AeroKey: Using Ambient Electromagnetic Radiation for Secure and Usable Wireless Device Authentication

KYUIN LEE, University of Wisconsin–Madison, USA
YUCHENG YANG, University of Wisconsin–Madison, USA
OMKAR PRABHUNE, University of Wisconsin–Madison, USA
AISHWARYA LEKSHMI CHITHRA, University of Wisconsin–Madison, USA
JACK WEST, Loyola University Chicago, USA
KASSEM FAWAZ, University of Wisconsin–Madison, USA
NEIL KLINGENSMITH, Loyola University Chicago, USA
SUMAN BANERJEE, University of Wisconsin–Madison, USA
YOUNGHYUN KIM, University of Wisconsin–Madison, USA

Wireless connectivity is becoming common in increasingly diverse personal devices, enabling various interoperation- and Internet-based applications and services. More and more interconnected devices are simultaneously operated by a single user with short-lived connections, making usable device authentication methods imperative to ensure both high security and seamless user experience. Unfortunately, current authentication methods that heavily require human involvement, in addition to form factor and mobility constraints, make this balance hard to achieve, often forcing users to choose between security and convenience. In this work, we present a novel over-the-air device authentication scheme named AeroKey that achieves both high security and high usability. With virtually no hardware overhead, AeroKey leverages ubiquitously observable ambient electromagnetic radiation to autonomously generate spatiotemporally unique secret that can be derived only by devices that are closely located to each other. Devices can make use of this unique secret to form the basis of a symmetric key, making the authentication procedure more practical, secure and usable with no active human involvement. We propose and implement essential techniques to overcome challenges in realizing AeroKey on low-cost microcontroller units, such as poor time synchronization, lack of precision analog front-end, and inconsistent sampling rates. Our real-world experiments demonstrate reliable authentication as well as its robustness against various realistic adversaries with low equal-error rates of 3.4% or less and usable authentication time of as low as 24 s.

CCS Concepts: • **Human-centered computing → Ubiquitous and mobile computing systems and tools**; • **Security and privacy →** *Authentication.*

Additional Key Words and Phrases: device authentication, key generation, EMR-based authentication

Authors' addresses: Kyuin Lee, University of Wisconsin–Madison, 1415 Engineering Dr, Madison, Wisconsin, USA, kyuin.lee@wisc.edu; Yucheng Yang, University of Wisconsin–Madison, 1415 Engineering Dr, Madison, Wisconsin, USA, yang552@wisc.edu; Omkar Prabhune, University of Wisconsin–Madison, 1415 Engineering Dr, Madison, Wisconsin, USA, oprabhune@wisc.edu; Aishwarya Lekshmi Chithra, University of Wisconsin–Madison, 1415 Engineering Dr, Madison, Wisconsin, USA, chithra@wisc.edu; Jack West, Loyola University Chicago, 1052 West Loyola Avenue, Chicago, Illinois, USA, jwest1@luc.edu; Kassem Fawaz, University of Wisconsin–Madison, 1415 Engineering Dr, Madison, Wisconsin, USA, kfawaz@wisc.edu; Neil Klingensmith, Loyola University Chicago, 1052 West Loyola Avenue, Chicago, Illinois, USA, neil@cs.luc.edu; Suman Banerjee, University of Wisconsin–Madison, 1210 West Dayton St, Madison, Wisconsin, USA, suman@cs.wisc.edu; Younghyun Kim, University of Wisconsin–Madison, 1415 Engineering Dr, Madison, Wisconsin, USA, younghyun.kim@wisc.edu.

2474-9567/2022/3-ART20 $15.00
https://doi.org/10.1145/3517254

ACM Reference Format:
Kyuin Lee, Yucheng Yang, Omkar Prabhune, Aishwarya Lekshmi Chithra, Jack West, Kassem Fawaz, Neil Klingensmith, Suman Banerjee, and Younghyun Kim. 2022. AeroKey: Using Ambient Electromagnetic Radiation for Secure and Usable Wireless Device Authentication. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 1, Article 20 (March 2022), 29 pages. https://doi.org/10.1145/3517254

1 INTRODUCTION

The trend of miniaturization of personal devices is more prevalent than ever. Enabled by rapid advances in low-power electronics technology and driven by emerging applications, small personal devices have become ubiquitous in the forms of various hand-held and wearable devices, such as smartwatches, wireless earbuds, and tablet stylus pens. Such devices seldom function alone but interoperate with others through wireless connection and have facilitated various applications that have not been possible as a single device. The expanding interoperability, as well as miniaturization of the devices, on the other hand, have rendered the security and usability of wireless connectivity more challenging. For instance, traditional device authentication methods that profoundly rely on manual human involvement (e.g., typing in a pin or password) to establish the authenticity of the connecting device can be considerably labor-intensive as the number of associating devices increases. Also, stringent constraints in the cost and the form factor have forced the devices to come with no usable interface, which makes password- or pin-based authentication more time-consuming and cumbersome through the usage of a secondary device, such as the user's smartphone.

The lack of usable authentication scheme has forced many personal devices to choose usability over security and resulted in failures in properly securing critical data. For instance, without a user interface to enter a password, Bluetooth 5 devices use a common default password to encrypt the communication messages used to establish an authentication token [10, 24]. If a malicious adversary manages to gain physical access to a Bluetooth 5 headset and unpair it, they can intercept the pairing messages and extract the authentication token, ultimately gaining perpetual access to the plaintext of all subsequent communications at a distance [35].

Recently, *context-based zero-involvement authentication* has emerged as a promising solution to the usability challenges of the authentication process. It enables secure key establishment between devices that wish to authenticate with one another, using correlated random noise from the environment to generate identical authentication or encryption keys [8, 27]. Environmental noises, such as ambient audio, light, or radio waves, are excellent source of randomness for key generation because they can be easily and inexpensively measured with sensors that are readily available on many Internet-of-Things (IoT) and mobile devices. If a secret key, independently generated by two devices from spatiotemporally unique environmental noise, closely matches, it is the evidence that they are located in the same place at the same time, implicitly implying that they are owned by the same user and thus trustworthy to each other. Its principal advantage is that, unlike password-based authentication where the entropy is derived from a user-provided text string, it leverages the randomness that exists in the environment and requires no input from the user. This property is particularly advantageous for enhanced security as every authentication session can be based on a new key, instead of repeatedly reusing the same password, or even periodically re-authenticating during a single session.

The benefits of the context-based authentication on personal wireless devices come with a set of practical challenges. First, it should be able to generate high-quality keys even from low-entropy environmental noise in the presence of substantial spatial variation that may cause discrepancies in independently generated keys. Furthermore, low-cost devices do not have the luxury of highly consistent device properties such as identical sensor modalities and precise time synchronization. For instance, previous solutions rely on common sensing capabilities such as microphones [34], luminosity sensors [26], and accelerometers [11, 19] to extract randomness, which may not be ubiquitously available across heterogeneous mobile devices [12]. Other solutions require the users to touch, wear, or interact with the devices to establish a common key based on the body potential [14, 47]

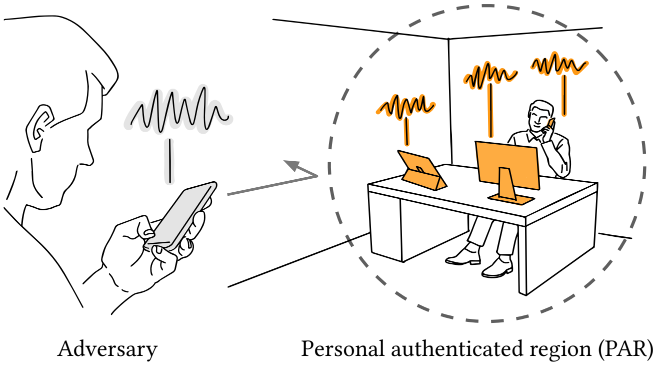


Fig. 1. Devices within the personal authenticated region (PAR) are autonomously authenticated based on ambient EMR that can only be observed within the region.

or other modalities [1, 22, 23, 46, 49]. Similarly, these protocols require some explicit or implicit user action and do not generalize to non-wearable devices. Therefore, the question of *how to provide a context-based and zero-involvement authentication protocol that is practical to deploy and use* is still an open one.

In this work, we present AeroKey as an affirmative answer to the above question. It represents a novel context-based authentication scheme for establishing secure wireless networks of personal devices using *ambient electromagnetic radiation (EMR)*[1] as a source of randomness. AeroKey extracts entropy from low-frequency ambient EMR generated by surrounding electrical appliances and powerlines that are ubiquitous indoors. We observe that the EMR noise tends to be spatially correlated only inside a small area of a few meters in radius yet temporally uncorrelated, making it ideal for authenticating spatiotemporally co-located devices in a personal area network. Fig. 1 is a conceptual illustration of the AeroKey's operation. A pair of AeroKey-enabled devices first independently measure the ambient EMR using readily available analog-to-digital converters (ADCs) without costly hardware-assisted signal conditioning. Then, from digitized sequences of the EMR measurements, two devices generate bit sequences which will be identical only within a small area called *personal authenticated region (PAR)*, around the user. The two devices can make use of this identical bit sequences to form a basis of a symmetric key to authenticate each other.

AeroKey can potentially be used to enhance the usability in almost all inter-device authentication scenarios. An example use case would be during the authentication process between the two headless devices (with no graphical user interface or other peripherals). For instance, two AeroKey-enabled Raspberry Pis can authenticate each other by simply placing them in a close distance without the use of external monitor and keyboard. Also, relatively minimal hardware requirements of AeroKey allows it to be implemented in many small mobile or personal devices with MCU to facilitate seamless authentication between devices (i.e., associating wireless earbuds or smart pens with smartphones).

The main contributions of this paper are summarized as follows:

- We present a secure and usable device authentication scheme named AeroKey, leveraging spatially correlated and temporally uncorrelated randomness in simultaneously measured ambient EMR using low-cost hardware.
- We propose and implement essential techniques (i.e., signal processing and noise extraction) to derive symmetric keys from EMR noise and to overcome the challenges in instrumenting AeroKey on low-cost

[1] The radiation in this context refers to non-ionizing radiation at frequencies below visible light (e.g., super and ultra low frequency ranges), contrary to the high-energy ionizing radiation (e.g., X-rays, Gamma rays).

microcontroller units (MCUs), such as lack of time synchronization, low-precision ADC measurement, and inconsistent sampling rates (Section 4).
- Our real-world experiments using commercial off-the-shelf MCUs in various environments demonstrate reliable authentication rate between co-located devices and its robustness against various realistic adversaries with low EER of 3.4% or less and authentication time of under 24 s (Section 5).

2 RELATED WORK

Context-based authentication has been actively investigated as it provides enhanced usability and security over traditional authentication mechanisms. Mathur et al. [25] utilizes received signal strength indicator of wireless signal (FM and TV) to generate symmetric keys for co-located devices. However, the need for a dedicated antenna limits its practical deployment in small-form-factor devices. Other ambient contexts such as audio, luminosity, and visual channel have been utilized to harvest random bits to pair proximate devices as well [17, 26–28, 33, 34, 42]. These methods usually rely on homogeneous sensor pairs (i.e., light sensor, camera, microphone) to capture matching contexts across authenticating devices. Similar works utilize more ubiquitously available context [18, 48]; they leverage the plugin power supply of devices to extract powerline noise to pair co-located devices within the same circuit breaker (thus only applicable to stationary devices). Additionally, Han et al. [12] pairs heterogeneous devices located in secure boundary (indoor) by leveraging inter-event timing contexts from different sensors types. While this method effectively addresses the limitations of requiring identical sensing modalities, its source of entropy still needs to be derived from human activities within the authenticating environment, which may not be observable at all times.

The schemes within the wearable device domain typically work under the assumption that the authenticating devices are being worn by the same user. Existing works leverage human body sensing capacity to derive identical secret keys from ambient RF signal [14, 47] or utilize human body as transmission medium to communicate secret key [30]. Other human-generated contexts, such as electrocardiography (ECG), electromyography (EMG), gait signatures, and body gestures can also be used to autonomously pair body-worn devices [1, 20, 22, 23, 44–46, 49]. For instance, the authentication schemes leverage inter-pulse intervals of ECG signals to facilitate body area network communications [23, 31]. Another line of work presents universal operation sensing [20], allowing IoT and wearable devices to pair through user's physical operations such as pressing a button or rotating a knob. While these works successfully explore secure and usable methods to autonomously pair devices, they require identical or at least one sensing capability that is not applicable to many miniature devices due to cost or form factor constraints. More importantly, the majority of the work in the wearable domain does not generalize to non-wearable devices.

EMR has been actively studied for detection and classification of the electrical appliances in the environments. These approaches rely on a unique transient and continuous noise that propagates back into the powerline during their powered operations [4, 6, 29]. Unlike AeroKey, these works utilize precise hardware (i.e., oscilloscope and spectrum analyzer) to physically measure the powerline signal to extract noise components that propagates through the line. Similar line of works have utilized EMR to identify or fingerprint different devices based on the device's unique EMR signatures [5, 37]. Contrary to our work, they leverage additional hardware components (i.e., software-defined radio or magnetic sensor) to capture the signal and further process them into a signal form that is temporally and spatially consistent. The drawbacks of these approaches are the requirement of the pre-training process and the extra hardware burden that may not be suitable for today's mobile devices. Overall, AeroKey efficiently addresses above mentioned limitations with a ubiquitously available measurement method (i.e., embedded ADC with a wire) enabled by efficient signal processing techniques. This can be strongly advantageous when implemented in low-cost, small form factor devices without the need of identical and specific sensing hardware such as microphone, luminosity sensors and etc. Furthermore, unlike the works within the

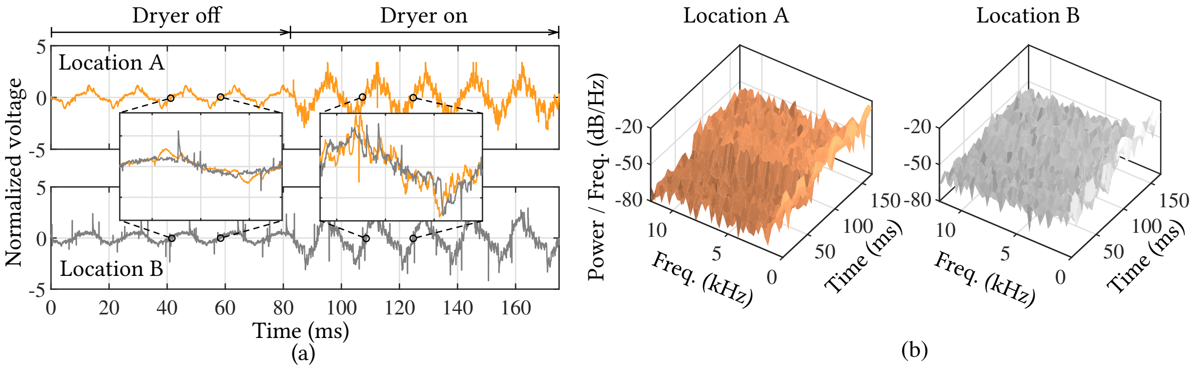


Fig. 2. (a) Raw ADC readings from two different locations (5 m apart) as a hair dryer (1875 W) switches on at 80 ms. (b) Spectrograms of the raw readings from two different locations.

wearable device domain [14, 20, 30, 47], AeroKey does not leverage external human body as an antenna (which requires direct contact) or other human activities to generate keys or extract entropy.

3 BACKGROUND AND SYSTEM OVERVIEW

In this section, we first discuss the underlying physical basis of AeroKey and further define the system as well as its threat models.

3.1 Ambient Electromagnetic Radiation

A powerline that delivers power to appliances, lighting, and other electrical loads is essentially a charged wire carrying an alternating current (AC) and radiates electromagnetic waves, which is called ambient EMR. The combined effect of the time-varying electromagnetic waves manifests in EMR, which oscillates at the same rate as the nominal AC frequency (50 or 60 Hz in most countries). As modern structures are built with complex networks of powerlines and wirings, the ambient EMR can be ubiquitously observed within the buildings. In fact, the effect is so pronounced that it is often easily detectable as an unwanted "60-Hz tone" on microphones, amplifiers, and other radio-related instruments [7].

At a specific location, the observed ambient EMR is a superposition of EMRs emitted by surrounding sources, including the powerlines and electric loads. While the AC powerlines are the main source of ambient EMR, various appliances produce random and transient electromagnetic waves on top of the powerline EMR based on their operating states and conditions (e.g., plugged or unplugged) [9, 21, 32, 47]. Therefore, EMR measurements from two locations (in the same structure) may vary significantly even when they are measured at the same time due to the different distance to the powerlines, different spatial geometry of the lines [9, 36], and transient operation of the appliances. Furthermore, even at a specific location, the ambient EMR may constantly vary depending on various electrical and human activities at or near the location. In summary, an ambient EMR signal at a certain location includes two major components: i) a periodic signal, typically a sinusoid, at the nominal powerline frequency, and ii) superimposed noises, which are higher frequency components that represent temporal and spatial variations in an environment.

AeroKey leverages the spatial and temporal uniqueness of the superimposed noises to allow a group of co-located devices that observe similar EMR patterns to *autonomously* generate near-identical bit sequences from the patterns. To provide an intuition of these properties, we conduct a controlled experiment in a typical home environment. First, we directly connect a short conductor wire to an ADC pin of the two off-the-shelf

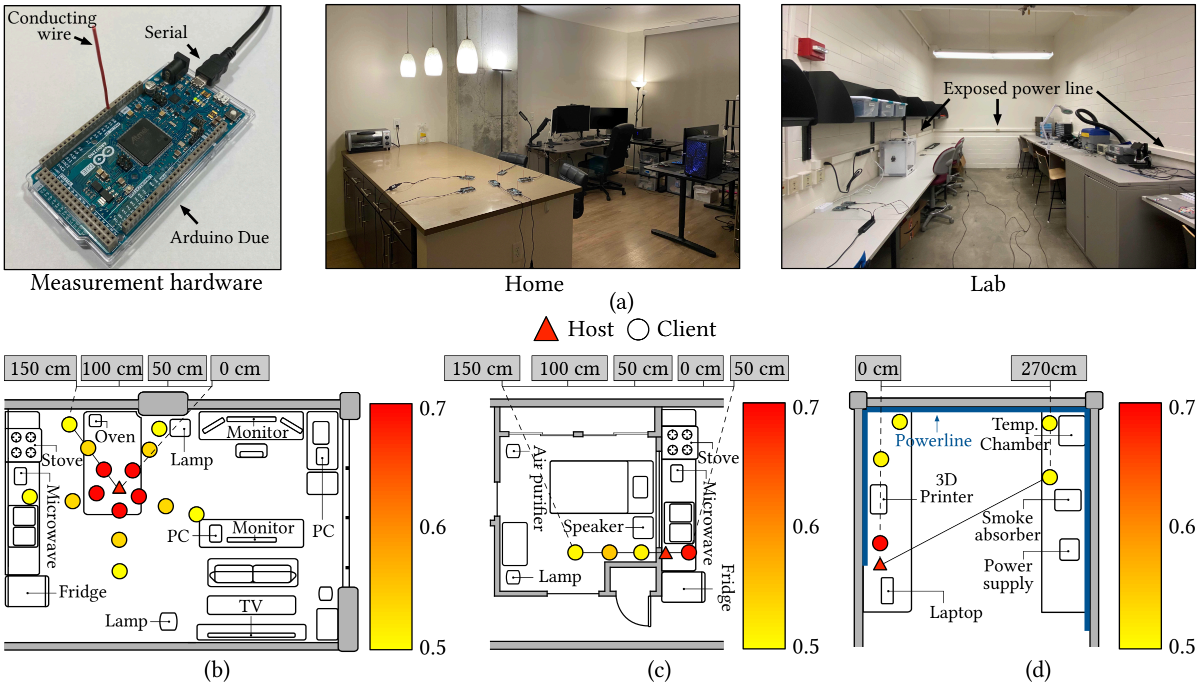


Fig. 3. (a) Measurement hardware with a conducting wire as an antenna. (b) Correlation heatmap of superimposed noise components between host and client devices within typical living room environment. (c) Correlation between devices located in the next room. (d) Correlation between devices located along the identical powerline.

MCUs (Arduino Due). Then, we read the ADC at two different locations that are 5 m apart, while power-cycling a 1875 W hair dryer in the middle between the two measuring points. As Fig. 2(a) shows, the ADC readings at both locations exhibit a synchronous 60 Hz sinusoidal structure due to the powerline EMR. However, a closer look at the pairs of periods captured from both signals reveals clear discrepancies in the shape and amplitude before and after turning on the hair dryer, indicating the spatial variability of EMR from two different locations at the distance of 5 m. If we compare the signals captured at the same location before and after turning on the hair dryer, the difference is even more dramatic, indicating the temporal variability of EMR caused by the state of the surrounding appliances. Fig. 2(b) presents the spectrograms of the signals that show the random and dynamic variabilities over time, location, and frequency. AeroKey leverages these spatiotemporal variations (on top of the 60 Hz signal) caused by various surrounding electromagnetic noises to harvest unique secret bits only between co-located devices.

To further explore the spatiality of the EMR signal, we extract the superimposed noise components and compare them between various locations within the home and lab environments. The details of the noise extraction method are presented in Section 4. Fig. 3(a) illustrates the measurement hardware (Arduino Due with a conducting wire connected to an input pin of the ADC) as well as the images of the experiment environments. In Figs. 3(b), (c), and (d), we show the correlation heatmap of the extracted noise components between the two devices (Host and Client) at varying locations. In typical living room environment (Fig. 3(b)), as the distance between the devices increases, the correlation gradually decreases from 0.7 at 50 cm down to 0.5 at 150 cm apart. The decrease in the correlation is observable in all directions with increase in the distance, which leads to demonstrate the spatial difference of the EMR noise between the two distinct locations. If the devices are separated by the wall (Fig. 3(c)), the correlation experiences slight decrease compared to an open environment, exhibiting correlation of 0.5 at 50 cm. This is because the observed signal between two devices is slightly different due to the distance variation from the surrounding appliances and the noise signal that gets attenuated by the wall. In Fig. 3(d), we illustrate the effect of the powerline to the observed signal by conducting an experiment in the lab, where we can visually identify the exposed powerline. Although all the devices are located close (within 20 cm) to the same powerline, the correlation decreases between devices due to the surrounding electrical loads producing varying noise signal at different locations. This leads to demonstrate that the EMR at a specific location is spatially unique, which allows AeroKey to only authenticate closely located devices.

3.2 System and Threat Models

In the following, we define the system and threat models of AeroKey.

3.2.1 System Model.

The main objective of AeroKey is to provide a mechanism for secure initial authentication between two headless commodity devices. As such, one of the use cases of AeroKey is to provide an alternative to the pairing in Bluetooth. The security level of pairing Bluetooth devices hinges on securely exchanging a six-digit code (32 bits of entropy) through a user utilizing the I/O interfaces. These pairing mechanisms, however, remain as pain points for users to suffer in the initial pairing stage. AeroKey brings at least this level of security to headless commodity devices without adding additional capabilities and user interaction.

For the design of AeroKey, we consider typical indoor environments where pairs of headless devices located within a short distance of each other attempt to establish a secure wireless connection (authentication). The user is able to keep legitimate devices within a close range and remove any suspicious devices away immediately. If a pair of devices are within close proximity, defined as the *personal authenticated region (PAR)*, they are considered trustworthy and safe to be connected. These assumptions are in line with previous literature in the context-based authentication domain [12, 18, 27, 28, 34]. The maximum radius of the PAR is limited to one meter, which is a typical range that the user can keep the devices under control (e.g., personal desk).

We assume a generic device model that is reasonable for most low-cost personal devices. In our model, headless devices are operated by a low-cost MCU with an integrated, typically low-precision, ADC. They can exchange messages over the established wireless channel (e.g., Bluetooth, Wi-Fi, or Zigbee) to process authentication requests.The authentication process is considered successful only when the pair of devices agree on a symmetric secret key. We assume no pre-established time synchronization between the devices; their system clock is not synchronized, and the ADC sampling frequency may be slightly offset. The powerline we assume throughout this paper is of 120 V at 60 Hz; the application of AeroKey, however, is not limited to a specific powerline voltage or frequency.

3.2.2 Threat Model.

We consider an adversary aiming to illegitimately authenticate its device to the victim's device as a steppingstone for additional threats. We assume that the adversary's device is not located within the victim's PAR at the same time during the authentication process. While it is possible for the adversary to perform an attack by placing or hiding an adversarial device within the PAR (e.g., underneath desk or chair), it would significantly raise the bar for the attack. Furthermore, such attack scenario can be thwarted by adopting a simple confirmation or selection interface (e.g., asking the user to confirm or select the associating device) with minimal user involvement.

The adversary, however, does have access to the victim's future expected PAR when legitimate devices are not within the vicinity. Furthermore, the adversary has full knowledge of the legitimate device's location as well as their authentication timestamps (i.e., protocol initiation time with fine granularity). We assume the adversary to be fully aware of AeroKey's underlying protocol; it is capable of eavesdropping on any plaintext wireless

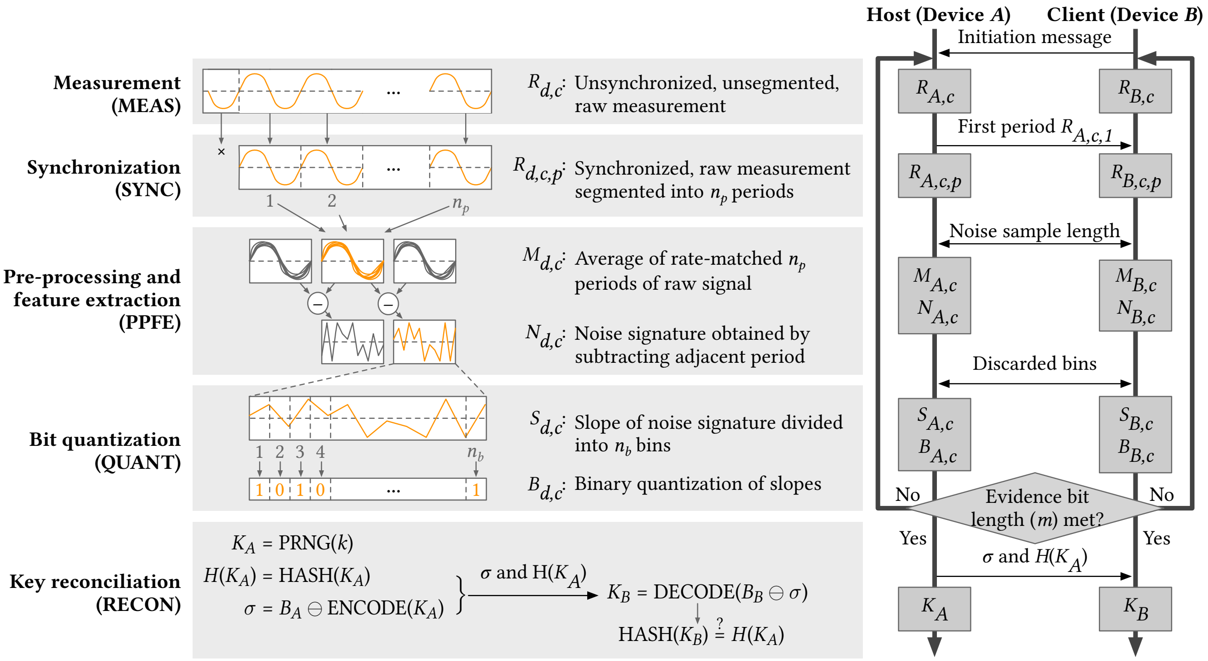

Host (Device A)
Client (Device B)
Initiation message
Measurement (MEAS)
Synchronization (SYNC)
Pre-processing and feature extraction (PPFE)
Bit quantization (QUANT)
Key reconciliation (RECON)
$R_{d,c}$: Unsynchronized, unsegmented, raw measurement
$R_{d,c,p}$: Synchronized, raw measurement segmented into n_p periods
$M_{d,c}$: Average of rate-matched n_p periods of raw signal
$N_{d,c}$: Noise signature obtained by subtracting adjacent period
$S_{d,c}$: Slope of noise signature divided into n_b bins
$B_{d,c}$: Binary quantization of slopes
1 2 n_p
1 2 3 4 n_b
1 0 1 0 ... 1
K_A = PRNG(k)
$H(K_A)$ = HASH(K_A)
σ = $B_A \ominus$ ENCODE(K_A)
σ and H(K_A)
K_B = DECODE($B_B \ominus \sigma$)
HASH(K_B) $\overset{?}{=}$ $H(K_A)$
$R_{A,c}$
$R_{B,c}$
First period $R_{A,c,1}$
$R_{A,c,p}$
$R_{B,c,p}$
Noise sample length
$M_{A,c}$ $N_{A,c}$
$M_{B,c}$ $N_{B,c}$
Discarded bins
$S_{A,c}$ $B_{A,c}$
$S_{B,c}$ $B_{B,c}$
No
Evidence bit length (m) met?
No
Yes
Yes
σ and $H(K_A)$
K_A
K_B

Fig. 4. Five-stage pipeline of the AeroKey protocol. All the symbol definitions are listed in the Appendix.

messages that are used in the legitimate authentication process and transmitting arbitrary packets to the devices. We consider denial of service attacks, such as jamming, to be outside the scope of AeroKey as such attacks are applicable to any wireless communication mechanisms and are not special to AeroKey. The following lists four types of attack scenarios which we evaluate later in Section 5.7.

- *Passive attack:* A nearby device that resides just outside of the victim's PAR tries to authenticate itself to the victim's device by generating bit sequences from its own EMR measurement. This scenario can be common where devices belonging to different users within the same house or space can accidentally authenticate with each other.
- *Replay attack:* An adversary with the knowledge of future PAR and their authentication timestamp gains partial access to the location and performs measurements from that specific location. For example, the adversary has access to a desk at which the victim will sit tomorrow. Afterwards, when legitimate authentication takes place at the similar time and location, the adversary uses previously measured EMR to generate bits to authenticate to the victim's device.
- *Injection attack:* An adversary is capable of inducing strong EMR to the environment to force a common bit sequence as the victim's device, for example, using a high-wattage signal generator and a load. This can also accidentally happen when such high-wattage load is active, and an inadvertent device is nearby.
- *Machine learning (ML) attack:* An adversary with the knowledge and access to the future PAR leverages previously measured EMR as well as any eavesdropped plaintext messages exchanged between legitimate devices to train an ML model and uses it to authenticate itself or directly predict unique bit sequences derived from legitimate authentication process.

4 AEROKEY PROTOCOL

This section details AeroKey's protocol between Device A as the host and Device B as the client. The underlying authentication mechanism of AeroKey requires two co-located devices to generate similar *evidence bit sequences*, B_d (d denotes device name, A or B) to be authenticated. Fig. 4 illustrates various notations used within the pipeline of our protocol consisting of five stages: i) Measurement (MEAS), ii) Synchronization (SYNC), iii) Pre-processing and feature extraction (PPFE), iv) Bit quantization (QUANT), and v) Key reconciliation (RECON). Each pass of the first four stages is denoted as a *cycle* and generates a partial evidence bit sequence (not necessarily by an equal amount). The cycle repeats until the number of evidence bits reaches the desired length, m. Afterwards, the RECON stage allows Devices A and B to establish an encrypted wireless channel using a symmetric key, K, of length k. The details of each stage are explained in the rest of this section.

4.1 Measurement (MEAS)

A primary design goal of AeroKey is to minimize hardware overheads. This goal is especially crucial for devices with stringent form factor constraints (e.g., wireless earbuds). In the MEAS stage, it achieves this goal by measuring ambient EMR using the ADC embedded in MCUs without any analog front-end circuitry for amplification or filtration. In particular, we consider a simple way to measure the ambient EMR: a short conductive wire (or a PCB trace) connected to a floating single-ended ADC pin in lieu of a frequency-matched antenna.

The client (Device B) commences the AeroKey protocol by sending an initiation message to the host (Device A). Both devices independently read their ADC for a pre-defined duration to capture the electric potential on the wire excited by the ambient EMR. The raw time-series measurement signals are denoted by $R_{A,c}$ and $R_{B,c}$, respectively, where c is the cycle count ($c = 1, 2, \ldots, n_c$), and n_c is the total number of cycles. Because the input impedance of a floating ADC pin is very high, the measured signal is weak and susceptible to various sources of electrical noises. Therefore, subsequent stages of the AeroKey protocol carefully synchronize and pre-process the obtained raw signals before extracting evidence bits, as described in the following subsections.

4.2 Synchronization (SYNC)

AeroKey avoids the need for tight clock synchronization that is costly and unavailable for low-cost wireless devices [39], and as a result, the two obtained raw signals, $R_{A,c}$ and $R_{B,c}$, are not temporally aligned. The timing mismatch is mainly attributed to the transmission delay of the initiation message, which almost always leads the raw signal of Device A to lag behind the signal of Device B. If we compare the worst-case Wi-Fi transmission time (up to 250 ms [38]) to the length of a single period in 60 Hz ambient EMR (16.7 ms), the starting point of the measured signal between each device can be off by several periods.

To achieve a common notion of time, the SYNC stage leverages the measured signals' sinusoidal property and the structural similarity of the signals measured at the same time. We first leverage the 60 Hz sinusoidal property of the ambient EMR to segment the raw signals into individual periods using zero-crossing detection. Although the raw signals exhibit a strong 60 Hz component, they are considerably noisy with occasional high amplitude peaks, which may result in multiple zero-crossing points within a single period and, in turn, lead to inaccurate detection of period boundaries. To deal with this, we condition the normalized raw signal with a software-based low-pass filter (cut-off at 60 Hz) and apply a Gaussian filter to further smooth the signal. Fig. 5 illustrates an example of raw and conditioned signals, as well as detected positive-going zero-crossings of two devices. From the zero-crossing indices, each device segments $R_{A,c}$ and $R_{B,c}$ into $R_{A,c,p}$ and $R_{B,c,p}$, respectively, where p denotes the period count ($p = 1, 2, \ldots, n_p$) within cycle c, and n_p is the number of periods per cycle. Note that the filtered and smoothed signal is only used to extract the zero-crossing indices used to mark the raw signal into individual periods.

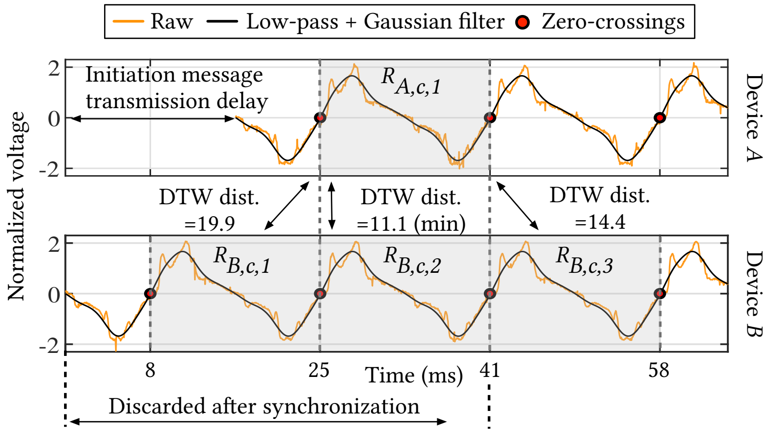


Fig. 5. Synchronization of raw signals between Device A and B. Among three periods of Device B, the second period, $R_{B,c,2}$, shows the minimum DTW distance against Device A's $R_{A,c,1}$.

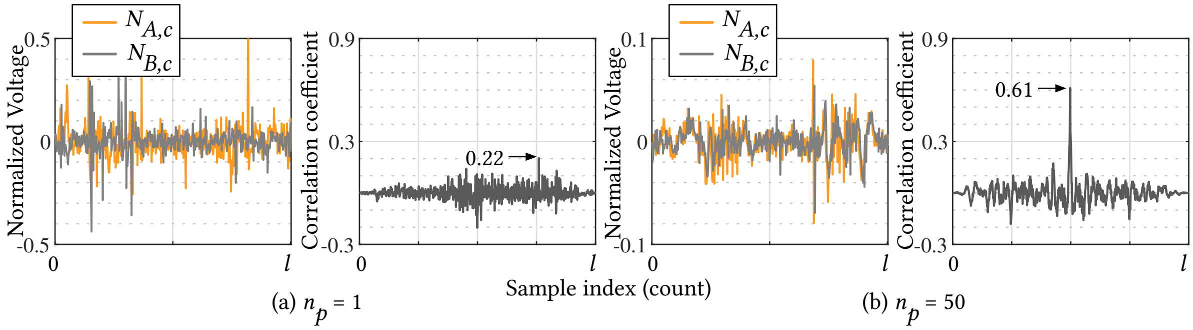
Next, Device A sends the first period of the measured raw signal, $R_{A,c,1}$, to Device B. Upon receiving $R_{A,c,1}$, Device B finds the closest matching period among $R_{B,c,p}$ by sequentially comparing measurements from each period against $R_{A,c,1}$. However, the number of samples in each period is not uniform even when both devices use the same hardware and measurement settings because: i) surrounding appliances inject transient and continuous noises that cause shifting of zero-crossing points, resulting in fluctuations in the frequency measurement, and ii) oscillators of low-cost MCUs suffer from large frequency variability ranging from −17% to +15% [13, 21]. As such, common signal similarity metrics such as root mean squared error (RMSE) or Pearson correlation coefficient are not applicable. To address this problem, we utilize the dynamic time warping (DTW) distance method, an algorithm for measuring the similarity between two time sequences that may vary in their speed or sampling rates [3]. It calculates the optimal time-warping path between two signals and outputs the closest Euclidean distance between them.

In the example shown in Fig. 5, the closest period in Device B is the second one, $R_{B,c,2}$, which exhibits the minimum DTW distance of 11.1 against $R_{A,c,1}$. Upon finding the closest period p (in this example, $p = 2$), both devices discard all measured samples up to end of $R_{A,c,1}$ and $R_{B,c,p}$ and re-segment the subsequent periods as $p = 1, 2, , \ldots, n_p$ as the publicly exchanged $R_{A,c,1}$ can be eavesdropped by the adversary and should not be used to generate evidence bits.

4.3 Pre-processing and Feature Extraction (PPFE)

The main goal of the following PPFE stage is to extract a portion of raw signals that is rich in randomness to be converted into evidence bits. Even between two closely located devices, the raw signals often exhibit low correlation due to slight location differences, environmental noise, and hardware variations. For a pair of signals to be used as a source for evidence bit generation, they should exhibit some correlation across co-located devices so that near-identical bit sequences can be extracted. To increase the correlation, we take the sample-wise mean of n_p raw period signals measured by device d in cycle c into a *mean signal* $M_{d,c}$:

$$M_{d,c} = \frac{\sum_{p=1}^{n_p} R_{d,c,p}}{n_p}. \tag{1}$$



(a) $n_p = 1$

(b) $n_p = 50$

Fig. 6. Two noise signals, $N_{d,c}$, and their cross-correlations calculated from mean signals with (a) $n_p = 1$ and (b) $n_p = 50$ on two co-located devices.

During this process, because each period contains a different number of samples, each device linearly re-samples all of its marked periods down to the minimum period length to make the sample number of each period equal. The signal $M_{d,c}$ is taken as the average of the signals $R_{d,c,p}$ over n_p periods of cycle c. It represents the general shape of the EMR waveform for the duration of the MEAS stage as observed by each device. For instance, when the MEAS stage in each cycle runs for 1 s, we can roughly expect the mean signal to capture the general shape of around 60 periods ($n_p = 60$), which will be more similar across co-located devices than a direct comparison of raw signals. The main reason that the mean signal mainly leverages 60 Hz over the higher frequency component (over 1 kHz range) is because: 1) typical low-cost IoT devices with on-chip ADC does not support high frequency sampling which only allows us to accurately measure signal bandwidth of below few KHz and 2) higher frequency component is difficult to correctly segment and measure due to lack of amplifier and precision analog front-end circuitry.

From the obtained mean signals, each device extracts the cycle-to-cycle temporal amplitude variation referred to as *noise signal*, $N_{d,c}$. This noise signal, mainly used to extract spatiotemporally unique evidence bit sequences, represents the index-wise subtraction of mean signals from two consecutive cycles:

$$N_{d,c} = M_{d,c+1} - M_{d,c}. \tag{2}$$

Because multiple mean periods, $M_{d,c}$, share a similar 60 Hz dominant structure, subtracting two subsequent mean signals essentially removes this dominance, and only captures the transient EMR variations between two subsequent cycles resulting from various electrical activities. Therefore, even if the adversary has access to the dominant structure of the raw signal (i.e., from publicly exchanged $R_{A,c,1}$ in the previous stage), it is not correlated to the resulting noise signal used for key generation. Finally, the two devices agree on the length of their noise signals by re-sampling them to match the length of the shorter one, denoted by l, called the *agreed noise sample length*.

To demonstrate the effectiveness of our signal processing technique in this stage, we show in Fig. 6 two pairs of noise signals obtained from two co-located devices. Fig. 6(a) represents the noise signal pair with the mean signals calculated with $n_p = 1$ (equivalent to no pre-processing), and Fig. 6(b) with $n_p = 50$. When $n_p = 1$, the two noise signals have a low max correlation of 0.22. On the other hand, when calculated from the mean of 50 periods ($n_p = 50$), they show a significantly higher max correlation of 0.61. In Fig. 7(a), we further investigate the relationship between n_p and correlation between noise signals on two devices. We can observe a significant increase in correlation from 0.20 to 0.74 as the n_p increases from 1 to 100. This suggests that the longer MEAS stage runs to measure a greater number of 60 Hz periods, the two noise signals independently extracted from two devices exhibit higher similarity, which can ultimately result in a higher probability of extracting identical

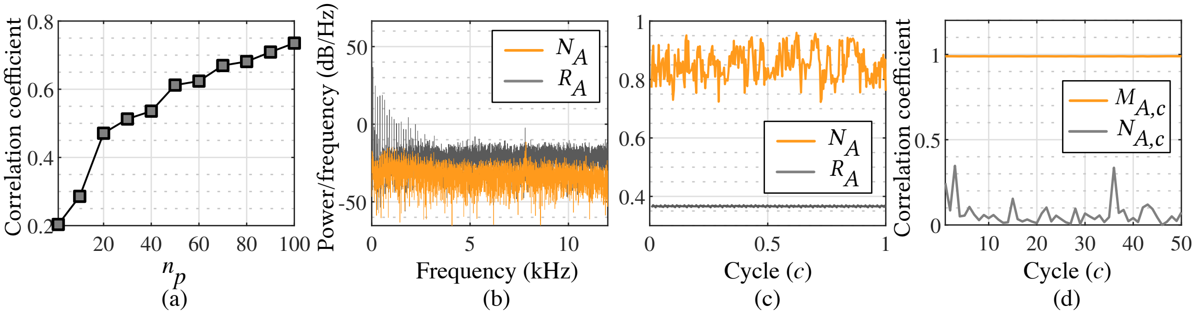


Fig. 7. (a) Relationship between n_p and correlation coefficient between two noise signals. (b) Periodogram of noise signal, N_A, and raw signal, R_A. 60 Hz component is removed in the noise. (c) Spectral entropy of the noise signal, N_A, and the raw signal, R_A.(d) Correlation achieved between $R_{A,c,1}$ and $M_{A,c}$, as well as $N_{A,c}$.

bits on a pair of authenticating devices. In Fig. 7(b), we show a periodogram (estimate of the spectral density) of the extracted noise signal, N_A, and raw signal, R_A. Because N_A retains relatively similar power between range of 0 kHz to 12 kHz, it successfully demonstrates that the PPFE stage effectively removes the 60 Hz sinusoidal property of the raw signal. In Fig. 7(c), we compare the spectral entropy of raw signal R_A against the noise signal N_A. The pre-processed noise signal exhibits more randomness and irregularities (mean of 0.85) compared to the raw signal that mainly fluctuates in sinusoidal behavior (mean of 0.36). This indicates that the extracted noise signal is more suitable for random bit generation purposes. It is important that the extracted noise signal is not similar to the raw signal since the first period of the raw signal, $R_{A,c,1}$, is exchanged over the public channel (during SYNC stage) and is assumed to be eavesdropped on by the adversary. In Fig. 7(d), we illustrate the correlation achieved between $R_{A,c,1}$ and the following mean signal, $M_{A,c}$, as well as the noise signal, $N_{A,c}$, with respect to subsequent cycle number. While the attacker can generally infer the shape of the mean signal with a high correlation of 0.98, there is relatively low correlation between $R_{A,c,1}$ and $N_{A,c}$ due to subtraction operation that removes the dominant shape. Because $N_{A,c}$ is the main source of entropy in generating bit sequences, this suggests that the attacker cannot leverage $R_{A,c,1}$ to infer $N_{A,c}$ used to extract the evidence bit sequences.

4.4 Bit Quantization (QUANT)

In the QUANT stage, both devices translate the time-aligned noise signals ($N_{d,c}$) into a sequence of evidence bits, B_d, used to derive a symmetric key in the following RECON stage. AeroKey leverages the *gradient of the amplitudes of noise signals* to extract bit sequences. Each noise signal is segmented into n_b *bins* of the same length, with b representing the bin count ($b = 1, 2, 3, \ldots, n_b$). Since the length of the noise signal is l, the length of each bin is $\lfloor \frac{l}{n_b} \rfloor$. Next, we determine the slope of each bin, denoted by $S_{d,c,b}$, by curve-fitting its noise signal as a linear function, which is subsequently converted into a bit $B_{d,c,b}$: a bit 1 for a positive slope or a bit 0 for a negative slope. To minimize bit discrepancies due to small differences in the noise around zero, we only use the bins with the absolute value of the slope higher than a *quantization threshold*, th; otherwise, the bins are discarded:

$$B_{d,c,b} = \begin{cases} 1 & \text{if } S_{d,c,b} > th \\ 0 & \text{if } S_{d,c,b} < -th \\ \text{X} & \text{discarded otherwise.} \end{cases} \tag{3}$$

Since this bit extraction is independently performed by two devices on their own noise signals, one device might discard bins (and corresponding bits) that are kept by the other. To use bits extracted from common bins, the

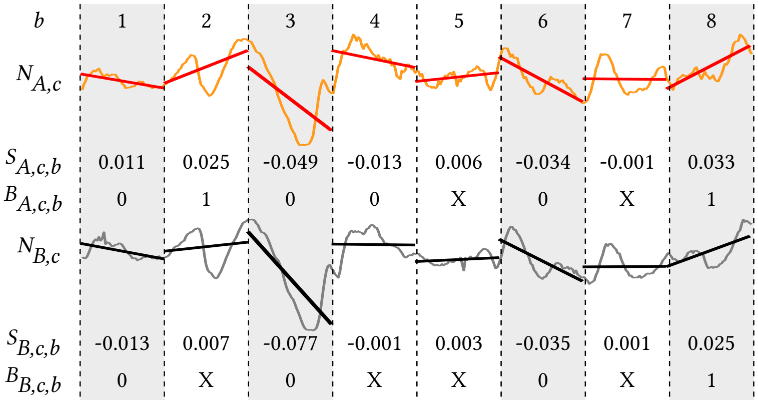


Fig. 8. In QUANT stage with $n_b = 8$, the bits are only extracted from bins with slopes greater or less than the quantization threshold of 0.01: bins 1, 3, 6, and 8.

two devices exchange the extracted bin numbers and agree to generate bits only from the bins that are kept by both devices. Note that exchanging the bin numbers is assumed to be eavesdropped by the adversary over the unencrypted channel, but it neither reveals information about the extracted bits nor makes it easier to guess them.

Fig. 8 illustrates an example of the bit quantization process with $n_b = 8$ and $th = 0.01$. In this example, Device A extracts six bits from bins 1, 2, 3, 4, 6, and 8, and Device B extracts four bits from bins 1, 3, 6, and 8, where the absolute value of the slope is greater or less than positive and negative th, respectively. Device A may generate a bit 1 from bin 2 ($S_{A,c,2} > th$), but since Device B discards it ($-th < S_{B,c,2} < th$), bin 2 is discarded by both of them. A similar observation occurs for bin 4 as well. As a result, a bit sequence "0001" is extracted from bins 1, 3, 6, and 8 that both devices agree to use. In this process, both th and n_b impact the bit agreement rate and the number of extracted bits. Intuitively, increasing th increases the bit agreement rate at the cost of a reduced number of extracted bits per cycle, hence a longer authentication time. In the case of n_b, it should be chosen to estimate the gradient with fine granularity. We empirically choose the values of n_b and th in Section 5. Note that the number of extracted bits varies each cycle due to the random nature of the noise. Therefore, the cycle from the MEAS stage through the QUANT stage is repeated until the total amount of extracted bit length m is reached by the generated evidence bit sequences B_A and B_B.

4.5 Key Reconciliation (RECON)

In the RECON stage, the two devices agree on an identical secret key, K, using the two evidence bit sequences, B_A and B_B. An underlying assumption in this stage is that two devices should authenticate if and only if their extracted evidence bit sequences have a small number of bit errors. AeroKey employs the *fuzzy commitment* scheme [16], where a key is turned into a commit/open pair using evidence bits, in such a way that only similar evidence bits can correctly open a key. This allows devices to transfer the secret data on the public channel as long as the adversary does not have similar evidence bits. The overview of the RECON stage between Device A and B is shown in Table 1. First, Device A generates a key K_A of length k, where $k < m$, using a pseudo-random number generator to be used as the symmetric key and calculates its hashed (e.g., SHA-256) value, $H(K_A)$. Then K_A is encoded into λ, using error correction code (e.g., Reed-Solomon coding), e.g., $\lambda = ENCODE(K_A)$. The commitment σ is then calculated, which is a finite field subtraction between the generated evidence bits and encoded codeword λ, i.e., $\sigma = B_A \ominus \lambda$. Then, σ and $H(K_A)$, which do not divulge any information about B_A or

Table 1. Overview of RECON stage between Devices A and B (Fuzzy commitment).

	Device A		Device B
①	$K_A = PRNG(k)$		
②	$H(K_A) = HASH(K_A)$		
③	$\lambda_A = ENCODE(K_A)$		
④	$\sigma = B_A \ominus \lambda_A$	$\xrightarrow{\sigma, H(K_A)}$	$\lambda_B = B_B \ominus \sigma$
⑤			$K_B = DECODE(\lambda_B)$
⑥			$H(K_A) \overset{?}{=} HASH(K_B)$

K_A, are sent to Device B. Upon receiving σ, Device B, derives K_B by decoding the subtraction result of σ and B_B, i.e., $K_B = DECODE(\lambda_B)$ where $\lambda_B = B_B \ominus \sigma$. To verify the equality of the derived key, Device B compares the hashed values of K_A and K_B, i.e., $H(K_A) \overset{?}{=} H(K_B)$; if identical, two devices are successfully authenticated and can establish a secure channel with the derived key.

The fuzzy commitment scheme ensures identical derivation of the key as long as the error (Hamming distance) between the two evidence bit sequences does not exceed $\frac{m-k}{2}$ bits. The latter condition ensures that the difference operation ($B_B \ominus \sigma$) when performed at Device B results in λ_B that is close to λ at A. The subsequent decoding of $B_B \ominus \sigma$ reveals K_B that should be the same as K_A. We evaluate the relationship between the error tolerance rate of the RECON stage and the length of evidence bit sequence, m, in Section 5.3.

5 EVALUATION

In this section, we evaluate the performance of AeroKey with varying parameter settings, deployed in various real-world environments. We also show the robustness of AeroKey against various strong adversaries as defined in Section 3.2.2.

5.1 Experimental Setup

We use the on-chip ADC of commercial off-the-shelf MCU on the Arduino Due with a conductive wire to measure the ambient EMR. The Arduino Due is equipped with Atmel's 32-bit SAM3X8E ARM Cortex-M3 processor that can operate at 84 MHz. The ADC is set at a sampling rate of 24 kSPS (samples per second) with a 10-bit resolution. The conductive wire has a length of 6 cm unless stated otherwise and wires are oriented parallel to one another to ensure that they both experience the same oscillations in the EMR. The number of periods measured in each cycle (n_p) is fixed at 30 period/cycle (2 cycle/second), which requires only about 30 kB of storage space that can easily be supported even by typical low-cost MCUs without external memory. Finally, the final symmetric key length, k, is set to 128 bits, and we evaluate different error tolerance rates ranging from 5% up to 45% by varying the length of evidence bit sequences, m.

We consider two common use environments: a home and a lab. The home represents an uncontrolled use environment with high-wattage uncontrolled loads, such as ceiling lights, televisions, computers and a fridge, which are switched on or off anytime. On the other hand, the lab represents a controlled use environment. It has several low-amp uncontrolled loads, including lighting, computers, and monitors; and three high-wattage controlled loads, including a fan (50 W), a temperature chamber (240 W), and a smoke absorber (110 W). The controlled loads are not power cycled when the protocol is in progress. The total amount of measurement time is 1093 hours in the home and 1045 hours in the lab.

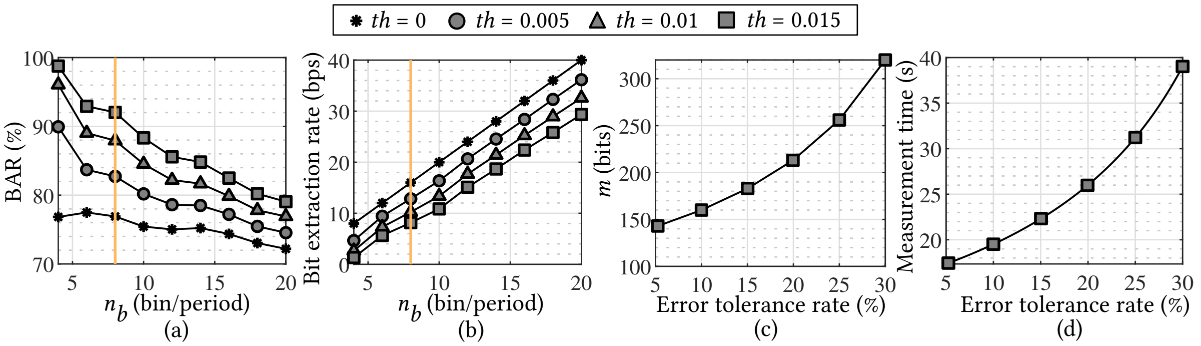


Fig. 9. (a) BAR and (b) bit extraction rate between co-located devices for varying n_b and th. (c) Evidence bit length (m) and (d) measurement time required for varying error tolerance rate.

5.2 Evaluation Metrics

To demonstrate generating common evidence bit sequences using the proposed bit quantization, we use the following metric:

- *Bit agreement rate (BAR):* Rate of matching bits between two evidence bit sequences when compared bit-by-bit.

To evaluate the security of the authentication process, we use the following metrics:

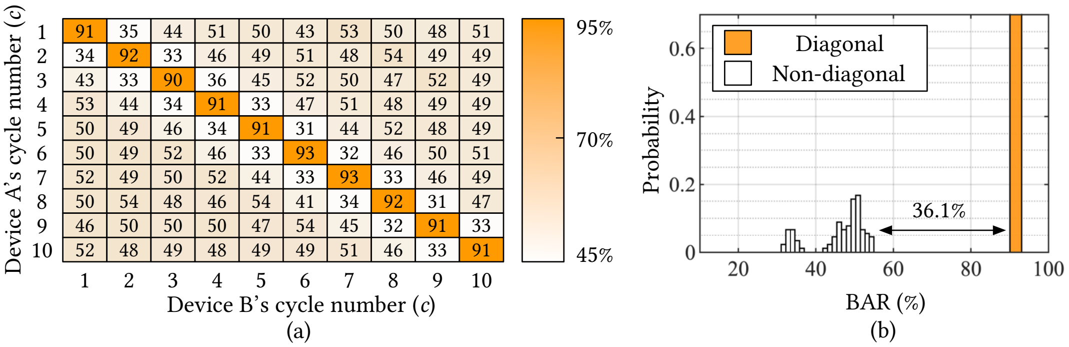
- *True acceptance rate (TAR):* Rate of successful authentication between the legitimate devices within PAR. The false rejection rate (FRR) equals to 100 - TAR.
- *False acceptance rate (FAR):* Rate of successful authentication from the passive, replay, injection and ML attacks.
- *Equal error rate (EER):* Intersection point where FAR is equivalent to FRR. Low EER represents higher accuracy of distinguishing legitimate over adversarial devices.

All acronyms are listed in the appendix section of the paper.

5.3 Protocol Parameters

We first investigate the effects of the main protocol parameters to determine the values that dictate the performance of bit quantization. As discussed in Section 4.4, th and n_b should be carefully chosen in the QUANT stage in order to achieve a high BAR and bit extraction rate. Fig. 9(a) shows BAR for varying n_b and th between devices that are located 20 cm apart. Overall, the BAR consistently decreases as the n_b increases from 4 bins to 20 bins. This is because the gradient of the noise signals becomes less distinctive as the bin width decreases, leading to slight signal differences to extract erroneous bit pairs. When the quantization threshold is not applied ($th = 0$), BAR can be as low as 72.1%–77.4%. On the other hand, using a quantization threshold significantly increases the overall BAR because the bins with less distinctive slopes are more likely to be discarded. Specifically, when $th = 0.015$, high BAR reaches above 90% for n_b up to 9.

High BAR alone, however, is not enough for robust and usable (fast) authentication. Increased BAR at low n_b and high th comes at the cost of reduced bit extraction rate, as shown in Fig. 9(b). Specifically, when $n_b = 4$, the extraction rate ranges from 1.3 bps to 8 bps, while $n_b = 20$ exhibits 29.3–40.0 bps. As th increases, the bit extraction rate decreases due to greater number of discarded bins. To achieve the balance between the BAR and bit extraction rate, we set $n_b = 8$ for the rest of the evaluation; this value exhibits a relatively high BAR while maintaining a reasonable bit extraction rate ranging from 8.1–16 bps.



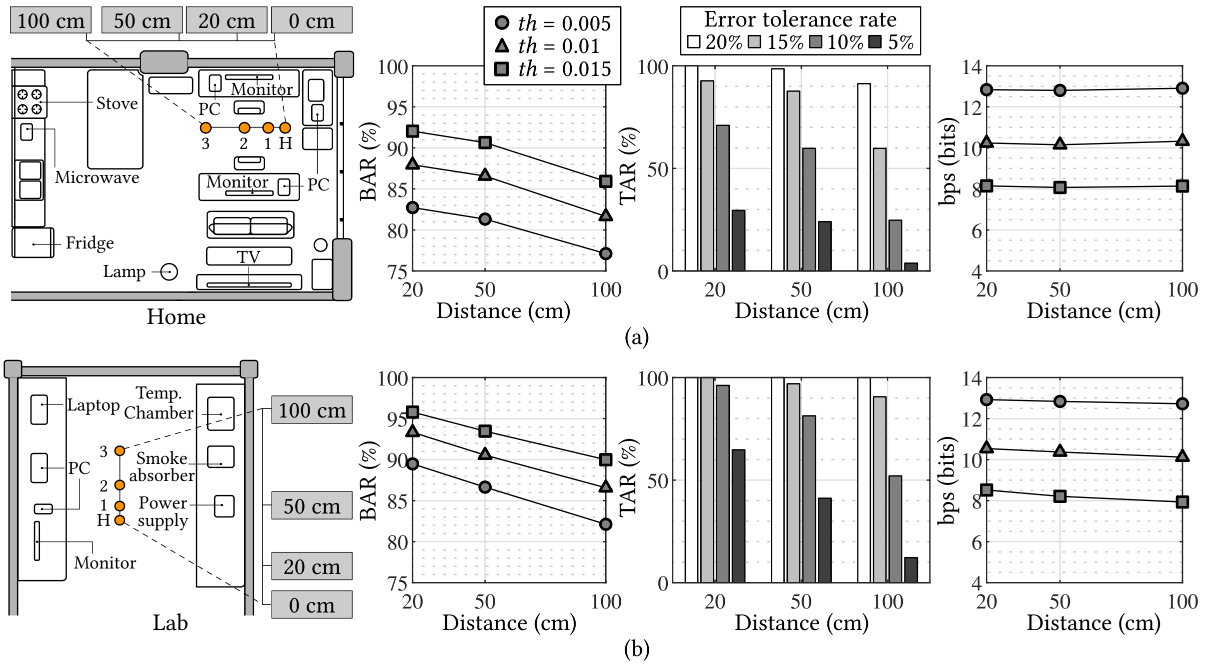
	1	2	3	4	5	6	7	8	9	10
1	91	35	44	51	50	43	53	50	48	51
2	34	92	33	46	49	51	48	54	49	49
3	43	33	90	36	45	52	50	47	52	49
4	53	44	34	91	33	47	51	48	49	49
5	50	49	46	34	91	31	44	52	48	49
6	50	49	52	46	33	93	32	46	50	51
7	52	49	50	52	44	33	93	33	46	49
8	50	54	48	46	54	41	34	92	31	47
9	46	50	50	50	47	54	45	32	91	33
10	52	48	49	48	49	49	51	46	33	91

Fig. 10. (a) 10-by-10 confusion matrix of average BAR between evidence bits generated at different cycles. (b) Distribution of BAR between diagonal and non-diagonal element pairs.

As mentioned in Section 4.5, the fuzzy commitment scheme guarantees identical key ($k = 128$ bits) reconciliation and successful authentication as long as the number of bit errors between two evidence bit sequences does not exceed $\frac{m-k}{2}$ bits. Fig. 9(c) shows the relationship between the error tolerance rate and the length of required evidence bit sequences, m. To tolerate up to 5% error between two evidence bit sequences, m needs to be at least 143 bits. To allow up to 30% error, at least 320 bits need to be extracted. While a higher tolerance rate increases the TAR between authenticating devices, it also allows adversaries to successfully authenticate to legitimate ones with fewer number of correct evidence bits. Furthermore, higher tolerance directly leads to longer measurement time due to higher number of required evidence bits. For instance, as shown in Fig. 9(d), tolerating 30% of error requires devices to execute MEAS stage for at least 39.4 s (with $th = 0.015$ and $n_b = 8$) while 5% tolerance requires only around 17.4 s. Therefore, the error tolerance rate should be kept minimal to guarantee low authentication time for a seamless user experience. In the subsequent evaluations, we vary the error tolerance rate to find the optimal balance between authentication reliability and security.

5.4 Temporal Uniqueness of Evidence Bit Sequences

While it is important for two very closely located AeroKey devices to exhibit high BAR for reliable authentication, it is also imperative that the extracted evidence bit sequences are temporally unique for secure authentication. In other words, the evidence bits obtained at different times (cycle) within the same location should be different enough to prevent adversaries from reusing the previously extracted bits. To investigate this temporal uniqueness, we deploy two devices (A and B located 20 cm apart) to periodically obtain 10 consecutive noise signals (from 10 cycles) under a daily home environment with typical usage of various surrounding electronic loads for three days. Then, all the pairs of the noise signals on the two devices are used to extract the evidence bits using previously obtained parameters of $th = 0.015$ and $n_b = 8$. In Fig. 10(a), we present a confusion matrix of average BAR achieved between evidence bits generated at varying cycles between devices, A and B. The diagonal elements, representing BAR between bits generated at the same cycles, consistently exhibit over 90% while the non-diagonal elements, representing BAR from bits generated at different cycles, show close to 50%. Note that the elements neighboring the diagonal ones exhibit mean BAR of 33.1%. While this can imply that the adversary with the noise signal from one cycle can reuse it to derive 66.9% (100%−33.1%) of the bits derived in the next single cycle by flipping the bits, only about 4 bits are extracted per cycle which is negligible considering the total length of the evidence bit sequence ($m > 128$). Furthermore, the BAR exhibits close to a random guess beyond one subsequent cycle, with mean of 49.4%, which significantly makes it difficult for the adversary to reuse the noise signal. The distribution of the rates shown in Fig. 10(b) illustrates the clear distinctions between diagonal against



(a)

(b)

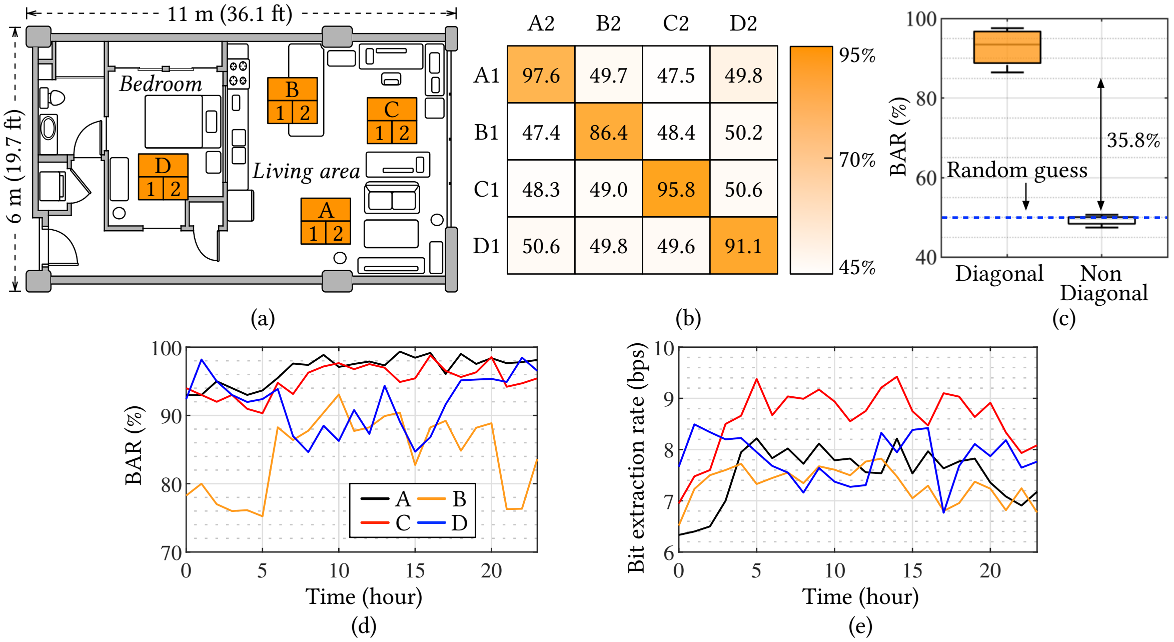
Fig. 11. BAR, TAR, and bit extraction rate with respect to varying distance between authenticating devices within (a) home and (b) lab environment.

non-diagonal elements by at least 36.1%. This result successfully demonstrates that the evidence bits (and noise signal) extracted from one cycle does not closely correlate to the ones generated at different cycle. Moreover, this temporally unique characteristic implies that the extracted bits are high in entropy, making it difficult for the adversary to reuse the previously obtained bits within the same location.

5.5 Authentication Distance

To demonstrate AeroKey's reliable authentication performance among devices within PAR, we attempt to simultaneously authenticate a set of multiple client devices to a single host device under varying distances within 100 cm. Figure 11(a) and (b) illustrate the floorplan of the deployment environments, locations and distances between each device pairs. Additionally, we present BAR (n_b = 8), bit extraction rates and TAR between each device pairs. In each scenario, over 500 authentication attempts are made over the span of three consecutive days. When evaluating TAR, we vary the error tolerance rate from 0% up to 20% while setting $th = 0.015$.

In both environments, the BAR decreases as the distance increases and, higher th leads to higher BAR. The decrease in the BAR is mainly due to the device's location variation that leads to slightly different observed noise signals between authenticating devices. In the home, with $th = 0.015$, BAR is maintained above 90% between devices located within 50 cm apart. Specifically, it achieves 92.1% and 90.6% as the distance increases to 20 and 50 cm, respectively. The high BAR at a close distance ultimately leads to high TAR as shown on the right. Specifically, when 20 cm apart, 92.6% of the attempts are successful when the error tolerance threshold is set to 15%. Even with 10% error tolerance, the success rate is as high as 70.9%. When the devices are 50 cm apart, over 98% TAR is achieved with an error tolerance of 20%. If the distance increases to 100 cm, relatively high BAR is still maintained at 85.9% with $th = 0.015$. This leads to high TAR of 91.3% with 20% of error tolerance. Authentication



(a) (b) (c)

(d) (e)

Fig. 12. (a) Four different regions of deployed device pairs. (b) Confusion matrix and (c) distribution of BAR between all evidence bit sequence pairs. (d) BAR and (e) bit extraction rate of devices within four locations with respect to different hours of the day.

attempts in the lab are more reliable than the attempts made in the home due to a fewer number of active electric loads and human activities. Within a 20 cm to 50 cm range, the devices exhibit a relatively higher BAR of over 93.4% with th = 0.015. This leads to high TAR of nearly 100% for devices within a distance of 20 cm, and 50 cm, respectively, under a 15% error tolerance rate. When the distance increases to 100 cm, a relatively high TAR is maintained with 90.6%. While the home environment achieves slightly less BAR and TAR than the lab, the bit extraction rate remains relatively equivalent. Additionally, the distance increase between the devices has no significant effect on the extraction rate since the noise signal remains relatively correlated. More specifically, in the home and lab environment, an average extraction rate of 8.2 bps and 8.3 bps is expected by the devices located inside PAR with th = 0.015. Overall, under both environments with 20% error tolerance and identical parameter settings (th = 0.015), AeroKey reliably authenticates all proximate devices with mean TAR of 98.3% within 100 cm, while maintaining similar bit extraction rates.

5.6 Spatial Variation

The size of PAR should be limited so that relatively distant devices, whether accidental or malicious, can be effectively rejected from authentication attempts. To validate this, we set a total of eight AeroKey devices within the home to attempt to authenticate with each other while deployed in four different regions that are at least 3 m apart. Over 1600 authentication attempts are made over the span of two consecutive days under typical electric load usages. Fig. 12(a) depicts four different regions: A, B, C, and D. In each region, we place a pair of devices (1 and 2) within 20 cm of each other to evaluate BAR between different regions. Figs. 12(b) and (c) illustrate the confusion matrix and the distribution of the BAR between evidence bit sequences generated by all pairs of deployed devices with $n_b = 8$ and $th = 0.015$. The diagonal elements in the matrix equate to the BAR between devices located in the same region whereas off-diagonal elements represent the BAR obtained between different regions. Clearly, devices within the same region, which are considered legitimate device pairs within PAR, show a high BAR ranging from 86.4% up to 97.6%. On the contrary, the BAR achieved between devices in different regions exhibits a mean of 49%, which differs from the legitimate pairs by at least 35.8%. This successfully indicates the spatially unique property of evidence bit sequences that can effectively allow AeroKey to distinguish proximate (legitimate) devices from the distant (adversarial) ones.

Because AeroKey strongly leverages the radiated EMR as the main source of entropy, the local environment including human activities and operating state of surrounding loads may impact the bit extraction rate, as well as BAR, achieved from devices within different locations. In Fig. 12(d) and (e), we present the BAR and bit extraction rate ($th = 0.015$) of devices within four locations with respect to varying hours of the day. Different locations exhibit slightly varying performance throughout the day depending on their surrounding load usage characteristics. For instance, devices within the bedroom (location D) suffer a slight loss in BAR and extraction rate during the daytime (6 a.m to 8 p.m) as the neighboring loads such as air purifier and cellphone charging are not active during work hours. On the contrary, devices in location B exhibit higher BAR and extraction rates during the daytime due to operating high wattage ceiling lighting nearby. The two locations A and C exhibit a relatively high and stable BAR throughout the day but the extraction rate is slightly higher during the daytime as they are surrounded by constantly running PCs, monitors and lighting. Nevertheless, regardless of time and varying human activities, devices in all four locations can experience a high BAR of above 85% and an extraction rate of above 6 bps.

5.7 Adversarial Scenarios

This section evaluates the robustness of AeroKey against the previously mentioned attack scenarios. The attacker aims to derive the legitimate device's symmetrical key by extracting similar evidence bit sequences (within the error tolerance level) using a combination of its previously observed and predicted signals. The attacker is aware of the parameter settings ($th = 0.015$ and $n_b = 8$) and is equipped with identical hardware and ADC settings as legitimate devices to further increase the attack success rate. We implement the five types of attacks (two types of ML attacks) as follows; for each attack type, over 100 authentication attempts are made:

- *Passive attack:* An adversary, just outside the boundary of a PAR (1.7 m and 2 m), attempts to authenticate in the home and lab environments in the presence of regular loads. We also investigate the effects of active high-wattage loads by activating a fan, a temperature chamber, or a smoke absorber placed within 10 cm of the victim device.
- *Replay attack:* The adversary with the knowledge of a future PAR and exact authenticating timestamp directly uses pre-measured EMR signals in the exact place at the same hour and minute of the day to authenticate itself in the home and lab environments.
- *Replay injection attack:* Replay injection attack is similar to replay attack, but the adversary activates high-wattage loads (fan, temperature chamber, and smoke absorber) within 10 cm of victim device in the lab environment.
- *Active injection attack:* An adversary activates high-wattage signal generator (with load) just outside the boundary of a PAR (1.7 m) to force a common environmental bits as the adversarial device located close (within 10 cm) to the generator. Unlike replay injection attack, the victim and the adversarial devices are time synchronized.
- *ML-raw attack:* An adversary with knowledge of the future PAR collects series of raw EMR signals over the course of one entire day and trains an ML model to use one raw 60 Hz period (input) to predict following subsequent 60 Hz (output) period. (i.e., $R_{A,1,1}$ to predict $R_{A,1,2}$). When legitimate authentication takes

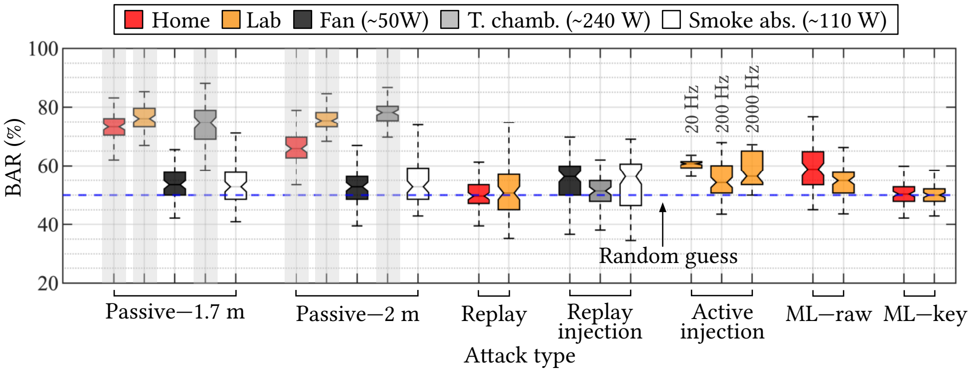
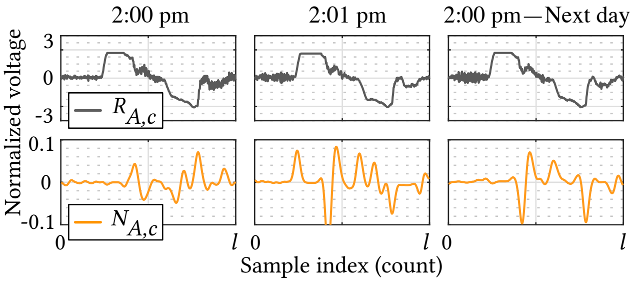


Fig. 13. BAR achieved from passive, replay, replay injection, active injection and ML attacks. Six most effective attacks (high BAR) are highlighted in gray columns.



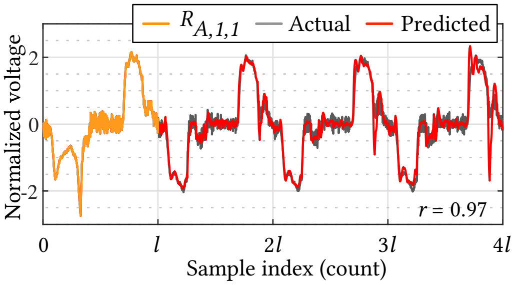


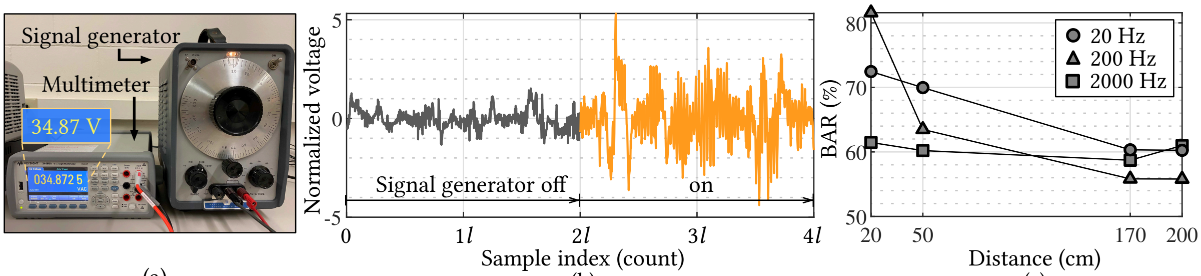
Fig. 14. Raw ($R_{A,c}$) and noise ($N_{A,c}$) signals extracted from the same location at different time instances with an operating temperature chamber nearby (replay injection attack).

Fig. 15. Measured and predicted raw signal using trained ML-model with ($R_{A,1,1}$) as an input (ML-raw attack). Two signals exhibit high correlation of $r = 0.97$.

place, the adversary uses eavesdropped $R_{A,1,1}$ and b (in SYNC and QUANT stage) to sequentially predict subsequent periods and ultimately extract evidence bits in both home and lab environment. Specifically, we use an artificial neural network (NN) composed of a one hidden layer with 100 nodes, ReLU activation function, and a softmax layer to predict subsequent raw EMR signals.

- *ML-key attack:* An adversary with the knowledge of the future PAR trains an ML model to predict the first two bits of evidence bits using $R_{A,1,1}$. The training occurs for one day and uses the trained model and eavesdropped $R_{A,1,1}$ to predict the first two bits of legitimate evidence bits the next day in the home and lab environment. We use a NN with one hidden layer, consisting of 28 nodes and an output layer with two nodes to predict the first two bits.

The distribution of the BAR achieved from all the attacks is shown in Fig. 13. In the passive attack at 1.7 m, mean of 72.8% and 76.0% of evidence bits can be derived by the adversary in the home and lab, respectively, under regular load usage. When the temperature chamber is turned on, the BAR remains relatively similar at 72.8%. On the other hand, when the fan or smoke absorber is turned on, they produce a unique EMR signal that does not propagate such a far distance, leading the attacks to achieve significantly lower BAR of 54.0% and 54.9%, respectively. Compared to the attacks from 1.7 m, attacks from 2 m achieve lower mean BAR of 66.0%



(a) (b) (c)

Fig. 16. (a) Signal generator (Hewlett-Packard audio oscillator 200AB) outputting 34.87 V AC signal. (b) Noise signal, N_A, observed from the victim device as the signal generator is turned off and on (~2000 Hz). (c) BAR achieved from the devices located in different distances from the generator signaling different frequency components.

and 75.9% in the home and lab, respectively, due to greater spatial variability of ambient EMR. Replay attack and replay injection attack result in a low BAR (close to random guess), ranging from 49.4% to 55.4%, either with or without active loads. This is because the raw EMR signal, $R_{A,c}$, even with its unique dominant structure, has temporal variability as illustrated in Fig. 14. The PPFE stage of AeroKey essentially removes this dominant structure, leaving the noise signal, $N_{A,c}$, with just temporally varying noise signals. Even when the high-power consuming temperature chamber is operating nearby, $N_{A,c}$, extracted from different time instant of one minute apart within the same day (2:00 p.m. and 2:01 p.m.) differs significantly. Further, the same time on the next day exhibits different noise signal even under identical surrounding appliances. This effectively prevents an adversary from reusing the previously derived (at same hour and minute under the same load on a prior day) evidence bit sequences.

The active injection attack with varying frequency range is also not very effective in forcing enough environmental bits to the victim device, achieving mean BAR of 60.3%, 55.7% and 58.9% at output frequency of 20 Hz, 200 Hz and 2000 Hz, respectively at distance of 170 cm. Fig. 16(a) illustrates our attack setup that outputs 35.87 V sine wave at different frequency. As shown in Fig. 16(b), when the signal generator is turned on, the victim device located at 1.7 m away observes significantly varying noise signal with high amplitude. However, the generated EMR signal does not identically propagate through environment or gets measured by the distant devices due to EMR signal rapidly dissipating with the increasing distance. Even for a pair of devices located close to the generator within 20 cm, the mean BAR decreases to 72.4%, 61.4% and 81.5%, respectively for increasing frequency, which exhibits less BAR than passive attack without signal generator in the same environment. Similarly, ML-based attacks are not favorable to the adversary. ML-raw attack performs close to a random guess, achieving only 58.3% and 54.2% BAR in home and lab, respectively. As illustrated in Fig. 15, although the ML model is able to closely predict the general structure of the subsequent raw signals with a correlation coefficient of 0.97, it was not effective in accurately predicting high frequency noise portion of the signal that AeroKey uses to extract bit sequences. Additionally, the small error from the early period number (i.e., $p = 2$) propagates to prevent reliable prediction of the later periods (i.e., $p = 10$). Lastly, the ML-key attack was also not sufficient enough to predict enough legitimate bits with its mean BAR of 49.9% and 49.8% in home and lab, respectively. While earlier period number can be assumed to have more impact on the first few bits of the evidence bit sequences, the model fails to capture enough information about the noise signal. This demonstrates that the adversary, even with the eavesdropped $R_{A,1,1}$, cannot acquire much information about the evidence bit sequence itself.

Overall, the passive attacks are the most successful in obtaining relatively higher portions of legitimate evidence bits, because the timestamp of the attacks is synchronized to the legitimate authenticating devices. In Fig. 17(a),

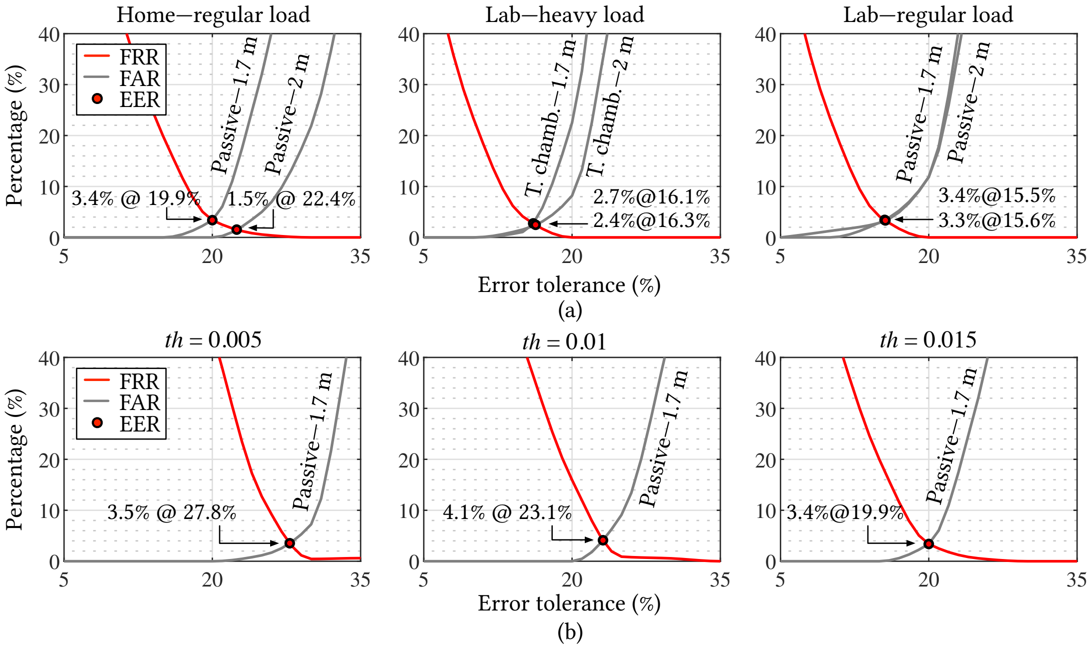


Fig. 17. (a) Resulting EER from six most effective passive attacks. (b) EER from varying *th* by passive attack within home environment.

we show the EER of the six most effective passive attacks: at 1.7 m and 2 m range in the home and lab (with and without an operating temperature chamber) environments; all other attacks exhibit EER of 0% due to low BAR. The home and lab environment accomplish similar EER of at most 3.4% under regular load usage. Compared to the attacks at 2 m distance, attacks from 1.7 m exhibit higher EER at lower error tolerance rates. Because home is less permissive in authenticating devices compared to the lab, EER of 3.4% occurs at 19.9% error tolerance rate, which is comparatively higher than that of the lab, which occurs at 15.5%. In the lab, when the temperature chamber is operating nearby a legitimate device, a relatively lower EER of 2.7% and 2.4% is achieved by the same attacks. The above observation suggests that depending on the context of the environment, the user may manually (one-time configuration) vary the error tolerance rate (19.9% for home and 15.5% for lab) to strike an optimal balance between usability and security. Nevertheless, selecting the lower threshold of 15.5% can effectively reject 96.6% of the passive attacks while maintaining over 80% TAR in both environments.

We next evaluate the optimal error tolerance rate for varying *th* within the home environment against passive attack from 1.7 m. As illustrated in Fig. 17(b), varying *th* results in different FRR and FAR due to lower *th* exhibiting lower BAR. Ultimately, this leads to different security and usability balancing point (EER) at a different error tolerance rate; higher *th* results in a lower rate. Specifically, the tolerance rate of the EER is achieved at 27.8%, 23.1% and 19.9% for increasing *th* of 0.005, 0.01 and 0.015, respectively, while maintaining a low EER of less than 4.1%. This sets the guideline error tolerance for varying *th*, which is particularly related to the usability aspect of AeroKey. While lower *th* results in a higher bit extraction rate, the requirement of a higher error tolerance rate increases the number of extracted bits to derive identical final keys on both devices. In further evaluation, we determine the optimal *th* to achieve the fastest authentication time.

Table 2. Measurement time required for varying *th* in home and lab.

	Home			Lab		
th	0.005	0.01	0.015	0.005	0.01	0.015
Error tolerance rate (%)	27.8	23.1	19.9	23.3	19.1	15.5
m (bits)	287	238	213	240	207	186
bit extraction rate (bps)	13.0	10.4	8.2	12.9	10.3	8.3
Measurement time (s)	22.0	22.8	25.9	18.6	20.0	22.4

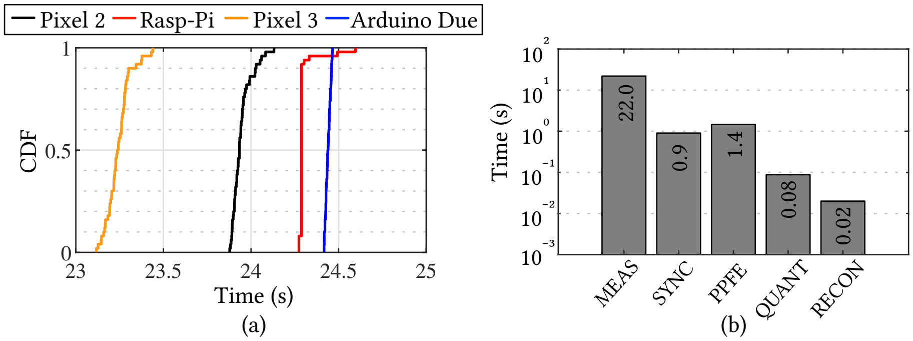


Fig. 18. (a) Total authentication time measured on four different devices. (b) Execution time of each AeroKey stages on Arduino Due (log scale).

5.8 Authentication Time and Energy Consumption

We next investigate the authentication time required for AeroKey to extract a 128-bit final key, *K*. As previously presented, each *th* requires a different optimal error tolerance rate that leads to varying *m*. Table 2 presents the required measurement time (time required to run MEAS stage) as well as the length of the evidence bit sequence, *m*, for corresponding *th* based on its bit extraction rate. As *th* increases from 0.005 to 0.015, an increase in the BAR and lower required error tolerance rate leads to shorter *m* in both environments. Specifically, in home, *m* reduces from 287 to 213, respectively. However, contrary to the shorter length of *m*, the overall time spent on the MEAS stage increases due to lower bit extraction rate on higher *th*; approximately 22.0 s up to 25.9 s of measurement time is necessary. In lab, a relatively lower number of bits are required that ultimately results in a measurement time of 18.6 s to 22.4 s as *th* increases.

To further accurately estimate realistic overall authentication time accounting for computation, we implement AeroKey on four different devices: Google Pixel 2 (2.35-GHz), Google Pixel 3 (2.5-GHz), Raspberry Pi 4 (1.5-GHz) and Arduino Due (84-MHz). On each device, we measure the authentication time 50 times with $m = 287$ and $th = 0.005$ which exhibits the lowest authentication time among all *th* in home environment. As Fig. 18(a) shows, Pixel 3 achieves the fastest mean authentication time of 23.2 s. While Pixel 2 and Raspberry Pi 4 exhibit a similar mean time of 23.9 s and 24.2 s, respectively, Arduino due exhibits the slowest mean authentication time of 24.4 s due to the relatively slower processor speed. The execution time of each AeroKey stage running on Arduino Due is illustrated in Fig. 18(b). The MEAS stage accounts for a majority of the authentication time (22.0 s), which is identical across all devices. The SYNC and PPFE stage takes 0.9 s and 1.4 s, respectively, due to the intensive computation (DTW) and filtering algorithms. The remaining stages (QUANT, and RECON) make up a negligible

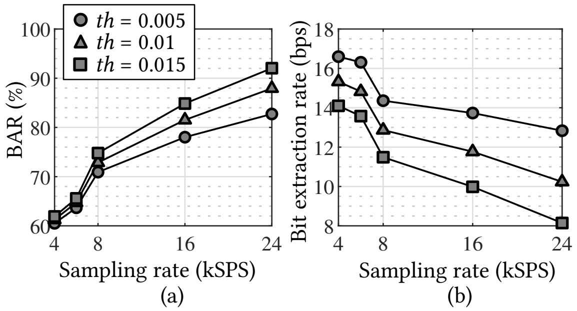


Fig. 19. (a) BAR and (b) bit extraction rate under varying sampling rate between two closely located devices.

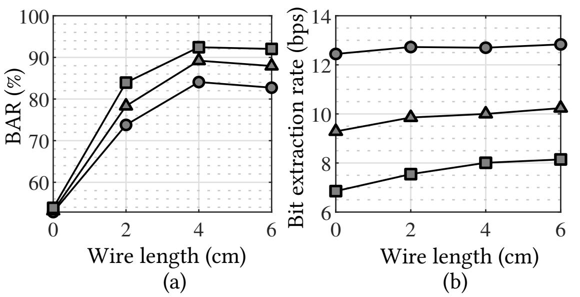


Fig. 20. (a) BAR and (b) bit extraction rate under varying wire length between two closely located devices.

portion of less than 0.1 s in total. Overall, the results suggest that the authentication can take place within 24.5 s on various processors.

Next, to show the feasibility of running AeroKey on various battery-operated mobile devices, we investigate the energy consumption of four major stages (SYNC, PFFE, QUANT and RECON) on Arduino Due that exhibits the slowest authentication time. Because the MEAS stage merely utilizes the on board ADC, the energy consumption is negligible compared to running other stages (0.5 mJ, assuming 10-bit resolution at 24 kSPS). In order to measure accurate power usage, we subtract the idle power from the power readings when running four stages. On average, SYNC, PFFE, QUANT and RECON stages consume 13 mW of power. This leads to a total energy consumption of approximately 30 mJ, which makes AeroKey suitable for running under a modern device's battery capacity (i.e., a typical smartphone battery can hold 40 kJ

5.9 Device Configurations

We next investigate the effect of the device configuration, including the ADC's sampling rate and length of the conductor wire connected to the input pin. We downsample the measured raw signals of two closely located devices (20 cm apart) from 24 kSPS down to 4 kSPS and compare its BAR and bit extraction rates. As Fig. 19(a) illustrates, at 24 kSPS and 16 kSPS, BAR is over 80% for $th \geq 0.01$, and falls below for 8 kSPS and 4 kSPS. To maintain BAR above 80% for reliable key reconciliation, the sampling rate should be at least 16 kSPS, which can easily be achieved by low-cost MCUs. In Fig. 19(b), we illustrates the affects on the bit extraction rate. As the accuracy of the measured signal increases with a higher ADC sampling rate, it tends to pick up more noise and high frequency components that decrease the bit extraction rate. Nevertheless, 16 kSPS and 24 kSPS range that exhibits reasonable BAR can expect bit extraction rate above 8.1 bps regardless of th.

Next, we examine the impact of the conductor wire's length. This is particularly important when AeroKey is to be implemented in various mobile devices with small form factor constraints. As shown in Figure 20(a), BAR is above 80% when the wire length is 4 cm or 6 cm for all $th \geq 0.005$. If the length decreases to 2 cm, BAR slightly drops, and when there is no wire connected to the ADC, significant degradation in BAR occurs to less than 60%. Therefore, to maintain high BAR, the conductor wire should be at least 4-cm long. As shown in Figure 20(b), contrarily to the sampling rate variation which is more sensitive to pick up high frequency signal, the wire length does not significantly impact the bit extraction rate. The length of 4 cm up to 6 cm can experience above 8 bps regardless of th.

6 DISCUSSION

In this section, we discuss some constraints of the current design of AeroKey and the ways to improve it.

Table 3. NIST test results ($p \geq 0.05$)

Test	p-value	Test	p-value
Frequency	0.3504	FFT	0.2133
Block frequency	0.5341	Nonperiodic template matching	0.7399
Cumulative sums	0.0668	Overlapping template matching	0.3504
Runs	0.3504	Serial	0.9114
Longest runs of ones	0.1223	Linear complexity	0.3504
Rank	0.3504		

Authentication time: Using a commercial off-the-shelf MCU with no hardware modifications, a pair of AeroKey-enabled devices can successfully authenticate each other within 24.5 s. The majority of this time is spent on the MEAS stage, and the bottleneck that constrains the overall authentication time of the system is mainly attributed to two reasons. First, because AeroKey targets low-cost IoT devices, typically with an on-chip ADC (low sampling frequency and resolution) with no hardware modifications (amplifier and analog front-end circuitry), the measured raw signal is very noisy and does not highly correlate between two devices. To solve this, MEAS stage has to collect an enough number of 60 Hz periods within each cycle for the PPFE stage to extract a highly correlating noise signal between two close devices as shown in Fig. 7(a). Secondly, to guarantee a robust security of 128-bit final key, the length of the environmental bit sequences needs to be significantly longer (over 200 bits) due to RECON stage that incurs some entropy loss. Nevertheless, if AeroKey is used for simple device pairing purposes (e.g., Bluetooth pairing mechanism requires two devices to agree on a 32-bit pin), users can expect to pair the devices with a relatively fast pairing time of under 5 s, regardless of th. This is significantly faster compared to the usability study that presented average typical pairing time of 27 s to copy and confirm 8-digit pin in IoT devices with limited user interface [41].

Randomness of generated bits: Just like every other cryptographic bit sequence or key, it is important for the AeroKey-generated evidence bits to be random and unpredictable. To examine the randomness, we conduct the NIST statistical suites test [2] on the generated bit sequences with a length of over one million bits. The NIST test consists of 15 randomness tests, each with an output p-value. Table 3 presents AeroKey's NIST test results. AeroKey passes 11 out of 15 tests with p-value greater than 0.05. However, to be considered as a true random number generating source, AeroKey should pass all 15 tests with $p \geq 0.05$. Therefore, further random bit corrections are necessary to produce higher quality evidence bit sequences [15, 43].

Radius of PAR: Our evaluation suggests that the typical range of the PAR can be as large as one meter. The concept of Transitivity of Trust (ToT) protocol [12] can be applied to increase the radius and authenticate devices within the entire household. In such case, the chain of authentication takes place through bridging devices deployed within different rooms. On the other hand, the user may wish to further decrease the range to be resilient against dedicated adversary. In these cases, AeroKey can be configured with a higher th and/or with a lower error tolerance rate to the point where the user feels comfortable with the tuned authentication range.

Outdoor applicability: The underlying assumption of the AeroKey protocol is that authenticating devices are located indoors where EMR is observable. Therefore, traditional password-based authentication should be used to authenticate devices in outdoor scenarios (e.g., in the vehicle or on the street).

Orientation of antenna: Our experiments are conducted in the setting where the conductive wires connected to the authenticating MCUs are oriented parallel to one another. However, because the EMR pattern propagates in three orthogonal planes (x, y and z) [40], the authenticating devices may experience signal correlation issues when one device is not properly oriented with respect to another authenticating device. To potentially deal with this issue, multiple orthogonally aligned antenna (with PCB trace or in the form of patch antenna) can be designed to receive spherical radiation pattern (isotropic measurement) while preserving the form-factor of the devices.

7 CONCLUSION

The ever-growing IoT ecosystem is demanding secure yet usable device authentication methods to ensure trustworthy wireless connectivity between a large number of small, low-cost and low-power devices. Context-based authentication is a promising solution to achieve both security and usability by generating random bit sequences to be used as a security key from an ambient source of randomness.

In this work, we present AeroKey, a secure and usable device authentication scheme leveraging spatiotemporally unique ambient EMR signature. This is the first work that realizes context-based authentication without any dedicated sensor, which makes it favorable for application in virtually any IoT device. With novel pre-processing and feature extraction techniques, AeroKey allows co-located devices to autonomously generate near-identical and high-entropy evidence bit sequences from noisy measurement signals on off-the-shelf MCUs with zero hardware modifications. Our evaluation successfully demonstrated high authentication success rates between devices within a small PAR deployed in home and lab environments with relatively low authentication time of as low as 24 s. We also showed its robustness against various adversaries leveraging time, space, electrical appliances and ML models with a low EER of 3.4% or less under different environments. To summarize, AeroKey enables low-cost IoT and mobile devices to periodically (re-)authenticate themselves to facilitate secure and usable wireless connection with no human involvement.

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their constructive feedback that has made the work stronger. This work was supported in part by NSF grants CNS-1845469 and CNS-2003129 and the Federal Highway Administration of United States Department of Transportation under contract number 693JJ318C000025.

REFERENCES

[1] Imtiaj Ahmed, Yina Ye, Sourav Bhattacharya, N. Asokan, Giulio Jacucci, Petteri Nurmi, and Sasu Tarkoma. 2015. Checksum Gestures: Continuous Gestures as an out-of-Band Channel for Secure Pairing. In *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '15)*. 391–401. https://doi.org/10.1145/2750858.2807521

[2] Lawrence E. Bassham, III, Andrew L. Rukhin, Juan Soto, James R. Nechvatal, Miles E. Smid, Elaine B. Barker, Stefan D. Leigh, Mark Levenson, Mark Vangel, David L. Banks, Nathanael Alan Heckert, James F. Dray, and San Vo. 2010. *SP 800-22 Rev. 1a. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications.* Technical Report.

[3] Donald J. Berndt and James Clifford. 1994. Using Dynamic Time Warping to Find Patterns in Time Series. In *Proceedings of the International Conference on Knowledge Discovery (AAAIWS '94)*. 359–370.

[4] Ke-Yu Chen, Sidhant Gupta, Eric C. Larson, and Shwetak Patel. 2015. DOSE: Detecting user-driven operating states of electronic devices from a single sensing point. In *2015 IEEE International Conference on Pervasive Computing and Communications (PerCom'15)*. 46–54. https://doi.org/10.1109/PERCOM.2015.7146508

[5] Yushi Cheng, Xiaoyu Ji, Juchuan Zhang, Wenyuan Xu, and Yi-Chao Chen. 2019. DeMiCPU: Device Fingerprinting with Magnetic Signals Radiated by CPU. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security (CCS '19)*. 1149–1170. https://doi.org/10.1145/3319535.3339810

[6] Miro Enev, Sidhant Gupta, Tadayoshi Kohno, and Shwetak N. Patel. 2011. Televisions, Video Privacy, and Powerline Electromagnetic Interference. In *Proceedings of the 18th ACM Conference on Computer and Communications Security (CCS '11)*. 537–550. https://doi.org/10.1145/2046707.2046770

[7] Mohammed Ferdjallah and Ronald E. Barr. 1994. Adaptive Digital Notch Filter Design on the Unit Circle for the Removal of Powerline Noise from Biomedical Signals. *IEEE Transactions on Biomedical Engineering* 41, 6 (June 1994), 529–536. https://doi.org/10.1109/10.293240

[8] Mikhail Fomichev, Max Maass, Lars Almon, Alejandro Molina, and Matthias Hollick. 2019. Perils of Zero-Interaction Security in the Internet of Things. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 1, Article 10 (March 2019), 38 pages. https://doi.org/10.1145/3314397

[9] Carlos Garrido, Antonio F. Otero, and Jose Cidras. 2003. Low-frequency Magnetic Fields from Electrical Appliances and Power Lines. *IEEE Transactions on Power Delivery* 18, 4 (October 2003), 1310–1319. https://doi.org/10.1109/TPWRD.2003.817744

[10] Core Specification Working Group. 2019. *Bluetooth Core Specification*. Rev. 5.2.

[11] Bogdan Groza, Adriana Berdich, Camil Jichici, and Rene Mayrhofer. 2020. Secure Accelerometer-Based Pairing of Mobile Devices in Multi-Modal Transport. *IEEE Access* 8 (January 2020), 9246–9259. https://doi.org/10.1109/ACCESS.2020.2964151

[12] Jun Han, Albert Jin Chung, Manal Kumar Sinha, Madhumitha Harishankar, Shijia Pan, Hae Young Noh, Pei Zhang, and Patrick Tague. 2018. Do You Feel What I Hear? Enabling Autonomous IoT Device Pairing Using Different Sensor Types. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P '18)*. 836–852. https://doi.org/10.1109/SP.2018.00041

[13] Microchip Technology Inc. 2018. SAM D5x/E5x Family Data Sheet. http://ww1.microchip.com/downloads/en/DeviceDoc/60001507C.pdf

[14] Wenqiang Jin, Ming Li, Srinivasan Murali, and Linke Guo. 2020. Harnessing the Ambient Radio Frequency Noise for Wearable Device Pairing. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS '20)*. 1135–1148. https://doi.org/10.1145/3372297.3417288

[15] Nick Jones and LluÍs Masanes. 2008. Key Distillation and the Secret-Bit Fraction. *IEEE Transactions on Information Theory* 54, 2 (2008), 680–691. https://doi.org/10.1109/TIT.2007.913264

[16] Ari Juels and Martin Wattenberg. 1999. A Fuzzy Commitment Scheme. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS '99)*. 28–36. https://doi.org/10.1145/319709.319714

[17] Nikolaos Karapanos, Claudio Marforio, Claudio Soriente, and Srdjan Capkun. 2015. Sound-Proof: Usable Two-Factor Authentication Based on Ambient Sound. In *Proceedings of the USENIX Security Symposium (USENIX Security '15)*. 483–498. https://www.usenix.org/conference/usenixsecurity15/technical-sessions/presentation/karapanos

[18] Kyuin Lee, Neil Klingensmith, Suman Banerjee, and Younghyun Kim. 2019. VoltKey: Continuous Secret Key Generation Based on Power Line Noise for Zero-Involvement Pairing and Authentication. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 3, Article 93 (September 2019), 26 pages. https://doi.org/10.1145/3351251

[19] Kyuin Lee, Neil Klingensmith, Dong He, Suman Banerjee, and Younghyun Kim. 2020. ivPair: Context-Based Fast Intra-Vehicle Device Pairing for Secure Wireless Connectivity. In *Proceedings of the Conference on Security and Privacy in Wireless and Mobile Networks (WiSec '20)*. 25–30. https://doi.org/10.1145/3395351.3399436

[20] Xiaopeng Li, Qiang Zeng, Lannan Luo, and Tongbo Luo. 2020. T2Pair: Secure and Usable Pairing for Heterogeneous IoT Devices. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS '20)*. 309–323. https://doi.org/10.1145/3372297.3417286

[21] Yang Li, Rui Tan, and David K. Y. Yau. 2017. Natural Timestamping Using Powerline Electromagnetic Radiation. In *Proceedings of the International Conference on Information Processing in Sensor Networks (IPSN '17)*. 55–66. https://doi.org/10.1145/3055031.3055075

[22] Qi Lin, Weitao Xu, Guohao Lan, Yesheng Cui, Hong Jia, Wen Hu, Mahbub Hassan, and Aruna Seneviratne. 2020. KEHKey: Kinetic Energy Harvester-Based Authentication and Key Generation for Body Area Network. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 1, Article 41 (March 2020), 26 pages. https://doi.org/10.1145/3381754

[23] Qi Lin, Weitao Xu, Jun Liu, Abdelwahed Khamis, Wen Hu, Mahbub Hassan, and Aruna Seneviratne. 2019. H2B: Heartbeat-based Secret Key Generation Using Piezo Vibration Sensors. In *Proceedings of the International Conference on Information Processing in Sensor Networks (IPSN '19)*. 265–276. https://doi.org/10.1145/3302506.3310406

[24] Mark Loveless. 2018. Understanding Bluetooth Security. https://duo.com/decipher/understanding-bluetooth-security.

[25] Suhas Mathur, Robert Miller, Alexander Varshavsky, Wade Trappe, and Narayan Mandayam. 2011. ProxiMate: Proximity-based Secure Pairing Using Ambient Wireless Signals. In *Proceedings of the International Conference on Mobile Systems, Applications, and Services (MobiSys '11)*. 211–224. https://doi.org/10.1145/1999995.2000016

[26] Markus Miettinen, Nadarajah Asokan, Thien Duc Nguyen, Ahmad-Reza Sadeghi, and Majid Sobhani. 2014. Context-Based Zero-Interaction Pairing and Key Evolution for Advanced Personal Devices. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS '14)*. 880–891. https://doi.org/10.1145/2660267.2660334

[27] Markus Miettinen, Thien Duc Nguyen, Ahmad-Reza Sadeghi, and N. Asokan. 2018. Revisiting Context-based Authentication in IoT. In *Proceedings of the Design Automation Conference (DAC '18)*. Article 32, 6 pages. https://doi.org/10.1145/3195970.3196106

[28] Shijia Pan, Carlos Ruiz, Jun Han, Adeola Bannis, Patrick Tague, Hae Young Noh, and Pei Zhang. 2018. UniverSense: IoT Device Pairing through Heterogeneous Sensing Signals. In *Proceedings of the International Workshop on Mobile Computing Systems and Applications (HotMobile '18)*. 55–60. https://doi.org/10.1145/3177102.3177108

[29] Shwetak N. Patel, Thomas Robertson, Julie A. Kientz, Matthew S. Reynolds, and Gregory D. Abowd. 2007. At the Flick of a Switch: Detecting and Classifying Unique Electrical Events on the Residential Power Line. In *Proceedings of the 9th International Conference on Ubiquitous Computing (UbiComp '07)*. 271–288. http://dl.acm.org/citation.cfm?id=1771592.1771608

[30] Marc Roeschlin, Ivan Martinovic, and Kasper Rasmussen. 2018. Device Pairing at the Touch of an Electrode. In *Network and Distributed Systems Security Symposium 2018 (NDSS '18)*. 1–15. https://doi.org/10.14722/ndss.2018.23076

[31] Masoud Rostami, Ari Juels, and Farinaz Koushanfar. 2013. Heart-to-Heart (H2H): Authentication for Implanted Medical Devices. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS '13)*. 1099–1112. https://doi.org/10.1145/

2508859.2516658

[32] Anthony Rowe, Vikram Gupta, and Ragunathan (Raj) Rajkumar. 2009. Low-Power Clock Synchronization Using Electromagnetic Energy Radiating from AC Power Lines. In *Proceedings of the ACM Conference on Embedded Networked Sensor Systems (SenSys '09)*. 211–224. https://doi.org/10.1145/1644038.1644060

[33] Nitesh Saxena, Jan-Erik Ekberg, Kari Kostiainen, and N. Asokan. 2006. Secure Device Pairing Based on a Visual Channel. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P '06)*. 308–313. https://doi.org/10.1109/SP.2006.35

[34] Dominik Schürmann and Stephan Sigg. 2013. Secure Communication Based on Ambient Audio. *IEEE Transactions on Mobile Computing* 12, 2 (February 2013), 358–370. https://doi.org/10.1109/TMC.2011.271

[35] Bluetooth SIG. 2020. Bluetooth SIG Statement Regarding the Exploiting Cross-Transport Key Derivation in Bluetooth Classic and Bluetooth Low Energy (BLURtooth) and the Security implications of key conversion between BR/EDR and BLE Vulnerabilities. https://www.bluetooth.com/learn-about-bluetooth/bluetooth-technology/bluetooth-security/blurtooth/.

[36] Mike Silva, Norm P. Hummon, D. Rutter, and C. Hooper. 1989. Power Frequency Magnetic Fields in the Home. *IEEE Transactions on Power Delivery* 4, 1 (January 1989), 465–478.

[37] Artur Souza, Ivo Carlson, Heitor Ramos, Antônio Loureiro, and Leonardo Oliveira. 2020. Internet of Things device authentication via electromagnetic fingerprints. *Engineering Reports* 2 (07 2020). https://doi.org/10.1002/eng2.12226

[38] Kaixin Sui, Mengyu Zhou, Dapeng Liu, Minghua Ma, Dan Pei, Youjian Zhao, Zimu Li, and Thomas Moscibroda. 2016. Characterizing and Improving WiFi Latency in Large-Scale Operational Networks. In *Proceedings of the International Conference on Mobile Systems, Applications, and Services (MobiSys '16)*. 347–360. https://doi.org/10.1145/2906388.2906393

[39] Bharath Sundararaman, Ugo Buy, and Ajay D. Kshemkalyani. 2005. Clock Synchronization for Wireless Sensor Networks: a Survey. *Ad Hoc Networks* 3, 3 (May 2005), 281–323. https://doi.org/10.1016/j.adhoc.2005.01.002

[40] Santi Tofani, Piero Ossola, Giovanni d'Amore, L Anglesio, Motohisa Kanda, and David R Novotny. 1996. A Three-loop Antenna System for Performing Near-Field Measurements of Electric and Magnetic Fields from Video Display Terminals. *IEEE Transactions on electromagnetic compatibility* 38, 3 (1996), 341–347. https://doi.org/10.1109/15.536064

[41] Ersin Uzun, Kristiina Karvonen, and N. Asokan. 2007. Usability Analysis of Secure Pairing Methods. In *Financial Cryptography and Data Security*. 307–324.

[42] Alex Varshavsky, Adin Scannell, Anthony LaMarca, and Eyal de Lara. 2007. Amigo: Proximity-based Authentication of Mobile Devices. In *Proceedings of the International Conference on Ubiquitous Computing (UbiComp '07)*. 253–270. https://doi.org/10.1007/978-3-540-74853-3_15

[43] Jack West, Kyuin Lee, Suman Banerjee, Younghyun Kim, George K. Thiruvathukal, and Neil Klingensmith. 2021. Moonshine: An Online Randomness Distiller for Zero-Involvement Authentication. In *Proceedings of the 20th International Conference on Information Processing in Sensor Networks (IPSN '21)*. 93–105. https://doi.org/10.1145/3412382.3458899

[44] Weitao Xu, Guohao Lan, Qi Lin, Sara Khalifa, Neil Bergmann, Mahbub Hassan, and Wen Hu. 2017. KEH-Gait: Towards a Mobile Healthcare User Authentication System by Kinetic Energy Harvesting. In *Network and Distributed Systems Security Symposium 2017 (NDSS '17)*. 1–15. https://doi.org/10.14722/ndss.2017.23023

[45] Weitao Xu, Guohao Lan, Qi Lin, Sara Khalifa, Mahbub Hassan, Neil Bergmann, and Wen Hu. 2019. KEH-Gait: Using Kinetic Energy Harvesting for Gait-based User Authentication Systems. *IEEE Transactions on Mobile Computing* 18, 1 (2019), 139–152. https://doi.org/10.1109/TMC.2018.2828816

[46] Weitao Xu, Girish Revadigar, Chengwen Luo, Neil Bergmann, and Wen Hu. 2016. Walkie-Talkie: Motion-Assisted Automatic Key Generation for Secure on-Body Device Communication. In *Proceedings of the International Conference on Information Processing in Sensor Networks (IPSN '16)*. Article 3, 12 pages. https://doi.org/10.1109/IPSN.2016.7460726

[47] Zhenyu Yan, Qun Song, Rui Tan, Yang Li, and Adams Wai Kin Kong. 2019. Towards Touch-to-Access Device Authentication Using Induced Body Electric Potentials. In *Proceedings of the Annual International Conference on Mobile Computing and Networking (MobiCom '19)*. Article 23, 16 pages. https://doi.org/10.1145/3300061.3300118

[48] Fangfang Yang, Mohammad A. Islam, and Shaolei Ren. 2020. PowerKey: Generating Secret Keys from Power Line Electromagnetic Interferences. In *Processing of International Conference on Network and System Security (NSS '20)*. 354–370.

[49] Lin Yang, Wei Wang, and Qian Zhang. 2016. Secret from Muscle: Enabling Secure Pairing with Electromyography. In *Proceedings of the ACM Conference on Embedded Network Sensor Systems (SenSys '16)*. 28–41. https://doi.org/10.1145/2994551.2994556

A ACRONYMS AND NOMENCLATURE

Table 4. Acronyms.

Acronym	Phrase
EMR	Electromagnetic radiation
PAR	Personal authenticated region
MCU	Microcontroller
ADC	Analog to digital converter
DTW	Dynamic time warping
BAR	Bit agreement rate
EER	Equal error rate
TAR	True acceptance rate
FAR	False acceptance rate
FRR	False rejection rate

Table 5. Nomenclature.

Acronym	Definition
d	Device name (A or B)
c	Cycle count $(1, 2, \ldots, n_c)$
n_c	Number of cycles
p	Period count within cycle $(1, 2, \ldots, n_p)$
n_p	Number of periods per cycle
$R_{d,c,p}$	Raw measurement of device d in cycle c, period p
$M_{d,c}$	mean signal of device d in cycle c
$N_{d,c}$	noise signal of device d in cycle c
l	Agreed sample length of noise signal
b	Bin count within cycle $(1, 2, \ldots, n_b)$
n_b	Number of bins per cycle
$S_{d,c,b}$	Slopes of bin b of noise $N_{d,c}$
th	Quantization threshold
$B_{d,c,b}$	Evidence bit sub-sequence in device d, cycle c, bin b
B_d	Evidence bit sequence extracted by device d
m	Length of desired number of bits for B_d
K_d	Final authentication key on device d
k	length of final authentication key K_d