating (7), (8), (22), and (24) into (25), $R_{\text{att\_}b}(n)$ can be changed into (28)

$$\begin{aligned} R_{\text{att\_}b}(n) &= S_b(n)H_{b\_\text{att}}(n)e^{-j2\pi k\varepsilon_{b\_\text{att}}/N} \\ &= \frac{A_b}{A_{\text{att}}}e^{j(-\Theta_{R\_b}(n)+\text{rand}_{\text{att}}*\pi)} \\ &\quad \cdot S_{\text{att\_}b}(n)H_{b\_\text{att}}(n)e^{-j2\pi k(\varepsilon_{b\_\text{att}}-\varepsilon_{\text{att\_}b})/N} \\ &= \frac{A_b}{A_{\text{att}}}e^{j\,\text{rand}_{\text{att}}*\pi}\eta_b S(n). \end{aligned} \tag{28}$$

According to the abovementioned analysis, we find that there are also three factors that have an impact on the phase of the received signals: $\varepsilon_{\text{att\_}a}$($\varepsilon_{\text{att\_}b}$), $\varepsilon_{a\_\text{att}}(\varepsilon_{b\_\text{att}})$, and the power ratio of the attack signal to the legitimate signal at the nth subcarrier at Alice (Bob).

IV. Simulation Analysis

In Sections III-C and III-D, we analyze the factors that have an impact on the generation of the shared keys. In this section, we utilize MATLAB to measure the impact of these factors on the final keys.

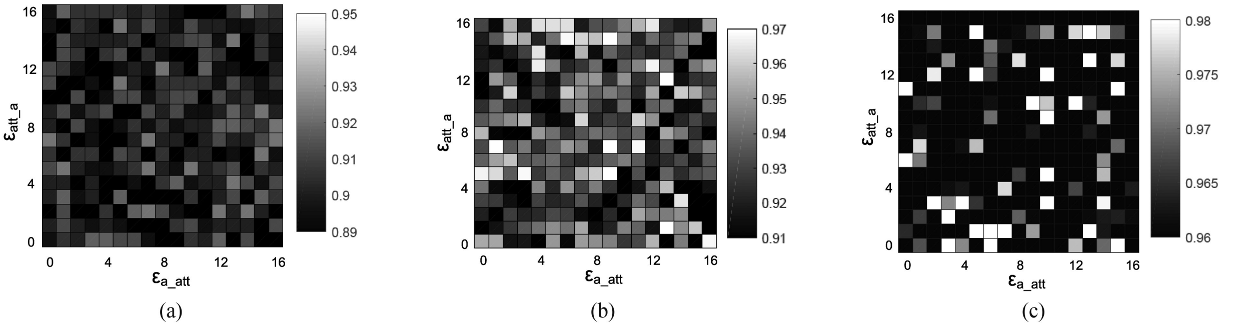


Fig. 5. Key similarity of RSS quantization between Alice and the attacker. (a) $\text{PowR}_{\text{att\_leg}} = 5$ dB. (b) $\text{PowR}_{\text{att\_leg}} = 10$ dB. (c) $\text{PowR}_{\text{att\_leg}} = 20$ dB.

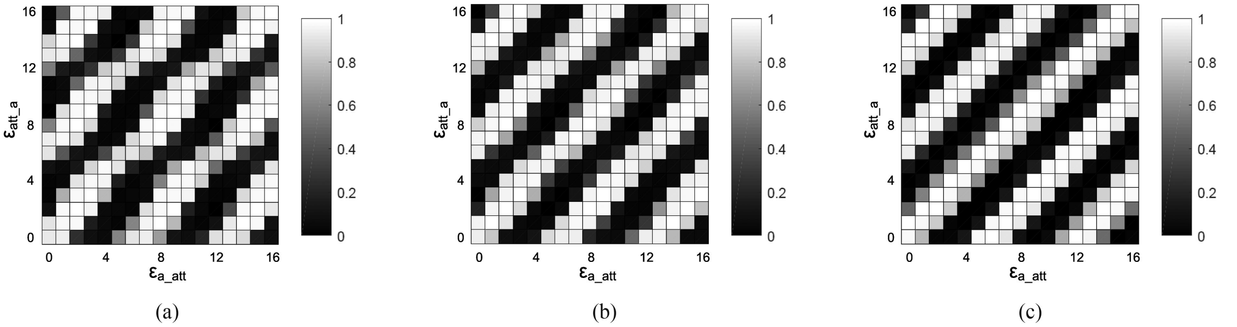


Fig. 6. Key similarity of phase quantization between Alice and the attacker. (a) $\text{PowR}_{\text{att\_leg}} = 5$ dB. (b) $\text{PowR}_{\text{att\_leg}} = 10$ dB. (c) $\text{PowR}_{\text{att\_leg}} = 20$ dB.

A. Setup

MATLAB is used to simulate the attack. All devices adopt the IEEE 802.11 OFDM protocol to communicate. Also, a multipath fading channel simulation scheme [27] is chosen to simulate the statistical wireless channel. The sampling rate is 20 MHz. The Doppler spread is 5 Hz and the root mean square delay spread is 150 ns. The power of the legitimate signals varies from 50 to 150% compared to the standard signal we choose to provide the user introduced randomness. The power of the attack signal is set to be 5, 10, and 20 dB higher than the standard signal. In 802.11a, there is a period called the guard interval that lasts 16 sample points to avoid Inter-Symbol Interference. So we specify that the time offset ε in our simulation is no larger than 16 sample points, or 0.8 ns. In our simulation, we test the key similarity rate under all possible time offset values for three kinds of power ratio. As the key is generated from both RSS quantization process and phase quantization process, we test these two processes separately.

B. RSS Quantization

As discussed in Section III-C, the key generated by Alice (Bob) and the attacker from RSS quantization process is affected by three factors: $\varepsilon_{\text{att\_}a}\,(\varepsilon_{\text{att\_}b})$, $\varepsilon_{a\text{\_att}}(\varepsilon_{b\text{\_att}})$, and the power ratio of the received attack signals to the received legitimate signals at the nth subcarrier at Alice (Bob). We use $\text{PowR}_{\text{att\_leg}}$ to represent this power ratio. We illustrate the result of how these three factors affect the shared key in Fig. 5. The unit of $\varepsilon_{\text{att\_}a}$ and $\varepsilon_{a\text{\_att}}$ is one sample point. From this figure, we can observe that the key similarity between Alice and the attacker is mainly affected by $\text{PowR}_{\text{att\_leg}}$. When $\text{PowR}_{\text{att\_leg}}$ is higher, the similarity of the shared key is higher.

C. Phase Quantization

Section III-D shows the three factors that have impact on the result of phase quantization: $\varepsilon_{\text{att\_}a}\,(\varepsilon_{\text{att\_}b})$, $\varepsilon_{a\text{\_att}}(\varepsilon_{b\text{\_att}})$, and $\text{PowR}_{\text{att\_leg}}$. We only simulate the impact of $\varepsilon_{\text{att\_}a}$, $\varepsilon_{a\text{\_att}}$, and $\text{PowR}_{\text{att\_leg}}$ for simplicity. We illustrate the result of how these three factors affect the shared key in Fig. 6. The unit of $\varepsilon_{\text{att\_}a}$ and $\varepsilon_{a\text{\_att}}$ is one sample point. This table shows that the difference between $\varepsilon_{\text{att\_}a}$ and $\varepsilon_{a\text{\_att}}$ is the key factor that affect the similarity between the key generated by phase quantization. With the increasing of the difference between $\varepsilon_{\text{att\_}a}$ and $\varepsilon_{a\text{\_att}}$, the key similarity varying periodically.

V. Performance Evaluation

To evaluate the performance of our session hijacking attack, we utilize software defined radios (USRP B200), to transmit and receive signals to acquire real CSI information as in Fig. 7. The two devices are about 7 m away from each other and there are furniture and people between them. The attacker is located at one of eight different places, from A1 to A8. The power of the signals transmitted by Alice and Bob vary from 50 to 150% compared to a standard signal. The power of the attack signal at the attacker is set to be 5, 10, and 20 dB higher than the power of the standard signals in three tests. As in [14], the carrier frequency of the RF signal is 2.4 GHz and the sampling rate is 200 kHz.

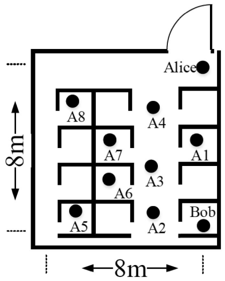


Fig. 7. Topology.

We implement the channel probing parts of PHY-UIR in both legitimate devices, as the main target of our attack is on this part. We also implement a timing synchronization scheme [28]. As we do not block the communication, a real-time reactive jamming technology is indispensable. In our experiment, Alice and Bob transmit signals at regular intervals. Our attacker records the arrival time of the first signals of Alice and Bob to estimate appropriate attacking time.

The result is shown in Fig. 8. The Y-axis represents the similarity between the shared keys extracted at the quantization step. PowerRatio is the power ratio of the attack signal to the legitimate signal. The X-axis indicates the proximity to the legitimate device. The proximity increases from left to the right on the X-axis. We can observe that when the attacker transmits the attack signal with identical power at all eight places, the similarity of the generated keys decreases as the proximity between the attacker and legitimate device increases. This is because longer distance brings higher attenuation, which leads to lower power signals. So the similarity of the generated shared keys mainly depends on the power of the attack signal, but we also observe a sharp decline when the power ratio is 5 dB. The reason is the rapid decreasing of the timing offset when the power ratio increasing from 5 to 10 dB[28], which leads to more stable high similarity of the key generated by phase quantization.

When the power of the attack signal is 5 dB higher than the standard signal, and if the attacker is located further away from Alice or Bob, the key similarity rate would be too low to achieve key agreement, due to the low power of the attack signal at the receiver. However, when the power ratio increases to 10 or 20 dB, our attack succeeds at all eight positions.

VI. Enhanced PHY-UIR

We have shown that our session hijacking attack can break PHY-UIR. We introduce a third kind of randomness to enhance PHY-UIR against the attack. We call this enhanced scheme PHY-UIR$^+$ and describe it further in this section.

A. Workflow of PHY-UIR$^+$

PHY-UIR$^+$ prevents our attack by adding an extra step to PHY-UIR where information is simultaneously exchanged between the honest parties to ensure that the keys they agreed on are the same. In this step, PHY-UIR$^+$ specifies that honest parties should adopt binary frequency-shift keying (BFSK) technology to modulate the data. A BFSK modulation scheme is shown as

$$s(x) = \begin{cases} \cos(2\pi f_0 t) & x = \text{“0”} \\ \cos(2\pi f_1 t) & x = \text{“1”} \end{cases}.$$

This scheme uses cosine waves with different frequency to represent different bits.

We show the workflow of PHY-UIR$^+$ in Fig. 9. As an example, we adopt the SHA-256 hash function.

Step 1: Alice and Bob execute PHY-UIR to generate the key: K_a and K_b.

Step 2: Alice and Bob generate signals S_a and S_b separately. $S_a(S_b)$ contains message $H(K_a)(H(K_b))$, the hash value of key $K_a(K_b)$. Then, they exchange the signals simultaneously.

Step 3: The devices perform a noise cancelation scheme to remove the self-interference of the signal they receive. It means that the rest of the received signals at Alice and Bob are S_b and S_a, respectively. Both devices recover the messages and compare the message with the hash value of their keys to check the existence of the attacker.

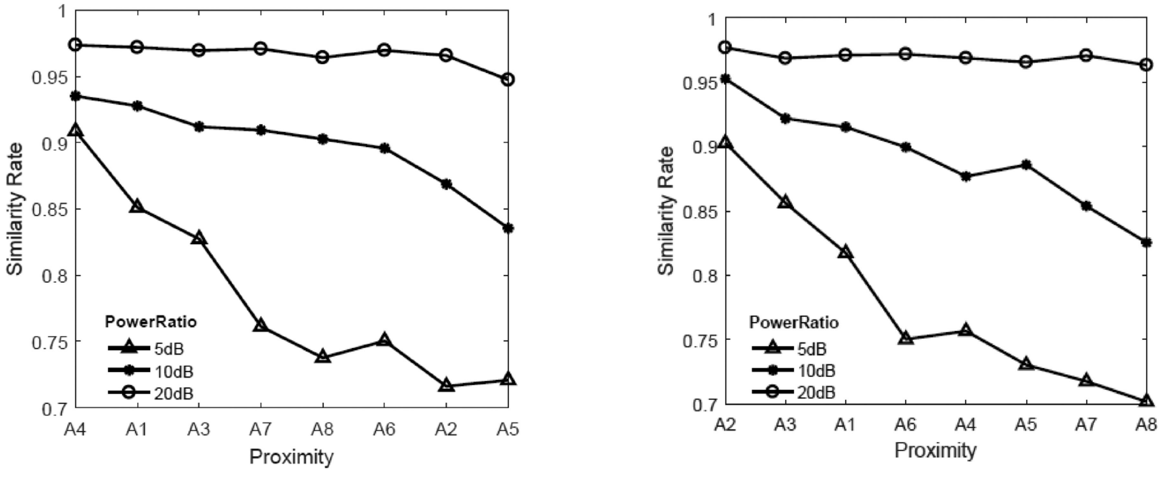
We can see that if K_a and K_b are the same, the results at Alice and Bob both are consistent. Inconsistency represents the existence of the attacker. Devices should transmit powerful noise to alert the other device.

If there exists an attacker, basically K_a and K_b are different from each other, then $H(K_a)$ is not equal to $H(K_b)$. The average amount of effort needed by the attacker to create two different keys amounts to a collision resistance of SHA-256, which is on average $2^{N/2} = 2^{128}$ hash attempts, which is considered infeasible to calculate. Given that a hash also exhibits a one-way property, an attacker cannot obtain additional information from the key based on the hash transmitted.

B. Performance Experiment

1) Attack Detection Performance: We conduct the experiment with the same setting in Section V and Fig. 7. As Alice and Bob detect the attacker by comparing the hash values of their keys, we measure the similarity between these hash values to reflect the effectiveness of PHY-UIR$^+$. The result is illustrated in Fig. 10. The Y-axis refers to the similarity rate measured at Alice and Bob. We can observe from this figure that the similarity rates always locate in the range of 45–57% at both Alice and Bob. This means Alice and Bob can detect the attacker effectively.

2) Energy Consumption Performance: To give an indication of energy consumption of PHY-UIR$^+$ relative to PHY-UIR, we implement the basic framework of PHY-UIR$^+$ and PHY-UIR in Sony Xperia e3 mobile phones. This system acts as a protocol emulator where we do the cryptographic functions and data transmissions as required, although not all signal processing functions are fully implemented, for example, the phone cannot determine CSI. This provides a good estimation of IoT devices as Qualcomm Snapdragon processors are promoted for embedded and IoT devices, and the mobile device also allows us to measure the cost of wireless transmission (802.11), which is the main cause of power consumption. We keep running these



(a)

(b)

Fig. 8. Similarity of the shared key. (a) Key similarity rate between Alice and the attacker. (b) Key similarity rate between Bob and the attacker.

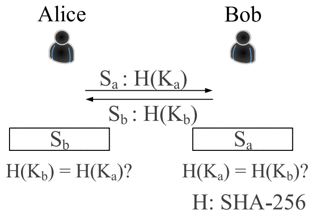


Fig. 9. Workflow of PHY-UIR$^+$.

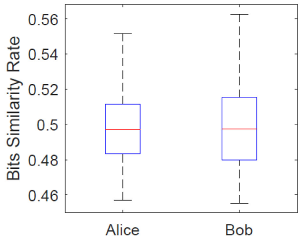


Fig. 10. Similarity rate between $H(K_a)$ and $H(K_b)$ at Alice and Bob.

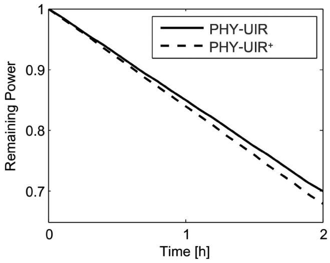


Fig. 11. Energy consumption of PHY-UIR and PHY-UIR$^+$.

two schemes on the phones and record the remaining power of the phones. The result is shown in Fig. 11. This figure shows that PHY-UIR$^+$ uses only a little more energy than PHY-UIR, which means PHY-UIR$^+$ offers improved security while being similar in term of energy efficiency.

C. Security Analysis

In this section, we analyze the security of PHY-UIR$^+$ against the session hijacking attack. We consider three potential strategies the attacker will adopt.

Security model: For the purpose of analyzing, we consider the following security model.

Definition 1 (Security Objectives): The proposed scheme has the following security requirements.

1) The scheme should detect a session hijacking attack.
2) The scheme should have no requirement of preshared secret information.

One of the main advantages of PHY-UIR is that it requires no preshared secret information to generate shared key. The solution should inherit this advantage.

Definition 2 (Adversary): The adversary is an attacker who can capture legitimate signals and send its own attack signals to overshadow target signals. The adversary has full knowledge of the design of PHY-UIR and PHY-UIR$^+$, but without unlimited computing resources.

As the protocol generates keys for each communication session, the adversary needs to launch session hijacking attacks in each communication session. Thus, security analysis considers only one protocol execution.

1) Brute-Force Search Attack:

To investigate the ability of PHY-UIR$^+$ to resist brute-force search, we present Lemma 1 as follows.

Lemma 1: If there is no session hijacking attacker, the possibility that the eavesdropper guesses the shared key correctly at the final key exchanging step is no higher than $(\frac{1}{2})^{N_{\text{key}}}$. N_{key} is the length of the shared key.

Proof: According to [24], if the channel in a rich scattering environment is divided into small time frames, each time frame can be treated as a WSS random process, which means each time frame is uncorrelated with the other one. We also treat the OFDM signals in this paper as a WSS random process. ■

Each signal can extract three bits, two bits by RSS and one bit by phase. There are eight possible values and we set the

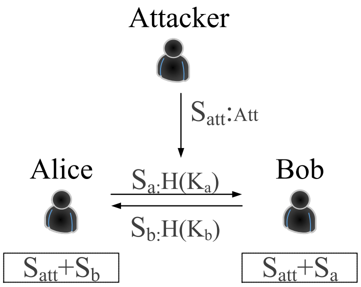


Fig. 12. Smart session hijacking attack.

possibility of these eight values as Pr_1 to Pr_8

$$Pr_1 + Pr_2 + Pr_3 + Pr_4 + Pr_5 + Pr_6 + Pr_7 + Pr_8 = 1. \tag{29}$$

The attacker guesses the value with equal possibility to choose one of the eight values. The possibility that the attacker recover the correct bits is

$$\begin{aligned}&\frac{1}{8}(Pr_1 + Pr_2 + Pr_3 + Pr_4 + Pr_5 + Pr_6 + Pr_7 + Pr_8)\\&\quad= \frac{1}{8}.\end{aligned} \tag{30}$$

Because the bits generated from each signal are uncorrelated with the others, the success possibility of the attacker is

$$\left(\frac{1}{8}\right)^{\frac{N_{\text{key}}}{3}} = \left(\frac{1}{2}\right)^{N_{\text{key}}}. \tag{31}$$

2) Signal Separation Attack: This attack tries to separate S_a and S_b from the mixed signal and then recover $H(K_a)$ and $H(K_b)$ with the purpose of recovering K_a and K_b. As a hash function is a one-way function, it is hard to deduce the key from its hash value.

3) Smart Session Hijacking Attack: Lets consider a smart session hijacking attack where the attacker tries to overshadow the extra step that PHY-UIR$^+$ adds after it launches the session attack. We illustrate this attack in Fig. 12.

The attacker transmits signal with malicious data Att, which has the same length with $H(K_a)$ and $H(K_b)$. Now we are going to prove the effectiveness of our method against the smart session hijacking attack.

Lemma 2: When the attacker and Alice (Bob) transmit the same bit at the extra step in PHY-UIR$^+$, Bob (Alice) receives the same bit.

Proof: When Alice and the attacker transmit the same bit, the signal Bob receives is expressed as (32). As they sent the same bit, both signals have identical frequency f, but they have different amplitude and phase shift

$$\begin{aligned}R_b &= S_{\text{att}} + S_a\\&= \alpha_{\text{att}}\cos(2\pi f t + \theta_{\text{att}}) + \alpha_a \cos(2\pi f t + \theta_a)\end{aligned} \tag{32}$$

where α_{att} and α_a are the amplitude of signals from the attacker and Alice; θ_{att} and θ_a are the phase offset of signals from the attacker and Alice. ■

Equation (32) can change into (33) [29]

$$R_b = \alpha\cos(2\pi f t + \theta) \tag{33}$$

where

$$\begin{aligned}\alpha &= \sqrt{\alpha_{\text{att}}^2 + \alpha_a^2 + 2\alpha_{\text{att}}\alpha_a\cos\Delta\theta}\\ \Delta\theta &= \theta_{\text{att}} - \theta_a\\ \tan\theta &= \frac{\alpha_{\text{att}}\sin\theta_{\text{att}} + \alpha_a\sin\theta_a}{\alpha_{\text{att}}\cos\theta_{\text{att}} + \alpha_a\cos\theta_a}.\end{aligned}$$

We can observe from (33) that R_b has the same frequency as S_{att} and S_a, which means Bob receives the same bit that is transmitted by Alice and the attacker. So Lemma 2 is proved. Similarly we can prove that when Bob and the attacker transmit the same bit, Alice receives the same bit.

Lemma 3: When a smart session hijacking attack has been launched successfully and there are difference between hash values $H(K_a)$ and $H(K_b)$ generated by two legitimate participants, PHY-UIR$^+$ can always detect the attack.

Proof: First, we specify the operation that legitimate participants use to recover data from received signal as $\bigotimes$. When the attacker and Alice (Bob) transmit different bits, the final bit recovered by Bob (Alice) can be either of those two different bits, usually the one with higher energy. Adding Lemma 2, we can write $\bigotimes$ as

$$A \bigotimes B = \begin{cases} A & \text{A} = \text{B}\\ A \text{ or } B & \text{A} \neq \text{B}.\end{cases}$$

■

As we set a precondition that $H(K_a)$ and $H(K_b)$ have differences, we only consider the differences. Analyzing at bit level, one bit transmitted by Alice is b and the corresponding bit transmitted by Bob is $\neg b$. This is because BFSK transmits binary bits. The result at Alice is Att $\bigotimes \neg b$ while the result at Bob is Att $\bigotimes b$. According to the workflow in Fig. 9, the adversary needs to satisfy the following two equations to avoid being detected

$$\text{Att} \bigotimes \neg b = b \tag{34}$$

$$\text{Att} \bigotimes b = \neg b. \tag{35}$$

If the adversary wants to satisfy (34), Att should be equal to b, which will make (35) false, vice versa. This means either Alice or Bob will detect the attack. So Lemma 3 is proved.

VII. Related Work

The signal manipulation attack is a well-known attack in wireless networks. There are three kinds of signal manipulation attacks [30]: the disruptive jamming attack, the manipulative jamming attack, and the channel manipulation attack. The disruptive jamming attack is the blocking attack we discussed in the introduction. The manipulative jamming attack injects a signal to manipulate the generated key while the channel manipulation attack controls the channel to infer the key.

There are five practical key manipulation attacks: three for manipulative jamming [13], [14], [31] and two for the channel manipulation [32], [33] by now.

In [14], an attack that aims at forcing both victims (devices) to generate the same shared key that is known to the attacker by injecting signals, is proposed. But it needs to have knowledge of the CSI of the channels between victims to the attacker. Another similar method [13] also wants to control the key but it requires waiting for an opportunity to inject the attack signals, which leads to a low key manipulation rate. The idea of [31] is relaying legitimate signals with higher power to overshadow the legitimate signal. Hence, there exists nontrivial probability that the receiver will extract they key from the information of the channel between the attacker and the receiver.

Instead of injecting signals to manipulate the key, Jin and Zeng [32] proposed a novel method to control the channel by controlling the movements of the intermediate objects or the devices themselves. Then, the CSI of the channel will change in a predictable way, which can be used to infer the shared key extracted from the controlled CSI. However, introducing an additional random variable that is unknown to the attacker [33] can stop this attack.

Our proposed attack has fewer constraints compared to [13], [14], [33]. Our attack has no need for prior knowledge of the CSI of the channels or wait for a specific moment to attack. Another advantage of our attack is that it can break the proposed security schemes that protects against the attacks in [14], [32], and [33].

VIII. Conclusion

In this paper, we introduced a novel active attack, called a session hijacking attack, against an existing PHY-UIR. Instead of attempting to reveal the key generated by legitimate participants, our attack forced participants to run the key generation scheme with the attacker by injecting attack signals.

We analyzed the impact of the session hijacking attack on two quantization schemes, RSS and signal phase, used in PHY-UIR. Then, we studied the effectiveness of our attack on these two schemes by simulation. Results showed that the power ratio between the attack signal and the legitimate signal at Alice (Bob) is the crucial factor to success or failure of the RSS quantization attack, while for the signal phase quantization attack the key factor is the time offset. Experiments were also conducted to validate the effectiveness of our attack and the findings of the simulation.

In the end, we proposed PHY-UIR$^+$ to detect our session hijacking attack by adding an extra step for both devices exchanging information about the shared keys at the same time. Experiments were shown that PHY-UIR$^+$ can detect the proposed session hijacking attack effectively and efficiently. Finally, we analyzed the security of this method and prove the resistance to brute-force attacks, signal separation attacks, and enhanced session hijacking attacks.

References

[1] D. Abbasinezhad-Mood and M. Nikooghadam, "Efficient anonymous password-authenticated key exchange protocol to read isolated smart meters by utilization of extended Chebyshev chaotic maps," *IEEE Trans. Ind. Informat.*, vol. 14, no. 11, pp. 4815–4828, Nov. 2018.

[2] R. Arul, G. Raja, A. K. Bashir, J. Chaudry, and A. Ali, "A console GRID leveraged authentication and key agreement mechanism for LTE/SAE," *IEEE Trans. Ind. Informat.*, vol. 14, no. 6, pp. 2677–2689, Jun. 2018.

[3] M. Wang and Z. Yan, "Privacy-preserving authentication and key agreement protocols for D2D group communications," *IEEE Trans. Ind. Informat.*, vol. 14, no. 8, pp. 3637–3647, Aug. 2017.

[4] N. Saxena and S. Grijalva, "Dynamic secrets and secret keys based scheme for securing last mile smart grid wireless communication," *IEEE Trans. Ind. Informat.*, vol. 13, no. 3, pp. 1482–1491, Jun. 2017.

[5] Y. Qiu and M. Ma, "A mutual authentication and key establishment scheme for M2M communication in 6LoWPAN networks," *IEEE Trans. Ind. Informat.*, vol. 12, no. 6, pp. 2074–2085, Dec. 2016.

[6] S. Mathur, W. Trappe, N. Mandayam, C. Ye, and A. Reznik, "Radiotelepathy: extracting a secret key from an unauthenticated wireless channel," in *Proc. 14th ACM Int. Conf. Mobile Comput. Netw.*, 2008, pp. 128–139.

[7] C. Ye, S. Mathur, A. Reznik, Y. Shah, W. Trappe, and N. B. Mandayam, "Information-theoretically secret key generation for fading wireless channels," *IEEE Trans. Inf. Forensics Secur.*, vol. 5, no. 2, pp. 240–254, Jun. 2010.

[8] R. Ahlswede and I. Csiszár, "Common randomness in information theory and cryptography. I. secret sharing," *IEEE Trans. Inf. Theory*, vol. 39, no. 4, pp. 1121–1132, Jul. 1993.

[9] Y. Wu, A. Khisti, C. Xiao, G. Caire, K.-K. Wong, and X. Gao, "A survey of physical layer security techniques for 5G wireless networks and challenges ahead," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 4, pp. 679–695, Apr. 2018.

[10] J. Zhang, T. Q. Duong, A. Marshall, and R. Woods, "Key generation from wireless channels: A review," *IEEE Access*, vol. 4, pp. 614–626, 2016.

[11] F. Pan, Z. Pang, M. Luvisotto, M. Xiao, and H. Wen, "Physical-layer security for industrial wireless control systems: Basics and future directions," *IEEE Ind. Electron. Mag.*, vol. 12, no. 4, pp. 18–27, Dec. 2018.

[12] S. M. MirhoseiniNejad, A. Rahmanpour, and S. M. Razavizadeh, "Phase jamming attack: A practical attack on physical layer-based key derivation," in *Proc. 15th Int. ISC (Iranian Soc. Cryptol.) Conf. Inf. Secur. Cryptol.*, 2018, pp. 1–4.

[13] S. Eberz, M. Strohmeier, M. Wilhelm, and I. Martinovic, "A practical man-in-the-middle attack on signal-based key generation protocols," in *Proc. Eur. Symp. Res. Comput. Secur.*, 2012, pp. 235–252.

[14] R. Jin and K. Zeng, "Physical layer key agreement under signal injection attacks," in *Proc. IEEE Conf. Commun. Netw. Secur.*, 2015, pp. 254–262.

[15] V. Brik, S. Banerjee, M. Gruteser, and S. Oh, "Wireless device identification with radiometric signatures," in *Proc. 14th ACM Int. Conf. Mobile Comput. Netw.*, 2008, pp. 116–127.

[16] B. Danev and S. Capkun, "Transient-based identification of wireless sensor nodes," in *Proc. Int. Conf. Inf. Process. Sensor Netw.*, 2009, pp. 25–36.

[17] M. Demirbas and Y. Song, "An RSSI-based scheme for sybil attack detection in wireless sensor networks," in *Proc. Int. Symp. World Wireless, Mobile Multimedia Netw.*, 2006, pp. 564–570.

[18] N. Patwari and S. K. Kasera, "Robust location distinction using temporal link signatures," in *Proc. 13th Annu. ACM Int. Conf. Mobile Comput. Netw.*, 2007, pp. 111–122.

[19] B. Danev, H. Luecken, S. Capkun, and K. El Defrawy, "Attacks on physical-layer identification," in *Proc. 3rd ACM Conf. Wireless Netw. Secur.*, 2010, pp. 89–98.

[20] Y.-C. Tung, K. G. Shin, and K.-H. Kim, "Analog man-in-the-middle attack against link-based packet source identification," in *Proc. 17th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2016, pp. 331–340.

[21] Q. Hu and G. P. Hancke, "A session hijacking attack on physical layer key generation agreement," in *Proc. IEEE Int. Conf. Ind. Technol.*, 2017, pp. 1418–1423.

[22] K. Zeng, D. Wu, A. Chan, and P. Mohapatra, "Exploiting multiple-antenna diversity for shared secret key generation in wireless networks," in *Proc. IEEE INFOCOM*, 2010, pp. 1–9.

[23] H. Minn, V. K. Bhargava, and K. B. Letaief, "A robust timing and frequency synchronization for OFDM systems," *IEEE Trans. Wireless Commun.*, vol. 2, no. 4, pp. 822–839, Jul. 2003.

[24] A. F. Molisch *et al.*, "IEEE 802.15. 4a channel model-final report," *IEEE P802*, vol. 15, no. 4, pp. 0662-1–0662-40, 2004.

[25] Q. Wang, H. Su, K. Ren, and K. Kim, "Fast and scalable secret key generation exploiting channel phase randomness in wireless networks," in *Proc. IEEE INFOCOM*, 2011, pp. 1422–1430.

[26] K.-C. Hsu, K. C.-J. Lin, and H.-Y. Wei, "Full-duplex delay-and-forward relaying," in *Proc. 17th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2016, pp. 221–230.

[27] C.-D. Iskander and H.-T. Multisystems, "A MATLAB-based object-oriented approach to multipath fading channel simulation," Hi-Tek Multisystems, Quebec, QC, Canada, White Paper, vol. 21, 2008.

[28] A. A. Nasir, S. Durrani, and R. A. Kennedy, "Performance of coarse and fine timing synchronization in OFDM receivers," in *Proc. 2nd IEEE Int. Conf. Future Comput. Commun.*, 2010, vol. 2, pp. V2–412.

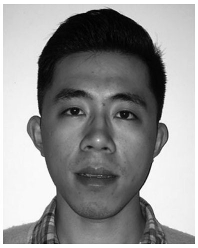
[29] W. Benenson, J. W. Harris, H. Stöcker, and H. Lutz, *Handbook of Physics*. Berlin, Germany: Springer, 2006.

[30] K. Zeng, "Physical layer key generation in wireless networks: challenges and opportunities," *IEEE Commun. Mag.*, vol. 53, no. 6, pp. 33–39, Jun. 2015.

[31] Y. Qiao, K. Srinivasan, and A. Arora, "Channel spoofer: Defeating channel variability and unpredictability," in *Proc. 13th Int. Conf. Emer. Netw. Experiments Technol.*, 2017, pp. 402–413.

[32] S. Jana, S. N. Premnath, M. Clark, S. K. Kasera, N. Patwari, and S. V. Krishnamurthy, "On the effectiveness of secret key extraction from wireless signal strength in real environments," in *Proc. 15th Ann. Int. Conf. Mobile Comput. Netw.*, 2009, pp. 321–332.

[33] R. Jin and K. Zeng, "Manipulative attack against physical layer key agreement and countermeasure," *IEEE Trans. Dependable Secure Comput.*, to be published, doi: 10.1109/TDSC.2019.2895325.



Qiao Hu received the B.S. degree in computer science and technology from Hunan University, Changsha, China, in 2011, the M.Sc. degree in information security from Wuhan University, Wuhan, China, in 2013, and the Ph.D. degree in information security from the Department of Computer Science, City University of Hong Kong, Hong Kong, in 2017.

He is currently an Assistant Professor with the College of Computer Science and Electronic Engineering, Hunan University. His research interests include RFID security and privacy, cloud computing, and wireless communication security.

Bianxia Du received the B.S. degree in computer science and technology from Wuhan University, Wuhan, China, in 2015 and the M.Sc. degree in computer applications from the Institute of Software, Chinese Academy of Sciences, Beijing, China, in 2018.

She is currently a Software Developer with the Baidu, Beijing, China. Her research interests include RFID security, trusted computing, and embedded security.

Konstantinos Markantonakis received the B.Sc. degree in computer science from Lancaster University, Lancaster, U.K., in 1995, the M.Sc. degree in information security from Royal Holloway, University of London, London, U.K., in 1996, the Ph.D. degree in information security from Royal Holloway, University of London, London, U.K., in 2000, and the M.B.A. degree in international management from School of Management, Royal Holloway, University of London, London, U.K., in 2005.

He is currently the Director of the Information Security Group Smart Card and IoT Security Centre (SCC). He has authored or coauthored more than 190 papers in international conferences and journals. His main research interests include smart card security and applications, embedded system security and trusted execution environments, cyber physical systems, and Internet of Things (IoT).

Gerhard P. Hancke (S'99–M'07–SM'11) received the B.Eng. and M.Eng. degrees in computer engineering from the University of Pretoria, Pretoria, South Africa, in 2002 and 2003, respectively, and the Ph.D. degree in computer science with the Security Group, Computer Laboratory, University of Cambridge, Cambridge, U.K., in 2008.

He is currently an Associate Professor with Department of Computer Science, the City University of Hong Kong, Hong Kong. His research interests include system security and reliable distributed sensing for the industrial Internet of Things.