Secure Device Trust Bootstrapping Against Collaborative Signal Modification Attacks

Xiaochan Xue[1], Shucheng Yu[2], Min Song[3]
Department of Electrical and Computer Engineering, Stevens Institute of Technology, Hoboken, NJ 07030, United States
Email: [1]xxue2@stevens.edu, [2]syu19@stevens.edu, [3]msong6@stevens.edu

***Abstract*—Bootstrapping security among wireless devices without prior-shared secrets is frequently demanded in emerging wireless and mobile applications. One promising approach for this problem is to utilize in-band physical-layer radio-frequency (RF) signals for authenticated key establishment because of the efficiency and high usability. However, existing in-band authenticated key agreement (AKA) protocols are mostly vulnerable to Man-in-the-Middle (MitM) attacks, which can be launched by modifying the transmitted wireless signals over the air. By annihilating legitimate signals and injecting malicious signals, signal modification attackers are able to completely control the communication channels and spoof victim wireless devices. State-of-the-art (SOTA) techniques addressing such attacks require additional auxiliary hardware or are limited to single attackers. This paper proposes a novel in-band security bootstrapping technique that can thwart colluding signal modification attackers. Different from SOTA solutions, our design is compatible with commodity devices without requiring additional hardware. We achieve this based on the internal randomness of each device that is unpredictable to attackers. Any modification to RF signals will be detected with high probabilities. Extensive security analysis and experimentation on the USRP platform demonstrate the effectiveness of our design under various attack strategies.**

***Index Terms*—Device pairing, in-band, MitM attack, signal cancellation attack**

I. Introduction

The proliferation of wireless and mobile devices in the era of Internet of Things (IoT) has introduced numerous promising applications wherein heterogeneous devices are securely grouped, temporarily or permanently, for a common task. For example, home devices like thermometers, humidity sensors, and smart locks can be paired to improve home energy efficiency and safety; wearable devices such as pacemakers, insulin pumps, and ECG sensors could be grouped to provide real-time health monitoring. In mission-critical scenarios, autonomous vehicles, unmanned aerial vehicles (UAV), and other wireless and mobile devices may form a field unit for tactical operations. In these application scenarios, wireless devices can be heterogeneous in the sense that they may be made by different vendors and equipped with different programming interfaces and various computation capabilities. Therefore, it is usually difficult to pre-install a shared secret to each device either at the manufacturing phase or by the user through a common programming interface.

To secure the communications between these devices, one important initialization step is to let them securely authenticate each other and establish a shared secret key to form a secure communication group. This process can be performed with or without the user's help and is usually known as *device trust bootstrapping* or *secure device pairing*. Conventionally, it can be achieved via an authenticated key agreement (AKA) mechanism with the support of public-key infrastructure (PKI). For example, the 3GPP 5G-AKA protocol [1] adopts this approach for initializing trust between the user (UE) and the base station (NB). However, this approach is not generally applicable to IoT applications wherein more flexible services (e.g., proximity-based services [2], [3]) are demanded through various devices-to-device (D2D) communication channels such as cellular, WiFi and near-field communications (NFC). The challenges exist in multiple folds including computing resource constraints on IoT devices, lack of uniform device registration processes (e.g., due to the complex vendor distribution channels), the extremely complex certificate management for massive amounts (tens of billions) of IoT devices, etc.

In the literature, there has been a body of research aiming at lightweight, secure, and highly usable device trust bootstrapping for heterogeneous wireless and mobile devices. Without relying on the public-key infrastructure (PKI), one important approach for device authentication is so-called *authentication through physical presence* (ATPP) [4]. ATPP relies on human verification of the physical presence of wireless devices in proximity. Device authentication is performed by verifying the common context information that can be confirmed by the user. AKA can be conveniently performed once device authentication is assured, e.g., through the Diffie-Hellman protocol or physical-layer secret key generation. Early research on ATPP mainly uses out-of-band (OOB) communication channels through audio, vibration, infrared, LED, or screen display interfaces [5], [6], [7], which are assumed relatively more difficult to tamper with. Despite the advantages of the OOB-based approach, it usually requires all participating devices to equip with common OOB interfaces and/or often involve non-trivial user interactions. An enhancement to the OOB-based approach is to utilize in-band communication channels for device authentication (e.g., Gollakota et al. [8] and Čapkun et al. [9]) with minimal involvement of OOB interactions. When OOB interfaces are needed, they are mainly on the authenticating devices such as smartphones but not on end devices to be authenticated. This is similar to many real-world (not necessarily secure) approaches adopted in commodity IoT systems. For example, to pair smart-home devices, a mobile App may require users to scan or manually enter IDs of IoT devices using smartphones as shown in Fig. 1. Existing in-band

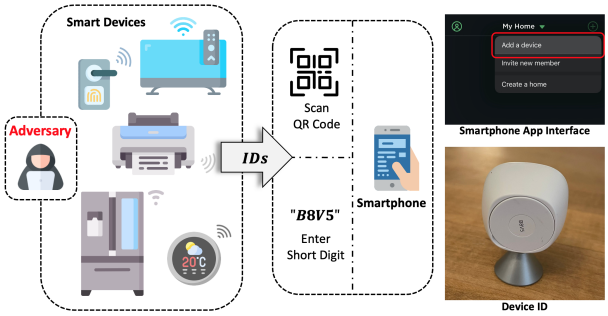


Figure 1: An example scenario of device pairing in the real world.

authentication (IBA) protocols in the literature mostly follow a similar methodology, e.g., by assuming the total number of devices is known via minimal initial OOB interactions.

Initial research on IBA [8] [9] [10] [11] mainly emphasizes on efficient protocol design under passive eavesdroppers and limited active man-in-the-middle (MitM) attackers. The former can only eavesdrop the communication channel while the latter has weak capabilities of RF signal manipulation (e.g., RF signal injection only). Strong active MitM attacks, such as over-the-air (OTA) RF signal cancellation, were conventionally assumed difficult in general settings because of the requirement on precise control of channel errors such as carrier frequency offset (CFO).

On the other hand, recent research [12][13][14] has also demonstrated the feasibility of OTA signal cancellation especially when wireless channels are more predictable, e.g., in open environments where multipath is not rich and devices are relatively static to each other. Considering strong MitM attacks, Pan et al. [15] utilized a re-configurable antenna to introduce channel randomness. In [16] Ghose et al. introduced an extra device called Helper to assist legitimate devices to detect attacker devices considering strong attackers that can perfectly cancel RF signals. In a subsequent work Ghose et al. [17] discovered an essential geolocation constraint of an RF signal cancellation attacker. Based on this observation, they utilized a third legitimate device to help the victim detect ongoing RF signal cancellation attacks. This technique saliently eliminates the need for an extra helping device. One limitation of this approach, however, is that it is aimed at single attackers and becomes ineffective when communication channels are controlled by multiple collaborating attackers as in the Dolev-Yao model. In practical applications, it is desired that a trust bootstrapping protocol is software-based without requiring any additional specialized hardware, efficient and scalable while secure against not only "lone wolves" but also collaborative attackers.

In this paper, we introduce a new trust bootstrapping protocol for wireless devices considering strong RF modification attacks (through signal cancellation and injection). Different from previous solutions, our design can defend against both single attackers and multiple colluding attackers without introducing any additional hardware. We achieve this by exploiting internal randomness of legitimate devices. Similar to well-adopted commodity applications and existing IBA techniques, our protocol only involves minimal one-time initial OOB interactions and enjoys high usability and scalability. A thorough security analysis shows that our solution is secure under various attack strategies. Our contributions are as follows:

- We develop a new security bootstrapping technique that can detect strong OTA RF signal modification attacks. Different from existing techniques, our solution is secure against multiple colluding attackers.
- Our design is software-based without requiring any additional hardware except for a user device such as a smartphone. Our protocol is highly usable because AKA is performed primarily in-band with minimum one-time initial OOB human interactions.
- We theoretically prove that the success probability of attacks is negligible even if only two legitimate devices but multiple malicious devices are presented.
- Extensive experiments on USRP devices validate the effectiveness of our design under single or multiple colluding attackers.

The rest of the paper is organized as follows. Section II describes system and adversary models. Section III expatiates the signal cancellation attack preliminaries. Our secure bootstrapping protocol is elaborated in Section IV. Section V presents security analysis. The experimental evaluations are illustrated in Section VI. Section VII discusses related work. Section VIII concludes this paper.

II. System Models and Assumptions

A. System Model

We consider a system with following wireless devices:

Legitimate Devices ($\boldsymbol{D}$): A set of K legitimate devices $\boldsymbol{D} = \{D_1, D_2, ..., D_K\}$ are newly introduced into the network. There is no prior shared secret key between any pair of these devices. To bootstrap initial trust (i.e., establishing pairwise authenticated secret keys), we assume all legitimate devices are synchronized, physically presented in the proximity of the user, and under the user's control. As widely adopted in real-world applications, we assume the public ID of each device is available, e.g., printed on the device (either in the form of digits or as a QR code) and can be easily accessed by the user.

Smartphone: We assume the user holds a smartphone with common computing capability and peripheral units including a camera. At the initial phase of the ATA protocol, the user can use the smartphone to conveniently collect each legitimate device's public ID, e.g., by scanning the QR code printed on the devices. It shall be noted that this is a one-time operation. A device's ID does not need to be scanned again even if there are other new devices introduced to the system in the future. Such OOB interaction has proven feasible in practice as shown in Fig. 1 and is a necessary step to establish the root of trust. We assume that the smartphone and other legitimate devices can communicate with each other through a common wireless protocol but they do not share any prior secrets.

B. Threat Model

We follow the Dolev-Yao model wherein the attacker can not only eavesdrop the communication but also modify the messages being exchanged. In particular, we consider one or multiple collaborative active adversaries that are able to annihilate OTA RF signals of the victim(s) and inject their own signals. We assume the attackers aim to spoof other legitimate devices to include themselves in the trusted communication group by replacing victims' messages with their own adversarial messages. This can be achieved through signal annihilation and injection. By this, we are assuming *strong attacks* as defined in [12] and adopted by existing research [16][17]. The attackers are also aware of the security bootstrapping protocol executed by legitimate devices. They can be presented in proximity to launch wireless attacks. But we do not consider physical attacks, e.g., by physically modifying the hardware or the printed ID of legitimate devices, nor do we consider Denial of Service (DoS) attackers. We assume the initial OOB interaction is secure and do not consider malware attacks against legitimate devices or the user's smartphone. We resort to orthogonal research on software security for malware attacks.

III. Technical Preliminaries

We first briefly introduce an ATPP device pairing protocol commonly used in existing works. Subsequently, we show feasible OTA signal modification attacks that have been considered and provide some insights into these attack mechanisms.

A. ATPP Device Pairing

In an ATPP device pairing protocol, each legitimate device D_k first broadcasts a beacon signal m_k that includes its unique ID, public key (for subsequent key establishment), and other public information needed by the protocol. Next, each device D_k concatenates all the beacons received, i.e., $s_k = m_1||m_2||\cdots||m_K$ assuming K legitimate devices in proximity, and broadcasts $h(s_k)$ as a short digest of s_k, where $h() : 1^* \rightarrow 1^l$ can be a cryptographic hash function. The received digests are then verified against the receiver's local knowledge of the context (e.g., the total number of devices in proximity) by checking if $h(s_k) = h(s_i)$ for each other receiver D_i. Alarm is raised when there is a mismatch. When MitM attackers are presented, however, the beacon signal of a legitimate device may be replaced by the attacker's beacon. This can be conveniently achieved via so-called *overshadowing attack* by which the attacker transmits signals with much higher power to overwrite the victim's signals over the air. The victim is thus muted even if it is in proximity. To thwart a such attack, Perkovic et al. [18] adopted the Manchester Coding (MC) to encode messages. With MC coding, each bit includes a pair of ON-OFF slots, i.e., bit 1 is encoded as (ON, OFF) and bit 0 as (OFF, ON).

One limitation with MC coding is that it doubles the message length because one bit takes two symbol slots after modulation. For efficiency, subsequent research [16][17] encodes messages using MC only for the short digest of all beacons, i.e., $h(s_k)$ at each device D_k. To further save communication time, the MC-encoded hash of all devices are synchronously transmitted over the air.

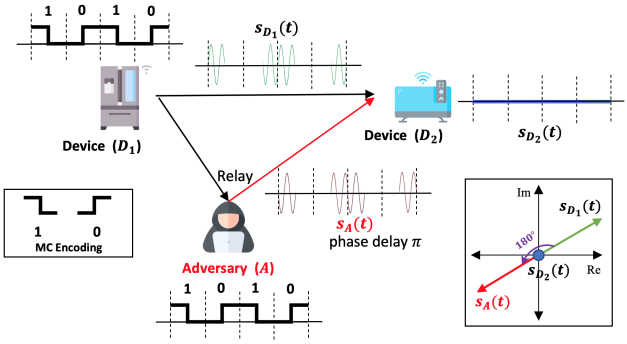


Figure 2: Relay-based signal cancellation attack.

B. Signal Modification Attack Against MC Coding

Despite the difficulties, recent research [12][13][14] has also demonstrated the feasibility of OTA signal cancellation attack. For illustration, we consider a MC-encoded message modulated with the ON-OFF Keying (OOK) modulation (though other modulation techniques such as BPSK and QPSK are also applicable). As shown in Fig. 2, a legitimate device D_1 is transmitting a modulated signal $s_{D_1}(t)$ to D_2 while the MitM attacker A wants to annihilate the signal. For this, attacker A sends another modulated signal $s_A(t)$ to D_2 simultaneously. Therefore, the resulting signal $s_{D_2}(t)$ received at D_2 is the superposition of both signals plus the noise $n(t)$:

$$s_{D_2}(t) = s_{D_1}(t) + s_A(t) + n(t) \tag{1}$$

Denoting the carrier frequency of device i as f_i, amplitude a_i, and the phase offset ϕ_{i-j} due to propagation from i to j, we know the received signals are as follows:

$$\begin{aligned} s_{D_1}(t) &= a_{D_1} \cdot cos(2\pi f_{D_1} \cdot t + \phi_{D_1-D_2}) \\ s_A(t) &= a_A \cdot cos(2\pi f_A \cdot t + \phi_{A-D_2}) \end{aligned} \tag{2}$$

To achieve perfect destructive interference, theoretically, the amplitude and frequency of both signals shall be identical ($a_{D_1} = a_A$, $f_{D_1} = f_A$), the noise be zero ($n(t) = 0$), and the phase offsets have a difference of π (i.e., $\phi_{D_1-D_2} = \phi_{A-D_2} \pm (2w+1)\pi, w \in \mathbb{W}$). However, this requires the attacker's perfect knowledge of both the sender's signal $s_{D_1}(t)$ and its channel status. To alleviate the challenges, Pöpper et al. [12] implemented the attack by letting the attacker located at a position that has a half-wavelength difference between paths $D_1 - A - D_2$ and $D_1 - D_2$. Instead of composing a new attack signal, attacker A simply relays the received signal from D_1 to D_2. Therefore, the signal relayed by A is the attenuated signal of $s_{D_1}(t)$ when received by D_2, i.e., $s_A(t) = \hat{a}_{D_1} \cdot cos(2\pi \hat{f}_{D_1} \cdot t + \phi_{D_1-A} + \phi_{A-D_2})$. In relatively static environments without rich multipath, the attacker may have a good estimation of the channel status including that of path $D_1 - D_2$. Equalization can be applied to the relayed signal $s_A(t)$. It shall be noted that the carrier frequency offset (CFO) caused by the mismatch between f_{D_1} and $\hat{f}_{D_1}$ incurs and increases linearly over time. Therefore, the attack is more

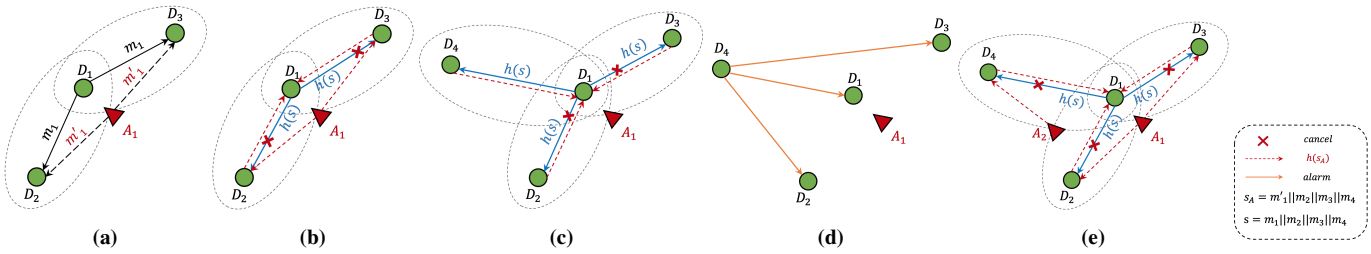


Figure 3: (a)-(b): three devices and one attacker. (a) D_1 transmits m_1 to D_2, D_3. Adversary A_1 replaces m_1 with m'_1. (b) D_1 broadcasts $h(s)$ for authentication. Adversary A_1 performs signal cancellation to annihilate $h(s)$ and injects $h(s_A)$. (c)-(d): four devices and one attacker. (c) $h(s)$ transmitted from D_1 to D_4 cannot be annihilated by attacker A_1. (d) D_4 raises an alarm to other devices. (e) four devices and two attackers: $h(s)$ transmitted from D_1 to D_4 is annihilated by the second attacker A_2.

effective for short message sequences than long sequences as reported in [12]. To modify a bit transmitted by the victim D_1, the attacker can annihilate the ON slot and inject a symbol for the corresponding OFF slot of that bit.

IV. Our Design

This section first discusses the vulnerability of the state-of-the-art defense technique under multiple collaborative attackers. Then we present our technique for thwarting signal cancellation attacks. Finally, we describe our ATA protocol for a group of devices based on the DH key agreement.

A. Vulnerability of Current Defense under Multiple Attacks

To defeat relay-based signal cancellation attacks, Ghose et al. [17] recently proposed a defense mechanism based on the constraint of the attacker's location. Specifically, an attacker A shall be located at a position with a distance difference of an odd multiple of half wavelength to assure a π phase shift to the original signal for any pair of TX-RX D_i and D_j:

$$d_{D_iA} + d_{AD_j} - d_{D_iD_j} = (2w+1)\frac{\lambda}{2},\ w \in \mathbb{W} \qquad (3)$$

Thus, the attacker's position must be on an ellipse with the two foci being D_i and D_j respectively. In the case of three legitimate devices, one attacker can be located at the intersection of two ellipses to perform the signal cancellation attack against the device being the focus of both ellipses. As shown in 3a, victim D_1 is broadcasting message m_1 to D_2 and D_3. The adversary A_1 can replace m_1 with its own message m'_1 through an overshadowing attack. Therefore, both D_2 and D_3 will obtain $s_A = m'_1||m_2||m_3$ while D_1 has $s = m_1||m_2||m_3$. During the authentication phase, D_1 transmits $h(s)$, while D_2 and D_3 transmit $h(s_A)$. As shown in Fig. 3b, since attacker A_1 is at the intersection of two ellipses, it is able to change the signal $h(s)$ transmitted by D_1 with its own signal $h(s_A)$ through the relay-based signal cancellation attack and injection attack as discussed in Section III. No alarm will be raised by D_2 and D_3 because they share the same view as the attacker A_1. To prevent D_1 from alarming, A_1 can modify $h(s_A)$ sent by D_2 or D_3 into $h(s)$. Please note that A_1 will launch the signal modification attack with a directional antenna and only the intended target will receive the modified message.

However, when there are more than three legitimate devices, a single attacker is not able to locate itself at the intersection of three ellipses. Consequently, it is not able to modify the message from/to the victim D_1 for all the three legitimate devices D_2, D_3, and D_4. As shown in Fig. 3c, the attacker A_1 cannot cancel the D_1's authentication message $h(s)$ on the channel h_{14}. Therefore, D_4 will detect $h(s) \neq h(s_A)$ and raise alarm to all other devices as shown in Fig. 3d. Based on this constraint of the attacker, Ghose et al. [17] is able to defend against single signal cancellation attackers. However, when there are multiple colluding attackers (which can also be thought of as one single attacker with multiple antennas), this defense becomes ineffective because the second attacker A_2 is able to modify the messages exchanged between D_1 and D_4 as shown in Fig. 3e. More generally, Ghose et al. [17] is ineffective when there are less than $2a+2$ legitimate devices but at least a attackers when all devices are within one-hop communication distance. However, it still remains a challenge to defend against signal cancellation attacks when the numbers of attackers and legitimate devices respectively are arbitrary.

B. Colluding Attack Detection with Internal Randomness

It shall be noted that collaborative signal cancellation represents a very strong attack in which attackers can mute any legitimate devices and arbitrarily modify messages being transmitted on any channel. In addition, the attackers have all the knowledge that legitimate devices have. Therefore, it is impossible to thwart such attacks without introducing additional advantages to legitimate devices. To this end, we utilize *internal randomness* at each legitimate device to gain the advantage over attackers. Intuitively, we let a legitimate device secretly flips some random bits before modulation and transmission. Without knowing the internal randomness of the device, the attacker would mistakenly "correct" some bit(s) that should have been flipped.

For illustration, we can consider the simplest scenario as shown in Fig. 4a wherein the attacker has complete control over channel h_{12} between D_1 and D_2 and can successfully launch a signal cancellation attack. To thwart the attack, D_1 randomly selects two bits, say the 5^{th} and 6^{th} (from left to right) bits of $h(s)$ (Fig. 4b), to flip, i.e., '01' (encoded as OFF-ON-ON-OFF) are flipped to '10' (ON-OFF-OFF-ON) during the actual transmission. To modify $h(s)$ to $h(s_A)$, the attacker A_1 needs to flip bits 3, 5, and 6 of $h(s)$ transmitted over the air (i.e., the symbols after bit flipping) as shown in Fig. 4b. This can be performed through relay-

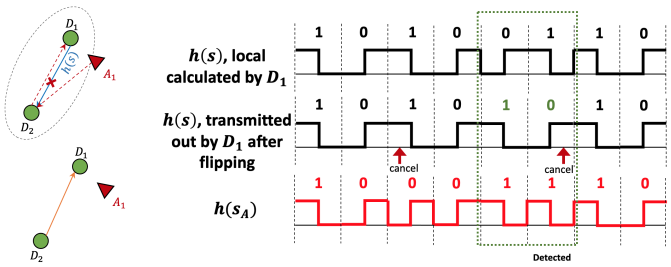


Figure 4: (a) A simple case of two legitimate devices with one attacker (D_1 as the victim). (b) D_1 randomly flipping the 5-th and 6-th bits of $h(s)$ during transmission while adversary injects $h(s_A)$.

based signal cancellation and injection attacks per Section III. After a successful cancellation attack by A_1, D_2 receives $h(s_A)$ as a bit sequence of '10001110', the 5^{th} and 6^{th} bits of which are '11' and different from the bit sequence transmitted by D_1. To detect the message modification attack, D_1 subsequently reveals the positions of the flipped bits to D_2 which will discover the signal cancellation attack because there should have been ON-ON symbols at bits 5 and 6 with a high probability if there were no attack but actually not. Intuitively, it is easy to understand that more random flipped bits will result in a higher successful detection probability. However, when multiple legitimate devices are presented, the probability of collision, i.e., multiple devices flip on the same bit position, will increase. This will reduce the attack detection probability because ON-ON symbols may still be detected due to the flipped bits transmitted by non-victim devices. We will present the detailed analysis of the detection probability under different system parameters in Section V.

C. Secure Device Trust Bootstrapping Protocol

With the basic idea of collaborative attack detection with internal randomness, we now investigate the detailed design of this technique and our secure trust bootstrapping protocol. In particular, as each legitimate device needs to reveal its internal randomness to assist others to detect attacks, attackers can modify the transmitted randomness to spoof legitimate devices given that the attackers are able to modify any messages over the air. To thwart such malicious modifications, one common approach is to send out a secure commitment before revealing the secret. Similarly, however, the commitment is also vulnerable to modification attack, through which an attacker can replace it with its own randomness-commitment pair that is consistent with the malicious digest message $h(s_A)$ that it injected. To defend against this attack, we pre-load each device D_k with a unique random number r_k, two counters c_k and $i_k = 1$, and a public identifier ID_k. All devices are installed with a cryptographic hash function $h() : 1^* \rightarrow 1^l$. It shall hold that

$$\underbrace{h(h(...h(}_{c_k \text{ times}} r_k))) = ID_k$$

where r_k, c_k, and i_k are stored in the memory of the device. ID_k, however, is made public, e.g., printed on the surface of the device in digits and/or QR code, and can be easily read by the user (or through smartphone scanning). We assume the device is physically safe so that r_k is not disclosed to the attacker by memory tampering. Our trust bootstrapping protocol can be carried out with the following steps as outlined in Fig. 5:

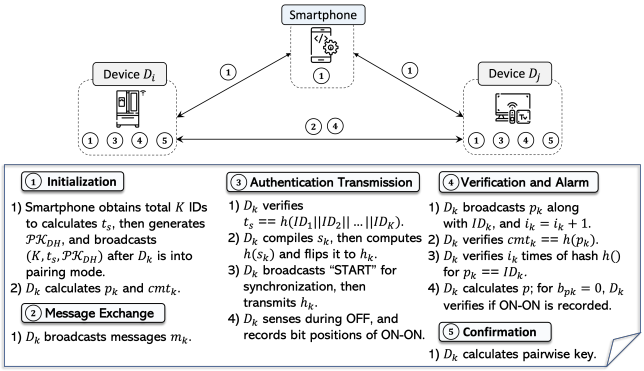


Figure 5: Secure pairing protocol steps for two legitimate devices.

1) **Initialization:** The user enters each legitimate device D_k's public ID_k into the smartphone, e.g., via QR code scanning. The smartphone calculates $t_s = h(ID_1||ID_2||...||ID_K)$, where K is the number of devices and $ID_1, ID_2, \cdots, ID_K$ are ordered by their binary values. Then the user lets the smartphone generate a tuple of public DH parameters $\mathcal{PK}_{DH} = (G, q, g)$, where G is a cyclic group of order q and g is a generator of G. Next, the user sets all legitimate devices to the pairing mode and lets the smartphone broadcast $(K, t_s, \mathcal{PK}_{DH})$. Each device D_k stores K, t_s and $\mathcal{PK}_{DH}$, and then computes the following:
$$p_k = \underbrace{h(h(...h(}_{c_k - i_k \text{ times}} r_k)))$$
p_k will be used as the internal randomness of D_k and is securely stored in the memory of the device. Then the device calculates a commitment $cmt_k = h(p_k)$ for this instance of the protocol execution.
2) **Message Exchange:** Each device D_k composes and broadcasts a beacon message $m_k = ID_k||z_k||cmt_k||i_k$ containing the ID of D_k, its one-time DH public key $z_k = g^{x_k}$, the commitment cmt_k and the counter i_k, where x_k is chosen from Z_q uniformly at random.
3) **Authentication Transmission:** On receiving all the K beacons, device D_k first verifies if $t_s = h(ID_1||ID_2||...||ID_K)$, where $ID_1, ..., ID_K$ are the IDs' extracted from the received beacon messages; if not, an alarm is raised. Otherwise, it compiles all received message as $s_k = m_1||m_2||...||m_K$ and computes the short digest of s_k, i.e., $h(s_k)$. Based on the internal randomness p_k, device D_k flips the i^{th} bit b_{hi} of $h(s_k)$ if the i^{th} bit $b_{p_k i}$ of p_k is 1, where $i \leq 2m \leq l$ and m is a system parameter. Denote the bit-flipped version of $h(s_k)$ as h_k. In a period τ, all legitimate devices with pairing mode broadcast a random message ended with a special START symbol sequence 'ON-ON-

ON-OFF-OFF-OFF' to synchronize each other and start pairing. This synchronization symbols sequence is invalid in MC encoding. Following synchronization, each device D_k synchronously transmits its MC-encoded authentication message h_k simultaneously. Meanwhile, each device senses the channel when it is in OFF slot. Bit position of any ON-ON slot pairs (two ON symbols for one bit) will be recorded.

4) **Verification and Alarm:** After the transmission of the authentication message, each device D_k broadcasts its internal randomness p_k along with ID_k; meanwhile, it increases its counter i_k by one. All the other devices verify p_k by checking if $cmt_k = h(p_k)$; if not, an alarm will be raised. Otherwise, each device will check if $\underbrace{h(h(...h(}_{i_k \text{ times}}p_k))) = ID_k$; if not, an alarm is raised. If all the devices pass the verification, D_k computes $p = p_1 \oplus p_2 \oplus \cdots \oplus p_K$. An alarm will be raised if $b_{pi} = 0$ and an ON-ON slot pair is recorded on the i^{th} bit position, where b_{pi} is the i^{th} bit of p.
5) **Confirmation:** Upon successful authentication, any two devices D_i and D_j calculate pairwise key $key_{i,j} = g^{x_i x_j}$.

V. Security Analysis

We now theoretically analyze the effectiveness of our design to defeat collaborative signal manipulation attacks. We consider a total of K legitimate devices and a colluding attackers, where $K < 2a + 2$. We consider only one victim, which is the easiest case for attackers but the hardest for defense. Per Section IV-A, the attackers are able to conduct signal modification attacks on all the channels from/to the victim. Therefore, the defense of Ghose et al. [17] is ineffective.

Successful attack: To understand the effectiveness of our design, we shall elaborate on how the attack is considered successful. It shall be easy to see that if there were no random bit flipping to the short digest $h(s_k)$ (see Step 3) of Section IV-C), an attacker with the help of other attackers is able to exclude the victim and replace this device with attacker itself by replaying the messages that the victim transmitted in a previous instance of the protocol execution. Therefore, we just need to focus on how the random bit flipping mechanism can help thwart the attacks. Taking the example of Fig. 4b for illustration, we see that the flipped bits 5 and 6 will be used to detect the signal modification attack. Bit 5 will not help because the flipped bit value 1 is intended by the attacker, and bit 6 is helpful. For convenience, we can call all bits selected for flipping (e.g., bits 5 and 6) as *detection bits*, and each bit transmitted over the air sharing the same value with the attacker's bit as *intended bit* (e.g., bits 1, 2, 4, 5, 7, 8). Please note, an intended bit is not necessarily a detection bit. In general, a successful attack happens when for each detection bit either of the following is true:

1) it is also an intended bit, or

2) it is not an intended bit, but there happens to be at least one other legitimate device selecting the same bit as the detection bit (considering all the other devices are spoofed by the attacker and will transmit the same bit string as the attackers).

In either case, a receiver will detect an ON-ON symbol pair on the detection bit. Please note when no ON-ON symbol pair is received for a detection bit, the receiver can alarm with a probability of 1 - $\frac{1}{2^K}$. This is because there is a probability of $\frac{1}{2^K}$ that all the legitimate devices happen to select this bit as the detection bit.

Another factor that affects the attack success probability is P_c, the probability that the attacker can successfully annihilate a given symbol (for a non-intended bit). For a non-detection non-intended bit, failure to annihilate will cause an alarm; for a detection non-intended bit, failure to annihilate will skip an alarm that shall have been raised. With the above analysis, we can obtain the attack detection probability as follows.

Theorem 1. *For a group of size K, there are $K-1$ verifiers. Our design is δ–secure against message modifications:*

$$\delta = (\frac{1}{8} + \frac{1}{2^{K+1}} + \frac{2^K + 2^{K-1} - 1}{2^{K+1}} \cdot P_c)^{2m} + \frac{1}{4^{l-2m}} \quad (4)$$

where δ is the probability that the adversary can successfully replace the authentication messages without being detected. m is the expected number of randomly flipped bits of $h(s)$. l is the output size of hash function $h()$. P_c is the probability of the attackers performing signal cancellation on a bit successfully. K is the total number of legitimate devices in the system.

Proof. For convenience, we define the following events for a given bit i:

- $iDct$: bit i is a detection bit;
- $iDctAll$: bit i for other $K-1$ devices are all detection bits;
- $iIntd$: bit i is an intended bit;
- $iFail$: signal cancellation attack failed for bit i.

Therefore, the probability that an alarm will not be raised at a given bit i as: $P_{na}(i) = P(\neg iDct)P(\neg iFail|\neg iDct) + P(iDct)P(iIntd \vee (\neg iIntd \wedge (iDctAll \vee iFail))|isDct)$

We know that $P(\neg iDct) = P(iDct) = \frac{1}{2}$ since detection bits are uniformly selected at random. $P(\neg iFail|\neg iDct) = P_c$. We also have the following: $P(iIntd \vee (\neg iIntd \wedge (iDctAll \vee iFail))|iDct) = P(iIntd)|iDct) + P(\neg iIntd \wedge (iDctAll \vee iFail)|iDct)$.

It is easy to see $P(\neg iIntd \wedge (iDctAll \vee iFail)|iDct) = P(\neg iIntd|iDct)P(iDctAll \vee iFail|iDct)$ because $iIntd$ is independent to $iDctAll$ and $iFail$. It is clear that $P(\neg iIntd|iDct) = P(\neg iIntd) = \frac{1}{2}$ because $iIntd$ and $iDct$ are independent. $P(iDctAll \vee iFail|iDct) = P(iDctAll \vee iFail)$ because both $iDctAll$ and $iFail$ are independent to $iDct$. We have $P(iDctAll \vee iFail) = P(iDctAll) + P(iFail) - P(iDctAll)P(iFail)$ because $iDctAll$ and $iFail$ are independent. $P(iDctAll) = \frac{1}{2^{K-1}}$. $P(iIntd)|iDct) = \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{4}$.

Therefore, we have $P_{na}(i) = \frac{1}{8} + \frac{1}{2^{K+1}} + \frac{2^K + 2^{K-1} - 1}{2^{K+1}} \cdot P_c$. For successful attacks, no alarm can be raised for all the first $2m$ bits from which devices select detection bits. Assume the hash function has l-bit output and $l \geq 2m$. Denote the probability that the first $2m$ bits will not raise alarm as

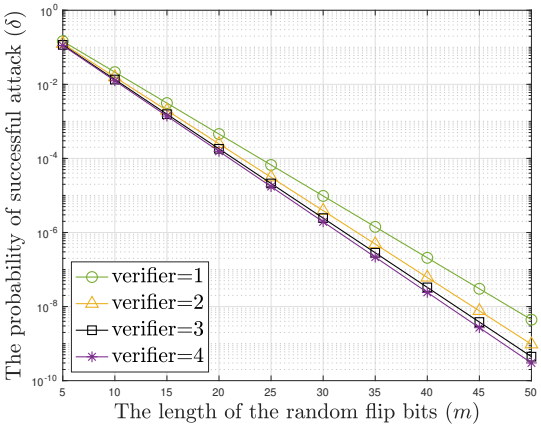


Figure 6: Attack success probability δ versus m with 1 - 4 verifiers.

$P_{2m} = P_{na}(i)^{2m}$. For the rest $l - 2m$ bits, the probability for no alarm is $P_{2m-l} = P(\neg iFail \vee \neg iIntd)^{l-2m} = (P(\neg iFail)P(\neg iIntd))^{l-2m} = \frac{1}{4^{l-2m}}$. The overall probability that there is no alarm raise, i.e., an attack is successful, is $\delta = P_{2m} + P_{2m-l} = (\frac{1}{8} + \frac{1}{2^{K+1}} + \frac{2^K + 2^{K-1} - 1}{2^{K+1}} \cdot P_c)^{2m} + \frac{1}{4^{l-2m}}$. □

The parameter P_c in **Theorem** 1 can be as large as 0.986 for one verifier (i.e., K=2) and 0.875 for two verifiers (K=3) according to our experiments in Section VI. Considering the cancellation probability $P_c = 0.9$ as the usual case, Fig. 6 shows δ as a function m. We consider the $l = 160$, which is the hash size after SHA-1. The success probability of attackers decreases with the increase of m. For 50 random flipped detection bits, the success probability δ is 4.92×10^{-9} for one verifier. Meanwhile, δ drops to 3.01×10^{-10} for four verifiers. With the increase of m, the success probability of attackers decreases log-linearly and will become negligible.

VI. Evaluation

In this section, we experimentally validate the effectiveness of our defense mechanism against signal or multiple collaborative signal modification attackers.

Experimental Setup: We use four sets of USRP N210 as four legitimate devices (i.e., K=4), three sets equipped with CBX daughter boards as three RXs (verifiers), one set equipped with SBX daughter board as one TX (victim). Two sets of USRP N200 equipped with CBX daughter boards act as two attackers (i.e., $a = 2$). This setting can simulate the worst case scenario for the victim because $2a + 2 > K$ and the attackers are able to launch the cancellation attack to every channel of the victim (cf. Section IV-A and Fig. 3). A directional antenna (LP0965 Log Periodic PCB Antenna, 850MHz to 6.5GHz) aims at the TX for listening function (for the relay attack), whereas either the directional antenna or the omni-directional antenna is used for transmitting function under different experiment setups. All antenna gains are high enough for flexible modification. We use GNURadio as the platform to implement our experiments. All devices were synchronized (with the clock of the same PC) and transmitting signals at 2.4GHz with 22MHz bandwidth. We transmit bits data to implement the short digest $h(s_k)$ in our design (e.g., 160 bits with SHA-1 encryption). Experiments were performed in an empty room to minimize Wi-Fi, Bluetooth, and multipath interference. Based on our experiment setup, the signal-to-noise ratio (SNR) is around 48dB for all devices. Our results show an attenuation of approximately 20dB can destroy the receiver's decoding capability. Therefore, we set the threshold for determining an ON slot to -41dB. The time period of one MC encoded message symbol sequence is 3ms. Each experiment was repeated over 10^6 times. We first conduct experiments to investigate the practical capability of signal cancellation attacks without any defense mechanism. Then we evaluate the effectiveness of our defense technique under collaborative signal modification attacks.

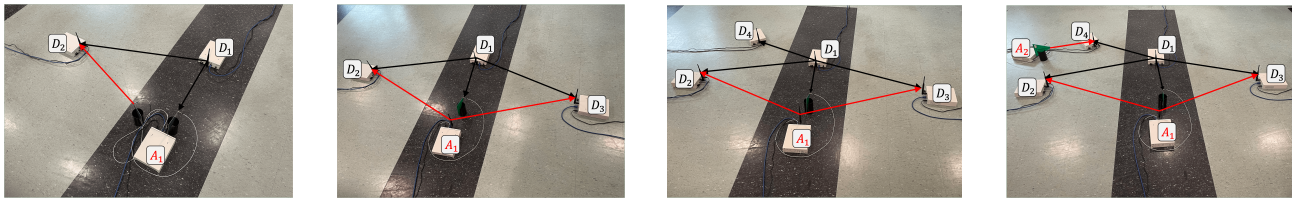
A. Signal Cancellation without Defense

Under all of our experiment setups, device D_1 acts as the transmitter and transmits MC-encoded 160 bits messages started with particular preamble symbols to other devices. Preambles are used to help devices estimate the channels and synchronize the carrier frequency offset (CFO). All adversarial devices are equipped with three capabilities: 1) estimate the respective channels, 2) modify the transmitting power online, and 3) coordinate with each other to achieve synchronization. We evaluate the signal cancellation probability P_c as following two aspects:

Signal cancellation from single attacker: We focus on one relay attacker annihilating signals at a total of three RX devices here. We first focus on one RX. The experimental setup is shown in Fig. 7a. D_2 receives the message from D_1 in the presence of an attacker A_1 who performs a relay cancellation attack. One USRP implements A_1 with two directional antennas for listening and transmitting functions. The front ends of two directional antennas satisfy the half-wavelength path difference that we discussed in Section IV-A. For two RX devices, we use the experimental setup shown in Fig. 7b. The transmitting antenna of A_1 is replaced with an omni-directional one to allow the simultaneous cancellation at devices D_2 and D_3. We make the vertical center of an omni-directional antenna and the front end of a directional antenna satisfy the same path difference for two ellipses (intersection of two ellipses). At last, we add the third RX device D_4 following the experimental setup Fig. 7c. A_1 is not able to locate the intersection formed by three ellipses. Therefore, A_1 cannot control the channel h_{12}, h_{13}, and h_{14} simultaneously.

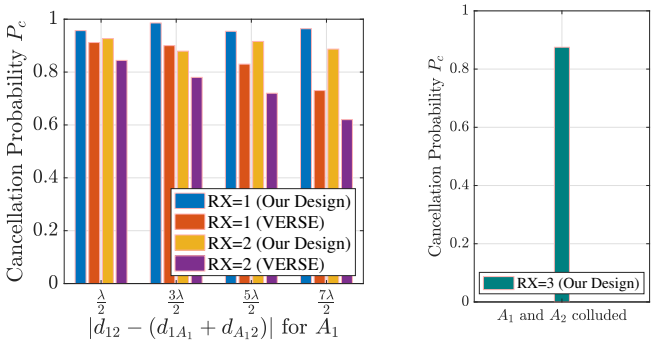
Signal cancellation from two colluding attackers: We then focus on two relay attackers annihilating signals at a total of three RX devices. The experiment setup is shown in Fig. 7d. The second attacker A_2 colludes with A_1 to compensate for the location constraint of A_1. A_2 controls the channel h_{14}.

Experimental results: Fig. 8 shows the cancellation probability as a function of path difference compared to existing results from Ghose et al. [17], called VERSE. Please note that results for the case RX=1 (i.e., the number of verifiers in addition to the victim) in Fig. 8a are for the setting of



(a) **(b)** **(c)** **(d)**

Figure 7: Signal cancellation experiment setups: (a) one TX to one RX with one attacker, (b) one TX to two RXs with one attacker, (c) one TX to three RXs with one attacker, and (d) one TX to three RXs with two attackers.



(a) **(b)**

Figure 8: (a) Signal cancellation probability as a function of the path difference with a single attacker. (b) Signal cancellation probability with two attackers.

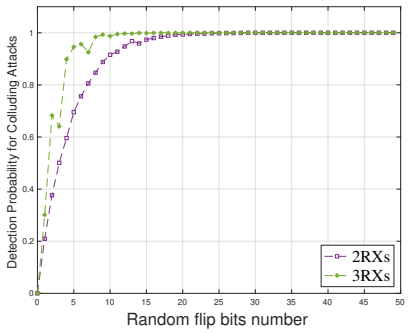


Figure 9: The detection probability for colluding attacks as a function of random flip number bits.

Fig. 7a, RX=2 for Fig. 7b; when RX=3 and there is only one attacker (Fig. 7c), the attack fails because $2a + 2 \not> K$, where $a = 1$ and $K = RX + 1 = 4$. Fig. 8a is for the setting of Fig. 7d, the first attacker A_1 is placed at different path differences according to Eq. (3) with $w = 0, 1, 2, 3$. The experimental results show that the cancellation performance for the case of Fig. 7b (RX=2) is worse than that for Fig. 7a (RX=1), because the attacker has to estimate channel h_{12} and h_{13} to cancel signals at both D_2 and D_3 simultaneously. The highest cancellation probability for Fig. 7a (RX=1) is $P_c = 0.986$ with path difference $\frac{3\lambda}{2}$. However, the cancellation probability with the same path difference for Fig. 7b (RX=2) is $P_c = 0.879$. Meanwhile, the attacker has the capability to modify the transmitting end antenna gain to a proper value to get high cancellation performance even at several wavelengths away. For example, P_c can reach 0.954 and 0.919 for cases of RX=1 and RX=2 respectively, with the one attacker located at $\frac{5\lambda}{2}$. For $\frac{7\lambda}{2}$, P_c can reach 0.964 and 0.887 for one RX and two RXs, respectively. When the third RX device D_4 is involved, the cancellation performance for A_1 drops significantly (to around 10^{-4}) for Fig. 7c. This means the signal cancellation attack is impossible for three RXs with only one attacker. However, the cancellation probability is increased to 0.875 with the help of the second attacker A_2 as in Fig. 7d. Due to attackers being capable of modifying the transmit power online, our cancellation probability is higher for each scenario as compared to VERSE. These experimental results validate that multiple collaborative attackers are able to successfully launch signal cancellation attacks against multiple devices wherein a single attacker is not able to. The results from these experiments also provide benchmarks P_c, an important parameter for security analysis (Section V).

B. Collaborative Signal Modification Attacks Detection

To evaluate the effectiveness of our defense mechanism, we implement signal modification attacks for 2 RX devices and 3 RX devices respectively but 2 colluding attackers A_1 and A_2. Take the case of 3 RX for example, we transmit flipped-bit authentication messages h_k from each legitimate device D_k to the other three RXs simultaneously. For example, D_1 is transmitting h_1 to RX D_k, where $k = 2, 3, 4$. D_2 is also transmitting h_2 to RX D_k, where $k = 1, 3, 4$. The same process is also for D_3 and D_4. Attackers A_1 and A_2 perform collaborative signal cancellation attacks and inject messages h_A during this process to all RXs simultaneously. We consider D_3 receives h_1, h_2, and h_4 from D_1, D_2, and D_4, respectively. D_3 senses the signal power during MC OFF slots and record any bit position of ON-ON slot pairs for the next step, attackers detection. D_3 calculates p by knowing p_k from each device. If the recorded ON-ON symbol pair is on the $b_{pi} = 0$ (the ith bit of p), D_3 reports the detection of the modification attack.

Fig. 9 shows the detection probability P_d as a function of randomly flipped-bit numbers. It can be seen the detection probability increases with an increasing number of flipped bits for both cases. For 2 RXs, 14 flipped bits will expose attackers with high probability ($P_d = 0.9591$). In the case of 3 RXs, RXs will report alarms with a high probability for 8 bits ($P_d = 0.9561$). After 11 flipped bits, two attackers almost have no chance to avoid the detection from three RXs. The experiment

results are as expected and validate our internal randomness protocol design for colluding attacks.

VII. Related Work

A plethora of trust bootstrapping techniques and wireless authenticated key agreements (AKA) have been proposed. To provide initial authentication of wireless devices, it is necessary to establish a root of trust. In contrast to traditional cryptographic certificates, authentication through physical presence (ATTP) is a simple yet flexible approach that relies on human verification of the common context information which cannot be altered or forged by attackers. With ATTP wireless devices presented within a physical context are considered legitimate. ATPP is usually performed through human-perceptible out-of-band (OOB) channels, such as vibration, visual interfaces, and audio channels. For example, ref. [19], [20], [21], [22], [23], [24] use mechanical vibration as a way for devices to confirm the shared context information. Visual interfaces (e.g., camera [6], LED blinking [25], and device screen [26]) are also used for the initial authentication of devices. Goodrich et al. [5] uses audio channels to share the public key via broadcasting sound among devices. A microphone is used in [27] to establish initial shared secret keys. These OOB channels are generally assumed relatively difficult to tamper with, especially when devices are in proximity, despite recent research on attacks against some OOB channels.

While OOB channels are convenient for ATPP, one practical issue is that they are not universally available on IoT devices especially when resourced-constrained devices are considered. For example, most IoT devices (e.g., sensors) lack output interfaces (e.g., a speaker, display), input interfaces (e.g., keypads), microphone or camera. Extensive reliance on OOB-based human verification also significantly increases the chance of human errors [28] in addition to the inconvenience incurred. Considering these constraints, one shift is to seek in-band exchange of authentication messages for initial trust establishment, i.e., by transmitting authentication messages over data communication channels. This approach is called "*in-band authentication*" (IBA) though it inevitably involves minimal OOB communications, especially at the initial phase to synchronize context information as the root of trust.

One challenge with IBA-based ATPP design is to assure security under powerful attacks. In particular, ATPP does not exclude the possible presence of malicious attackers in proximity, especially in public and hostile environments. In such scenarios, attackers not only eavesdrop wireless channels but may also launch MitM attacks. Existing research makes different assumptions to address the challenge. For example, the tamper-evident pairing (TEP) protocol [8] and the integrity codes (I-codes) [9] technique encode any message to binary bits and then implement the ON-OFF keying for verification of the integrity of authentication messages transmitted over RF channels. These techniques assume passive attackers or limited active MitM attacks such as signal injection or overshadowing attacks. However, recent research has also suggested the feasibility of a more powerful MitM attack - *signal cancellation attack* in which the attacker can annihilate signals being transmitted over the air. Despite of the difficulties of implementing such an attack because of the stringent requirement on channel estimation, it has also been shown feasible especially when wireless channels are relatively more predictable. For example, Pöpper et al. [12] shows the practicality of signal cancellation using carefully placed relay nodes and directional antennas. Moser et al. [14] qualitatively demonstrate that RF signal cancellation attacks are practically feasible and can be considered as an exceptional case of correlated jamming. Ghose et al. [17] further analyzes the practical geographic placement of OTA signal cancellation attackers and demonstrates the feasibility on real devices.

Various methods have been proposed to detect and thwart OTA RF signal cancellation attacks. For example, Move2Auth [29] and SFIRE [30] detect signal manipulation attacks based on RSS (Received Signal Strength) change patterns caused by user movements. These techniques can achieve authentication without OOB but require non-trivial human interactions to perform special gestures for authentication, which impair system usability. Moreover, the effectiveness of this approach is highly dependent on the quality of the user's physical movements (e.g., strict straight-line move toward given orientations). Another work by Ghose et al. [16] proposes a pairwise protocol by using an extra helping device to detect signal cancellation attacks considering perfect signal cancellation. The main limitation of this approach, however, is that the special helping device is not readily available on commodity devices. To address this issue, the authors subsequently proposed another technique VERSE [17] that utilizes a third legitimate communicating node as the helper. While this work saliently eliminates the need for specialized helping hardware, it only considers one single attacker and is vulnerable when multiple attackers are presented.

VIII. Conclusion

In this paper we address the problem of secure physical-layer trust bootstrapping among a group of wireless devices. Different from existing research, we consider the strong colluding RF signal cancellations attacks. We achieve this by letting each legitimate device randomly flip some bits of the message to be transmitted. Without knowing the internal randomness of legitimate devices, attackers will mistakenly annihilate some of the flipped bits, which will be detected by the former. Theoretical analysis shows that the success probability of colluding attackers decreases at a log-linear rate as the number of flipped bits increases. Experimental results using a USRP testbed validate the increased capability of colluding signal modification attackers as compared to single attackers, and that our design is able to detect such powerful attacks with a high probability under various attack strategies.

Acknowledgment

This work was supported in part by the National Science Foundation under grants ECCS-1923739 and CNS-1817438.

References

[1] 3GPP. Study on Authentication and Key Management for Applications (AKMA) phase 2. Technical Report (TR) 33.737, 3rd Generation Partnership Project (3GPP), 04 2022.

[2] Andre Kalamandeen, Adin Scannell, Eyal de Lara, Anmol Sheth, and Anthony LaMarca. Ensemble: Cooperative proximity-based authentication. In *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services*, MobiSys '10, page 331–344, New York, NY, USA, 2010. Association for Computing Machinery. ISBN 9781605589855. doi: 10.1145/1814433.1814466. URL https://doi.org/10.1145/1814433.1814466.

[3] Suhas Mathur, Robert Miller, Alexander Varshavsky, Wade Trappe, and Narayan Mandayam. Proximate: Proximity-based secure pairing using ambient wireless signals. In *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services*, MobiSys '11, page 211–224, New York, NY, USA, 2011. Association for Computing Machinery. ISBN 9781450306430. doi: 10.1145/1999995.2000016. URL https://doi.org/10.1145/1999995.2000016.

[4] Srdjan Čapkun, Mario Čagalj, Ghassan Karame, and Nils Ole Tippenhauer. Integrity regions: Authentication through presence in wireless networks. *IEEE Transactions on Mobile Computing*, 9(11):1608–1621, 2010. doi: 10.1109/TMC.2010.127.

[5] M.T. Goodrich, M. Sirivianos, J. Solis, G. Tsudik, and E. Uzun. Loud and clear: Human-verifiable authentication based on audio. In *26th IEEE International Conference on Distributed Computing Systems (ICDCS'06)*, pages 10–10, 2006. doi: 10.1109/ICDCS.2006.52.

[6] J.M. McCune, A. Perrig, and M.K. Reiter. Seeing-is-believing: using camera phones for human-verifiable authentication. In *2005 IEEE Symposium on Security and Privacy (S P'05)*, pages 110–124, 2005. doi: 10.1109/SP.2005.19.

[7] Volker Roth, Wolfgang Polak, Eleanor Rieffel, and Thea Turner. Simple and effective defense against evil twin access points. In *Proceedings of the First ACM Conference on Wireless Network Security*, WiSec '08, page 220–235, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781595938145. doi: 10.1145/1352533.1352569. URL https://doi.org/10.1145/1352533.1352569.

[8] Shyamnath Gollakota, Nabeel Ahmed, Nickolai Zeldovich, and Dina Katabi. Secure in-band wireless pairing. In *20th USENIX Security Symposium (USENIX Security 11)*, San Francisco, CA, August 2011. USENIX Association.

[9] Srdjan Čapkun, Mario Čagalj, Ramkumar Rengaswamy, Ilias Tsigkogiannis, Jean-Pierre Hubaux, and Mani Srivastava. Integrity codes: Message integrity protection and authentication over insecure channels. *IEEE Transactions on Dependable and Secure Computing*, 5(4):208–223, 2008. doi: 10.1109/TDSC.2008.11.

[10] Wenlong Shen, Yu Cheng, Bo Yin, Jin Du, and Xianghui Cao. Diffie-hellman in the air: A link layer approach for in-band wireless pairing. *IEEE Transactions on Vehicular Technology*, 70(11):11894–11907, 2021. doi: 10.1109/TVT.2021.3116619.

[11] Wenlong Shen, Bo Yin, Lu Liu, Xianghui Cao, Yu Cheng, Qing Li, and Wenjing Wang. Secure in-band bootstrapping for wireless personal area networks. *IEEE Internet of Things Journal*, 3(6):1385–1394, 2016. doi: 10.1109/JIOT.2016.2604221.

[12] Christina Pöpper, Nils Ole Tippenhauer, Boris Danev, and Srdjan Capkun. Investigation of signal and message manipulations on the wireless channel. In *Proceedings of the 16th European Conference on Research in Computer Security*, ESORICS'11, page 40–59, Berlin, Heidelberg, 2011. Springer-Verlag. ISBN 9783642238215.

[13] Yantian Hou, Ming Li, Ruchir Chauhan, Ryan M. Gerdes, and Kai Zeng. Message integrity protection over wireless channel by countering signal cancellation: Theory and practice. In *Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security*, ASIA CCS '15, page 261–272, New York, NY, USA, 2015. Association for Computing Machinery. ISBN 9781450332453. doi: 10.1145/2714576.2714617. URL https://doi.org/10.1145/2714576.2714617.

[14] Daniel Moser, Vincent Lenders, and Srdjan Capkun. Digital radio signal cancellation attacks: An experimental evaluation. In *Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks*, WiSec '19, page 23–33, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450367264. doi: 10.1145/3317549.3319720. URL https://doi.org/10.1145/3317549.3319720.

[15] Yanjun Pan, Yantian Hou, Ming Li, Ryan M. Gerdes, Kai Zeng, Md. A. Towfiq, and Bedri A. Cetiner. Message integrity protection over wireless channel: Countering signal cancellation via channel randomization. *IEEE Transactions on Dependable and Secure Computing*, 17(1):106–120, 2020. doi: 10.1109/TDSC.2017.2751600.

[16] Nirnimesh Ghose, Loukas Lazos, and Ming Li. HELP: Helper-Enabled In-Band device pairing resistant against signal cancellation. In *26th USENIX Security Symposium (USENIX Security 17)*, pages 433–450, Vancouver, BC, August 2017. USENIX Association. ISBN 978-1-931971-40-9.

[17] Nirnimesh Ghose, Loukas Lazos, and Ming Li. Secure device bootstrapping without secrets resistant to signal manipulation attacks. In *2018 IEEE Symposium on Security and Privacy (SP)*, pages 819–835, 2018. doi: 10.1109/SP.2018.00055.

[18] Toni Perkovic, Mario Cagalj, Toni Mastelic, Nitesh Saxena, and Dinko Begusic. Secure initialization of multiple constrained wireless devices for an unaided user. *IEEE Transactions on Mobile Computing*, 11(2): 337–351, 2012. doi: 10.1109/TMC.2011.35.

[19] Alexander De Luca, Emanuel von Zezschwitz, and Heinrich Hußmann. Vibrapass: Secure authentication based on shared lies. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, page 913–916, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605582467. doi: 10.1145/1518701.1518840. URL https://doi.org/10.1145/1518701.1518840.

[20] Younghyun Kim, Woo Suk Lee, Vijay Raghunathan, Niraj K. Jha, and Anand Raghunathan. Vibration-based secure side channel for medical devices. In *2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC)*, pages 1–6, 2015. doi: 10.1145/2744769.2744928.

[21] S. Abhishek Anand and Nitesh Saxena. Vibreaker: Securing vibrational pairing with deliberate acoustic noise. In *Proceedings of the 9th ACM Conference on Security amp; Privacy in Wireless and Mobile Networks*, WiSec '16, page 103–108, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450342704. doi: 10.1145/2939918.2939934. URL https://doi.org/10.1145/2939918.2939934.

[22] Kyuin Lee, Vijay Raghunathan, Anand Raghunathan, and Younghyun Kim. Syncvibe: Fast and secure device pairing through physical vibration on commodity smartphones. In *2018 IEEE 36th International Conference on Computer Design (ICCD)*, pages 234–241, 2018. doi: 10.1109/ICCD.2018.00043.

[23] S Abhishek Anand and Nitesh Saxena. Noisy vibrational pairing of iot devices. *IEEE Transactions on Dependable and Secure Computing*, 16 (3):530–545, 2019. doi: 10.1109/TDSC.2018.2873372.

[24] Sougata Sen and David Kotz. Vibering: Using vibrations from a smart ring as an out-of-band channel for sharing secret keys. *Pervasive and Mobile Computing*, 78:101505, 2021.

[25] Longfei Wu, Xiaojiang Du, Wei Wang, and Bin Lin. An out-of-band authentication scheme for internet of things using blockchain technology. In *2018 International Conference on Computing, Networking and Communications (ICNC)*, pages 769–773, 2018. doi: 10.1109/ICCNC.2018.8390280.

[26] Dirk Balfanz, Glenn Durfee, Rebecca E. Grinter, Diana K. Smetters, and Paul Stewart. Network-in-a-box: How to set up a secure wireless network in under a minute. In *13th USENIX Security Symposium (USENIX Security 04)*, San Diego, CA, August 2004. USENIX Association.

[27] Claudio Soriente, Gene Tsudik, and Ersin Uzun. Hapadep: human-assisted pure audio device pairing. In *International Conference on Information Security*, pages 385–400. Springer, 2008.

[28] Ronald Kainda, Ivan Flechais, and A. W. Roscoe. Usability and security of out-of-band channels in secure device pairing protocols. SOUPS '09, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605587363. doi: 10.1145/1572532.1572547. URL https://doi.org/10.1145/1572532.1572547.

[29] Jiansong Zhang, Zeyu Wang, Zhice Yang, and Qian Zhang. Proximity based iot device authentication. In *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, pages 1–9, 2017. doi: 10.1109/INFOCOM.2017.8057145.

[30] Nirnimesh Ghose, Loukas Lazos, and Ming Li. Sfire: Secret-free-in-band trust establishment for cots wireless devices. In *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*, pages 1529–1537, 2018. doi: 10.1109/INFOCOM.2018.8486417.