Hardening and Speeding Up Zero-interaction Pairing and Authentication

Mikhail Fomichev
Technical University of Darmstadt
mfomichev@seemoo.tu-darmstadt.de

Timm Lippert
Technical University of Darmstadt
timm.lippert@gmail.com

Matthias Hollick
Technical University of Darmstadt
mhollick@seemoo.tu-darmstadt.de

Abstract

Establishing and maintaining secure communications in the Internet of Things (IoT) is vital to protect smart devices. *Zero-interaction pairing (ZIP)* and *zero-interaction authentication (ZIA)* enable IoT devices to establish and maintain secure communications without user interaction by utilizing devices' *ambient context*, e.g., audio. For autonomous operation, ZIP and ZIA require the context to have enough entropy to resist attacks and complete in a timely manner. Despite the low-entropy context being the norm, like inside an unoccupied room, the research community has yet to come up with ZIP and ZIA schemes operating under such conditions. We propose HARDZIPA, a novel approach that turns commodity IoT actuators into injecting devices, generating high-entropy context. Here, we combine the capability of IoT actuators to impact the environment, e.g., emitting a sound, with a pseudorandom number generator (PRNG) featured by many actuators to craft hard-to-predict context stimuli. To demonstrate the feasibility of HARDZIPA, we implement it on off-the-shelf IoT actuators, i.e., *smart speakers*, *lights*, and *humidifiers*. We comprehensively evaluate HARDZIPA, collecting over *80 hours* of various context data in real-world scenarios. Our results show that HARDZIPA is able to thwart *advanced active attacks* on ZIP and ZIA schemes, while doubling the amount of context entropy in many cases, which allows *two times faster* pairing and authentication.

Categories and Subject Descriptors

K.6.5 [**Management of Computing and Information Systems**]: Security and Protection

General Terms

Security, Design, Experimentation, Data collection

Keywords

Zero-interaction, context-based security, pairing, authentication, sensing, Internet of Things

1 Introduction

The proliferation of the Internet of Things (IoT) urges the need to *establish and maintain* secure communications between smart devices to protect not only the data, which they exchange, like sensor readings, but also devices themselves, e.g., from unauthorized access. This ensures the user *privacy* and *trustworthiness* of IoT systems [10, 15, 17, 38]. Device *pairing* and *authentication* are approaches to (1) establish a secure communication channel between two devices, and (2) maintain it afterwards. By means of pairing two unassociated devices derive a *shared secret key* without any trusted third party, bootstrapping their secure channel [9, 42], while authentication allows one device to assure the legitimacy of another device on such a channel [6, 26].

Traditionally, pairing and authentication rely on *user interaction* (e.g., entering a password) to fulfill their purposes. Yet, the rapidly increasing number of IoT devices demands a *prohibitive user effort* to pair and authenticate them [17, 31]. Even worse, many IoT devices lack user interfaces, making user-assisted pairing and authentication infeasible [6, 10]. To address these issues, research proposes *zero-interaction pairing (ZIP)* and *zero-interaction authentication (ZIA)* [17, 28]. They allow *colocated devices*, that reside inside an enclosed physical space, e.g., a room, to pair and authenticate *without user involvement*, utilizing devices' ambient context, like audio, captured by their on-board sensors [31, 38]. Despite ZIP and ZIA offer improved *usability* by minimizing user interaction and *deployability* by using off-the-shelf sensors of IoT devices, their *security* relies upon unpredictability of context, which depends on the intensity and variety of ambient activity (e.g., sound) happening in the environment [10].

Recent studies scrutinizing ZIP and ZIA schemes find that the *insufficient entropy of context*, which is common in many scenarios, like an unoccupied smart home, results in *attacks* against the schemes while prolonging their *time to complete* pairing and authentication [11, 4, 33, 34]. To date, there exist *few solutions* to address such issues, stemming from the low entropy of context, in the domain of ZIP and ZIA. We review these solutions in Section 2.

In this work, we propose HARDZIPA—a novel approach that allows ZIP and ZIA to prevent advanced active attacks while shortening schemes' completion time. The idea behind HARDZIPA is simple, yet powerful: we exploit off-the-shelf IoT actuators, like a robotic vacuum cleaner, to *generate context which is hard to predict*. We are inspired by the ubiquity of IoT actuators found in home and office spaces, e.g., voice assistants, smart lights, and cleaning robots [25, 15]. Unlike prior ZIP and ZIA works that relied on a human to generate context, like by walking or talking [31, 17], we are the first, to the best of our knowledge, to research the applicability of *commodity IoT actuators* to produce hard-to-predict context. Our motivation for utilizing IoT actuators is twofold: (1) in many scenarios, e.g., a smart home or office, *humans can be absent* for extended amounts of time, like from home during working hours or from office during lunch breaks, rendering the scenario context predictable due to low entropy [33, 15]; (2) human actions that affect context, e.g., motion or speech, are sufficiently *deterministic*, allowing an adversary to either approximate the context of legitimate devices [4, 33, 34] or even dominate it, like by playing a loud sound [15, 11].

Considering our motivation, we focus on *unattended scenarios*,[1] e.g., a smart home without any residents, to evaluate HARDZIPA for two reasons: (1) *attacks* against ZIP and ZIA schemes are *feasible* in these scenarios due to adversary's reduced effort to guess and manipulate context, as well as the lack of legitimate users who can notice the attack [33, 34, 6]; (2) human absence *lowers the amount of entropy* in context, requiring ZIP and ZIA schemes to accumulate more context data to maintain their security, leading to pairing and authentication time on the order of *minutes or even hours*, which is prohibitive for many IoT applications [41, 10]. Despite addressing the unattended scenarios, we also study the impact of human presence on the efficacy of HARDZIPA, providing the first insights into how users may perceive it.

In HARDZIPA, we leverage (1) the capability of IoT actuators to "stimulate" the context as well as (2) the presence of a pseudorandom number generator (PRNG) on many actuators [18]. Combining these two points, we devise an approach enabling IoT actuators to *inject hard-to-predict context stimuli*, like audio, into the environment of colocated devices (e.g., a room), boosting the amount of entropy in context, which hardens ZIP and ZIA schemes against attacks and reduces their completion time. For realizing HARDZIPA, we investigate how different context stimuli, like light or audio, can be crafted in a *generic fashion* and then be instantiated on real devices. To demonstrate the efficacy of HARDZIPA, we design and implement it on off-the-shelf IoT actuators, namely *smart speakers*, *lights*, and *humidifiers*. We evaluate HARDZIPA in real-world home and office scenarios, collecting over *80 hours* of various sensor data, capturing context. Our findings reveal that HARDZIPA thwarts *active attacks* on ZIP and ZIA, which can hardly be prevented by the state-of-the-art schemes [6, 42], while it also increases the amount of context entropy by up to *two times*, allowing ZIP and ZIA schemes to speed up their completion by the same factor.

In summary, we make the following contributions:

- We design HARDZIPA, a novel approach that leverages IoT actuators to improve security and shorten the completion time of ZIP and ZIA schemes.
- We implement HARDZIPA on off-the-shelf IoT actuators and evaluate it by collecting real-world sensor data, demonstrating the effectiveness of HARDZIPA.
- We publicly release our collected sensor dataset and the source code of HARDZIPA implementation.

[1]As part of our system model (cf. Section 3), we exemplify why unattended scenarios are relevant for ZIP and ZIA schemes.

2 Background and Related Work

We first explain the working principles of ZIP and ZIA schemes and then review related work.

Background. ZIP and ZIA schemes utilize the *similarity of context* observed by colocated devices (e.g., inside the same room) to either establish a shared secret key or verify devices' physical proximity [29, 11, 38]. In ZIP, two devices agreeing to pair (1) sense their shared context, like audio, using on-board sensors, (2) translate the captured context into bit sequences called *fingerprints*, and (3) input these fingerprints into a key agreement protocol to establish the shared secret key. In ZIA, (1) one device requests an authentication from another device, (2) both devices sense their shared context for a predefined timeframe (e.g., 5 seconds), and (3) the requesting device sends its context readings to the authenticator device, which compares the received context readings with its own, to make the authentication decision. Note that the two devices performing ZIA are assumed to share a secret key, protecting the transmitted sensor readings from eavesdropping and tampering with—this key can be preloaded or established via device pairing [6, 17, 9].

Related Work. To date, a few dozen ZIP and ZIA schemes utilizing different sensor modalities, like audio and illuminance, to capture context have been proposed—these are surveyed in [42] and [6]. Several studies disclose the perils of low-entropy context, reducing the security of ZIP and ZIA schemes while prolonging their completion time [11, 4, 33, 34]. Currently, few solutions exist to address these issues. *FastZIP* presents a novel ZIP architecture resisting advanced attacks (e.g., similar-context attack) and shortening the pairing time simultaneously [10]. However, this scheme needs various sensors to capture context, decreasing its deployability. *Moonshine* distills ZIP fingerprints obtained from low-entropy context, to output the high-entropy fingerprints [41]. While this approach improves the security, it operates by discarding parts of the fingerprint that have low entropy. Hence, the fingerprints input to Moonshine need to contain redundancy (i.e., more bits), which is achieved by collecting extra context data, thus increasing the pairing time.

In ZIA, *DoubleEcho* utilizes sound emission, allowing colocated devices to observe a similar room impulse response, which captures unique characteristics of the environment [39]. While being conceptually close to HARDZIPA, DoubleEcho focuses on preventing strong *colocated adversaries*, whom we do not consider (cf. Section 3). Such stringent security comes at the expense of the scheme's practicality, i.e., DoubleEcho only works for colocated devices residing within half a meter distance. *Proximity-Proof* is similar to DoubleEcho regarding the used audio context and threat model, but demands colocated devices to have both a speaker and microphone, reducing the scheme's applicability [14].

We view HARDZIPA as a complementary contribution to the above ZIP and ZIA schemes. Despite sharing some of their weaknesses, like the need for IoT actuators similar to extra devices and sensors, HARDZIPA improves the security and pairing time of existing schemes (cf. Section 5.6). Using ubiquitous IoT actuators allows generating *various types of context* (e.g., audio, illuminance) accommodating ZIP and ZIA schemes that rely on single-sensor [17], multi-sensor [38, 10], and heterogeneous [15] contexts.

We find one system—*Listen!* [27]—that has a similar goal to HARDZIPA. While this research is an important preliminary work for context stimuli injection, there are two notable differences with HARDZIPA: (1) Listen! is customized for audio context injection, whereas HARDZIPA is a generic approach extensible to diverse context stimuli (cf. Section 4); (2) HARDZIPA is evaluated under a stronger threat model, where an active adversary can inject their stimuli via an open door (cf. Section 3), while in Listen! such an attacker is separated from legitimate devices by a solid wall.

3 System and Threat Models

We present our system model, detailing the goal, requirements, and assumptions of HARDZIPA, as well as our threat model, describing the goal and capabilities of the adversary.

System Model. The *goal* of HARDZIPA is to inject hard-to-predict context stimuli, such as audio, into the environment (e.g., a room), where colocated devices perform ZIP or ZIA, leveraging off-the-shelf IoT actuators. These stimuli seek to increase the similarity and entropy of context observed by colocated devices, allowing ZIP and ZIA schemes to withstand attacks and speed up their completion time. We design HARDZIPA to fulfill the following *requirements*: (1) perform without user interaction after the start-up (*usability*); (2) execute on off-the-shelf IoT actuators without adding major modifications to their software stacks (*deployability*). To achieve HARDZIPA's goal while satisfying its requirements, we make the following *assumptions*: (1) IoT actuators are trusted, i.e., *not* compromised with malware; (2) they feature cryptographically secure PRNGs,[2] producing *unpredictable* random numbers suitable for security purposes [18].

Unattended Scenarios. Despite ZIP and ZIA are initiated by a user, there exist cases when devices decide to (re-)pair or (re-)authenticate themselves. For example, if there are many devices, ZIP may need a few pairing iterations to ensure confidence in the shared key to exclude passing by devices [28]; meanwhile, a user who initiated such ZIP can leave, trusting that pairing will succeed. Another case is when devices re-pair or reauthenticate if one of them gets compromised or as part of a key refreshing routine [41]. Finally, IoT robots can autonomously change their location (e.g., different rooms inside a smart building), prompting the need to pair or authenticate with in-room devices, irrespective of user's presence.

We envision HARDZIPA to be mainly used in *unattended scenarios*, like a smart home without occupants but IoT actuators within, which challenge security and completion time of ZIP and ZIA schemes [27, 11, 33, 29]. To produce hard-to-predict context, HARDZIPA can be invoked periodically or on-demand, based on the needs of ZIP and ZIA schemes.

Threat Model. The *goal* of an adversary is to break a ZIP or ZIA scheme by pairing or authenticating with legitimate (i.e., colocated) devices while residing *outside* their environment, such as a room. To accomplish this goal, the adversary tries to obtain context readings akin to that of legitimate devices', utilizing similar sensing hardware. The adversary carries out either a *passive* or *active attack* to achieve their goal. In the former attack, the adversary is located right outside the environment of legitimate devices, like in front of a *closed door* to a room; this threat model is followed by most of ZIP and ZIA schemes [38, 29, 15, 19, 17]. In the latter attack, the adversary, on top, uses a *half-open door* to the environment of legitimate devices to actively inject their own context stimuli (e.g., audio), employing commodity IoT actuators or household appliances. We note that active attacks are difficult to prevent by existing ZIP and ZIA schemes [4, 33, 27].

We consider *colocated adversaries* and *denial-of-service (DoS) attacks* to be outside the scope of this work. While the former is the most difficult attack to defend against in ZIP and ZIA, colocated adversaries (e.g., in the same room as legitimate devices), by definition, undermine the notion of physical security assumed in many use cases, like a smart home [15]. Despite DoS attacks being feasible for ZIP and ZIA [11], they can only prevent pairing and authentication but *not* circumvent them, leading to a false sense of security.

[2]Most operating systems and programming languages have such PRNGs built-in, e.g., *random* in Linux and *secrets* in Python.

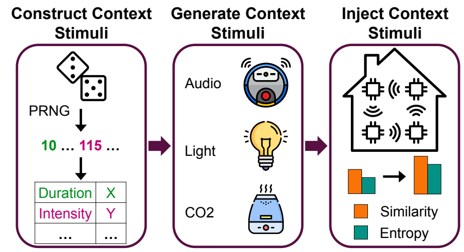


Figure 1: HARDZIPA injects hard-to-predict context stimuli into the environment of colocated devices (e.g., inside a room) performing ZIP or ZIA to increase the similarity and entropy of context observed by these devices. HARDZIPA constructs such stimuli by shaping their form and occurrence using a PRNG and generates the stimuli utilizing off-the-shelf IoT actuators.

4 System Design and Implementation

We detail the design and implementation of HARDZIPA.

System Overview. HARDZIPA produces hard-to-predict context stimuli to *secure* ZIP and ZIA schemes against attacks and *shorten* their completion time as follows (cf. Figure 1). First, HARDZIPA constructs a context stimulus that can be produced by a specific actuator, such as audio by the speaker. For this, we use a PRNG to randomize the stimulus parameters, e.g., frequency, duration, intensity, and pattern (cf. Table 1). Second, different off-the-shelf IoT actuators generate context stimuli constructed in this fashion, affecting various types of context, like light or audio. Third, when launched in the environment, where colocated devices execute a ZIP or ZIA scheme (e.g., a room), HARDZIPA continuously injects context stimuli for the duration required by the scheme (cf. Section 3), increasing the similarity and entropy of context captured by these colocated devices.

4.1 Context Stimuli Injection Algorithm

Before describing how HARDZIPA injects stimuli into the context, we first rationalize the used IoT actuators influencing various types of context captured by specific sensors.

Table 1: Parameters of context stimuli used in HARDZIPA.

Stimuli parameter	Context stimuli		
	Audio	Light	CO2
Frequency	1–3 min	30–90 sec	5–10 min
Duration	4.5–75 sec	5–60 sec	10–15 min
Intensity	Loudness: 0.1–1	Brightness: 1–254	Mist emission: low/high
Pattern	Speech + noise	Blink/constant	n/a
Color	n/a	RGB: 0–255	n/a
Spectrum	50–22100 Hz	n/a	n/a

Selected IoT Actuators. We design HARDZIPA in a generic way, choosing various IoT actuators that (1) are ubiquitous and (2) can impact specific context (e.g., audio) utilized by existing ZIP and ZIA schemes. Based on these two criteria, we select a *smart speaker*, *light*, and *humidifier* as actuators to implement and evaluate HARDZIPA. The smart speakers are on the rise due to the proliferation of voice assistants, like Alexa, while being built into many other smart devices, e.g., vacuum cleaning robots. Hence, the *audio* played by speakers can be captured by microphones of multiple user-end IoT devices, like smartphones or watches. It is thus unsurprising that the audio context is most frequently used by existing ZIP and ZIA schemes [29, 17, 38, 39, 14, 15].

The popularity of smart lights is increasing due to energy-saving and sustainability concerns [25]. The *illuminance* produced by smart bulbs can be captured by low-power ambient light sensors that are pervasive in IoT devices, while the *light's color* can be recorded by RGB sensors, which are also widespread [42]. Hence, the light context has been extensively utilized by ZIP and ZIA schemes [28, 22, 24].

The number of smart humidifiers is skyrocketing, boosted by the Covid-19 pandemic, as they can reduce viruses spread and enable a healthier in-door environment [1]. A humidifier emitting vapor affects not only the *humidity* of the environment but also its *temperature* and *carbon dioxide (CO2)* concentration [23]. These three modalities can be captured by integrated environmental sensors that become ubiquitous [3]. We identify one ZIA scheme that relies on humidity, temperature, and CO2 contexts [32]. However, recent concerns for public health and climate protection should prompt the massive adoption of environmental sensors, enabling future ZIP and ZIA schemes. In HARDZIPA, we focus on the CO2 context due to its major importance for human health as well as pollution and climate monitoring.

Overview of Context Stimuli Injection. To produce hard-to-predict context stimuli with HARDZIPA, we rely upon the following two observations: (1) actions performed by actuators, i.e., context stimuli, are *configurable* like sound loudness; and (2) many IoT actuators *feature a PRNG* for functional and security purposes [18]. Thus, by using the PRNG, we are able to parameterize the actions of IoT actuators to be non-deterministic, generating hard-to-predict context stimuli. However, a *main challenge* here is to identify the stimuli parameters that are generic for various IoT actuators and set them to yield *higher context similarity* and *entropy* observed by colocated devices in their environment, e.g., a room. Table 1 shows such stimuli parameters and their values used in HARDZIPA. In the following, we explain how these parameters have been selected based on our empirical findings.

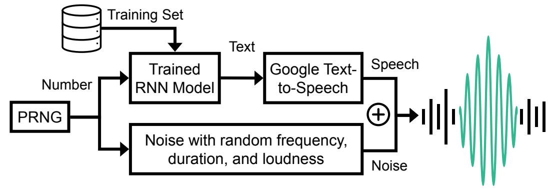


Figure 2: Structure of HARDZIPA's audio stimulus generator. It produces sound comprised of the randomized speech which is interleaved with noise signals of random frequency, duration, and loudness.

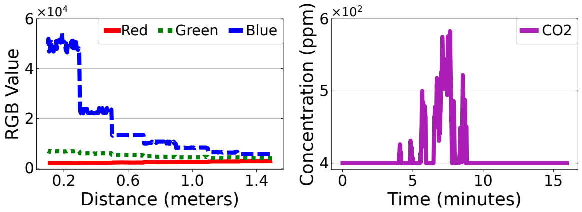
The *algorithm* for context stimuli injection in HARDZIPA works as such (cf. Figure 1): (1) each actuator uses its PRNG to produce the sequence of random numbers, where (2) every random number controls the assigned stimuli parameter, like duration in Table 1, to shape the context stimulus executable by this actuator (e.g., audio by speakers), which (3) generates such crafted stimulus, repeating the complete algorithm, i.e., steps (1)–(3), after a random pause.

Audio Injection. We use smart speakers to inject audio stimulus. Since microphones have a short response time, sensing sound almost instantaneously, we set the duration of stimulus to 4.5–75 seconds, allowing for both shorter and longer audio injections. The duration parameter *impacts* the frequency of stimulus occurrence: we choose it 1–3 minutes to keep a balance between the time required by the speaker to inject the stimulus and its regularity. We express the intensity of audio as loudness, normalized between 0.1–1, reflecting the minimum and maximum *loudness output* by a specific speaker. As for the pattern of stimulus, we combine *speech and noise*; the latter is generated in a wide frequency spectrum from 50 Hz to 22 kHz, which can be produced and captured by off-the-shelf speakers and microphones, respectively [17].

Figure 2 shows the structure of HARDZIPA's audio stimulus generator. To produce speech, we use a real-world text corpus[3] from the robotic vacuum cleaner, which is one of the actuators in our experiments (cf. Section 5.1). Based on this corpus, we train a recurrent neural network (RNN) to generate random text; such a technique is popular [8]. Our RNN has *three layers* as inspired by [36]: (1) embedding with 256 dimensions, (2) long short-term memory (LTSM) with 1024 units, and (3) dense. We set the RNN *temperature parameter* to 0.45, yielding the best trade-off between unpredictability and comprehensibility of a text generated by our model, that we implement in Keras. Then, such generated text is input to the Google's Text-to-Speech converter [7], producing speech at the conversational rate of 120 words per minute (wpm) [2].

To further increase the *unpredictability* of audio injection, we insert into the speech 1–3 evenly distributed noise signals with varying duration (0.5–5 seconds), loudness (0.1–1), and spectrum (50–22100 Hz). The number of inserted noise signals depends on the speech length, which is found as the duration of one word, i.e., 2 seconds at 120 wpm, multiplied by the word count in the RNN generated text, i.e., 2–30 words. Other noise parameters—duration, loudness, spectrum—are chosen randomly using the PRNG output. To select the shape

[3] Contains 96 sentences in English with minimum 2, average 7.12, and maximum 30 words [12].



(a) RGB sensor: smart bulb (b) CO2 sensor: humidifier

Figure 3: Impact of (a) colored light produced by a smart bulb and (b) water vapor produced by a humidifier on RGB and CO2 sensors, respectively.

of noise signal, we experiment with sine, square, and sawtooth waveforms, finding that the latter two introduce harmonics [5], which complicates the comparison of audio contexts captured by devices (cf. Section 4.2). Thus, we choose the *sine waveform*, allowing us to generate noise at a specific frequency, such as 500 Hz, without any harmonics.

Light Injection. We utilize smart bulbs to inject light stimulus in terms of illuminance and light color. As both ambient light and RGB sensors respond within several milliseconds, we set the stimulus duration to 5–60 seconds to enable brief and extended light injections. Given this duration, we inject the stimulus with the frequency of 30 to 90 seconds, ensuring that our hardware can perform it correctly while running uninterrupted. We represent the intensity of light as *brightness*, which often has actuator-specific ranges (e.g., 1–254)—these can, in principle, be unified, like from 0 to 1. For the pattern of stimulus, we consider *constant and blinking light*, finding that commodity smart bulbs cannot blink faster than once per second. Hence, we choose the blinking frequency from the interval 0.2–1 Hz, providing both reliable light injection and various blinking modes.

To set the color of light, we randomly select R, G, and B values from their 0–255 range, mapping the resulting color onto the *color space of a smart bulb*, i.e., CIE 1931 space in our experiments (cf. Section 5.1). Figure 3a shows how colored light generated by HARDZIPA impacts the RGB sensor that we use. This sensor returns undocumented RGB values when being exposed to colored light. Therefore, we reverse-engineered its working principle, allowing us to reliably distinguish between red, green, and blue colors in the range of distances from the source of light to the sensor.

CO2 Injection. We employ a smart humidifier to inject CO2 stimulus, i.e., by evaporating water, the humidifier changes the CO2 concentration in the environment. Figure 3b depicts this process using the off-the-shelf humidifier and CO2 sensors (cf. Section 5.1). Here, the humidifier starts to operate after minute 1, working for about five minutes. We see that (1) it takes 2–3 minutes for the stimulus to affect the sensor; (2) the CO2 concentration gradually returns to its ambient level once the stimulus is removed. Based on these findings, we set the stimulus frequency and duration to 5–10 minutes and 10–15 minutes, respectively. This allows the CO2 stimulus to reach multiple sensors in the environment and settle back to the background level of CO2. The humidifier used in our experiments has two levels of intensity: low or high *mist emission*. We omit setting the pattern of stimulus, as our humidifier features a single nozzle that allows spraying mist in *only one direction*. Yet, this stimulus pattern can be realized on advanced humidifiers with multiple mist intensity levels and nozzles, making the CO2 context more unpredictable.

4.2 Context Similarity and Entropy

We first justify the similarity metrics to compare context captured by two devices and then present our method for estimating the amount of entropy in context.

Similarity Metrics. In HARDZIPA, we want to compare the similarity of different types of context, like audio or CO2, in a *generic way* to abstract from specifics of concrete ZIP and ZIA schemes, whose context similarity metrics often depend on the use case (e.g., smart home vs. wearables) and implementation [11, 41]. To achieve this, we consider the *dynamic time warping (DTW)* algorithm which measures the distance between two time series, like sensor readings, handling misaligned and different-size data [20]. We discover that DTW suits well for comparing the similarity of illuminance, RGB, and CO2 data—confirming the results of prior work [25]; in HARDZIPA, we use DTW's Python implementation [13].

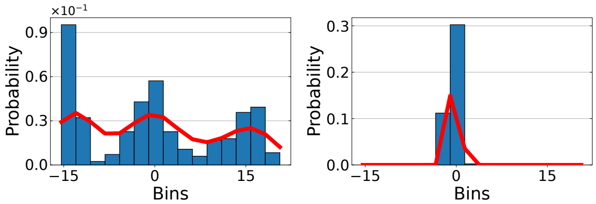
We find DTW to be less practical to compare audio data, as it imposes high computational overhead while only allowing the comparison in the time domain. Thus, we experiment with other metrics measuring *audio similarity* proposed by prior research: similarity score, maximum cross-correlation, and time-frequency distance [17, 38]. Our findings show that the first metric can best capture audio similarity in both time and frequency domains simultaneously, and it behaves stably across different environments (e.g., rooms); these results are in accord with [11]. Hence, we leverage the *similarity score*, which is found as the average of maximum cross-correlations computed in each of 20 one-third octave bands from 50 Hz to 4 kHz, to compare the similarity of two audio recordings. In HARDZIPA, we use the Matlab implementation of the similarity score metric from [11].

Entropy Estimation. There exist two ways to estimate entropy in the ZIP and ZIA domain: (1) is to apply NIST tests on the fingerprints, i.e., sequences of bits, generated by ZIP schemes from context data [40]; (2) is to compute the amount of entropy in raw context data using its distribution [41]. The former method is *not generic*, because it only works for ZIP schemes, whose fingerprints often have *entropy biases* introduced during the translation of context data into the sequence of bits [4, 11], while the NIST tests can assess the entropy inaccurately if the amount of input data (e.g., ZIP fingerprints) is insufficient. Thus, we adopt the latter approach and utilize the following formula to calculate the amount of *entropy in raw sensor data* that captures context:

$$H(X) = -\frac{\sum_{i=1}^{b} P(x_i)\log_2 P(x_i)}{\log_2 b} \quad (1)$$

Here, $H(X)$ is the entropy of time series X (i.e., sensor data), treated as a random variable. To use this representation, we quantize sensor readings into the b number of bins, obtaining the distribution of such data. Therefore, x_i is the center of the bin, $P(x_i)$ is the probability that the random variable X falls inside bin i, while $\log_2 b$ acts as a normalization factor.

Figure 4 depicts two distributions obtained in this manner for RGB data when HARDZIPA (1) works and (2) is idle. In



(a) HardZiPA works (data Y) (b) HardZiPA is idle (data Z)

Figure 4: Distribution of RGB data in two cases, i.e., (a) HardZiPA works, and (b) it is idle. In the former case, data Y has a significantly more uniform distribution than in the latter case of data Z, hence Y contains more entropy.

the former case (cf. Figure 4a), the data distribution is much closer to *uniform* than in the latter case (cf. Figure 4b). Using Equation 1, we calculate the entropy of these two time series: $H(Y) = 0.88$ and $H(Z) = 0.23$, which exhibits the impact of HardZiPA. It is agreed in the ZIP and ZIA domain that the amount of entropy in context data is *directly proportional* to pairing and authentication time [17, 10, 15]. Intuitively, from more unpredictable context ZIP and ZIA schemes can *faster* extract a distinct fingerprint and context feature, respectively. Thus, utilizing data Y allows speeding up ZIP and ZIA by the factor of 3.8, as compared to data Z (cf. Figure 4).

To use our entropy estimation method, we need to choose the number of bins for each sensor data, such as CO2. While considering existing bin selection criteria,[4] we find that data properties, i.e., range and behavior, to be decisive in choosing the number of bins. Based on our empirical findings, we set the following *number of bins* for each sensor data: 19 (audio), 20 (illuminance), 100 (RGB), and 20 (CO2) to provide consistent and reliable entropy estimation.

5 Evaluation of HardZiPA

We evaluate HardZiPA based on the real-world data that we collect to demonstrate its feasibility.

5.1 Experiment Setup

Apparatus. We prototype HardZiPA using the following IoT actuators. To inject audio, we utilize two types of speakers: (1) a built-in speaker of the *Roborock S5* vacuum cleaning robot and (2) a standalone *JBL Flip 5* speaker. We replicate the procedure in [12] to gain access to the Roborock S5, allowing us to play customized audio on it using the *python-miio* package.[5] The JBL speaker is controlled by connecting it to a laptop that runs the HardZiPA logic. For light injection, we make use of popular *Philips Hue* lights by connecting smart bulbs with the Hue bridge and managing them via the Philips API [35]. To inject CO2, we leverage an affordable *Maxcio Smart* humidifier, which is commanded through the Tuya API [16].

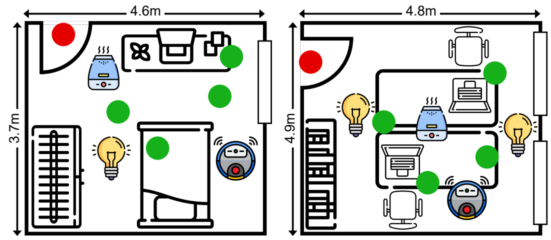
Similar to HardZiPA, we employ the following IoT actuators and household appliances to inject adversarial context stimuli during the active attack (cf. Section 3): a budget *Lenrue A2* speaker and JBL Flip 5 for audio, a colored lamp and flashlight for light, and a pedestal fan for CO2.

We build the following platform to collect sensor data. To record audio, we utilize a *Samson Go* USB microphone connected to the *Raspberry Pi 3 Model B*. The audio is captured in a raw PCM16 format at 44.1 kHz sampling rate, and it is stored in a WAV file. To collect illuminance and RGB data, we utilize the light sensor of the *Samsung Galaxy S6* smartphone that records these data at 5 Hz sampling rate, streaming it to the Raspberry Pi via USB. For CO2 collection, we employ an *SGP30 multigas sensor* attached to the Raspberry Pi via its pins, capturing the data at 1 Hz sampling rate. The recordings of our light and gas sensors are stored in text files, where each data point is supplied with a timestamp.

Table 2 summarizes the apparatus, i.e., both actuators and sensors, used in our experiments. We note that legitimate and adversarial devices utilize the *same sensing hardware*, which is described above, to collect context data.

Data Collection. We capture audio, illuminance, RGB, and CO2 data using our sensing platform in two real-world scenarios: *home* and *office*, as shown in Figure 5. In the former, we deploy four legitimate devices and three actuators (i.e., Roborock S5, two Hue bulbs, Maxcio humidifier) within an apartment room, while the adversarial device resides in front of an entrance door that can be open or closed (cf. Section 3). In the latter scenario, we equip an office in a similar fashion: with four legitimate devices and one adversarial, but we use four Hue bulbs to account for an office's larger area and more obstacles. To further assess the capability of HardZiPA to resist attacks and boost context entropy, we experiment with additional actuators in the office scenario, i.e., the JBL Flip 5 speaker and *iTvanila Mist* humidifier.

Following our threat model given in Section 3, we collect context data in *four experimental settings*—the first two correspond to passive and active attacks while HardZiPA *does not* execute: labeled as *PA* and *AA*, respectively. The second two settings are for passive and active attacks during which HardZiPA *works*, namely *PA+H* and *AA+H*. In our evaluation of HardZiPA, we focus on *unattended* use cases, i.e., without humans, as motivated by Section 3. Hence, the home scenario remains unoccupied in all experiments (cf. Table 3), yet we expose it to *realistic ambient conditions*, like external noise from neighbouring rooms / streets, flicker of a display, and drafts from a tilted window. While in the office, we introduce two people in the *PA* experiment who actively move and converse. Thus, we can assess the *impact of human presence* on context and compare it with that of HardZiPA. The rest of office experiments are unoccupied, but we conduct them



(a) Home (b) Office

Figure 5: Data collection settings in our experiments. The legitimate colocated devices (●) and IoT actuators reside inside a home or office environment, while a non-colocated adversarial device (●) stays outside it, i.e., in front of an entrance door which is closed in the passive attack and half-open during the active attack (cf. Section 3).

[4] https://www.statisticshowto.com/choose-bin-sizes-statistics/.

[5] https://github.com/rytilahti/python-miio.

Table 2: Apparatus, i.e., actuators and sensors, used for data collection and evaluation of HARDZIPA.

Context data	Actuators: HARDZIPA	Actuators: Adversarial	Sensors: *same* for legitimate and adversarial devices (sampling data rate)
Audio	Speaker: Roborock S5 / JBL Flip 5	Speaker: Lerne A2 / JBL Flip 5	Microphone: Samson Go USB (44.1 kHz)
Illum./RGB	Smart light bulbs: Philips Hue	Colored bulbs: lamp / flashlight	Light sensor: Samsung Galaxy S6 (5 Hz)
CO2	Humidifier: Maxcio Smart / iTvanila Mist	Pedestal fan	Multigas sensor: SGP30 (1 Hz)

Table 3: Details on HARDZIPA data collection and evaluation settings.

Experimental setting	Attack type	HARDZIPA works?	Human presence? Home	Human presence? Office
PA	Passive	No	No	Yes
AA	Active	No	No	No
PA+H	Passive	Yes	No	No
AA+H	Active	Yes	No	No

Table 4: Similarity and entropy metrics used for HARDZIPA evaluation.

Context data	Similarity: Metric	Similarity: Range	Similarity: High similarity?	Entropy: *same* for all context data
Audio	Sim. score	$[0,1]$	Big sim. score	Metric: Equation 1
Illum./RGB	DTW dist.	$[0,1]$	Small DTW dist.	Range: $[0,1]$
CO2	DTW dist.	$[0,1]$	Small DTW dist.	

under realistic conditions similar to the home (cf. Table 3).

To understand whether human presence can interfere with HARDZIPA, lowering its efficacy, and gain the *first insights* into how users may perceive HARDZIPA (cf. Section 6), we additionally repeat the office *PA+H* and *AA+H* experiments with two people present in the scenario whom we ask to follow their normal working routine, implying occasional conversations and motion of our participants. In total, we collect over 80 hours of sensor data in our scenarios.

Reproducibility. Our collected sensor data, intermediate results, and the codebase of HARDZIPA are publicly available: https://github.com/seemoo-lab/hardzipa.

Ethical Considerations. This research was approved by our institutional review board, the participants residing in experimental settings (cf. Table 3) gave informed consent for the collection, use, and release of sensor data.

5.2 Methodology

Data Preprocessing. To estimate the similarity and entropy of context (cf. Section 4.2), we preprocess our collected sensor data as follows. We synchronize the audio recordings of colocated legitimate devices via cross-correlation as in [11]. For illuminance, RGB, and CO2 data, we first perform mean subtraction to eliminate offsets between sensors, e.g., due to hardware variation, and then conduct signal smoothing and noise reduction in two steps: (1) applying a Savitzky-Golay filter using a window length 3 and degree 2 polynomial, followed by (2) a Gaussian filter with a sigma of 1.4. We adapt the filter parameters for signal smoothing and noise reduction based on best practices from related work [10, 21].

Similarity and Entropy Estimation. Recall that we choose the similarity score as our metric to compare audio similarity (cf. Section 4.2). This metric is best suited for audio snippets [17], hence we split our audio data into *recordings* of 10, 30, and 60 seconds. We then compute the similarity score using such recordings of colocated and non-colocated devices, and find the average result. The similarity score ranges *between 0 and 1*, with a bigger number showing higher audio similarity. As for illuminance, RGB, and CO2 similarity, we compute it as DTW distance between the *full data* of colocated and non-colocated devices collected in each experiment (cf. Table 3), finding the average result. We normalize our DTW distances utilizing min-max scaling to be *from 0 to 1*. Here, a smaller distance value indicates higher data similarity. Moreover, we study DTW distances on snippets of illuminance, RGB, and CO2 data to shed light on recording sizes of such modalities that most benefit ZIP and ZIA aided by HARDZIPA.

To estimate the entropy of audio, illuminance, RGB, and CO2 data, we use Equation 1 from Section 4.2. Specifically, we input the *full data* of each colocated device (per experiment, cf. Table 3) into this equation to calculate its entropy, averaging the results. By considering the full data per device, we keep the entropy estimation *generic*, avoiding parameters of specific ZIP and ZIA schemes, e.g., size of the input data. Our resulting entropy is bounded *between 0 and 1*, where a bigger number means higher entropy of sensor data.

The summary of our similarity metrics and entropy estimation is provided in Table 4.

End-to-end Comparison. To evaluate how HARDZIPA can benefit existing solutions, we *prototype* a state-of-the-art ZIP *and* ZIA scheme, *comparing* their error rates as well as completion time on the context data (1) impacted by HARDZIPA and (2) affected by humans.

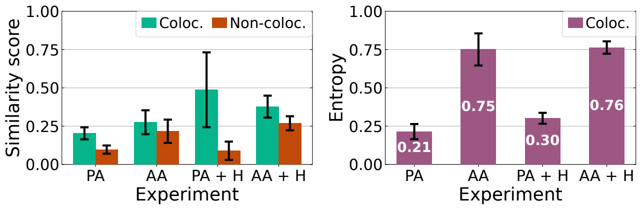
5.3 Results: Audio Context

We present our findings on the similarity and entropy of audio context in the home and office scenarios.

Similarity. Figure 6a depicts the similarity scores (y-axis) of colocated and non-colocated devices in our four experiments (x-axis, cf. Table 3) for the home scenario. We see two main trends here: (1) *without* HARDZIPA, the passive attack (*PA*) is avoided by a narrow margin, while the active attack (*AA*) *succeeds*;[6] (2) using HARDZIPA allows preventing the passive attack (*PA + H*) and mitigating the active attack (*AA + H*). In the office scenario, the similarity results are alike.

The similarity scores of Figure 6a are obtained on the audio snippets of 60 seconds (cf. Section 5.2). With the shorter snippets of 10 and 30 seconds, we have the same trends, but the similarity of colocated devices is 15–30% *higher* while it *grows by* up to 50% for non-colocated devices during active attacks, i.e., *AA* and *AA + H*. Hence, longer audio recordings can help ZIP and ZIA schemes to resist attacks, corroborating findings from prior work [11].

[6] The distributions of colocated and non-colocated similarity scores significantly overlap, as indicated by intersecting error bars of *AA* in Figure 6a.



(a) Similarity scores (home)

(b) Audio entropy estimation (home)

Figure 6: Evaluation of context (a) similarity and (b) entropy for audio data.

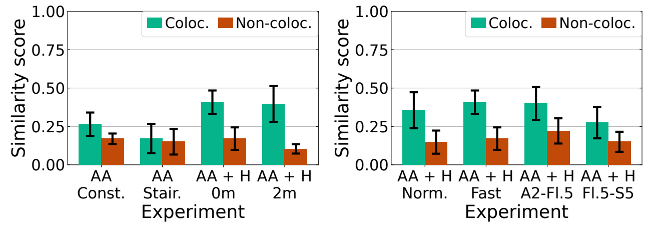
Exploring the high deviation of similarity scores between colocated devices when HARDZIPA executes (cf. *PA + H* in Figure 6a), we find that it stems from *varying perception* of higher-frequency audio spectrum by obstructed devices. The similarity score captures audio spectrum till around 4.5 kHz. In the home scenario, the one device which is blocked by the bed, as presented in Figure 5, senses slightly distorted audio, injected by the built-in speaker of a Roborock S5, than other colocated devices, causing discrepancies in similarity scores. We do not observe such behavior in the office scenario, however it reveals obstacles to using higher audio frequencies for audio injection in HARDZIPA.

In Figure 6a, we conduct the active attacks utilizing an affordable A2 speaker that plays *constant* sine-waveform noise at 150 Hz (cf. Section 4.1). We use such low frequency, as it allows the noise to easily propagate into the environment of colocated devices without saturating the adversarial microphone situated near the A2 speaker. While this active attack already succeeds (cf. *AA* in Figure 6a), we further advance it by generating noise on various frequencies, i.e., *sequentially* on each one-third octave band starting from 40 Hz to 20 kHz. Such resulting noise resembles a frequency *staircase*.

Figure 7a presents the similarity scores for the *AA* and *AA + H* cases in the office scenario. We see that the active attack using the staircase strategy is far more efficient than with the constant noise (cf. *AA*: Const. vs. Stair. in Figure 7a). Still, HARDZIPA prevents the staircase attack which becomes *impractical* if we increase the distance between the adversarial speaker and colocated devices by two meters, as depicted by *AA + H*: 0m vs. 2m in Figure 7a.

To assess the impact of HARDZIPA's audio injection parameters on the efficacy of active attack prevention, we *halve* the frequency of audio occurrence from 1–3 to 0.5–1.5 minutes (cf. Table 1). Figure 7b shows that such a change alone can not only increase the similarity scores of colocated devices but also make them more consistent (lower error bars), thwarting active attacks, i.e., *AA + H*: Norm. vs. Fast. In the same fashion, we evaluate how *high-quality speakers* affect the efficiency of both HARDZIPA and the active attack. For this, we compare the similarity scores of colocated and non-colocated devices when (1) HARDZIPA uses the better Flip 5 speakers while the adversary—A2; (2) vice versa: with the adversary utilizing Flip 5, whereas HARDZIPA relies on the built-in Roborock S5 speaker. Indeed, higher-quality speakers benefit both HARDZIPA and adversaries by either mitigating or aggravating the active attacks (cf. *AA + H*: A2–Fl.5 vs. Fl.5–S5 in Figure 7b).

Entropy. Figure 6b provides the estimated audio entropy (y-axis) of colocated devices in our four experiments (x-axis)—listed in Table 3—for the home scenario. We see that running HARDZIPA *boosts* the amount of entropy by over 40%, i.e., from 0.21 in *PA* to 0.30 in *PA + H*. This allows ZIP and ZIA schemes to reduce their completion time by the *same factor*. Interestingly, that during active attacks (cf. *AA* and *AA + H* in Figure 6b) performed using constant noise at 150 Hz, the amount of entropy is more than doubles, as compared to the *PA + H* case. Studying this phenomenon, we find that the A2 speaker (employed by the adversary) vibrated when it played the noise while lying down on the tiled floor, producing additional clinking sounds. We attribute the high entropy figures of *AA* and *AA + H* cases to such sound artifacts.



(a) Similarity scores for different adversarial signals and distances (office)

(b) Similarity scores for different injection speed of HARDZIPA and hardware (office)

Figure 7: Detailed analysis of audio context similarity under active attacks.

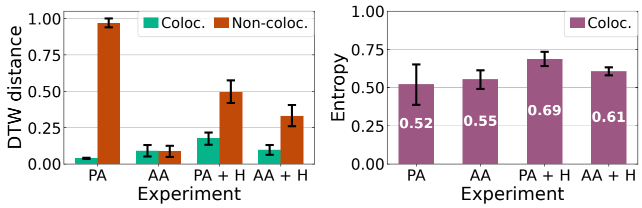
We obtain a more consistent picture of assessed audio entropy in the office scenario. Recall that the office's *PA* experiment includes two persons, who actively influence the audio context (cf. Section 5.1). Thus, our entropy estimates for the *PA* and *PA + H* cases account for 0.47 and 0.46, respectively. This result is crucial, indicating that HARDZIPA can *replace* intense human interaction in producing high-entropy context for ZIP and ZIA. In the office scenario, we do not notice any sound artifacts, seen in the home, resulting in consistent entropy figures of 0.48 and 0.54 in the active attack cases of *AA* and *AA+ H*, respectively.

Following our evaluation for the audio similarity, we find that utilizing high-quality speakers (i.e., Flip 5 vs. S5) allows HARDZIPA to increase the entropy from the office's *PA + H* estimate by an extra 20%. While injecting the audio stimulus twice as often, boosts the same entropy estimate by an added 40%, already with the built-in Roborock S5 speaker.

5.4 Results: Illuminance and RGB Context

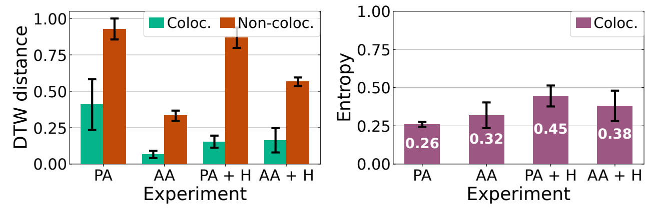
We report on the similarity and entropy of illuminance as well as RGB context found in the home and office scenarios.

Similarity. Figure 8a shows the DTW distances (y-axis) between colocated and non-colocated devices for the RGB context in our four experiments (x-axis, cf. Table 3) for the home scenario; the illuminance results are similar. We see that the passive attack *fails* due to varying lighting conditions in- and outside the apartment room (e.g., daylight vs. electric), while the active attack, conducted with a colored lamp mounted on top of a half-open door, *succeeds*—as depicted by *PA* vs. *AA* in Figure 8a. HARDZIPA prevents such an active attack (cf. *AA + H* in Figure 8a), yet it slightly enlarges context dissimilarity (i.e., bigger DTW distances) among colocated devices, suggesting that the coverage and directionality of light injections need to be considered.



(a) RGB DTW distances (home) (b) RGB entropy estimation (office)

Figure 8: Evaluation of context (a) similarity and (b) entropy for RGB data.



(a) CO2 DTW distances (office) (b) CO2 entropy estimation (home)

Figure 9: Evaluation of context (a) similarity and (b) entropy for CO2 data.

In the office scenario, we observe the same DTW distance trends as in the home for both illuminance and RGB context, except that the active attack without HARDZIPA, i.e., the *AA* case, *does not* succeed. This happens because the adversarial light stimuli injected towards colocated devices are blocked by obstacles in the office, like displays.

We study the DTW distance on snippets of 15, 30, 60, 90, 120, and 300 seconds for illuminance and RGB data in both home and office scenarios. Our results show that the snippets of *15–60 seconds* universally attain smallest DTW distances, thus capturing the context similarity most efficiently. Hence, ZIP and ZIA schemes that rely on illuminance and RGB data can most benefit from these recording sizes, given the current light injection parameters of HARDZIPA (cf. Table 1).

Entropy. Figure 8b depicts the found RGB entropy (y-axis) of colocated devices in our four experiments (x-axis, cf. Table 3) for the office scenario. We see that HARDZIPA *raises* the amount of entropy by more than 30%—from 0.52 to 0.69 in the *PA* and *PA + H* cases, respectively; allowing ZIP and ZIA schemes using such data to *proportionally* shorten their completion time. In the *AA* and *AA + H* cases, the estimated RGB entropy behaves as expected, according to presence or absence of context stimuli, whether from HARDZIPA or the adversary. Our results for the illuminance entropy are alike.

In the home scenario, the entropy estimates for RGB and illuminance follow the same trends as in the office while attaining comparable figures.

5.5 Results: CO2 Context

We provide the results for the CO2 context similarity and entropy obtained in our home and office scenarios.

Similarity. Figure 9a shows the DTW distances (y-axis) between colocated and non-colocated devices for our four experiments (x-axis, cf. Table 3) in the office scenario. We observe that the passive attack (*PA*) *fails*, as the ambient CO2 levels in- and outside the office room differ significantly being affected by various factors, e.g., temperature and ventilation. Such factors also contribute to *reduced* CO2 similarity between colocated devices, as indicated by the bigger DTW distance and higher error bars (cf. *PA* in Figure 9a). We see that the active attack (*AA*), carried out by blowing air using a pedestal fan via a half-open door, is also *unsuccessful*. Still, non-colocated DTW distances of *AA* are much smaller than in the *PA* case, suggesting the feasibility of active attacks on the CO2 context, like through more advanced actuators, e.g., air pumps or industrial humidifiers.

Using HARDZIPA *benefits* the CO2 context by (1) making it *more similar* and *consistent* between colocated devices, as seen from smaller and less deviating DTW distances of the *PA + H* case than *PA* in Figure 9a; and (2) further *raising the bar* for active attacks (cf. *AA + H* vs. *AA* in Figure 9a).

In the home scenario, the DTW distance trends as well as figures are akin to the office results.

We explore the DTW distance on snippets of 30, 60, 150, 300, 600, and 900 seconds for CO2 data in both home and office scenarios. Our findings suggest that the snippets in the range *from 300 to 600 seconds* universally result in smallest DTW distances, thus capturing CO2 context most similarly. Hence, such recording sizes can significantly profit ZIP and ZIA schemes utilizing CO2 data, for the existing parameters of CO2 injection in HARDZIPA (cf. Table 1).

Entropy. Figure 9b gives the assessed CO2 entropy (y-axis) of colocated devices for our four experiments (x-axis, cf. Table 3) in the home scenario. HARDZIPA attains the *entropy increase* of over 70%, i.e., from 0.26 in *PA* to 0.45 in *PA + H*. This enables ZIP and ZIA schemes based on CO2 data to *comparably* decrease their completion time. We also see that the active attack, which does not succeed in terms of context similarity (cf. *AA + H* in Figure 9a), can slightly reduce the entropy boost provided by HARDZIPA, as shown by the *AA + H* case of Figure 9b.

In the office scenario, we observe similar entropy behavior with marginally higher figures, when using the same affordable Maxcio humidifier (cf. Table 2). With the more advanced iTvanila Mist, we can further raise the amount of entropy by an additional 20% from the home's *PA + H* estimate (cf. Figure 9b), accounting for 0.54.

5.6 End-to-end Comparison with Prior Work

We now *compare how* a state-of-the-art ZIP [28] and ZIA [17] scheme perform in terms of security, usability, and completion time on context data collected in two office scenarios: (1) occupied by humans [11] and (2) devoid of them, but with HARDZIPA working. To provide a meaningful comparison, we pick the schemes utilizing different context data, i.e., illuminance and audio. In [11], these data are captured in three office rooms, encompassing four colocated devices as well as between one to three humans each, throughout eight hours. While our data are collected in one office with four colocated devices inside and no people for six hours (cf. Figure 5b).

ZIP Scheme. We replicate ZIP by *Miettinen et al.* [28] that relies on illuminance, using the scheme's implementation of [11], allowing for a direct comparison. As reported by [11], this scheme achieves lowest error rates when one fingerprint bit (i.e., 0 or 1) is obtained from 30 seconds of illuminance data—we follow such parametrization in our comparison. To assign a 0- or 1-bit, the ZIP scheme utilizes two thresholds, which are set to 10 and 0.1 by prior works [28, 11]. We find

Table 5: Evaluation of ZIP scheme from [28] on illuminance data affected by (1) humans [11] and (2) HARDZIPA in the office scenario.

Illum. data	EER	Fingerprint quality (colocated)		
		% of 1-bits	Min-entropy	% of *balanced* keys
[11]	0.54	8.9 ± 0.6	0.004	3.2
HARDZIPA	0.07	25.3 ± 4.1	0.26	9.1

EER – Equal Error Rate.
Min-entropy is assessed in one bit; the balanced keys are 20 bits in size.

these threshold values to be conservative, as they only allow capturing *profound illuminance changes*, like turning on / off electric lighting or shading daylight. Hence, the fingerprints produced by this scheme contain mostly 0-bits, rendering it insecure; this limitation is also noted by [11]. We thus adjust the thresholds to 8 and 0.01, respectively, making it possible to catch less extreme changes in illuminance (e.g., brightness alteration) within our fingerprints.

Table 5 shows our evaluation results of ZIP by Miettinen et al. [28] obtained on illuminance data impacted by humans [11] and HARDZIPA. We find that the *average difference* in fingerprint similarity of colocated and non-colocated devices is 12.4% for the HARDZIPA data, while it reaches only 0.1% on the illuminance from [11]. To assess the *security and usability* of this scheme, we compute Equal Error Rate (EER), that is the intersection point of False Acceptance Rate (FAR) and False Rejection Rate (FRR). The EER thus balances security (FAR) and usability (FRR) metrics, which are equally important for ZIP and ZIA schemes [42, 6]. We obtain EERs of 0.07 and 0.54 on the data from HARDZIPA and [11], respectively. Hence, HARDZIPA *greatly* improves the security and usability of the ZIP scheme under consideration.

We further explore the quality of fingerprints produced by this ZIP scheme to evaluate not only its security—as *unpredictability* of fingerprints—but also the pairing time. Table 5 reveals that fingerprints derived from the illuminance data of [11] contain less than 10% of 1-bits, explaining the *striking similarity* of colocated and non-colocated fingerprints, since both are mainly comprised of 0-bits. Even worse, these fingerprints are prone to include long sequences of 0-bits which are rarely interleaved with short series of consecutive 1-bits. Such fingerprints are easily predictable—we confirm this by assessing their *min-entropy* in one bit using the NIST SP800-90B test suite [40] (cf. Table 5). Recall that the low entropy of fingerprints leads to a prolonged pairing time [10].

Inspecting fingerprints obtained on the HARDZIPA data, we see that (1) the ratio of 1-bits exceeds 25% while (2) there exist few long sequences of consecutive 0- and 1-bits. Thus, our fingerprints have *65-times more entropy* in one bit, estimated with the NIST suite, than those derived from the illuminance data of [11] (cf. Table 5). Hence, the pairing times, that are achievable on the data by HARDZIPA and [11], can potentially differ by the same factor.

The NIST entropy tests can be *inaccurate* requiring a significant amount of data for a fair assessment [40]. Hence, we also estimate the amount of *balanced* keys, i.e., fingerprints of a fixed size containing roughly the same number of 0- and 1-bits. We set the size of such fingerprints to 20-bits, which suffices to pair securely, using state-of-the-art cryptographic protocols [10]. Moreover, we demand that balanced keys lie within the 40:60 ratio between 0- and 1-bits (or vice versa), following prior work [4]. To obtain such balanced keys, we traverse our fingerprints with an overlapping sliding window, using the step of 4-bits. We see that the HARDZIPA data allows producing *three times more* balanced keys that the illuminance from [11] (cf. Table 5), shortening the pairing time of the studied ZIP scheme by the similar degree.

Table 6: Evaluation of ZIA scheme from [17] on audio data affected by (1) humans [11] and (2) HARDZIPA in the office scenario.

Audio data	EER	% of enough-power snippets (colocated)
[11]	0.08	90.0
HARDZIPA	0.06	99.5

EER – Equal Error Rate.

ZIA Scheme. For ZIA by *Karapanos et al.* [17], we already adapt the scheme implementation from [11] (cf. Section 4.2), enabling a fair comparison. Both our findings in Section 5.3 and those of [11] indicate that this scheme can best separate colocated and non-colocated devices on audio snippets of 60 seconds. Thus, we use the similarity scores obtained on such snippets to compute EERs for audio data affected by humans [11] and HARDZIPA. Table 6 shows that on the HARDZIPA data, which is taken under active attacks, we achieve a lower EER than for the audio from [11], where only passive attacks are considered (cf. Section 3). Thus, HARDZIPA can *further harden* ZIA by Karapanos et al., improving its security and usability, even during active attacks.

To discard audio snippets that have low entropy, thwarting predicable context attacks (e.g., in a quiet room), this scheme uses a *power threshold* of 40 dB [17, 11]. Thus, the snippets, whose power is *below* this threshold, are rejected prolonging the authentication time. To estimate this authentication time for HARDZIPA data and audio from [11], we find how many audio snippets (60-second) of colocated devices have *enough power*, lying above the power threshold. Table 6 reports that using such estimate, HARDZIPA achieves almost *10% faster* authentication time for the studied ZIA scheme.

6 Discussion

We present relevant discussion points for HARDZIPA.

Generalizability. We design HARDZIPA to generalize for various types of IoT actuators. The stimuli parameters of Table 1: frequency, duration, intensity, and pattern are *generic*, hence they apply to different actuators, yet this parameter list is non-exhaustive. We also identify *actuator-specific* parameters, e.g., color and spectrum, to consider the heterogeneity of IoT actuators. Thus, HARDZIPA can be *extended* to other actuators, using a mixture of generic and specific stimuli parameters found in this work and by future research.

To set the stimuli parameters of HARDZIPA injection (cf. Table 1), we *need to* balance the capabilities of IoT actuators, i.e., how fast they can act, and the dynamics with which their stimuli affect context. Increasing the stimulus frequency, as shown in Section 5.3, boosts both the context similarity and entropy. However, one should consider that in the *real-world* settings, HARDZIPA must not disrupt the main routine of an actuator or consume too much power, depleting its battery.

Deployment Considerations. We intend HARDZIPA to operate in *unattended* scenarios which are prevalent in the IoT, like a smart home without inhabitants during working hours. Still, in many scenarios humans are present, yet hardly affect the context, e.g., by sitting quietly or sleeping. In such cases, HARDZIPA can help to produce high-entropy context, but it may be *perceived as invasive* by people; we elaborate on this issue in the next discussion point. We envision HARDZIPA can be tailored to work in human presence by using *less obtrusive* context stimuli, e.g., ultrasound, and applying them in a *directional manner*, like emitting light towards colocated devices. Such directionality can be achieved by utilizing mobile IoT actuators, e.g., robotic vacuum cleaners. Still, more research is required to study the feasibility of these methods.

Another concern is the security of HARDZIPA due to its *reliance on* the PRNG. PRNGs are plagued with issues, producing numbers in a deterministic fashion [18]. Recent work accentuates this problem for IoT devices whose PRNGs are *commonly* insecure [37]. Thus, one should consider such an issue when prototyping HARDZIPA on real hardware.

Our findings indicate that the *capabilities* of IoT actuators (e.g., speaker quality) and their *placement* (e.g., light coverage) play an important role for the efficiency of HARDZIPA. Hence, to leverage HARDZIPA, the actuators should be deployed such that their stimuli could reach colocated devices, performing ZIP or ZIA, as seamlessly as possible.

Insight: HARDZIPA and Human Presence. Our extra experiments, where we have *two persons* within the office scenario whilst HARDZIPA *works* (cf. Section 5.1), inform this discussion point. First, we check if human presence may interfere with HARDZIPA, reducing context similarity and / or entropy. We see *no* significant difference between the results of these experiments and those provided in Sections 5.3–5.5, for all types of context data that we study. Thus, HARDZIPA *maintains its efficiency* in the face of human presence.

Second, we ask our participants, who are two males of 32 and 28 years old as well as tech-savvy, about their perception of HARDZIPA in terms of intrusiveness. Interestingly, both respondents found HARDZIPA to be *less intrusive* than we had anticipated. Specifically, they *hardly noticed* the humidifier (blowing vapor) that worked "soundlessly" according to one person, while another—had a similar humidifier at home and was used to it. Our participants reported the same experience with smart lights, clarifying that the blinking bulbs *did not bother* them, as they were not the only light source in the room, since the main ceiling lights were on.

Unsurprisingly, both respondents rated the audio injection as the *most intrusive*. When inquired about the level of intrusiveness, one person compared it to a busy day in the office, where visitors drop by, while another—imagined a gathering of several people nearby. We further asked for how long our participants are able to endure the audio injection, providing they fully understand the HARDZIPA purpose. Both agreed that *2–4 minutes* of audio injection are acceptable.

We consider the above experience of our participants with HARDZIPA to be the first step to understanding of how users view it, paving the way for the HARDZIPA deployment.

Limitations and Future Work. In our audio injection based on speech, we do not consider *vocal registers* occupying different frequencies (e.g., fry–low vs. whistle–high). This can diversify speech produced by HARDZIPA, *raising entropy* of the audio context. Now, we generate random speech utilizing the state-of-the-art RNN (cf. Section 4.1). Yet, in this RNN, we only use English language, hence adding more languages and mixing them would make the produced speech more *unpredictable*. As an alternative to RNNs, we can leverage the GPT-4 mechanism[7] for generating random text—from which HARDZIPA will synthesize the speech.

A different approach to audio injection can rely on *white noise*, where the phase of different frequencies is controlled by the PRNG, to attain higher entropy of audio context. We leave the investigation of such an approach to future work.

In our evaluation, we were unable to perform a successful active attack on the CO2 context while showing its feasibility (cf. Section 5.5). We envision this limitation to be addressed, as part of future research, by leveraging more *advanced actuators*, like air pumps and industrial humidifiers.

Another avenue for improvement is to develop a *generic method* to select bins in our entropy estimation for different types of context data (cf. Section 4.2). So far, we rely on data properties, like range and behavior, to choose the number of bins. This allows us to evaluate the relative change in entropy per sensor data type (e.g., audio), but *does not* compare the entropy of various data, like CO2 vs. illuminance. Such bin selection is prone to *over- or underestimate* the entropy if the bins are chosen without considering certain artifacts of data behavior, as seen for audio in Section 5.3. Note that devising the generic method to select bins can be inspired by [30, 41].

7 Conclusion

Zero-interaction pairing (ZIP) as well as zero-interaction authentication (ZIA) allow Internet of Things (IoT) devices to establish and maintain secure communication *without user assistance*, leveraging their *ambient context*, like audio. The amount of entropy in this context is *decisive* for the security and completion time of ZIP and ZIA schemes. Such context entropy is often low, especially in *unattended scenarios* (e.g., empty smart home) that are prevalent in the IoT, threatening both security and utility of the schemes. To address these issues, we propose HARDZIPA, a novel approach that utilizes off-the-shelf IoT *actuators* and their pseudorandom number generators (PRNGs) for producing high-entropy context. We implement HARDZIPA on commodity actuators, i.e., smart speakers, lights, humidifiers, and conduct real-world experiments, showing the capability of HARDZIPA to prevent advanced *active attacks* on ZIP and ZIA schemes, while *speeding them up* by up to two times.

8 Acknowledgments

We would like to thank the anonymous reviewers for their insightful comments. This work has been co-funded by the Research Council of Norway as part of the project Parrot (311197) and the Deutsche Forschungsgemeinschaft (DFG) within the Collective Resilient Unattended Smart Things (CRUST) project as part of the DFG Priority Program SPP 2378 – Resilient Worlds.

[7] https://openai.com/product/gpt-4

9 References

[1] A. Ahlawat, A. Wiedensohler, S. K. Mishra, et al. An Overview on the Role of Relative Humidity in Airborne Transmission of SARS-CoV-2 in Indoor Environments. *Aerosol and Air Quality Research*, 20(9):1856–1861, 2020.

[2] D. Barnard. Average Speaking Rate and Words per Minute, 2022. https://virtualspeech.com/blog/average-speaking-rate-words-per-minute.

[3] Bosch Sensortec GmbH. IoT-ready Environmental Sensor BME680 from Bosch, 2021. https://www.bosch-sensortec.com/news/iot-ready-environmental-sensor-bme680.html.

[4] A. Brüsch, N. Nguyen, D. Schürmann, S. Sigg, and L. Wolf. Security Properties of Gait for Mobile Device Pairing. *IEEE Transactions on Mobile Computing*, 19(3):697–710, 2019.

[5] J. Comeau. Let's Learn about Waveforms, 2018. https://pudding.cool/2018/02/waveforms/.

[6] M. Conti and C. Lal. Context-based Co-presence Detection Techniques: A Survey. *Computers & Security*, 88:101652, 2020.

[7] P. N. Durette et al. gTTS, 2022. https://pypi.org/project/gTTS/.

[8] W. Fedus, I. Goodfellow, and A. M. Dai. Maskgan: Better Text Generation via Filling in the_. *arXiv preprint arXiv:1801.07736*, 2018.

[9] M. Fomichev, F. Álvarez, D. Steinmetzer, P. Gardner-Stephen, and M. Hollick. Survey and Systematization of Secure Device Pairing. *IEEE Communications Surveys & Tutorials*, 20(1):517–550, 2018.

[10] M. Fomichev, J. Hesse, L. Almon, T. Lippert, J. Han, and M. Hollick. FastZIP: Faster and More Secure Zero-interaction Pairing. In *Proceedings of the 19th Annual International Conference on Mobile Systems, Applications, and Services*, page 440–452, 2021.

[11] M. Fomichev, M. Maass, L. Almon, A. Molina, and M. Hollick. Perils of Zero-Interaction Security in the Internet of Things. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 3(1):1–38, 2019.

[12] D. Giese et al. dustcloud, 2022. https://github.com/dgiese/dustcloud.

[13] T. Giorgino. Computing and Visualizing Dynamic Time Warping Alignments in R: the DTW Package. *Journal of Statistical Software*, 31(1):1–24, 2009.

[14] D. Han, Y. Chen, T. Li, R. Zhang, Y. Zhang, and T. Hedgpeth. Proximity-proof: Secure and Usable Mobile Two-factor Authentication. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*, pages 401–415, 2018.

[15] J. Han, A. J. Chung, M. K. Sinha, M. Harishankar, S. Pan, H. Y. Noh, P. Zhang, and P. Tague. Do You Feel What I Hear? Enabling Autonomous IoT Device Pairing Using Different Sensor Types. In *2018 IEEE Symposium on Security and Privacy (SP)*, pages 836–852, 2018.

[16] M. Isom et al. TuyAPI, 2022. https://github.com/codetheweb/tuyapi.

[17] N. Karapanos, C. Marforio, C. Soriente, and S. Capkun. Sound-proof: Usable Two-factor Authentication Based on Ambient Sound. In *24th {USENIX} Security Symposium ({USENIX} Security 15)*, pages 483–498, 2015.

[18] P. Kietzmann, T. C. Schmidt, and M. Wählisch. A Guideline on Pseudorandom Number Generation (PRNG) in the IoT. *ACM Computing Surveys (CSUR)*, 54(6):1–38, 2021.

[19] K. Lee, N. Klingensmith, S. Banerjee, and Y. Kim. 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):1–26, 2019.

[20] K. Lee, N. Klingensmith, D. He, S. Banerjee, and Y. Kim. ivPair: Context-based Fast Intra-vehicle Device Pairing for Secure Wireless Connectivity. In *Proceedings of the 13th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, pages 25–30, 2020.

[21] Q. Lin, W. Xu, J. Liu, A. Khamis, W. Hu, M. Hassan, and A. Seneviratne. H2B: Heartbeat-based Secret Key Generation Using Piezo Vibration Sensors. In *Proceedings of the 18th International Conference on Information Processing in Sensor Networks*, pages 265–276, 2019.

[22] D. Liu, J. Chen, Q. Deng, A. Konate, and Z. Tian. Secure Pairing with Wearable Devices by Using Ambient Sound and Light. *Wuhan University Journal of Natural Sciences*, 22(4):329–336, 2017.

[23] Y. Liu, X. Ni, Y. Wu, and W. Zhang. Study on Effect of Temperature and Humidity on the CO2 Concentration Measurement. In *IOP Conference Series: Earth and Environmental Science*, volume 81, page 012083, 2017.

[24] Y. Lu, F. Wu, Q. Huang, S. Tang, and G. Chen. Telling Secrets in the Light: An Efficient Key Extraction Mechanism Via Ambient Light. *IEEE Transactions on Wireless Communications*, 20(1):186–198, 2020.

[25] A. Maiti and M. Jadliwala. Light Ears: Information Leakage via Smart Lights. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 3(3):1–27, 2019.

[26] R. Mayrhofer and S. Sigg. Adversary Models for Mobile Device Authentication. *ACM Computing Surveys (CSUR)*, 54(9):1–35, 2021.

[27] S. Mei, Z. Liu, Y. Zeng, L. Yang, and J. F. Ma. Listen!: Audio-based Smart IoT Device Pairing Protocol. In *2019 IEEE 19th International Conference on Communication Technology (ICCT)*, pages 391–397, 2019.

[28] M. Miettinen, N. Asokan, T. D. Nguyen, A.-R. Sadeghi, and M. Sobhani. Context-based Zero-interaction Pairing and Key Evolution for Advanced Personal Devices. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pages 880–891, 2014.

[29] M. Miettinen, T. D. Nguyen, A.-R. Sadeghi, and N. Asokan. Revisiting Context-based Authentication in IoT. In *Proceedings of the 55th Annual Design Automation Conference*, pages 1–6, 2018.

[30] B. C. Ross. Mutual Information between Discrete and Continuous Data Sets. *PloS One*, 9(2):e87357, 2014.

[31] D. Schürmann, A. Brüsch, S. Sigg, and L. Wolf. BANDANA—Body Area Network Device-to-device Authentication Using Natural Gait. In *2017 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 190–196, 2017.

[32] B. Shrestha, N. Saxena, H. T. T. Truong, and N. Asokan. Drone to the Rescue: Relay-resilient Authentication Using Ambient Multi-sensing. In *International Conference on Financial Cryptography and Data Security*, pages 349–364, 2014.

[33] B. Shrestha, N. Saxena, H. T. T. Truong, and N. Asokan. Sensor-based Proximity Detection in the Face of Active Adversaries. *IEEE Transactions on Mobile Computing*, 18(2):444–457, 2018.

[34] B. Shrestha, M. Shirvanian, P. Shrestha, and N. Saxena. The Sounds of the Phones: Dangers of Zero-effort Second Factor Login Based on Ambient Audio. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 908–919, 2016.

[35] Signify Holding. Lights API, 2022. https://developers.meethue.com/develop/hue-api/lights-api/.

[36] Tensorflow.org. Text Generation with an RNN, 2022. https://www.tensorflow.org/text/tutorials/text_generation.

[37] J. Tillmanns, J. Classen, F. Rohrbach, and M. Hollick. Firmware Insider: Bluetooth Randomness is Mostly Random. In *Proceedings of the 14th USENIX Conference on Offensive Technologies*, 2020.

[38] H. T. T. Truong, X. Gao, B. Shrestha, N. Saxena, N. Asokan, and P. Nurmi. Comparing and Fusing Different Sensor Modalities for Relay Attack Resistance in Zero-interaction authentication. In *2014 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 163–171, 2014.

[39] H. T. T. Truong, J. Toivonen, T. D. Nguyen, C. Soriente, S. Tarkoma, and N. Asokan. DoubleEcho: Mitigating Context-manipulation Attacks in Copresence Verification. In *2019 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 1–9, 2019.

[40] M. S. Turan, E. Barker, J. Kelsey, K. A. McKay, M. L. Baish, M. Boyle, et al. Recommendation for the Entropy Sources Used for Random Bit Generation. *NIST Special Publication*, 800(90B):102, 2018.

[41] J. West, K. Lee, S. Banerjee, Y. Kim, G. K. Thiruvathukal, and N. Klingensmith. Moonshine: An Online Randomness Distiller for Zero-involvement Authentication. In *Proceedings of the 20th International Conference on Information Processing in Sensor Networks*, pages 93–105, 2021.

[42] W. Xu, J. Zhang, S. Huang, C. Luo, and W. Li. Key Generation for Internet of Things: A Contemporary Survey. *ACM Computing Surveys (CSUR)*, 54(1):1–37, 2021.