Secure Device Bootstrapping without Secrets Resistant to Signal Manipulation Attacks

Nirnimesh Ghose, Loukas Lazos, and Ming Li
Department of Electrical and Computer Engineering, University of Arizona, Tucson, AZ
Email: {nghose, llazos, lim}@email.arizona.edu

***Abstract*—In this paper, we address the fundamental problem of securely bootstrapping a group of wireless devices to a hub, when none of the devices share prior associations (secrets) with the hub or between them. This scenario aligns with the secure deployment of body area networks, IoT, medical devices, industrial automation sensors, autonomous vehicles, and others. We develop VERSE, a physical-layer group message integrity verification primitive that effectively detects advanced wireless signal manipulations that can be used to launch man-in-the-middle (MitM) attacks over wireless. Without using shared secrets to establish authenticated channels, such attacks are notoriously difficult to thwart and can undermine the authentication and key establishment processes. VERSE exploits the existence of multiple devices to verify the integrity of the messages exchanged within the group. We then use VERSE to build a bootstrapping protocol, which securely introduces new devices to the network.**

Compared to the state-of-the-art, VERSE achieves in-band message integrity verification during secure pairing using only the RF modality without relying on out-of-band channels or extensive human involvement. It guarantees security even when the adversary is capable of fully controlling the wireless channel by annihilating and injecting wireless signals. We study the limits of such advanced wireless attacks and prove that the introduction of multiple legitimate devices can be leveraged to increase the security of the pairing process. We validate our claims via theoretical analysis and extensive experimentations on the USRP platform. We further discuss various implementation aspects such as the effect of time synchronization between devices and the effects of multipath and interference. Note that the elimination of shared secrets, default passwords, and public key infrastructures effectively addresses the related key management challenges when these are considered at scale.

I. Introduction

It is predicted that approximately five billion IoT devices–wearable sensors, pacemakers, insulin pumps, blood pressure and heart monitors, smart occupancy sensors and locks, Internet-enabled appliances, sensors for autonomous vehicles–will be deployed by 2020 [1]. These devices become sensing instruments for our physical world, collecting a plethora of data that enhances the understanding of our surroundings and improves our interactions with the environment. The collected data is typically offloaded to a hub or a base station that provides connectivity to the Internet backbone and enables remote device access and control. For instance, the insulin dosage of a network-enabled implanted pump can be remotely adjusted according to the patient's vitals, without implant removal.

On many application scenarios, the devices need to be securely bootstrapped to the hub because they collect and communicate sensitive information. Often bootstrapping needs to occur in the presence of passive and active adversaries who may also attempt to pair with the hub or impersonate its functions. Establishing trust between two or more devices is one of the most fundamental problems in security that can be decomposed to achieving device authentication and key agreement. The first property is used to verify the device's identity (or legitimacy), whereas the second establishes a secure channel over a public medium. Conventional solutions include the use of default passwords, the preloading of secrets to the relevant parties [2] or the establishment of a public key infrastructure [3]. However, such solutions pose serious key management, scalability, and interoperability challenges. Often, manufacturers opt to preload devices with default keys that are easily leaked. The largest DDoS attack launched to date exploited default passwords preloaded to IoT devices such as IP cameras, digital video recorders, etc. to form the Mirai botnet and attack the DNS infrastructure [4]. Moreover, many IoT devices do not have advanced interfaces such as keyboards, screens, etc. to easily change default passwords.

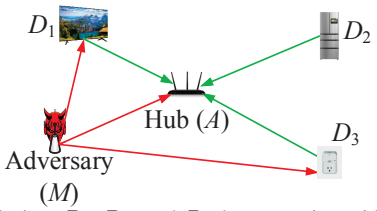


Fig. 1: Multiple devices D_1, D_2, and D_3 bootstrapping with the hub (A) in presence of an MitM adversary (M).

To address these challenges, recent works have proposed secure device pairing methods that do not rely on pre-shared secrets [5]–[9], [9]–[21]. Most rely on out-of-band (OOB) human verification to provide authentication and verify the protocol success. Human-dependent solutions scale poorly with the number of devices. Some in-band solutions have also appeared, but they almost unanimously derive security from the *infeasibility of advanced wireless signal manipulations, signal cancellation in particular*. To preserve the message integrity during the execution of a key agreement protocol, messages are encoded using Manchester-coded ON-OFF keying (MC ON-OFF), as shown in Fig. 2. In MC ON-OFF keying, a zero bit is represented with an OFF-ON signal sequence over two slots, whereas the one bit is represented by an ON-OFF signal. ON slots are realized by transmitting random symbols from the constellation plane, whereas OFF slots are realized by no transmission. A Man-in-the-Middle (MitM) adversary attempting to replace m with m' has to completely annihilate

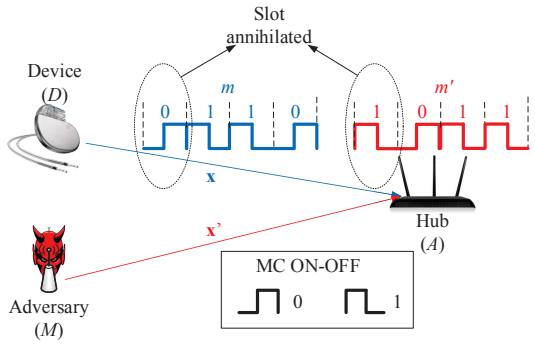


Fig. 2: D transmits an MC ON-OFF message x to A in the presence of M. To modify x to x', M has to annihilate ON slots of D's transmission.

the ON slots of m on those bit positions that the two messages differ. This is generally difficult to achieve under a rich scattering environment due to the unpredictability of the wireless channel between the legitimate parties. At the same time, device authentication is achieved via the verification of co-presence when the user interacts with the devices.

However, it was recently demonstrated that signal cancellation and injection is possible under more predictable channel conditions [22], [23]. For many applications, devices are paired in a relatively static indoor environment, where the channels between devices are predictable and slow fading. Under such conditions, an MitM attack over wireless becomes possible, and the adversary can inject his own messages to spoof a legitimate device or the hub. Note that the lack of an authenticated channel between the legitimate parties (due to the absence of prior trust) makes the legitimate transmissions and the injected ones indiscernible. Recently, the first protocol resistant to wireless MitM attacks utilizing signal cancellation was presented [6]. The protocol relied on the detection of signal cancellation attacks using a secondary device called the helper, that maintained an authenticated channel to the hub. The helper was placed in close proximity to the legitimate device to ensure the indistinguishability between the helper's and the legitimate device's transmissions. However, this approach does not scale with the number of devices. In a group setting, the helper would have to be manually moved to multiple locations and device pairing must occur sequentially.

In this work, we address the problem of securely bootstrapping multiple devices with a single entity such as a hub or a base station. Our goal is not to differentiate between legitimate and malicious devices. Such a proposition is infeasible in the absence of any prior trust and without the existence of out-of-band channels for verification, or some unique advantage of the legitimate devices (proximity, superior channel conditions, unique contextual information, etc.). Rather, we aim to guarantee protocol soundness in the absence of an adversary, and abort the bootstrapping process if any active protocol manipulation is detected. Moreover, we investigate if the presence of multiple legitimate devices can be leveraged to strengthen resistance to signal cancellation and therefore improve the security of the pairing process. We theoretically and experimentally characterize the limits of the adversary's capability based on geometric constraints and exploit those limits to construct a secure bootstrapping protocol for multiple devices. Our main contributions are four-fold:

- We develop a scalable PHY-layer group message integrity verification primitive called VERSE that achieves bootstrapping in-band (using only a common RF interface) and does not rely on pre-shared secrets. The key idea is to simultaneously verify the integrity of a transmitted message at multiple receivers, thus forcing the adversary to perform signal cancellation/injection at multiple locations simultaneously. This requirement dramatically degrades the success of MitM over wireless.
- We use VERSE to construct a secure in-band bootstrapping protocol for multiple devices based on the Diffie-Hellman (DH) key agreement. Our protocol securely pairs and then establishes pairwise keys with the hub. Such keys can then be used to establish group keys, if necessary.
- We analyze the security of VERSE and theoretically establish that a successful attack becomes infeasible if three or more verifiers are present when a single malicious device launches the attack. Moreover, the effort of a multi-device adversary must scale linearly with the group size.
- We carry out extensive USRP experiments to evaluate the effectiveness of our PHY-layer integrity verification against signal manipulations. First, we demonstrate the effectiveness of cancellation and injection attacks over a single channel. We then evaluate signal manipulations when multiple devices are used as receivers and/or transmitters and validate our theoretical findings. We then evaluate the adversary's ability to defeat VERSE.

Paper organization: The paper is organized as follows. We discuss related work in Section II. In Section III, we describe the system and adversary models. We present the VERSE primitive and the secure bootstrapping protocol for multiple devices in Section IV. We analyze the protocol's security in Section V. The experimental evaluation of MitM attacks over wireless and of the security of our protocol are detailed in Section VI. We conclude the paper in Section VII.

II. Related Work

Most prior methods for bootstrapping multiple devices that do not share prior secrets involve some degree of human intervention and OOB verification [24]–[27]. Perković *et al.* [24] proposed a group key establishment technique in which each participating device sequentially transmits its ID, public key, and a short random string. Each device compiles the XOR of the short strings of all the devices. For integrity verification, this short authenticated string is simultaneously transmitted using ON-OFF keying through a visual light channel or LEDs. The user performs integrity verification by pressing a button on each device individually. Li *et al.* proposed a DH-based group key exchange protocol, where integrity verification is derived by a human performing visual comparison of an ON-OFF keyed string [25]. Here, the string is the hash of all the transmitted messages. Nguyen *et al.* proposed a group bootstrapping protocol where each device computes and transmits

a long hash of its key or key primitive. This is followed by the transmission of the actual key or key primitive. Message integrity is verified by a human compiling the verification result from each device. Valkonen *et al.* proposed the use of a trusted node to verify the number of participating devices, preventing an adversary to pair in the group [28]. The user was responsible for compiling the verification result from individual nodes onto a single trusted device. Farb *et al.* proposed a group message transmission protocol over Bluetooth, which is initiated through a trusted device [29]. Similar to prior methods, the user validated the successful pairing of each device. Wong *et al.* proposed a multichannel verification scheme which required pre-shared secrets between the devices and the hub, in addition to a trusted device performing verification [30], [31].

There are many key-agreement protocols both OOB and in-band for a pair of devices. The OOB channel is used to protect the communication against an MitM attack because the OOB channel is assumed to be inaccessible to the attacker [9]–[21]. Therefore, verification is performed over a private and authenticated channel. However, OOB channels usually require non-trivial human support and advanced user interfaces. To reduce the human interaction there have been few past attempts to design in-band message integrity protection mechanisms, which assume that signal cancellation over the wireless channel is not possible [5], or occurs with bounded success [7], [23]. For example, the Tamper-Evident Pairing (TEP) protocol proposed by Gollakota *et al.* [8], and the integrity codes (I-codes) proposed by Čapkun *et al.* [5] both assumed the infeasibility of signal cancellation. Message authentication was achieved by assuming the presence of a legitimate device is known (a.k.a. authentication through presence).

However, the infeasibility of signal cancellation assumption does not always hold. Pöpper *et al.* first showed the feasibility of signal cancellation attacks using carefully placed relay nodes and directional antennas [22], Recently, Hou *et al.* [23] showed that success probability of signal cancellation attack in the one-to-one setting depends on the randomness of the legitimate channel. A typical indoor environment may not be sufficient because the devices are static and the channel is usually stable. In a recent work [6], Ghose *et al.* presented the first pairwise protocol called HELP that detects signal cancellation even if the adversary is assumed to have a perfect cancellation capability. The key idea of HELP is to place a helper device in close proximity to the legitimate pairing device so that their concurrent transmissions become indistinguishable. Therefore, if the adversary cancels part of the helper's signal, the hub can detect the cancellation, as the helper later reveals its signal to the hub via an authenticated channel. HELP does not scale well with multiple devices attempting to pair at once because the user will have to move the helper device manually to multiple locations. Moreover, the opposite authentication direction (hub-to-device) is not resistant to a cancellation attack, when the adversary can selectively cancel the signal at the device and not the helper (though this occurs with bounded probability due to the close proximity between the helper and the device).

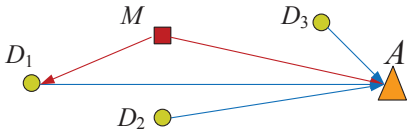


Fig. 3: System model depicting all entities.

Key advancements of VERSE relative to the state-of-the-art: Pairing methods can be extended to associate multiple devices. But there are two major issues with such extensions. First, the user effort becomes significant with OOB channel pairing, if it has to be repeated multiple times. Second, as it was shown by Mirzadeh *et al.* [32], suppose the success probability of an adversary pairing with the system to be p_S. With N pairing repetitions, the adversary's success probability of pairing one device becomes $1-(1-p_S)^N$ which approaches one with N. In our work, we leverage the existence of multiple devices to actually reduce the probability of a successful attack. Moreover, orthogonal to these works, our method requires the least user interaction (powering of devices and initialization of pairing from the hub). The message integrity verification is done in-band for all the participating devices without requiring any other interface (led lighting, microphone, speaker) other than the common RF interface. Also, most prior works do not address the possibility of MitM attacks, where the adversary can hijack the session of a legitimate entity by performing signal cancellation and injection. Compared to HELP, the only other work that addresses an MitM over wireless without pre-shared secrets, VERSE does not require a helper with an authenticated channel to the hub that also needs to be manually moved by the user. Moreover, VERSE improves security with a group of devices. Finally, the security of VERSE does not hinge on the close proximity of some devices, the randomness of the channel, nor the placement of the adversary outside a protected zone. Rather, it is derived from the fundamental constraints posed by the geometry and basic signal propagation properties.

III. Model and Assumptions

A. System Model

We consider the system model shown in Fig. 3. The system consists of the following entities:

Hub (A): The hub coordinates and verifies the bootstrapping process. It is assumed to be under user control.

Legitimate Devices ($\mathbf{D}$): We consider a set of legitimate devices $\mathbf{D} = \{D_1, D_2, \ldots, D_{N-1}\}$ that are newly introduced into the network. The devices attempt to pair with A, but do not share any prior secrets. They are assumed to be under user control. The devices and A are synchronized to a common slotted system with a bounded synchronization error ϵ. Synchronization is achieved with any known method such as [33], and it is already a necessary requirement for many standardized MAC protocols that follow a time slotted system [34]–[37].

Adversary (M): We consider an active adversary that aims at (a) pairing with A as a legitimate device and (b) spoofing a rogue hub that is joined by at least one legitimate device. We do not address DoS attacks such as jamming, simply aiming at preventing the pairing of legitimate devices without gaining

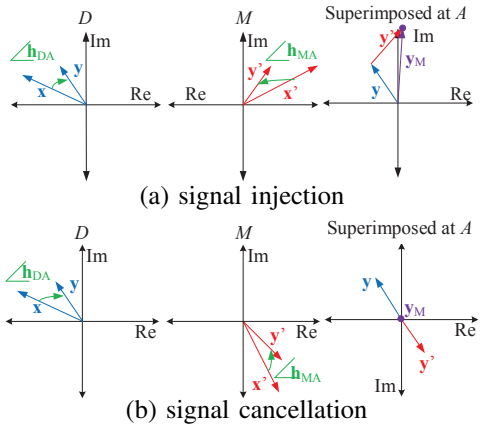


Fig. 4: (a) A signal injection attack and (b) a signal cancellation attack.

access to the system. All entities are located within the same collision domain and can overhear broadcast transmissions.

B. Threat Model

We consider an adversary that is aware of the protocol executed by the legitimate parties but does not have physical access to any of the devices. Because the bootstrapping process is initiated by the user, the adversary can only hijack an ongoing session. This can be achieved by launching an MitM attack and modifying the wireless transmissions during the bootstrapping session. We analyze the feasibility of the MitM attack when the adversary deploys a single device. We further discuss the feasibility and complexity of a multi-device MitM attack.

1) MitM attack by a single device: Let a legitimate device D transmit a message m to the hub A. To perform an MitM attack, the adversary has to replace m with m'. Let $\mathbf{x} = \{x(1), x(2), \ldots, x(k)\}$ denote the transmitted symbols modulating m and $\mathbf{y} = \{y(1), y(2), \ldots, y(k)\}$ the received symbols at A. Then,

$$\mathbf{y} = \mathbf{h}_{DA}\mathbf{x}, \tag{1}$$

where $\mathbf{h}_{DA} = \alpha_{DA} \cdot e^{j\phi_{DA}}$ is the impulse response of the D-A channel, α_{DA} is the channel attenuation factor, and ϕ_{DA} is the channel's phase shift. Here, we have assumed that the entire transmission of $\mathbf{x}$ completes within the channel's coherence time, so the channel remains constant. To modify $\mathbf{y}$, the adversary M must transmit $\mathbf{x}'$, modified by the M-A channel to $\mathbf{y}' = \mathbf{h}_{MA}\mathbf{x}'$ such that the superposition $\mathbf{y}_M = \mathbf{y} + \mathbf{y}'$ decodes to m'. In other words, M must compute

$$\mathbf{x}' = \frac{1}{\mathbf{h}_{MA}}(\mathbf{y}_M - \mathbf{h}_{DA}\mathbf{x}), \tag{2}$$

and transmit $\mathbf{x}'$ in a timely fashion such that $\mathbf{y}$ and $\mathbf{y}'$ are superimposed as shown in Fig. 4(a). According to equation (2), the computation of $\mathbf{x}'$ requires the knowledge of the signal $\mathbf{x}$ transmitted by D and of the channels $\mathbf{h}_{DA}$ and $\mathbf{h}_{MA}$. Moreover, the reception of $\mathbf{y}'$ must be synchronized with the reception of $\mathbf{y}$ such that $\mathbf{y}'$ arrives at A within an acceptable delay spread τ_A for correct symbol superposition [38]. Synchronization can be achieved using the preambles or the pilot symbols from the device; such methods are discussed in detail in [33]. The delay spread requirement imposes an important physical constraint on M's locations. The difference between the adversary's path, and the direct path must satisfy

$$d_{DM} + d_{MA} - d_{DA} \leq \tau_A \cdot c, \tag{3}$$

where d_{XY} denotes the distance between X and Y and c is the speed of light.

When the signal $\mathbf{x}$ is MC ON-OFF encoded, denoted by $[\mathbf{x}]$, modification of the received signal to $\mathbf{y}_M$ requires some ON slots of $[\mathbf{x}]$ to be annihilated, i.e., the amplitude of $\mathbf{y}_M$ must be below the signal detection threshold (typically 10s of dBms below zero) in some slots. Hence, the adversary must be capable of carrying out a signal cancellation attack. We primarily focus on the cancellation scenario, because it is more challenging to achieve than shifting the original constellation point closer to another point in the I-Q plane. The latter can be achieved by launching an overshadowing attack [39].

Practically, obtaining $\mathbf{x}$ in advance to compute $\mathbf{x}'$ is not possible. This is because D can transmit random symbols to implement an ON slot when ON-OFF keying is used. These symbols do not need to belong to a particular modulation mode such as BPSK, QPSK, etc. Alternatively, the adversary can avoid the requirement of knowing $\mathbf{x}$, by performing a relay attack. In this attack, the adversary's position is strategically selected such that the path difference between the direct path and the adversary's path satisfies:

$$d_{DM} + d_{MA} - d_{DA} = (2w+1)\frac{\lambda}{2}, \quad w = 0, 1, 2, \ldots \tag{4}$$

where λ denotes the wavelength. This guarantees that the inverse of $\mathbf{y}$ will be received at A when the incoming signal at M is compensated for the respective channel attenuation factors. Because the path difference is an odd multiple of $\lambda/2$, $\mathbf{y}$ and $\mathbf{y}'$ arrive at A with opposite phases, thus canceling each other ($\mathbf{y}_M = 0$). The signal superposition at A for a cancellation attack is shown in Fig. 4(b). To enable a fast and error-free relay operation, the adversary may be equipped with directional antennas, one for receiving the transmission of D and one for relaying $\mathbf{x}'$. (4) can be generalized for the adversary who is capable of modifying the phase (ϕ_{MA}) of the relayed signal in real time. From a geometric standpoint, modifying the phase of the incoming signal only changes the set of ellipses that yield cancellation. The new set of ellipses must satisfy,

$$d_{DM}+d_{MA}-d_{DA} = (2w+1)\frac{\lambda}{2}+\frac{\phi_{MA}}{\pi}, \quad w = 0, 1, 2, \ldots \tag{5}$$

We note that the phase calculations in (5) assume a strong Line-of-Sight (LoS) environment between all three entities. This is the best-case scenario for M, as it allows the calculation of a location from where cancellation via relaying becomes possible, without knowing $\mathbf{x}$ and by modeling $\mathbf{h}_{DA}$, since the latter cannot be directly measured. In the general case, $\mathbf{x}$ arrives at A via multiple paths which hardens channel modeling. In our model, we consider this best-case scenario for the attacker, where the channel is predictable with a strong LoS.

When M's placement satisfies (4) or (5) and assuming stable LoS channels, the symbols traveling over the relay path are copies of the symbols received via the LoS path but

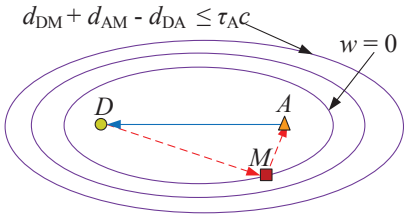


Fig. 5: To perform signal cancellation, the adversary is placed on an ellipse, centered at D and A that satisfies a path difference of $(2w+1)\lambda/2$ and does not violate the maximum delay spread τ_A.

shifted by $(2w+1)\pi$ and attenuated differently. Therefore, M does not need to know the transmitted symbols a priori. To compensate for the attenuation difference, M must only know the attenuation factors α_{DA}, α_{DM}, and α_{MA} in the impulse responses $\mathbf{h}_{DA}$, $\mathbf{h}_{DM}$, and $\mathbf{h}_{MA}$, respectively. Some of these channels ($\mathbf{h}_{DM}$, and $\mathbf{h}_{MA}$) can be measured, whereas the $\mathbf{h}_{DA}$ channel can be modeled after a path loss model.

We now examine the candidate set of M's locations that lead to successful cancellation via relaying. The adversary's location ℓ_M must satisfy the phase difference equation in (5) and the delay spread constraints in (3). For (4) or (5), candidate ℓ_M form a series of ellipses with D and the A placed at the two focal points. The set of such ellipses is shown in Fig. 5 and is computed by considering all odd integer values of w in (4) or (5). Finally, the delay spread constraint (3) upper bounds w.

2) MitM attack by multiple coordinated devices: When the adversary has multiple devices at his disposal, he can deploy them at multiple locations to perform simultaneous signal cancellation at more than one receivers. For instance, each adversarial device may target a single legitimate device. However, this attack requires online coordination among the different devices (timely channel sensing, time synchronization, power coordination, etc.) and the use of highly-directional transmissions to avoid unintended interference. For IoT scenarios, pairing devices are relatively close, which requires the use of very narrow beams. Antennas that can achieve such narrow beams are bulky with many antenna elements and therefore easily discernible in an IoT environment. Moreover, the attacker's cost increases linearly with the number of legitimate devices that are deployed. We primarily focus on the single device scenario and comment on the security and limitations of our scheme under a multi-device adversary.

IV. The Secure Bootstrapping Protocol

In this section, we present an in-band secure bootstrapping protocol for a group of devices. We first describe VERSE, a PHY-layer message integrity protection primitive that exploits multiple verifiers to detect signal manipulation attacks launched by an MitM adversary. We then use VERSE to construct an authenticated pairwise key establishment protocol between a group of devices and the hub, based on DH key agreement.

A. The VERSE Primitive

Consider a general group protocol in which multiple legitimate devices sequentially exchange a set of messages. Let s denote the *protocol transcript*. In VERSE, all legitimate devices

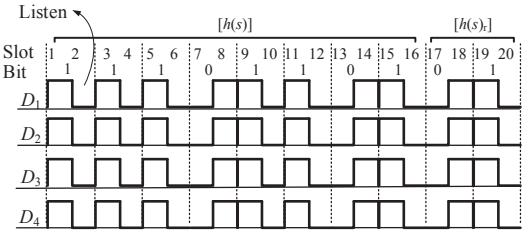


Fig. 6: D_1, D_2, D_3 and D_4 synchronously transmit $[h(s)]$. The devices sense the channel during the OFF slots.

operate as verifiers by recording the over-the-air messages. Each device compiles s and contributes in the integrity verification process by broadcasting a transcript digest $h(s)$, where $h(\cdot)$ is a non-cryptographic hash function. Specifically, all verifiers synchronously transmit the MC ON-OFF modulated message $[h(s) \mid\mid h(s)_r]$ where $h(s)_r$ is a repetition of the last r bits of $[h(s)]$. The synchronous transmission $[h(s) \mid\mid h(s)_r]$ is shown in Fig. 6. During the OFF slots of the $[h(s)]$ transmission, verifiers sense the wireless channel. If any device D_i compiled an $s' \neq s$, there will be at least one OFF slot for which D_i will sense an ON slot, as $h(s') \neq h(s)$ with overwhelming probability. Upon sensing this discrepancy, D_i will raise an alarm by sending only ON slots, essentially jamming the remainder of the $[h(s) \mid\mid h(s)_r]$ transmission, leading to further alarms being raised by the rest of the verifiers. The addition of $h(s)_r$ guarantees that an alarm will be raised, even if an integrity violation is detected at the last bit in $h(s)$.

Formally, the VERSE primitive involves the following steps:

1) **Compilation of the protocol transcript:** Each D_i broadcasts a message m_i using its default modulation mode. These messages are recorded by all D_is. Every D_i compiles the protocol transcript as $s = m_1||m_2||\ldots||m_N$.
2) **Device Synchronization:** A lead device (e.g., the hub) sends a delimiter to synchronize the clocks of all D_is. We set the delimiter to be an ON-ON-OFF-OFF-ON-ON sequence, which is not a valid MC-coded sequence.
3) **Transcript digest transmission:** Following synchronization, D_is transmit $[h(s) \mid\mid h(s)_r]$ synchronously using MC ON-OFF keying, where $h(\cdot)$ is a non-cryptographic uniform hash function and $h(s)_r$ are the last r bits of $h(s)$.
4) **Transcript verification:** While $[h(s) \mid\mid h(s)_r]$ is being transmitted, each D_i plays the role of a verifier. During the OFF slots of $[h(s) \mid\mid h(s)_r]$ D_is senses the wireless channel. If any OFF slot is sensed as ON by D_i, then D_i raises an alarm by transmitting ON slots for rest of the slots in $[h(s) \mid\mid h(s)_r]$. The $[h(s)]_r$ is appended to $[h(s)]$ to ensure there are sufficient slots to raise an alarm even if a mismatch is detected at the last ON-OFF bit of $[h(s)]$. The minimum value of r is two.

An example of VERSE for four devices is shown in Fig. 7. Initially, the devices exchange messages sequentially, creating a protocol transcript s. The transmission of m_1 is shown in Fig. 7(a). In the transcript verification step shown in Fig. 7(b), all devices synchronously broadcast $[h(s) \mid\mid h(s)_r]$ and use the OFF slots to verify the integrity of $h(s)$.

We provide a sketch of VERSE's security (a detailed analysis

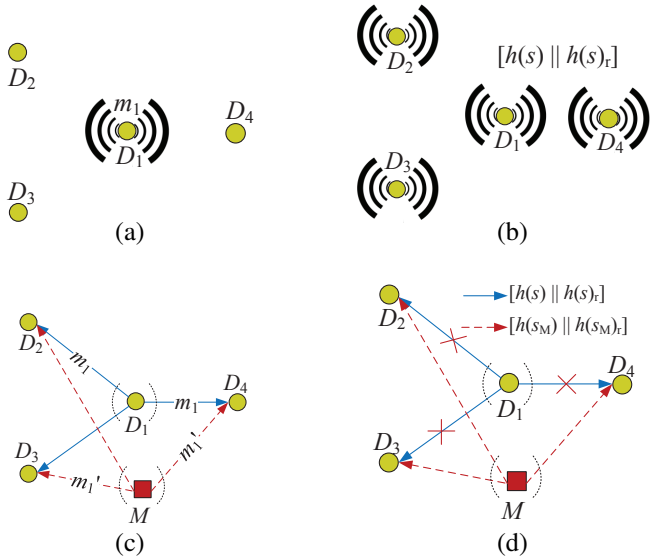


Fig. 7: (a) Transmission of m_1, (b) synchronous transmission of $[h(s) \mid\mid h(s)_r]$ during the integrity verification phase, (c) M replaces m_1 with m_1' by launching an overshadowing attack, and (d) M attempts a signal cancellation at D_1, D_2 and D_3 while D_1 transmits $[h(s) \mid\mid h(s)_r]$.

is presented in Section V). To successfully launch an MitM attack against VERSE, the adversary must ensure that no alarm is raised. Consider M modifying the protocol transcript from s to s_M by modifying m_i. In Fig. 7(c), we show M replacing m_1 with m_1'. Even if M launches an overshadowing attack against all devices and successfully replaces m_i, the device D_i that originated m_i compiles s. Because $s \neq s_M$, it follows with overwhelming probability that $[h(s) \mid\mid h(s)_r] \neq [h(s_M) \mid\mid h(s_M)_r]$, due to the collision resistance property of $h(\cdot)$. In fact, for a uniform hash function, the two hashes will differ in approximately half the bits. For the bits where $[h(s) \mid\mid h(s)_r] \neq [h(s_M) \mid\mid h(s_M)_r]$, D_i transmits (receives) when the rest of the devices are sensing (transmitting). To avoid the detection of s by the devices that compiled s_M, the adversary must perform signal cancellation from one TX to many RXs, which becomes increasingly difficult with the number of RXs. Similarly, to avoid detection of $[h(s_M) \mid\mid h(s_M)_r]$, at D_i, the adversary must perform signal cancellation from many TXs to one RX, which also becomes increasingly difficult with the number of simultaneous TXs.

B. Secure Bootstrapping using VERSE

To bootstrap a set of new devices with the hub, we execute a DH key exchange [40] for establishing pairwise keys over the public channel and use VERSE to protect the integrity of the protocol execution. The bootstrapping protocol consists of the following steps, which are also outlined in Fig. 9.

1) **Initialization:** A total of $N-1$ legitimate devices D_1, $D_2, \ldots, D_{N-1}$ participate in the group. The protocol is initialized when the user sets the hub (A) to pairing mode and loads the total number of devices N (including A) to A. For a period τ (e.g., two mins), the hub broadcasts a random MC ON-OFF sequence that ends in delimiter ON-ON-OFF-OFF-ON-ON. During that period, the user turns on each D_i to set it to pairing mode, and all D_i's synchronize to the MC ON-OFF sequence. Initialization terminates with the delimiter, allowing each device to note the beginning of the DH message exchange phase. Figure 8 shows the initialization step for four legitimate devices.

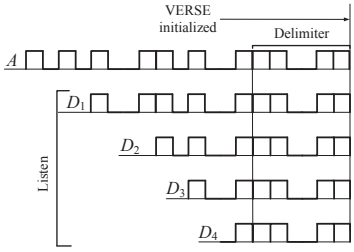


Fig. 8: Protocol initialization. The hub broadcasts an MC ON-OFF sequence during device activation. This sequence terminates with a known delimiter.

2) **DH message exchange:** All devices use public DH parameters $(\mathbb{G}, q, g)$, where $\mathbb{G}$ is a cyclic group of order q and g is a generator of G. Each D_i broadcasts a message $m_i = ID_i||z_i$ containing ID of D_i and the DH primitive $z_i = g^{X_i}$, where X_i is chosen from $\mathbb{Z}_q$ uniformly at random. The hub also broadcasts $m_A = ID_A||z_A$.
3) **Integrity Verification:** The integrity verification phase is initiated by the transmission of the delimiter by the hub, which serves as a SYNC message for all D_is. The D_is use VERSE to verify the integrity of the protocol transcript $s = m_1||m_2||\ldots||m_{N-1}||m_A$. The hub records the total number of public DH primitives N' exchanged during the protocol execution. The hub verifies that $N \stackrel{?}{=} N'$ to ensure that the correct number of devices participated in the protocol. If verifications is passed, D_is and A participate in VERSE by transmitting $[h(s) \mid\mid h(s)_r]$. Otherwise, D_is and A raises an alarm by transmitting all ON slots in the remaining of the sequence. The devices stay in pairing mode for a period $\tau' > \tau$ even if the integrity verification is completed. This is to ensure that they paired with the legitimate hub and no other pairing operation takes place. If a second MC ON-OFF sequence is overheard by a device D_i, the device raises an alarm.
4) **Confirmation:** Upon successful verification, each device calculates a pairwise key $k_{D_i,A} = g^{X_i \cdot X_A}$. Moreover, A displays a "SUCCESS" message. Else, A displays "FAILURE" and broadcasts a "RESTART" message.

We emphasize that the message integrity verification can be integrated with any group association protocol, such as the group Diffie-Hellman key exchange [41]. For this work, we establish pairwise keys with A. Once pairwise keys are established, A can securely distribute a group key to each device.

V. Security Analysis

We first analyze the security of VERSE by demonstrating the infeasibility of signal cancellation when multiple verifiers are used to verify the integrity of the protocol digest. We then evaluate the security of the DH-based protocol presented in Section IV-B.

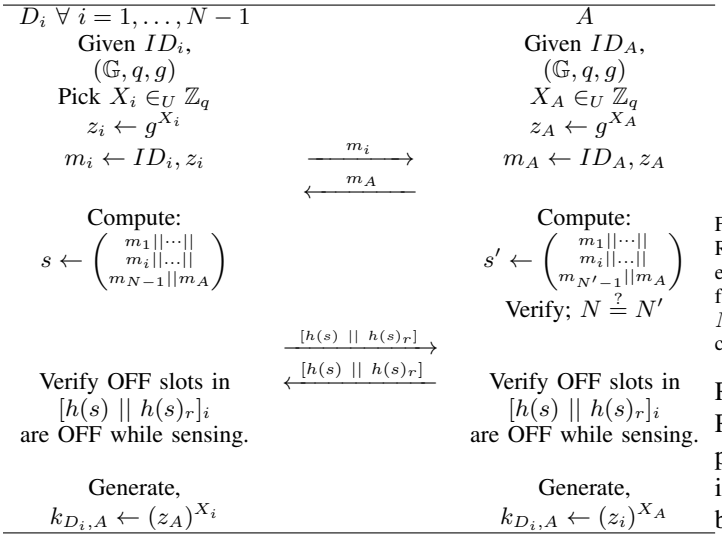


Fig. 9: Diffie-Hellman key agreement using VERSE after the initialization step.

A. Signal Cancellation from One TX to Multiple RXs

In this section, we analyze the signal cancellation attack for the adversary introduced in Section III. We consider the transmission of an MC ON-OFF sequence from one TX to multiple RXs and show that when at least three RXs act as verifiers, signal cancellation becomes infeasible.

Consider the scenario of Fig. 10(a), where a transmitter TX broadcasts an MC ON-OFF coded message m_1, which is received by RX_1, RX_2, and RX_3. Let $\mathbf{x}$ denote the symbols of the transmitted message, and $\mathbf{y}_1$, $\mathbf{y}_2$ and $\mathbf{y}_3$ denote the received symbols at RX_1, RX_2, and RX_3, respectively. The ON slots of m_1 are realized by a series of random symbols from the constellation plane, whereas the OFF slots are realized by no transmission. To cancel any ON slot at all three receivers, an adversary M must find a location ℓ_M such that it can simultaneously annihilate $\mathbf{y}_1$, $\mathbf{y}_2$ and $\mathbf{y}_3$, at the respective RXs. This is because $\mathbf{x}$ contains random selected symbols that do not allow the prediction of $\mathbf{y}_1$, $\mathbf{y}_2$ and $\mathbf{y}_3$. Therefore, M must perform a relay attack by being positioned at a location that cancels the received signal at each RX, independently of $\mathbf{x}$.

Let M transmit $\mathbf{x}'$ and RX_1, RX_2, and RX_3 receive $\mathbf{y}'_1$, $\mathbf{y}'_2$, and $\mathbf{y}'_3$. The cancellation attack is successful if $\mathbf{y}'_1 = -\mathbf{y}_1$, $\mathbf{y}'_2 = -\mathbf{y}_2$ and $\mathbf{y}'_3 = -\mathbf{y}_3$. That is, M's transmission arrives at each RX location with an inverse phase and the same amplitude as $\mathbf{y}_1$, $\mathbf{y}_2$ and $\mathbf{y}_3$. For each RX, M's location must satisfy the phase difference equation (4). The solution to (4) is an ellipse with TX and RX located at the focal points. For three RXs, ℓ_M must lie in the intersection of three ellipses, as shown in Fig. 10(a). However, the following proposition shows that no such location exists.

Proposition 1. *Three distinct ellipses sharing one focal point irrespective of the plane they lie in, do not have a common point of intersection.*

Proof. The proof is included in Appendix A. □

Based on Proposition 1, there is no location such that M can perform simultaneous cancellation of the TX's signal at three RXs with a single transmission. There are some degenerate RX arrangements that make cancellation from a single location possible. This is when two of the RXs are at the same location, in which case only the intersection of two ellipses needs to be considered. We consider such cases to be point-specific, which could be avoided by requesting distinct RX locations or including additional verifiers. Moreover, cancellation becomes possible if M is positioned at the common focal point, i.e., at the same location as the TX, which is detectable by the user.

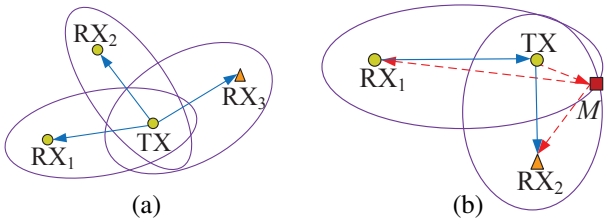


Fig. 10: (a) TX placed on the shared focus of three ellipses which have RX_1, RX_2 and RX_3 on the other foci respectively. An adversary positioned on one ellipse can cancel the TX signal at the RX positioned at the ellipse's other focal point. No common intersection point exist among three ellipses, and (b) M is placed on the intersection point between two ellipses to simultaneously cancel the signal at RX_1 and RX_2.

Extending Proposition 1, no common intersection point exists for $n > 3$ if such point cannot be found for $n = 3$. Furthermore, common intersection points between two ellipses exist as shown in Fig 10(b), and any point over a set of ellipses can be selected when $n = 1$ (see Section III-B). This sets the minimum requirement to thwart signal cancellation to three. For the proposed bootstrapping protocol, it is expected that at least three verifiers (e.g., the hub plus two other legitimate devices) will be available, as our work targets a group setting. If not, auxiliary devices can be added for verification purposes. We emphasize that there is no need for an authenticated channel between any auxiliary device and legitimate device.

Signal cancellation by a multi-device adversary: A multi-device adversary may be capable of canceling a transmission at more than two RXs. To scale this attack to more RXs, the adversary can deploy additional devices that lie on the intersection of the respective ellipses defined by TX-RX pairs. For instance, Fig. 11(a) shows the deployment of two devices to perform cancellation at RX_1, RX_2 and RX_3. The device at location A targets at RX_1 and RX_2, whereas from B to RX_3.

However, such a coordinated attack poses significant challenges. First, the transmission of the cancellation signal at location A contaminates the reception of the TX's signal at location B. The latter is necessary to compute the cancellation signal for RX_3. Second, the cancellation signal at locations A and B superimpose at RX_1 and RX_2, thus significantly degrading the cancellation capability. This multi-device attack can be successful only if the interference caused by multiple cancellers is minimal, which is only possible with close placement to the respective RXs when omnidirectional antennas are used. Such a close placement may be apparent to the user.

A higher-cost approach for performing cancellation to multiple RXs without causing unintended interference is to deploy devices with highly directional antennas. This scenario is depicted in Fig. 11(b). Three devices are deployed at locations A,

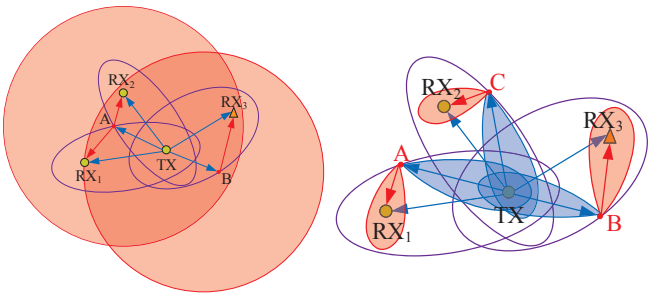


Fig. 11: (a) The adversary place two colluding devices one at A with omnidirectional transmission antenna and highly directional receiving antenna and other at B with highly directional antenna, the attack fails due to self-interference, and (b) the adversary places three colluding devices at A, B and C with highly directional antenna.

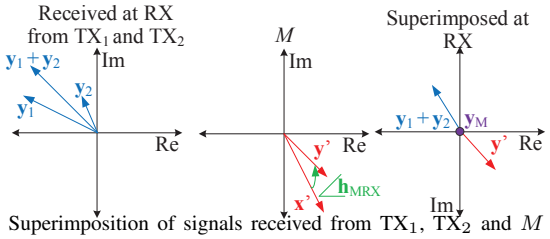


Fig. 12: Superimposition of signals received from TX_1, TX_2 and M at RX. M must be able to relay $-y_1 - y_2$ form a single location.

B, and C. Each device is equipped with two directional antennas. One is pointed to the TX to receive the transmitted signal and the other is pointed to the RX to perform cancellation. For a group of n verifiers, $2n$ directional antennas are needed. For a typical device separation of 10-30 ft. with an adversary located at a distance of 60 ft. he is required to achieve 9°-26° beamwidth. Such narrow beamwidths can be created by an antenna array [42] or a parabolic antenna [43]. A 9° beamwidth or a 26° beamwidth antenna array requires approximately 30 antenna elements or 17 antenna elements, respectively.

Our scheme does not provide protection against a multi-device adversary that can perfectly cancel MC ON-OFF sequences with highly-directional non-interfering transmissions from devices located at ideal locations. For all practical purposes, such a potent adversary is in full control of multiple wireless channels and can erase/inject any message at will.

B. Signal Cancellation from Multiple TXs to One RX

We now consider the inverse scenario where an MC ON-OFF message m is synchronously transmitted by n TXs and is received at a single RX. For this scenario, we examine whether signal cancellation at the RX is possible. A key observation for this case is that although the n TXs convey the same ON-OFF message m, ON slots are realized using different and randomly selected symbols at each TX. Therefore, Let $\mathbf{x}_i = \{x_i(1), x_i(2), \ldots, x_i(k)\}$ denote the transmitted symbols from one TX_i modulating m and $\mathbf{y}_i = \{y_i(1), y_i(2), \ldots, y_i(k)\}$ the received symbols at RX. To cancel the incoming signal at RX, M has to transmit the inverse signal,

$$\mathbf{x}' = -\frac{\sum_{i=1}^{n} \mathbf{h}_{\mathrm{TX}_i\mathrm{RX}}\mathbf{x}_i}{\mathbf{h}_{M\mathrm{RX}}} = -\frac{\sum_{i=1}^{n} \mathbf{y}_i}{\mathbf{h}_{M\mathrm{RX}}}. \tag{6}$$

The superposition of $\sum_{i=1}^{n} \mathbf{y}_i$ and $\mathbf{y}'$ for two TXs is shown in Fig. 12. According to (6), the computation of $\mathbf{x}'$ requires

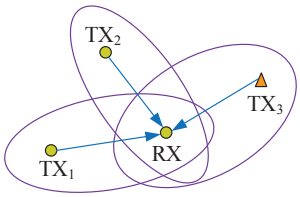


Fig. 13: RX placed on the shared focus of three ellipses which have TX_1, TX_2 and TX_3 on the other foci respectively. An adversary positioned on one ellipse can cancel the TX signal at the RX positioned at the ellipse's other focal point. No common intersection point exist among three ellipses.

the knowledge of the transmitted signals $\mathbf{x}_i$ from all the TXs and of the channels $\mathbf{h}_{\mathrm{TX}_i\mathrm{RX}}$ and $\mathbf{h}_{M\mathrm{RX}}$. However, the adversary does not have knowledge of the randomly transmitted symbols by each TX in advance. Moreover, it receives the superposition of the $\mathbf{x}_i$s, modified by the individual channels. For successful cancellation irrespective of the values of the $\mathbf{x}_i$s, the adversary must be positioned such that it cancels each individual $\mathbf{x}_i$.

For example, consider the scenario of Fig. 13, where TX_1, TX_2, and TX_3 transmit $\mathbf{x}_1$, $\mathbf{x}_2$, and $\mathbf{x}_3$ respectively and RX receives $\mathbf{y}$ as the superposition of $\mathbf{y}_1$, $\mathbf{y}_2$ and $\mathbf{y}_3$. As this superposition randomly changes with each transmitted symbol, to cancel any ON slot at RX, the adversary must find a location ℓ_M such that it can simultaneously annihilate $\mathbf{y}_1$, $\mathbf{y}_2$, and $\mathbf{y}_3$ by relaying the received signal.

Similarly to the case of one TX and multiple RXs, the adversary must attempt to cancel the symbols from each individual transmission, such that the aggregated symbol is canceled at RX. For each TX, M's location must satisfy the phase difference equation (4). The solution to each individual equation is an ellipse with the respective TX and RX located at the focal points of the ellipse. Therefore, ℓ_M must lie in the intersection of three ellipses, as shown in Fig. 13. These ellipses have RX as a common focal point, with TX_1, TX_2, and TX_3 being the other three focal points. However, Proposition 1 states that no such common intersection point exists. Hence, an adversary cannot find a valid location to perform cancellation from three TXs to one RX. Similarly to the case of one TX and multiple RXs, there are some degenerate TX arrangements that make cancellation from a single location possible. For the case of signal cancellation from multiple TXs to one RX the same complexity arguments as in the previous section. The best approach for the adversary is to cancel the signal of each TX individually using highly directional antennas to avoid unintended interference. The number of devices that need to be deployed grows linearly to the number of legitimate devices.

C. Security Analysis of the VERSE Primitive

The security of the VERSE primitive is derived from the difficulty in canceling a signal of one TX at multiple verifiers when the number of verifiers is greater than two and canceling the signal from more than two TXs at one verifier. We discuss a basic scenario with three verifiers for each transmission (four devices in total). In this example, M attempts to inject m_1' while D_1 transmits m_1 and pass the verification at the other three devices D_2, D_3, and D_4. The adversary must be capable of injecting m_1' at D_2, D_3, and D_4 simultaneously. This can be

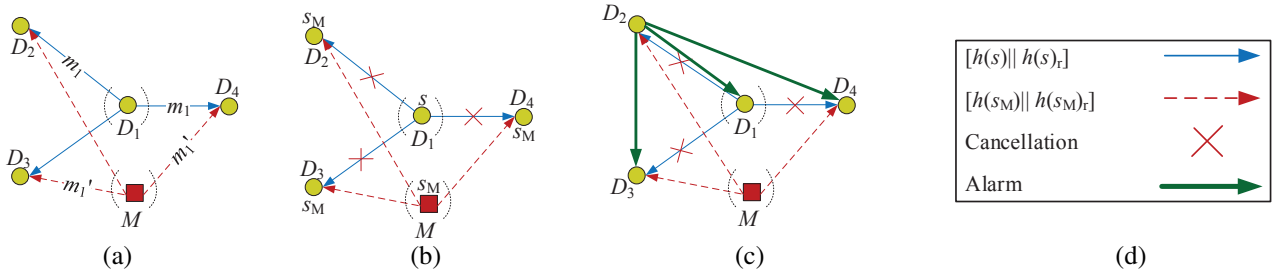


Fig. 14: (a) M replacing m_1 with m_1' during overshadowing attack, (b) M attempting to perform signal cancellation on D_1's transmission to D_2, D_3 and D_4 during the verification phase of the VERSE primitive (c) D_2 raises the alarm after detecting error during the verification phase of the VERSE primitive, and (d) legends for the figure.

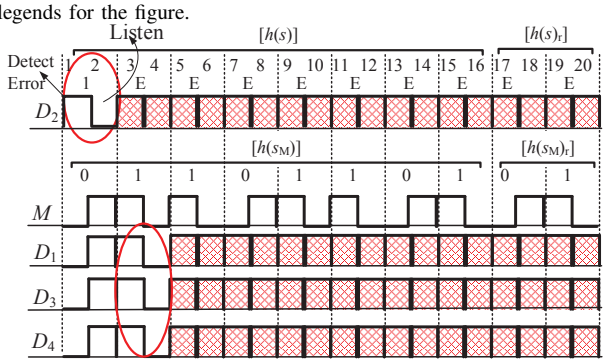


Fig. 15: M performing signal manipulation attack on D_2's transmission to flip bits where $[h(s)\ ||\ h(s)_r] \neq [h(s_M)\ ||\ h(s_M)_r]$ to pass the verification. D_2 followed by D_1, D_3 and D_4 transmits "Alarm" or error bits by sending all ON slots after detection of energy during its OFF slot.

achieved by launching an overshadowing attack [39], as shown in Fig. 14(a). Because m_1 is not ON-OFF modulated and a signal cancellation is not necessary, the adversary can inject a signal with large enough energy that causes demodulation to a desired constellation point. This is plausible for low order constellations (e.g., BPSK, QPSK), where the received constellation point needs to fall within a specific plane or quadrant. Note m_is are not protected with MC ON-OFF keying to improve the time efficiency of the bootstrapping process.

According to the VERSE primitive, D_2, D_3, and D_4 compile $s_M = m_1'||m_2||m_3||m_4$, whereas D_1 compiles $s = m_1||m_2||m_3||m_4$. During the integrity verification phase of VERSE, D_1 transmits $[h(s)\ ||\ h(s)_r]$, while D_2, D_3, and D_4 transmit $[h(s_M)\ ||\ h(s_M)_r]$. To prevent an alarm at D_2, D_3, and D_4, the adversary has to perform signal cancellation on D_1's transmission to replace $[h(s)\ ||\ h(s)_r]$ with $[h(s_M)\ ||\ h(s_M)_r]$ at all the three verifiers. This attack is shown in Fig. 14(b). However, in Section V-A, we showed that it is infeasible to perform such signal cancellation at more than three verifiers.

Since the adversary is unable to perform signal cancellation on D_1's signal, at least one of D_2, D_3, and D_4, will detect the error when $[h(s)\ ||\ h(s)_r] \neq [h(s_M)\ ||\ h(s_M)_r]$ and raise an alarm. In Fig. 14(c), we show D_2 raising an alarm during the verification phase. This alarm will be now heard by the rest of the devices because the adversary is not positioned to cancel the signal from D_2 to the remaining three devices. The sequential raising of an alarm by each of the devices is shown in Fig. 15. We note that even if the adversary is positioned such that it can achieve cancellation to a subset of devices, it cannot cancel the raised alarms as the number of TXs raising alarms increase because it is infeasible to perform signal cancellation from more than two TXs to one RX. There might be other attack vectors where the adversary chooses to overshadow a different combination of messages during the protocol execution phase. For instance, for the scenario of four devises, it could choose to inject m_1' only at D_3 and D_4. In this case, D_1 and D_2 compile s, whereas D_3, and D_4 compile s_M. Hence, to pass the verification the adversary has to perform signal cancellation on the transmissions from D_1 and D_2 to D_3 and D_4 and replace $[h(s)\ ||\ h(s)_r]$ with $[h(s_M)\ ||\ h(s_M)_r]$.

To guarantee the secure operation of VERSE under any possible attack vector we need to have *at least three verifiers for any direction*. This can be achieved by requiring at least four legitimate devices and the hub participate in the group (a total of five devices). Then, irrespective of the set of devices selected by M to perform the overshadowing attack, M will have to perform signal cancellation attack from at least one TXs to at least three RXs, or from at least three TXs to at least one RX. We have shown that neither of these attacks is feasible, due to the impossibility of finding a location to concurrently perform successful cancellation at multiple verifiers.

Even though we have that cancellation attacks to multiple RXs or from multiple TXs are theoretically infeasible, in practice, such attacks could have some limited success probability. This is because the adversary does not have to completely annihilate the incoming signal at a given verifier, but has to reduce it below the detection threshold for an ON slot. This threshold is typically larger than the receiver sensitivity, to account for ambient noise from other devices. Therefore, there could be some location for which M has a cancellation probability p_n for each slot. To guarantee the security of VERSE, we use the length of the hash value used for integrity verification to drive the overall success probability for M to negligible values. This is formalized in the following proposition, where we show that the probability of M successfully modifying any (or multiple) message(s) without being detected by all the legitimate devices is bounded by δ.

Proposition 2. *For a group of size N, the VERSE is δ–secure against message modifications with*

$$\delta \quad \leq \quad (p_H + (1 - p_H)p_n)^{\ell}\,, \tag{7}$$

where δ is the probability that M can replace any m_i sent by D_i with m_i' at any subset of remaining devices without being detected at every $D_{i'} \in \mathcal{D}$ (where $\mathcal{D}$ is the set of all legitimate devices), p_H is the probability for a bit of $h(s)$ to equal a bit of $h(s_M)$, and p_n is the probability of successfully flipping one

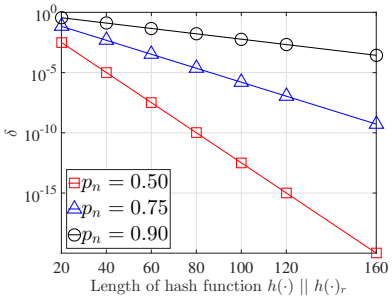


Fig. 16: The probability that M can replace m_i with m_i' without being detected at $D_{i'}$ $\forall\ i' \neq i$.

bit in $[\cdot]$ *during transmissions from* n *TXs to one RX or from one TX to* n *RXs where* $n = \lceil N/2 \rceil$*, and* ℓ *is the length of the hash function* $h(\cdot) \mid\mid h(\cdot)_r$*. We show that* δ *is a negligible function of* ℓ*.*

Proof. The proof is included in Appendix B. □

Note that the above proposition is general and applies to any group size $N > 1$. However, for different values of N, we have different concrete guarantees since p_n depends on the minimum number of devices n (number of transmitters for many-to-one, or receivers for one-to-many) that the adversary needs to launch a cancellation attack against, among all possible cases of group partitioning. For example, when $N = 2$, the minimum number of cancellation targets is 1; in general, for $N \geq 5$, $n = \lceil N/2 \rceil$. In addition, according to our experiments in Section VI, we show that for $n = 1, 2$, p_n can be as large as 0.9. However, p_n dramatically drops to a very small value when $n = 3$. Thus, the security guarantee of the VERSE primitive is stronger with an increasing n and also the group size N.

Figure 16 shows δ as a function of the hash length ℓ for various values of p_n when $p_H = 0.5$ (i.e., the bits of the $h(s)$ and $h(s_M)$ are random). As expected, a higher p_n yields higher δ values for the adversary. For instance, when $p_n = 0.9$ we have $\delta = 0.00027$ for $\ell = 160$. But when the cancellation probability is significantly low, for instance when $n = 3$, $p_n = 8.7 \times 10^{-5}$, we have $\delta = 6.9 \times 10^{-49}$ for $\ell = 160$. We note that this is an *online* attack that has to be performed while the pairing session is ongoing security. Similar standards are used for other existing pairing protocols [18]. Moreover, a $p_n = 0.9$ is difficult to achieve in the presence of multiple D_i's. δ is a negligible function of ℓ, the adversary's success probability can always be driven to any desired value by choosing a long enough ℓ.

D. Security of the Bootstrapping Protocol using VERSE

We now analyze the security of the bootstrapping protocol shown in Fig. 9 against MitM attacks, which can be reduced to the security of VERSE (Corollary 1). Basically, we need to show that the adversary can neither join the group as an additional device and pair with any existing legitimate device nor can the adversary carry out an MitM attack against any legitimate device(s) to pair itself with the hub A or any D_i.

Corollary 1. *The bootstrapping protocol protected by the VERSE primitive is* δ*–secure against active attacks with*

$$\delta \quad \leq \quad (p_H + (1 - p_H)p_n)^{\ell}. \tag{8}$$

Here, δ *is the probability that* M *can replace any DH public number* m_i *(sent by any device or* A*) with* m_i' *at any subset of remaining devices, without being detected at every device* $D_{i'} \in \mathcal{D}$ *(including the hub). Notations are defined in the same way as in Proposition 2.*

Proof. The only differences between our bootstrapping protocol and the VERSE primitive are: (a) the addition of an initialization phase, where the devices are synchronized and the group count is pre-loaded to A, and (b) the messages being exchanged are the DH public numbers. The message content does not affect the security because of hash function's collision-resistance property. We analyze the security of the bootstrapping protocol in two parts. First, we address the case of a malicious device attempting to pair with the legitimate hub. We then analyze the case where a rogue hub attempts to pair with a legitimate device. Note that, an adversary targeting the synchronization phases of the protocol will fail to pair with either the legitimate hub or devices, as we will show in Proposition 3 later. In the following we assume that the adversary does not attempt a desynchronization attack.

Malicious device pairing with the legitimate hub: Any malicious device that simply participates in the protocol will appear as an additional device beyond the $N - 1$ legitimate devices indicated by the user. The extra device count leads to the abortion of the protocol according to Step 3. The legitimate hub raises an alarm by broadcasting all ON slots during the MC ON-OFF transmission of the protocol transcript digest. As we showed in Proposition 1, this broadcast cannot be canceled and eventually propagates to all legitimate devices.

An alternative approach for the adversary would be to hijack the pairing session of a legitimate device so that the total number of participating devices is not violated. The integrity verification phase prevents this hijacking because the transmission of the protocol transcript digest is protected by the VERSE primitive. According to Proposition 2, as long as any subset of devices computes different transcripts, all devices will detect the attack with probability no less than $1 - \delta$.

Rogue hub pairing with a legitimate device: The adversary can attempt to pair with a legitimate device by posing as the hub and hijacking the pairing session with the legitimate hub. To carry out this attack against a device D_i, the adversary has to perform a signal overshadowing attack and replace the legitimate DH primitive m_A with m_M at D_i. Moreover, the adversary has to replace the protocol transcript digest $[h(s) \mid\mid h(s)_r]$ transmitted by the remaining legitimate devices and A to D_i, with $[h(s') \mid\mid h(s')_r]$. Proposition 2, states that as long as any subset of devices computes different transcripts, all devices will detect the attack with probability no less than $1 - \delta$. Hence, the adversary will fail to pose as a legitimate hub. □

Moreover, in Proposition 3, we show that an adversary targeting the initialization phase to either desynchronize the legitimate devices or make them synchronize with a rogue

hub leads to a protocol failure. Therefore, we do not need to introduce a secure synchronization mechanism.

Proposition 3. *The bootstrapping protocol protected by VERSE fails under a desynchronization attack during the initialization phase.*

Proof. The proof is provided in Appendix C. □

VI. Evaluation

In this section, we experimentally evaluate the effectiveness of signal cancellation under different number of verifiers. We also discuss practical implementation details.

Experimental Setup: We performed all the experiments using NI-USRP 2921 devices. Each device and the hub was realized by one USRP device. The adversary was implemented using two USRP devices one for listening and one for relaying. The listening adversarial device was equipped with a directional antenna (LP0965 Log Periodic PCB Antenna, 850MHz to 6.5GHz) aimed at the TX, whereas the adversarial transmitting device was equipped with either a directional antenna aiming at one RX, or an omnidirectional antenna targeting multiple RXs. All devices were synchronized (with the clock of the same computer) and transmitted at 2.4GHz with 22MHz bandwidth. The slot duration was fixed to 1ms. An ON slot was realized with the transmission of 250 random symbols with 4μs duration, whereas an OFF slot was realized with silence. Experiments were performed at night to minimize Wi-Fi interference although Wi-Fi beacon signals were present during the experiments. The threshold for determining an ON slot was set to -50dBm, which is significantly higher than the receiver sensitivity (typically at or less than -70dBm). This higher value was selected to minimize false positives due to ambient wireless activities at the 2.4GHz band. Each experiment was repeated 10^6 times.

A. Effectiveness of the Signal Cancellation

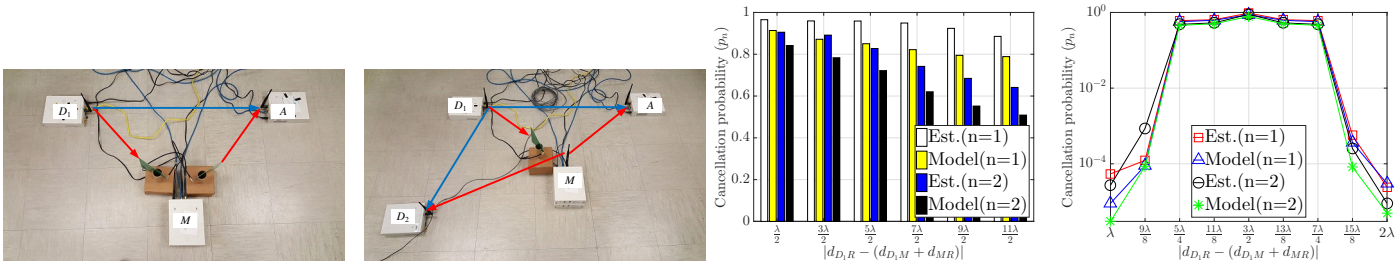
Signal cancellation when $n = 1, 2$. In the first set of experiments, we evaluated the probability p_n (used in Proposition 2 and Corollary 1) of successful signal cancellation via a relay attack for $n = 1$ and $n = 2$. For $n = 1$, we used the experimental setup shown in Fig. 17(a). A device D_1 sent 10^6 MC ON-OFF modulated bits to a hub A in the presence of M who performed a relay cancellation. The two USRPs implementing M were stacked on top of each other at a location on one ellipse that satisfied (3) and (4). For $n = 2$, we used the experimental setup shown in Fig. 17(b). The adversary was placed at the intersection of the two ellipses that satisfied (3) and (4). The transmitting antenna of M was replaced with an omnidirectional one to allow the simultaneous cancellation at two locations.

The receiver at M played three roles: (a) estimate the respective channels, (b) quickly detect ON slots using energy detection, and (c) determine the symbols being transmitted from D_1 during ON slots in an online fashion as M is not aware of the pseudo-random symbols transmitted by D_1. The estimated channel was used to craft the amplitude of the symbol relayed by M's transmitter to cancel D_1's signal at the receivers (the phase was matched based on M's location). The transmitting signal at M was crafted using two approaches. In the first approach, M estimated the $\mathbf{h}_{D_1M}$ and $\mathbf{h}_{MA}$ channels based on the transmissions of D_1 and A, respectively. The $\mathbf{h}_{D_1A}$ channel was modeled after a Rician model with a K factor equal to two, which represents an indoor environment with a strong LoS component. In the second approach, no channel estimation took place at M. All channels were modeled after a free-space path loss model with an attenuation exponent $\alpha = 2$.

Figure 17(c) shows the cancellation probability (p) as a function of the difference between the direct and relay paths. The adversary was placed at the different ellipses dictated by eq. (4), and for w =1, 2, 3, 4, 5, and 6. We observe that when the adversary is close and therefore, has a dominant LoS channel to D_1 and H, the cancellation probability is quite high (94.56% and 91.17% for estimated channel and modeled channel attenuation, respectively for $n = 1$ and 90.57% and 84.42% for estimated channel and modeled channel attenuation, respectively for $n = 2$). Even at several wavelengths away, signal cancellation remains possible with non-negligible probability. The cancellation performance is worse for $n = 2$ because M has to perform simultaneous cancellation at both A and D_2 and more channels need to be estimated. Moreover, the channel estimation yields a stronger cancellation capability compared to channel modeling for both $n = 1$ and $n = 2$.

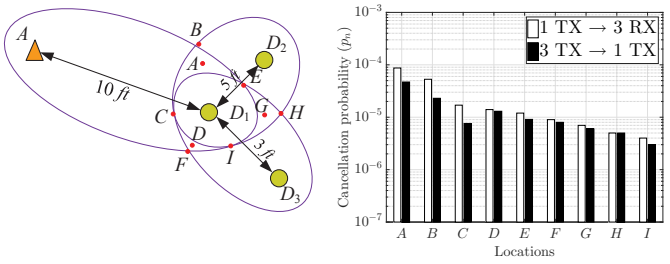
Sensitivity to location placement: In the next set of experiments, we studied the sensitivity of the cancellation attack to M's location. The adversary was placed at a set of ellipses with a path difference between λ to 2λ and incremented by a step of $\lambda/8$. Figure 17(d) shows the cancellation probability as a function of the difference between the direct and relay path. As expected, the cancellation probability is maximized when the path difference equals $(3\lambda/2)$, which satisfies eq. (4). The cancellation probability drops significantly when M's location deviates more than $\lambda/2$ from the optimal location for both $n = 1$ and $n = 2$. From this experiment, we verify that signal cancellation attacks are sensitive to the adversary's location due to the short wavelength of the carrier frequency. A location perturbation of just a few centimeters is sufficient to reduce the effectiveness of the attack, as M's signal no longer arrives at the targeted RXs with the opposite phase.

Signal cancellation when $n = 3$**:** We also evaluated the signal cancellation capability for the one TX/three RX scenario and the three TX/one RX scenario. These two cases serve as the basis for the security of VERSE. We used the topology shown in Fig. 18(a). In the first scenario, D_1 broadcasted MC ON-OFF signals that were simultaneously received by three RXs. According to Proposition 1, there is no single location that allows M perform signal cancellation to all three RXs. Therefore, we selected a set of locations that could likely succeed in canceling some of the received signals. Specifically, the adversary is placed in all locations marked by dots. Locations (A, B, C, E, F, H) correspond to the intersection of two ellipses whereas locations (D, G, I) are the centroids of the



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

Fig. 17: (a) Experimental setup for signal cancellation from one TX to one RX, (b) experimental setup for signal cancellation from one TX to two RXs, (c) cancellation probability as a function of the distance difference between the direct and the relay paths when M is placed at an ellipse satisfying eq. (4), and (d) cancellation probability as a function of the distance difference between the direct and the relay paths, when M is perturbed from the location with path difference equal to $3\lambda/2$.



(a) (b)

Fig. 18: (a) Experimental topology for the evaluation of security primitive of VERSE, and (b) cancellation probability for the experimental setup of (a).

areas created by the three closest intersection points. In the second scenario, A, D_2, and D_3 synchronously transmitted an MC ON-OFF signal that was received by D_1.

Figure 18(b) shows the cancellation probability for the two different scenarios and for each location. We observe that for any scenario, the cancellation probability is below 10^{-4}. Moreover, the cancellation probability was non-zero in all cases due to the relatively high threshold value (-50dBm) that was used to detect ON slots. Although the adversary's signal was not the exact inverse to annihilate legitimate transmissions, on certain occasions, there was sufficient alignment to drop the received power below -50dBm for the respective RX(s). It should be noted here that this experiment is not the proof of the adversary's inability in performing cancellation when $n > 2$, but the proof is derived from Proposition 1.

Alarm raising probability: We further evaluated the security of VERSE in terms of raising an alarm. We replicated the experimental setup of Fig. 18 and implemented the verification phase where every device transmits the hash of the protocol transcript using MC ON-OFF modulation. We considered an adversary that successfully replaced m_1 of D_1 with m_1' leading to the compilation of s_M at D_2, D_3, A and the compilation of s at D_1. To account for a varying number of bits that must be canceled by M, we varied the Hamming distance between $h(s_M)$ and $h(s)$ from 0.1 of the hash length (160 bits) to 0.8 of the hash length. This is done by randomly generating two 160-bit strings with the desired Hamming distances. An alarm was raised by any device that detected a transmitted sequence different than the one it was transmitting. In all scenarios tested and for all adversary locations, *all verifiers* detected the message manipulation and raised an alarm. The attack was detected with probability one for all 10^6 hash transmissions.

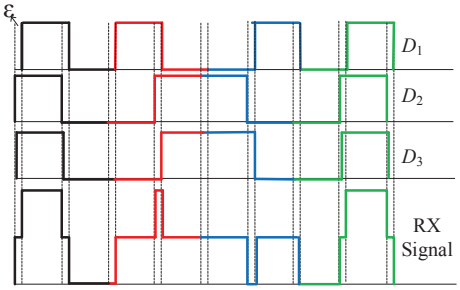


Fig. 19: An example of superimposed received signals from D_1, D_2 and D_3 which are misaligned by an offset of ϵ.

B. Practical Considerations

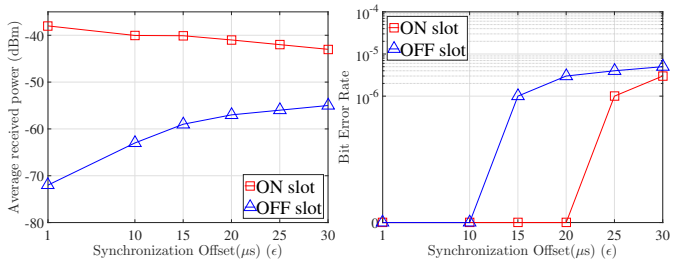
We now analyze the time synchronization requirement, interference effect for the VERSE protocol and its timing overhead.

Synchronization: During the verification phase of VERSE, multiple devices must simultaneously transmit an ON-OFF sequence. Possible misalignment between the clocks of each device may lead to false alarms. To address the possible time misalignment, the hub broadcasts a delimiter just before the start of the verification phase, to synchronize The clock of each device. Despite this synchronization, there is still possible time misalignment between the devices due to clock drift and the different path delays caused by multipath or NLoS channels to each receiver. There have been extensive studies on synchronization of independent wireless nodes [33], but practically it is impossible to reach perfect synchronization.

Figure 19 shows an example, where D_1, D_2, and D_3 transmit simultaneously, with the transmissions being misaligned by a time offset ϵ. Misalignment causes some energy from ON slots "bleed" into OFF slots and some silent period of the OFF slot "bleed" into ON slots. However, the offset ϵ is much smaller (a few μsec) than the slot duration for the ON-OFF sequence which is set to 1ms. The state $s(j)$ of the j^{th} slot is decided according to the following rule:

$$s(j) = \begin{cases} \text{OFF}, & \text{if } p(j) \leq \gamma_D, \\ \text{ON}, & \text{if } p(j) > \gamma_D. \end{cases} \quad (9)$$

where γ_D is the detection threshold (set to -50dBm in our experiments), and $p(j)$ is the average received power over the j^{th} slot. To resolve the time misalignment problem, a solution similar to [44] can be adopted. Rather than averaging the power of all the samples in slot j, an RX eliminates the samples corresponding to an interval $\epsilon_{\max}$ from the beginning and the end of each j^{th} slot, where the slot boundaries are computed according to the RX's own clock. This strategy leaves a time



(a) (b)

Fig. 20: (a) Average received power of superimposed signal from D_1, D_2, and D_3 on ON and OFF slots as a function of synchronization offset (ϵ), and (b) bit error rate as a function of synchronization offset (ϵ) in μs.

interval of $T - 2\epsilon_{\max}$ for estimating the received power, where T is the slot duration, and, $\epsilon_{\max}$ is the maximum time offset between any of the devices.

Experimental evaluation of synchronization: We set up three USRP devices to transmit ON-OFF messages simultaneously, while a fourth USRP was acting as the intender RX. We placed the TXs that simultaneously transmitted the random MC ON-OFF sequence at different locations in the laboratory with both LoS and NLoS channels to the RX. TX_1 was placed behind a bookshelf inside the room, TX_2 was placed outside the room to ensure an NLoS channel, whereas TX_3 was placed at a LoS to the RX. The transmit power for an ON slot was set to 20dBm with a symbol duration of 1ms. An artificial clock misalignment from $\epsilon = 1\mu$s to $\epsilon = 30\mu$s was induced between D_1, D_2, and D_3 to emulate the maximum time offset error. The experiment lasted for the transmission of 10^6 sequences of 40 bits each.

The first experiment was performed to select the detection threshold γ_D. Figure 20(a) shows that average received power during an ON slot varied from -42dBm to -38dBm. The received power during an OFF slot varied from -72dBm to -55dBm indicating the presence of some ambient noise. The detection threshold was set to $\gamma_D = -50dBm$.

In the second experiment, we used the same setup with experiment one and evaluated the slot detection error rate as a function of the synchronization offset. To cope with the time misalignment, the RX excluded the first 30μs from the beginning and end of each slot. The results for the ON slot error rate and the OFF slot error rate are shown in Fig. 20(b). We observe that ON slots are always correctly detected for any time offset. For the OFF slots, a very small number (seven slots out of 10^6) were wrongly estimated. This indicates that excluding the samples at the beginning and end of each slot effectively addresses the synchronization problem.

Interference Effect: To make VERSE robust to interference from co-existing wireless systems, we set the detection threshold for ON slots significantly higher than the typical receiver sensitivity. In the experiments, we selected the detection threshold for ON slots to be -50dBm, which is orders of magnitude higher than the average noise level (typically at -120dBm). The security of VERSE could be impacted because the adversary no longer has to cancel a transmission to the noise floor, but achieving cancellation below the detection threshold is sufficient. To account for this tradeoff, the system security, as expressed by Proposition 2 and Corollary 1, incorporate the probability p_n of successfully flipping a bit during cancellation. This probability parametrizes the success of the adversary in performing cancellation due to considering a higher than the noise floor detection threshold.

Timing Analysis: The timing overhead of VERSE includes the following components (a) the initialization step, (b) exchanging the public DH parameters, and (c) transmitting in MC ON-OFF mode the digest of the protocol transcript. The initialization step can be maximum of τ for powering of all the group devices by the user so they can be set to pairing mode, which can be set to 120s [45]. From the remaining two components, the verification phase dominates the protocol's timing performance, since the ON-OFF mode is significantly slower than nominal transmission speeds. However, the ON-OFF keying time is constant to the group size. For a hash with length ℓ, a total of $2(\ell + r)$ slots of duration T are necessary to complete the verification phase. Assuming typical values of $\ell = 256$, $r = 256$ (in the worst case) and $T = 1ms$ [5], the verification phase requires 1.024s to complete which is acceptable for all practical uses and it is independent of the number of participating devices.

VII. Conclusions

We addressed the problem of securely bootstrapping a group of devices to a hub when none of the devices share any prior security associations. We propose VERSE, a new PHY-layer group message integrity verification primitive resistant to MitM attacks over the wireless channel. We exploit the existence of multiple devices that act as verifiers of the protocol transcript for integrity protection. When three or more devices perform an integrity check, it is infeasible for the adversary to simultaneously manipulate the wireless signal at all devices, based on geometrical constraints. We presented a DH-based device bootstrapping protocol that utilized VERSE, which only requires in-band communications with minimal human effort during initialization. We formally prove the security of both VERSE and the bootstrapping protocol against active attacks. With a real-world USRP testbed, we experimentally validated our theoretical results by showing that an increasing number of devices significantly weakens the adversary's ability to successfully manipulate wireless signals. This is in contrast to prior state-of-the-art where the attacker's success probability increases with the number of devices.

Acknowledgments

We thank the anonymous reviewers and the program committee for their insightful comments. This research was supported in part by the NSF under grants CNS-1409172, CNS-1731164 and CNS-1410000. Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the author(s) and do not necessarily reflect the views of the NSF.

REFERENCES

[1] L. Columbus. (2016) Roundup of internet of things forecasts and market estimates, 2016. [Online]. Available: https://www.forbes.com/sites/louiscolumbus/2016/11/27/roundup-of-internet-of-things-forecasts-and-market-estimates-2016/#2b025ea0292d

[2] J. Zhang and V. Varadharajan, "Wireless sensor network key management survey and taxonomy," *Journal of Network and Computer Applications*, vol. 33, no. 2, pp. 63–75, 2010.

[3] A. Leung and C. Mitchell, "Ninja: Non identity based, privacy preserving authentication for ubiquitous environments," *UbiComp 2007: Ubiquitous Computing*, pp. 73–90, 2007.

[4] The Guardian. (2016) DDoS attack that disrupted internet was largest of its kind in history, experts say. [Online]. Available: https://www.theguardian.com/technology/2016/oct/26/ddos-attack-dyn-mirai-botnet

[5] S. Capkun, M. Cagalj, R. Rengaswamy, I. Tsigkogiannis, J.-P. Hubaux, and M. Srivastava, "Integrity codes: Message integrity protection and authentication over insecure channels," *IEEE Transactions on Dependable and Secure Computing*, vol. 5, no. 4, pp. 208–223, 2008.

[6] N. Ghose, L. Lazos, and M. Li, "HELP: Helper-enabled in-band device pairing resistant against signal cancellation," in *Proc. of 26th USENIX Security Symposium*, 2017, pp. 433–450.

[7] Y. Hou, M. Li, and J. D. Guttman, "Chorus: Scalable in-band trust establishment for multiple constrained devices over the insecure wireless channel," in *Proc. of the WiSec Conference*, 2013, pp. 167–178.

[8] S. Gollakota, N. Ahmed, N. Zeldovich, and D. Katabi, "Secure in-band wireless pairing." in *Proc. of USENIX security symposium*. San Francisco, CA, USA, 2011, pp. 1–16.

[9] D. Balfanz, D. K. Smetters, P. Stewart, and H. C. Wong, "Talking to strangers: authentication in ad-hoc wireless networks," in *Proc. of NDSS Symposium*, 2002.

[10] F. Stajano and R. J. Anderson, "The resurrecting duckling: Security issues for ad-hoc wireless networks," in *Proc. of IWSP*, 2000, pp. 172–194.

[11] M. Cagalj, S. Capkun, and J.-P. Hubaux, "Key agreement in peer-to-peer wireless networks," *Proceedings of the IEEE*, vol. 94, no. 2, pp. 467–478, Feb. 2006.

[12] S. Pasini and S. Vaudenay, "SAS-based authenticated key agreement," in *Proc. of PKC Conference*, ser. LNCS, vol. 3958, 2006, pp. 395 – 409.

[13] S. Laur and S. Pasini, "SAS-based group authentication and key agreement protocols," in *Proc. of PKC Conference*, ser. LNCS, 2008, pp. 197–213.

[14] T. Perković, M. Čagalj, T. Mastelić, N. Saxena, and D. Begušić, "Secure Initialization of Multiple Constrained Wireless Devices for an Unaided User," *IEEE transactions on mobile computing*, 2011.

[15] J. M. McCune, A. Perrig, and M. K. Reiter, "Seeing-is-believing: Using camera phones for human-verifiable authentication," in *Proc. of Security and Privacy Symposium*, 2005, pp. 110–124.

[16] R. Nithyanand, N. Saxena, G. Tsudik, and E. Uzun, "Groupthink: Usability of secure group association for wireless devices," in *Proc. of ACM international conference on Ubiquitous computing*. ACM, 2010, pp. 331–340.

[17] A. Kumar, N. Saxena, G. Tsudik, and E. Uzun, "Caveat eptor: A comparative study of secure device pairing methods," *Proc. of PRECOM Conference*, pp. 1–10, 2009.

[18] L. H. Nguyen and A. W. Roscoe, "Authentication protocols based on low-bandwidth unspoofable channels: a comparative survey," *Journal of Computer Security*, vol. 19, no. 1, pp. 139–201, 2011.

[19] C.-H. O. Chen, C.-W. Chen, C. Kuo, Y.-H. Lai, J. M. McCune, A. Studer, A. Perrig, B.-Y. Yang, and T.-C. Wu, "GAnGS: gather, authenticate 'n group securely," in *Proc. of MOBICOM Conference*, 2008, pp. 92–103.

[20] Y.-H. Lin, A. Studer, H.-C. Hsiao, J. M. McCune, K.-H. Wang, M. Krohn, P.-L. Lin, A. Perrig, H.-M. Sun, and B.-Y. Yang, "SPATE: small-group pki-less authenticated trust establishment," in *Proc. of MOBISYS Conference*, 2009, pp. 1–14.

[21] M. T. Goodrich, M. Sirivianos, J. Solis, G. Tsudik, and E. Uzun, "Loud and clear: Human-verifiable authentication based on audio," in *Proc. of ICDCS Conference*, 2006, p. 10.

[22] C. Pöpper, N. O. Tippenhauer, B. Danev, and S. Capkun, "Investigation of signal and message manipulations on the wireless channel," in *Proc. of European Symposium on Research in Computer Security*. Springer, 2011, pp. 40–59.

[23] Y. Hou, M. Li, R. Chauhan, R. M. Gerdes, and K. Zeng, "Message integrity protection over wireless channel by countering signal cancellation: Theory and practice," in *Proc. of the AsiaCCS Symposium*, 2015, pp. 261–272.

[24] T. Perkovic, M. Cagalj, T. Mastelic, N. Saxena, and D. Begusic, "Secure initialization of multiple constrained wireless devices for an unaided user," *IEEE transactions on mobile computing*, vol. 11, no. 2, pp. 337–351, 2012.

[25] M. Li, S. Yu, J. D. Guttman, W. Lou, and K. Ren, "Secure ad hoc trust initialization and key management in wireless body area networks," *ACM Transactions on sensor Networks (TOSN)*, vol. 9, no. 2, p. 18, 2013.

[26] L. H. Nguyen and A. W. Roscoe, "Authenticating ad hoc networks by comparison of short digests," *Information and Computation*, vol. 206, no. 2-4, pp. 250–271, 2008.

[27] L. H. Nguyen and A. Roscoe, "Efficient group authentication protocols based on human interaction." *IACR Cryptology ePrint Archive*, vol. 2009, p. 150, 2009.

[28] J. Valkonen, N. Asokan, and K. Nyberg, "Ad hoc security associations for groups," in *Proc. of European Workshop on Security in Ad-hoc and Sensor Networks*. Springer, 2006, pp. 150–164.

[29] M. Farb, Y.-H. Lin, T. H.-J. Kim, J. McCune, and A. Perrig, "Safeslinger: easy-to-use and secure public-key exchange," in *Proc. of international conference on Mobile computing & networking*. ACM, 2013, pp. 417–428.

[30] F. L. Wong and F. Stajano, "Multichannel security protocols," *IEEE Pervasive Computing*, vol. 6, no. 4, 2007.

[31] M. N. Mejri, N. Achir, and M. Hamdi, "A new group diffie-hellman key generation proposal for secure vanet communications," in *Proc. of CCNC Conference*. IEEE, 2016, pp. 992–995.

[32] S. Mirzadeh, H. S. Cruickshank, and R. Tafazolli, "Secure device pairing: A survey." *IEEE Communications Surveys and Tutorials*, vol. 16, no. 1, pp. 17–40, 2014.

[33] A. Sampath and C. Tripti, "Synchronization in distributed systems," in *Advances in Computing and Information Technology*. Springer, 2012, pp. 417–424.

[34] "IEEE standard for information technology–telecommunications and information exchange between systems local and metropolitan area networks–specific requirements - part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications," *IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012)*, pp. 1–3534, Dec 2016.

[35] "IEEE standard for low-rate wireless networks - amendment 5: Enabling/updating the use of regional sub-ghz bands," *IEEE Std 802.15.4v-2017 (Amendment to IEEE Std 802.15.4-2015, as amended by IEEE Std 802.15.4n-2016, IEEE Std 802.15.4q-2016, IEEE Std 802.15.4u-2016, and IEEE Std 802.15.4t-2017)*, pp. 1–35, June 2017.

[36] "IEEE standard for information technology–telecommunications and information exchange between systems - local and metropolitan area networks–specific requirements - part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment 2: Sub 1 ghz license exempt operation," *IEEE Std 802.11ah-2016 (Amendment to IEEE Std 802.11-2016, as amended by IEEE Std 802.11ai-2016)*, pp. 1–594, April 2017.

[37] "IEEE standard for information technology– telecommunications and information exchange between systemslocal and metropolitan area networks– specific requirements–part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications–amendment 4: Enhancements for very high throughput for operation in bands below 6 ghz." *IEEE Std 802.11ac-2013 (Amendment to IEEE Std 802.11-2012, as amended by IEEE Std 802.11ae-2012, IEEE Std 802.11aa-2012, and IEEE Std 802.11ad-2012)*, pp. 1–425, Dec 2013.

[38] A. M. Tonello, N. Laurenti, and S. Pupolin, "Analysis of the uplink of an asynchronous multi-user dmt ofdma system impaired by time offsets, frequency offsets, and multi-path fading," in *Vehicular Technology Conference, 2000. IEEE-VTS Fall VTC 2000. 52nd*, vol. 3. IEEE, 2000, pp. 1094–1099.

[39] M. Wilhelm, J. B. Schmitt, and V. Lenders, "Practical message manipulation attacks in ieee 802.15. 4 wireless networks," in *Proc. of Workshop MMB12*, 2012, pp. 29–31.

[40] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, 1976.

[41] E. Bresson, O. Chevassut, D. Pointcheval, and J.-J. Quisquater, "Provably authenticated group diffie-hellman key exchange," in *Proceedings of the*

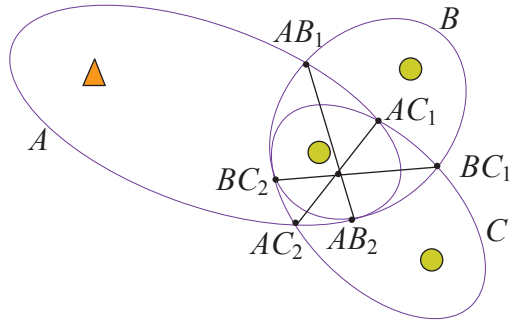


Fig. 21: Three eclipses sharing one focus point. The lines join the intersection points between pairs of ellipses are concurrent, with the common intersection point inside all three ellipses.

8th ACM conference on Computer and Communications Security. ACM, 2001, pp. 255–264.

[42] V. Rabinovich and N. Alexandrov, "Typical array geometries and basic beam steering methods," in *Antenna Arrays and Automotive Applications*. Springer, 2013, pp. 23–54.

[43] H. J. Visser, *Array and phased array antenna basics*. John Wiley & Sons, 2006.

[44] A. Dutta, D. Saha, D. Grunwald, and D. Sicker, "SMACK: a smart acknowledgment scheme for broadcast messages in wireless networks," in *ACM SIGCOMM Computer Communication Review*, vol. 39, no. 4. ACM, 2009, pp. 15–26.

[45] W.-F. Alliance, "Wi-fi protected setup specification," *WiFi Alliance Document*, vol. 23, 2007.

[46] I. I. BOGDANOV, "Two theorems on the focus-sharing ellipses: a three-dimensional view," *Journal of Classical Geometry Volume 1 (2012)*, vol. 1, p. 1, 2012.

APPENDIX

APPENDIX A-PROOF OF PROPOSITION 1

Proposition. *Three distinct ellipses sharing one focal point irrespective of the plane they lie in, do not have a common point of intersection.*

Proof. Let A, B, and C be three ellipses sharing a focal point, with the three ellipses being distinct. Each pair of ellipses will have a minimum of two intersection points. Let AB_1, AB_2, BC_1, BC_2, AC_1, and AC_2 be the respective intersection points between A, B, B, C, and A, C. These points are shown in Fig. 21. According to Theorems 1 and 2 in [46], the lines connecting the intersection points between each pair of ellipses *are concurrent* at a common intersection that lies inside all three ellipses, irrespective of the planes the ellipses lie in. Assume now that there is a common intersection point between all three ellipses. Without loss of generality, assume that AB_1 is the same as AC_1. Then the lines AB_1-AB_2 and AC_1-AC_2 will have a common origin point. The only way that the two lines AB_1-AB_2 and AC_1-AC_2 are concurrent with the BC_1-BC_2 line is if also AB_2 is the same point as AC_2. In the latter case, B and C become the same ellipse or A and B become the same ellipse, and there are no longer three distinct ellipses. Hence, A, B, and C sharing a focal point cannot have a common point of intersection.

The proof states that three ellipses sharing a common focus point cannot have a common intersection point, regardless of the plane that the lie in. This is sufficient for our purposes. Without attempting a formal proof, it is natural to conjecture that the proof does extend to the case of ellipsoids. Ellipsoids consist of an infinite number of ellipses on different planes that have common foci. If three of these ellipsoids share a single focal point, then we can treat their intersection as the intersection of an infinite number of combinations between three ellipses sharing the focal point on different planes. Applying the proof on those ellipses shows that three ellipsoids sharing one focal point do not have a common intersection point. □

APPENDIX B-PROOF OF PROPOSITION 2

Proposition. *When the group size is N, the VERSE is δ–secure against active message modifications with*

$$\delta \quad \leq \quad (p_H + (1 - p_H)p_n)^{\ell}, \tag{10}$$

here, δ is the probability that M can replace any m_i sent by D_i with m'_i at any subset of remaining devices without being detected at every $D_{i'} \in \mathcal{D}$ (where $\mathcal{D}$ is the set of all legitimate devices), p_H is the probability for a bit of $h(s) \mid\mid h(s)_r$ to equal a bit of $h(s_M) \mid\mid h(s_M)_r$, and p_n is the probability of a successfully flipping one bit in $[\cdot]$ during transmissions from n TXs to one RX or from one TX to n RXs where $n = \lceil N/2 \rceil$, and ℓ is the length of the hash function $h(\cdot) \mid\mid h(\cdot)_r$. We show that δ is a negligible function of ℓ.

Proof. Let's consider an adversary that targets to modify one message m_i sent by D_i[1]. In the simplest case, the adversary replaces m_i with m'_i at all other legitimate devices $\mathcal{D} \setminus D_i = \mathcal{D}_{-i}\{D_{i'}|i' \neq i\}$, where $\mathcal{D}$ denotes the set of all legitimate devices in the group. During the VERSE verification phase, all the $D_{i'}$ compiles $s_M = m_1||\ldots||m'_i||\ldots||m_n$, whereas D_i compiles $s = m_1||\ldots||m_i||\ldots||m_n$. Then to pass the transcript verification M has to replace $[h(s) \mid\mid h(s)_r]$ with $[h(s_M) \mid\mid h(s_m)_r]$ at all the $D_{i'}$ on transmission from D_i, so that none of the verifiers raise an alarm. If any one other verifier $D_{i'}$ raises an alarm, then all the others will detect the MitM attack and raise an alarm, since a single M can only be set to cancel the transmissions from one TX (D_i) to other RXs at one time, but not from $D_{i'}$ to those RXs. Hence, the adversary has to perform signal cancellation on transmission of one TX to multiple (all other) RXs in this case.

In general, M might choose to replace m_i with m'_i, at a subset of other legitimate devices, $\mathcal{D}_M = \{D_{i'}|, i' \in 1, 2, ...N, i' \neq i\} \subset \mathcal{D}_{-i}$, such that during the VERSE verification phase D_i and all the $D_{i''} \in \mathcal{D}_{-i} \setminus \mathcal{D}_M$ compile the same communication transcript as s, whereas every $D_{i'} \in \mathcal{D}_M$ compiles s_M. Then to pass the transcript verification M has to replace (cancel and inject) $[h(s) \mid\mid h(s)_r]$ with $[h(s_M) \mid\mid h(s_m)_r]$ at all the $D_{i'} \in \mathcal{D}_M$ on transmissions from D_i and every $D_{i''} \in \mathcal{D}_{-i} \setminus \mathcal{D}_M$, and vice versa, to replace the ON-OFF signals from $D_{i'} \in \mathcal{D}_M$ to all devices in $D_{i''} \in \mathcal{D}_{-i} \setminus \mathcal{D}_M$ and D_i, such that none of the verifiers raise the alarm. Hence, the adversary has to perform signal cancellation on transmissions of multiple TXs to multiple RXs simultaneously.

[1]Modifying multiple messages is more difficult, in which case the success probability is upper bounded by that of modifying a single message.

In any of the above cases, the success of the adversary is upper-bounded by the capability to replace $[h(s) \mid\mid h(s)_r]$ with $[h(s_M) \mid\mid h(s_m)_r]$ on transmission from one TX to multiple RXs, or from multiple TXs to one RX. Next, we compute the probability of replacing $[h(s) \mid\mid h(s)_r]$ with $[h(s_M) \mid\mid h(s_m)_r]$. First, we compute the probability that the k^{th} bit is received as $h(s_M) \mid\mid h(s_M)^k_r$ at all $D_{i'} \in \mathcal{D}_M$ (say, from D_i). This occurs if one of the following two conditions is met: either the k^{th} bit is the same in $h(s) \mid\mid h(s)_r$ and $h(s_M) \mid\mid h(s_M)_r$ or M is able to perform cancellation and injection of k^{th} at all $D_{i'} \in \mathcal{D}_M$:

$$\begin{aligned} \Pr[k^{th} = h(s_M)^k] &= \Pr[h(s)^k = h(s_M)^k] + \\ &\quad \Pr[h(s)^k \neq h(s_M)^k] \Pr[\text{Cancel}] \\ &= p_H + (1 - p_H)p_n, \end{aligned} \tag{11}$$

where p_H is the probability for a bit of $h(s) \mid\mid h(s)_r$ to equal a bit of $h(s_M) \mid\mid h(s_M)_r$, and p_n is the probability upper bound of successfully flipping one bit in $[\cdot]$ during transmissions from multiple TXs to one RX or from one TX to multiple RXs (it is reasonable to assume the same p_n applies to both scenarios).

For a strictly universal hash function, the hashes for two different inputs differ at each bit with probability $1/2$. The probability δ of accepting the modified message of M at A is computed by taking into account the total number of bits (ℓ) generated by the hash function $h(\cdot) \mid\mid h(\cdot)_r$. The adversary's modified message is accepted by all the $D_{i\prime}$ if M has replaced m_i with m'_i and $[h(s_M) \mid\mid h(s_M)_r]$ is received at all $D_{i'}$ instead of $[h(s) \mid\mid h(s)_r]$. We argue that successful cancellation of every ON-slot occurs independently, as each ON slot symbol transmitted by each device is randomly generated (i.i.d). This is because, if the attacker is located at a fixed location, the resulted aggregated signal relayed by the attacker will be randomly distributed (and independent across symbols), so the probability of each aggregated received symbol's power being less than a threshold is also independent from each other. Thus, δ is the product of the probability of successfully manipulating each bit:

$$\begin{aligned} \delta &\leq \Pi_{k=1}^{\ell} \Pr[k^{th} = h(s_M)^k] \\ &\leq \Pi_{k=1}^{\ell} (p_H + (1 - p_H)p_n) \\ &\leq (p_H + (1 - p_H)p_n)^{\ell}. \end{aligned} \tag{12}$$

where p_H is the probability for a bit of $h(s) \mid\mid h(s)_r$ to equal a bit of $h(s_M) \mid\mid h(s_M)_r$, and p_n is the probability of a successfully flipping one bit in $[\cdot]$ during transmissions from multiple TXs to one RX or from one TX to multiple RXs, and ℓ is the length of the hash function $h(\cdot) \mid\mid h(\cdot)_r$. It is easy to show that δ is a negligible function of ℓ, since $p_H + (1 - p_H)p_n < 1$ (as long as $p_n < 1$ in general, for any number of verifiers). Since for each possible sub-case (of adversary choosing to modify one message from any device to any subset of remaining devices), we have the same success probability bound δ, we can conclude that the adversary's overall success probability is also upper bounded by δ, meaning with probability at least $1 - \delta$, all the devices in the group will detect the MitM attack. □

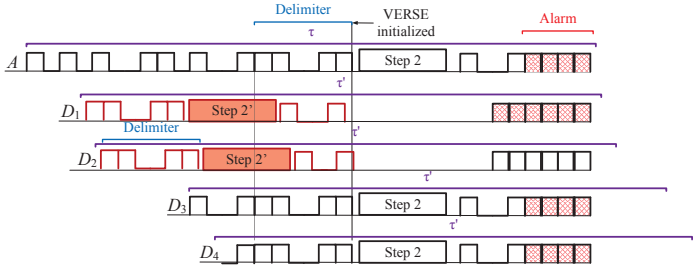


Fig. 22: Attack on the initialization step of VERSE.

Appendix C-Proof of Proposition 3

Proposition. *The bootstrapping protocol protected by VERSE fails under a desynchronization attack during the initialization phase.*

Proof. The attack on the synchronization between the legitimate entities during the simultaneous MC ON-OFF transmission can be mitigated by initiating VERSE simultaneously at all the legitimate entities. We first discuss the initialization step, followed by the security analysis of it. Finally, we discuss the security analysis on the attack on synchronization.

According to Step 1 of the pairing protocol, the protocol is initiated by the user by powering ON all the legitimate devices and setting the hub to pairing mode. This step is followed by the transmission of the DH primitives. To inform each device when all other devices are powered ON and ready to pair, we have added the coordination process.

Initially, the user sets the hub to pairing mode by pressing a button on the hub device. When in this mode, the hub broadcasts a random MC ON-OFF sequence while waiting for other devices to be turned ON. This mode lasts for a pre-specified time period τ sufficient for pairing all other devices, or until the users press the pairing button again. This phase terminates by transmitting a known delimiter (ON-ON-OFF-OFF-ON-ON). When legitimate devices are powered ON, they listen to the ON-OFF sequence broadcasted by the hub and wait for the known delimiter to synchronously initiate Step 2. Note that the known delimiter further allows the devices to time synchronize with the clock of the hub.

To combat possible active attacks on initialization and/or time synchronization, each device remains in pairing mode for a period τ' which is slightly longer than τ, even if it has already paired with the hub.

We now demonstrate that an adversary targeting the initialization and/or synchronization of the protocol will fail to pair with the legitimate hub or a legitimate device. Consider the device activation sequence shown in Fig. 22. Because the delimiter used to denote the end of the initialization phase is public, an adversary can attempt to pair with a legitimate device by performing a signal cancellation and injection attack. In this attack, the adversary cancels the ON-OFF sequence of the hub and injects a delimiter sequence to cause the initiation of the pairing process sooner than the time intended by the legitimate

hub. According to Proposition 1, the adversary is able to cancel the ON-OFF sequence at most at two devices, say D_1 and D_2. These two devices may complete the pairing process with the malicious hub before other legitimate devices are activated or execute the protocol with the legitimate hub. However, they remain in pairing mode for a period $\tau' > \tau$.

When devices D_3 and D_4 execute the VERSE protocol with the legitimate hub, the adversary has to replace the expected messages from D_1 and D_2 with his own messages to satisfy the group count. This can be done by a simple message injection. However, during the confirmation stage, all devices synchronously transmit the ON-OFF sequence of the protocol transcript digest. In our example, at least A, D_3, and D_4 will transmit that sequence. As a result, D_1 and D_2 will overhear a second integrity verification phase (Step 3) within their pairing period τ'. Based on Proposition 1, the adversary cannot perform cancellation from three transmitters to one receiver to prevent the overhearing of the legitimate confirmation phase at D_1 and D_2. The two latter devices will raise an alarm by transmitting all ON slots during the integrity verification phase and the protocol will terminate in FAILURE.

This delimiter is sent by the hub before the synchronous transmission of the protocol digest is initiated (Step 3). We clarify that we have not assumed a secure synchronization protocol between the hub and the legitimate devices. We have simply stated that under a benign setting, the devices are capable of achieving synchronization with a bounded error ϵ. This error has been assumed to be fairly large in our experimentations relative to typical clock drifts of wireless devices and topology scenarios considered in this work (we set ϵ between 1μs to 30μs). Such a value demonstrates that VERSE operates correctly even in worst-case time misalignment scenarios. If the adversary attacks the second SYNC message to misalign the legitimate transmitters, the ON-OFF sequence transmitted during the integrity verification phase will be misaligned leading to the sounding of the alarm by transmitting all ON slots. Therefore, the adversary cannot successfully join the group, by causing time misalignment between legitimate devices.

Now we will present the security analysis on the attack of synchronization between legitimate entities. Two attack scenarios can weaken the security of the proposed group pairing protocol: (a) a malicious device pairs with the legitimate hub, or (b) a legitimate devices pairs with a rogue hub.

Malicious device pairing with the legitimate hub: The device synchronization is initiated by the hub, by sending the delimiter message in Step 3, when the VERSE primitive is used to secure the transmission of $[h(s) \mid\mid h(s)_r]$. To pair with hub A, the malicious device must follow the timing set by the end of the delimiter sent from A. Any message received by A at a different timeline will be aborted. The adversary can attempt to cancel the delimiter message sent by A at a target device D_i, so as to prevent D_i from broadcasting the protocol digest with other devices. The goal is to reduce the number of devices where cancellation should take place when $[h(s) \mid\mid h(s)_r]$ is transmitted using ON-OFF mode by the remaining devices. However, device D_i will overhear the MC ON-OFF sequence transmitted by the remaining of devices, without having received the delimiter. This sequence from many simultaneous transmitters to one receiver cannot be canceled by the adversary. Device D_i will raise an alarm by transmitting continuous ON slots, leading to the protocol failure. So attacking the synchronization protocol can only lead to a DoS and does not provide the adversary with an additional capability to compromise the protocol.

Malicious device posing as a legitimate hub: The adversary can also attempt to synchronize the legitimate device to his own delimiter message rather than the legitimate hub. If the desynchronized device transmits when the MC ON-OFF sequence of the protocol transcript is transmitted by legitimate devices, the legitimate devices and the hub will detect energy during the OFF slots and abort the protocol.

This proves that the VERSE is protected against any attack on the synchronization between the legitimate entities. □