Not-so-Secret Authentication: The SyncBleed Attacks and Defenses for Zero-Involvement Authentication Systems

Isaac Ahlgren*, Rushikesh Shirsat*, Omar Achkar†, George K Thiruvathukal*, Kyu In Lee† and Neil Klingensmith*
†Department of Information Science Technology, University of Houston, Houston, Texas 77204
*Department of Computer Science, Loyola University Chicago, Chicago, Illinois 60660
Email: †{oachkar, klee48}@cougarnet.uh.edu, *{iahlgren, rshirsat, gthiruvathukal, nklingensmith}@luc.edu

2025 IEEE International Conference on Cyber Security and Resilience (CSR) | 979-8-3315-3591-9/25/$31.00 ©2025 IEEE | DOI: 10.1109/CSR64739.2025.11130070

***Abstract*—Zero-involvement authentication (ZIA) offers a promising solution for autoprovisioning large IoT device networks by enabling devices to extract identical authentication keys from ambient environmental signals without user intervention. However, we demonstrate that existing ZIA systems leak critical information during key negotiation when they exchange synchronization messages over public wireless channels. Our novel passive attack, SyncBleed, exploits these leaked messages to reconstruct ZIA-generated keys, successfully cracking approximately 50% of keys in under one second in our testbed experiments. To address this vulnerability, we introduce TREVOR (Time shift REsistant VEctor ExtractOR), which generates nearly identical bit sequences from environmental signals without exchanging any synchronization information. TREVOR produces keys in under 4 seconds with 90–95% bit agreement rates between legitimate devices across various environmental sources, while maintaining complete resistance to SyncBleed attacks.**

I. INTRODUCTION

Internet of Things (IoT) devices must establish secure wireless communication channels to protect sensitive data and enable reliable coordination. Security is not optional for these systems—without proper safeguards, IoT networks become vulnerable to man-in-the-middle attacks [9], [14] that can compromise user privacy, expose confidential information, and undermine the entire system's trustworthiness. The foundation of secure communication begins with device authentication, where IoT devices must first establish shared cryptographic keys before they can communicate securely with one another.

Traditionally, IoT devices establish security by individually authenticating with a trusted central hub or gateway. This process typically requires human intervention, where a user must manually enter passwords for each device—a method that becomes increasingly inefficient and error-prone as networks scale. Zero-involvement authentication (ZIA) addresses these limitations by eliminating the need for human-generated passwords. In ZIA networks, devices establish their legitimacy by demonstrating physical co-presence within the same secured environment, such as an office or home. These devices simultaneously sample ambient environmental signals including electromagnetic radiation, audio patterns, voltage fluctuations, and other contextual data, that are naturally confined to their shared physical space. This approach creates a security boundary based on physical presence, as external entities

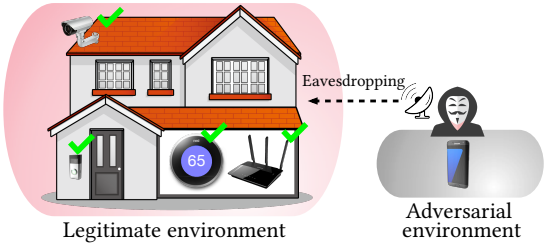

Fig. 1: Threat model of SyncBleed where external adversary attempts to eavesdrop on synchronization messages to gain unauthorized access.

located outside the protected area experience fundamentally different environmental contexts and cannot generate identical authentication keys.

ZIA systems agree on a key by first sampling a time series of an environmental signal, then synchronizing their sample buffers, and finally extracting a key from the synchronized buffers. Without perfectly synchronized sample buffers, legitimate devices often fail to authenticate, as we demonstrate in §II. The synchronization protocol used by most ZIA systems is extremely reliable as it works by sharing a snippet of the environmental context sampled over a public wireless channel [8], [7], [17], [15], [16]. Although external eavesdroppers can intercept these synchronization snippets, previous researchers have assumed that this technique remains secure because the snippet is discarded and never incorporated directly into the key. This security model assumes that the synchronization messages do not leak information about the key or the legitimate space.

In this work, we first present a novel attack that reveals ZIA keys solely by intercepting synchronization messages on the unencrypted public channel and recording environmental context from a distance (shown in Fig. 1). Our attack, which we call SyncBleed, specifically targets the synchronization phase of ZIA key generation and extracts crucial information from synchronization messages to substantially narrow the key space. In our testbed using state-of-the-art ZIA algorithms, our attack successfully identified approximately 50% of the generated keys in less than a second.

To mitigate this security vulnerability, we developed TREVOR (Time shift REsistant VEctor ExtractOR), a novel authentication approach that fundamentally reimagines key generation for IoT devices. Unlike existing solutions, TREVOR requires no time synchronization mechanisms—eliminating dependence on atomic clocks, GPS, or other specialized hardware that would be impractical for low-cost IoT deployments[18], [6]. By operating without exchanging any synchronization messages, TREVOR removes the attack vector exploited by SyncBleed while simultaneously addressing privacy concerns associated with transmitting environmental context samples. Our evaluation demonstrates that TREVOR achieves exceptional performance metrics: it generates authentication keys in under five seconds, requiring no inter-device communication during the bit quantization process. This silent operation not only enhances security but also significantly improves efficiency compared to conventional ZIA protocols. TREVOR maintains high reliability with 90–95% bit agreement rates between legitimate devices while completely rejecting unauthorized authentication attempts. These performance characteristics make TREVOR competitive with leading ZIA implementations [15], [17], [27], [19] while providing complete immunity to SyncBleed attacks. The contributions of this work include the following:

- *Threat Model:* We introduce a realistic threat model for ZIA systems that assumes attackers can remotely measure environmental context in the legitimate space.
- *SyncBleed:* We introduce a passive attack on ZIA systems that exploits synchronization messages to guess the key.
- *Mitigation:* We present TREVOR, a practical, general-purpose mitigation for SyncBleed. TREVOR is a bit quantization technique for ZIA systems that tolerates misalignment. TREVOR is not susceptible to SyncBleed because it does not exchange synchronization messages.
- *Evaluation:* We present a comprehensive evaluation TREVOR on a diverse group of existing datasets.

II. THE SYNCBLEED ATTACK

In this Section, we present SyncBleed, a passive attack that exploits synchronization messages in ZIA systems to reconstruct authentication keys. We demonstrate that synchronization messages, previously assumed to be secure, leak substantial information that allows an adversary to accurately estimate the environmental context within a legitimate space.

A. Threat Model

SyncBleed operates under a threat model where legitimate devices reside within a physically secured environment (such as an office, home, or apartment), while adversaries have access to adjacent spaces. Unlike previous security assumptions in ZIA systems that presumed adversaries have no access to environmental context, our model recognizes that attackers can measure approximations of the legitimate context through shared walls, windows, or other boundaries. Common attack scenarios include: (1) Office Suite Neighbor—a tenant attempting to access a neighboring suite's ZIA network; (2) Trespasser—an attacker placing recording devices near building windows; and (3) Party Scenario—an adversary in a neighboring apartment recording audio through shared walls during a loud event. Adversaries in our model can intercept all public wireless transmissions, masquerade as legitimate devices, and possess computational resources including GPUs for neural network training. Their goal is either passive key discovery through eavesdropping or active participation in the authentication protocol while appearing legitimate.

B. Attack Implementation

SyncBleed is implemented as a passive attack targeting the synchronization phase of ZIA systems. To test its effectiveness, we established a controlled test environment with a legitimate device placed inside an office and an adversarial device positioned outside the office's closed door (Fig. 5). The legitimate device used the Schurmann & Sigg [29] bit quantization algorithm, which is commonly implemented in ZIA systems. Inside the office, we played a YouTube video [24] simulating conversation, while the adversarial device captured the audio as it filtered through the door barrier. The physical barrier between these environments acts as a selective filter that differentially affects the frequency components of the environmental signal. This filtering effect is key to the attack as it creates a predictable transformation that can be modeled mathematically.

The adversary leverages two data sources to implement the attack: the distorted environmental signal recorded from outside the legitimate space and the synchronization messages intercepted from public wireless channels. Using these inputs, we trained a four-layer fully connected generative neural network on the Fast Fourier Transform (FFT) of the adversarial audio, with the legitimate audio's FFT as the target. After training on 1024 synchronization samples, the model can accurately estimate the transfer function between spaces. This function enables the adversary to reconstruct the legitimate environmental signal using the formula $E(\omega) = \frac{M(\omega)}{H(\omega)}$, where $E(\omega)$ represents the estimated audio inside the legitimate space, $M(\omega)$ is the muffled signal measured outside, and $H(\omega)$ is the estimated transfer function. The adversary then applies an inverse FFT to recover the time-domain signal and generates a key using the same bit quantization algorithm as legitimate devices. To validate the generated key without communication with legitimate devices, the adversary can apply Reed-Solomon reconciliation locally, providing confirmation of successful key extraction.

C. Attack Effectiveness

Our experimental results demonstrate SyncBleed's considerable effectiveness. As illustrated in Fig. 2, an adversary without using our technique generates keys with a mean bit error rate of 41.7% compared to legitimate devices. With SyncBleed, this error rate drops significantly to 28.7%, enabling adversaries to generate keys with less than 30% bit error rate more than 50% of the time—sufficient for successful

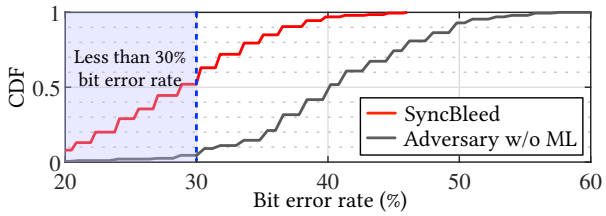

Fig. 2: CDF of of bit error rates between standard adversarial approaches and SyncBleed. The SyncBleed attack significantly reduces bit error rates to below 30%.

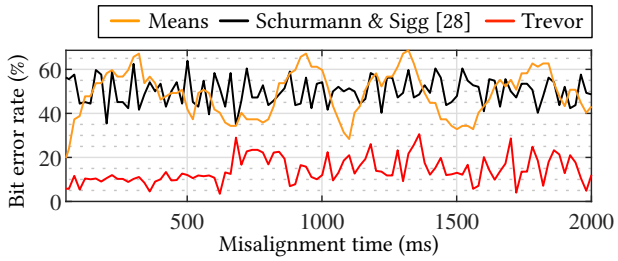

Fig. 3: The Means and Schurmann & Sigg bit quantization algorithms produce high bit error rate between misaligned bit sequences while TREVOR achieves lower bit error rate.

authentication after error correction through Reed-Solomon reconciliation.

The attack requires only basic knowledge of neural networks and signal processing, making it accessible to many potential adversaries. Moreover, the estimated transfer function remains effective even as legitimate devices rotate their keys, creating a persistent vulnerability. While our passive data collection required approximately 17 hours to gather 1024 training samples, we found that by actively inducing key generation events in the legitimate network, an adversary could collect the same training data in under five minutes.

D. Shift Intolerance of Existing ZIA Protocols

The fundamental vulnerability exploited by SyncBleed stems from existing ZIA bit quantization algorithms' sensitivity to time shifts. Fig. 3 demonstrates how even minimal misalignments between audio signals result in unacceptably high bit error rates for common algorithms like Means and Schurmann & Sigg. This temporal sensitivity forces ZIA systems to exchange synchronization information over public channels—precisely the vulnerability that SyncBleed exploits. Our TREVOR solution, which we present in subsequent section, addresses this core issue by generating shift-invariant keys that remain consistent despite timing misalignments. By eliminating the need for synchronization messages entirely, TREVOR removes the attack vector exploited by SyncBleed while maintaining the usability and convenience benefits of ZIA.

III. THE TREVOR PROTOCOL

TREVOR (Time shift REsistant VEctor ExtractOR) establishes shared cryptographic keys between two un-trusting devices A and B that seek to authenticate each other based on their shared physical environment. While two devices must be within radio transmission range, we assume adversaries can also access this channel, intercepting all messages. In this scenario, A and B use TREVOR to prove to one another that they are located in the same physical space and therefore legitimate, as illustrated in Figure 4.

Noise Harvesting The TREVOR key generation process begins with a simple initiation message followed by concurrent environmental sampling also known as noise harvesting. Device A typically initiates the process by transmitting a message to device B, after which both begin recording ambient audio through their microphones. Because of inherent latency in wireless communications, two devices will not start sampling at precisely the same moment—a misalignment that would cause traditional authentication schemes to fail. TREVOR overcomes this challenge through its frequency-domain approach.

Bit Quantization The sequence of time-domain audio samples is then subdivided into blocks of length d. Each block of time-domain samples is then Fourier transformed, discarding phase. Frequency components of the Fourier spectrum are binned together to build a coarse-grained estimate of the frequency spectrum. Binning reduces the effects of measurement error and time domain shifts between nearby legitimate devices. The remaining magnitude information is shift-invariant for shifts that are small compared to the buffer length. Adversaries located outside the physically-secured space will measure substantially different frequency spectra because walls and doors will selectively muffle frequencies. The Fourier transformed blocks are arranged as rows of a data matrix X. TREVOR discards the first frequency bin, which usually contains some high-amplitude low-frequency components that are common to many kinds of signals. Low-frequency high-amplitude frequency components tend to dominate the eigenvectors without providing unique context. The number of rows in X is controlled by the amount of audio acquired during warmup and sampling. In general, more rows in X give better pairing reliability for legitimate devices.

The Fourier transformed observation matrix X is used to calculate the covariance matrix $C = X^T X$ between each frequency component. Shifts in the original time-domain signal will not propagate to significant differences in C because the rows of X do not vary much under small time shifts. From C, we extract 4-8 dominant eigenvectors by repeatedly using the power method followed by deflating the matrix. Although in general eigenvector problems tend to be computationally intensive, in practice it is manageable on a modest CPU. TREVOR computes a covariance matrix on the order of dimension 32×32 which can comfortably fit in memory even on a microcontroller.

Covariance matrices built by devices A and B are slightly different. The resulting eigenstructure of C_A and C_B will also be slightly different. But as long as the underlying audio signals are similar, the dominant eigenvectors will also be similar. The first 4–8 eigenvectors are usually similar enough to be useful in key generation. By including more eigenvectors,

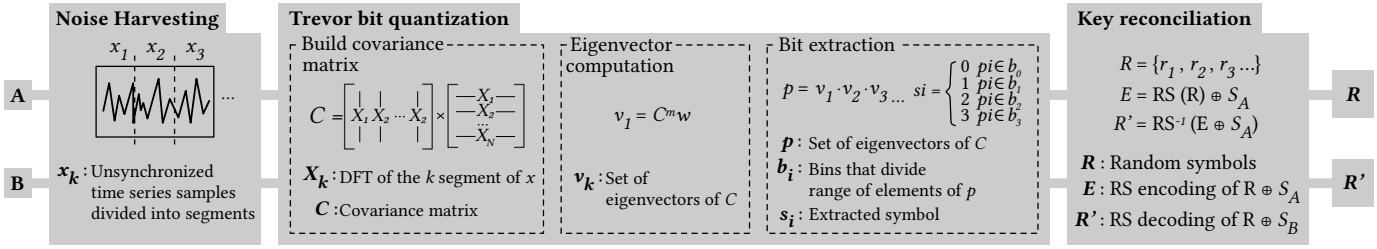

Fig. 4: TREVOR protocol diagram to establish a key.

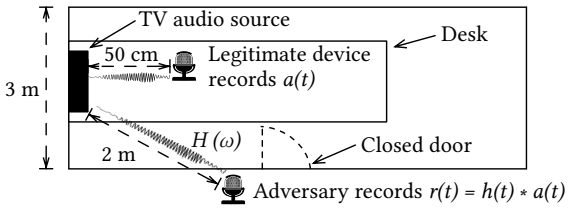

Fig. 5: A floorplan of testbed used to evaluate TREVOR.

we can extract more bits at the expense of a higher bit error rate. The most common problem that arises is that two similar covariance matrices have corresponding eigenvectors that point along the same dimension in opposite directions. Our solution to this problem is to force the sign of the largest component in absolute value of each eigenvector to be positive.

To build a bit sequence from the dominant eigenvectors of the covariance matrix, we append the components of the eigenvectors together into one large array p. We bin the elements of the array into four equally-sized bins that span the range of p. We assign a two-bit symbol s_i to each component of p based on which bin it lives in.

Key Reconciliation We use the sequence of symbols produced by TREVOR to encrypt a randomly-generated key R, and transmit that key from one device to another. Key R is encoded with a Reed-Solomon[12] error correcting code. Then the coded symbols produced by TREVOR are added to the encoded R to form an encrypted message that can only be decrypted with a similar sequence of symbols.

$$E = RS(R) \oplus S_A$$

The resulting sequence E is then transmitted over a public channel. The other device then reconstructs an estimate of R by adding its locally-generated sequence of symbols:

$$R' = RS^{-1}(E \oplus S_B)$$

If $S_A \approx S_B$, the Reed-Solomon decoder will correct the bit stream to get $R' = R$, which can be used as an encryption or authentication token.

IV. EVALUATION

We developed a comprehensive prototype system to evaluate TREVOR in real-world conditions using ambient audio. Our prototype implementation consists of Raspberry Pi 4 [2] modules running Debian Linux, carefully chosen to mirror the capabilities of mid-range IoT devices. Each Pi features an ARM Cortex-A72 CPU operating at 1.8 GHz with 2 GB of RAM, providing sufficient computational power while maintaining realistic resource constraints. For environmental sensing, we equipped each node with a HyperX SoloCast USB microphone [4] sampling at 48 kHz. Fig. 5 illustrates our experimental testbed setup, showing the physical arrangement of legitimate and adversarial devices used in our evaluation.

Throughout these experiments, we generated environmental audio by playing two types of content through room speakers: conversation audio from YouTube videos [24] and music selections [30], representing common acoustic environments where IoT devices operate. To ensure comprehensive testing beyond our controlled setup, we supplemented these experiments with evaluations using the public DEMAND dataset [32]. DEMAND is a diverse dataset with realistic ambient audio recordings from various residential environments. This additional testing allowed us to validate TREVOR's effectiveness in authentic home settings without requiring additional physical deployments.

A. Resistance to Shifting

We first evaluate TREVOR's resilience to timing misalignments by measuring bit error rates between devices with deliberately misaligned audio buffers. This test simulates real-world conditions where perfect synchronization is unachievable due to network latency and processing variations. As shown in Fig. 6, we examine three device configurations: closely positioned legitimate devices (0.5 m), moderately distant legitimate devices (2 m), and an adversarial device placed outside the secured environment. The results demonstrate TREVOR's robust performance across various environmental signals and timing offsets.

For conversation audio (Fig. 6(a)), legitimate devices maintained mean bit error rates below 25% even with substantial misalignments, while the adversary consistently produced error rates above 40%. Music audio (Fig. 6(b)) showed even better performance, with legitimate devices achieving remarkably consistent mean error rates under 15% across the entire two-second misalignment range. The DEMAND dataset results (Fig. 6(c)) confirmed similar patterns in natural home environments. The clear separation between legitimate and adversarial device performance persists across all tested conditions, providing strong evidence that TREVOR can reliably authenticate legitimate devices while rejecting unauthorized

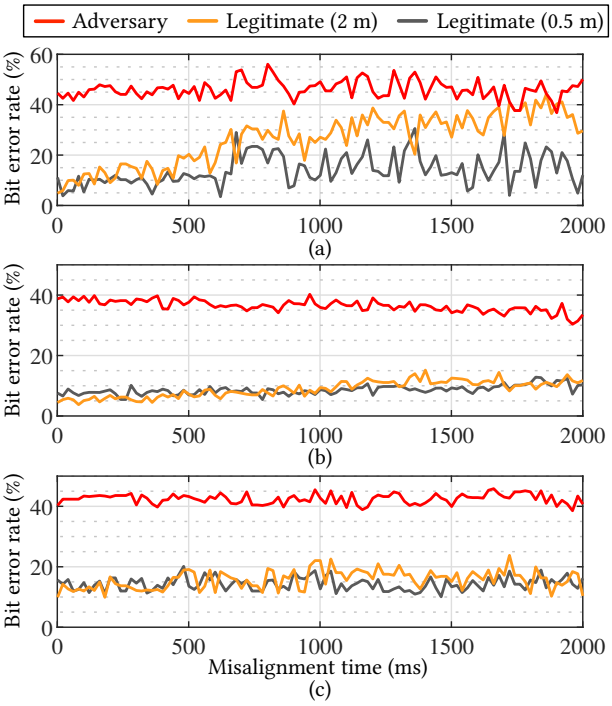

Fig. 6: Bit error rate as a function of shift amount for (a) conversation audio, (b) music audio, (c) DEMAND living room ambient audio.

TABLE I: Authentication success rate of TREVOR.

SIGNAL TYPE	Legitimate (0.5m)	Legitimate (2m)	Adversary
Conversation Audio	100%	87%	0%
Cooking Audio	100%	70%	0%

access attempts, even without the synchronization messages that create vulnerabilities in existing ZIA systems.

B. Authentication Success Rate

We next implemented Reed-Solomon key reconciliation to evaluate authentication outcomes in practical deployment scenarios. Using a symbol length of 8 bits and an error correction threshold of 12.5%, we measured the success rates for legitimate and adversarial authentication attempts across different environmental contexts.

Table I presents the authentication success rates for TREVOR across various audio environments. For conversation audio, nearby legitimate devices (0.5 m) achieved perfect authentication success (100%), while more distant legitimate devices (2 m) maintained a strong 87% success rate. Similarly, with cooking audio, close-proximity devices maintained 100% success rates while devices at 2 m distance achieved 70% successful authentications. Most importantly, adversarial devices were completely blocked in all test scenarios, with 0% authentication success regardless of the audio environment. These results confirm that TREVOR maintains a substantial security margin between legitimate and adversarial authentication attempts. The clear separation in bit error rates (shown in Fig. 6) translates directly to binary authentication outcomes: legitimate devices can reliably authenticate while adversaries are consistently rejected. This performance profile makes TREVOR particularly valuable for real-world IoT deployments where devices may be positioned at varying distances within a secured space.

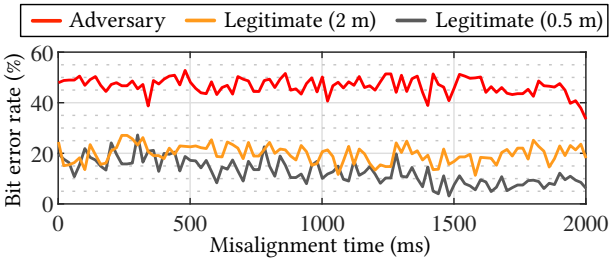

Fig. 7: Bit error rates under replay attack. Legitimate devices successfully authenticate using current ambient audio, while an adversary with replay attack fails to authenticate.

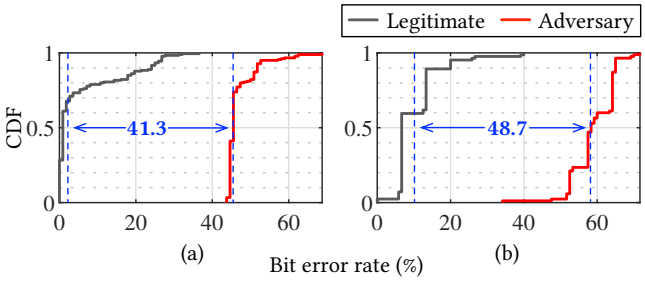

Fig. 8: CDF of bit error rates on (a) conversation audio and (b) cleaning audio

C. Replay Attack

We next evaluated TREVOR's resilience against replay attacks, a sophisticated threat where adversaries with temporary legitimate access attempt to authenticate later using previously recorded environmental data. To simulate this attack vector, we recorded ambient cooking audio within our secure test environment on one day, then attempted to use these recordings for authentication on a subsequent day. As shown in Fig. 7, the results demonstrate TREVOR's strong resistance to such attacks. Legitimate devices operating within the secured environment maintained bit error rates between 10-25%, well below our Reed-Solomon correction threshold. In contrast, the adversary attempting to authenticate using previously recorded audio consistently produced bit error rates around 50%—essentially random chance—regardless of timing alignment attempts. This robust security property stems from TREVOR's frequency-domain analysis approach. TREVOR successfully identifies these temporal differences and rejects authentication attempts based on outdated environmental context, even when the recording quality is high and the physical location is identical.

TABLE II: Bit error rate for other ZIPA systems. Our prototype of TREVOR achieves comparable bit error rates without exchanging synchronization messages.

SYSTEM	Bit Error Rate
VOLTKEY [15]	5%
AEROKEY [17]	9%
SECRET FROM MUSCLE [34]	8%
IVPAIR [16]	5%
PROXIMATE [21]	3-50%
TREVOR	**5-10%**

D. Bit Error Rate in Real-Time Prototype Implementation

To validate TREVOR's performance under realistic conditions, we conducted extensive testing using our hardware prototype. We configured two experimental scenarios: a legitimate device-reference pair positioned inside our lab with the door closed, and an adversary-reference pair with the adversary positioned immediately behind the closed door. All devices remained within 2 meters of each other to eliminate distance as a variable. During testing, we played approximately 40 minutes of diverse YouTube content inside the lab while continuously generating authentication keys. Fig. 8 presents the cumulative distribution functions (CDFs) of bit error rates for both legitimate and adversarial authentication attempts. For conversation audio Fig. 8(a), legitimate devices consistently achieved bit error rates below 20%, while the adversary's attempts clustered around 45–50%. This created a substantial 41.3% separation in mean bit error rates, providing a robust security margin. When testing with cleaning audio (Fig. 8(b)), which features more monotonous sound patterns from continuous vacuuming, legitimate devices maintained bit error rates below 20% while the adversary's performance worsened further. The resulting 48.7% difference in mean bit error rates demonstrates TREVOR's ability to maintain security even with less varied environmental audio. Table II shows that TREVOR compares favorably to other ZIA systems in terms of bit error rate.

These results confirm that TREVOR can effectively distinguish between legitimate and adversarial devices in real-time deployments, even when the adversary is physically adjacent to the secured space.

E. TREVOR Protocol Latency

We evaluated TREVOR's computational efficiency to assess its suitability for resource-constrained IoT devices. Fig. 9 illustrates TREVOR's significant performance advantage compared to existing ZIA systems like VoltKey [15]. TREVOR's streamlined communication architecture requires only two messages (initiation and key reconciliation), while VoltKey exchanges 11 messages including kilobyte-sized data transfers. This reduction in network communication provides a substantial speed advantage in real-world deployment scenarios where network interfaces may be unreliable. Excluding audio acquisition time,

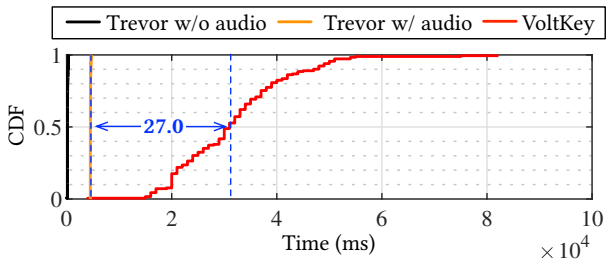

Fig. 9: Evaluation of TREVOR's overall run time on our Raspberry Pi based system.

TABLE III: Evaluation of TREVOR running on Cortex-M4 microcontroller without collecting audio.

TREVOR Runtime	3200 ms
Code (flash) size	161 kbytes
Data (RAM) size	11 kbytes

TREVOR processes keys in under 10 milliseconds demonstrating a 27.0 millisecond mean improvement over VoltKey.

F. Runtime on Cortex-M4 Microcontroller

Here we evaluate the time to compute a key on a Cortex-M4 microcontroller, not including time taken to acquire audio. We provided a pre-recorded time-domain audio buffer to our algorithm and timed the key generation process on the microcontroller. Our goal in this experiment is to determine if TREVOR is capable of running on a heavily resource-constrained platform.

We implemented the TREVOR bit extraction algorithm on a SAME54 Xplained Pro board, which has an ARM Cortex-M4 CPU. The SAME54 microcontroller [3] we used for this experiment has a single-precision floating point unit and runs at 50 MHz. The platform has 256 kbytes of SRAM and 1 MByte of flash. We implemented the entire TREVOR protocol except for audio sampling and message exchange, which requires an audio input and network interface that are not available on the SAME54 Xplained platform. Our microcontroller software generates a key from 2.7 seconds of audio with 128k audio samples. The software we evaluated is a C-language port of the one shown in Figure 5. We used ARM's CMSIS-DSP library [1] for basic linear algebra operations, and we ported our eigenvector decomposition software to work with that library.

Table III shows a summary of TREVOR's performance on the SAME54 microcontroller platform. Our algorithm takes about 3.2 seconds to compute a key on the Cortex-M4 platform. Although it takes much longer to compute a key on the microcontroller, it appears that TREVOR could be made to work on a full microcontroller-based system.

V. RELATED WORK

Bit Distillation Bit distillation techniques transform input bit streams into more random output sequences, a process fundamental to many cryptographic applications. Bit distillation algorithms take a bit stream as input and outputs a

more random bit stream [33]. Common uses for bit distillation are applications where the platform is restricted to sampling randomness from a low-entropy source [19]. The problem that bit distillation aims to solve is low randomness found in bit streams. While these approaches focus primarily on enhancing randomness quality, TREVOR addresses a different challenge by creating consistently matching bit sequences from environmental signals despite timing variations.

Fuzzy Extractors When sampling from an environmental noise source, there are cases where noise gathered from the same scene from two different perspectives does not match. Fuzzy extractors are algorithms that can account for different perspectives or slightly different readings from the environment to allow for stable use in cryptographic algorithms [13], [11]. These algorithms are commonly used for biometric identification [20], [28]. TREVOR differs from the work in fuzzy extractors in that it generates a single matching key rather than allowing the key to differ.

Zero Involvement and Authentication (ZIA) Our work is similar to other ZIA works because we rely on shared context to generate a key. Other schemes rely on shared entropy seen by at least two devices. ZIA is commonly used in IoT systems[23], [8], [10], autonomous vehicles[35], and smart home security [31] to name a few applications. TREVOR diverges from conventional implementations by eliminating time synchronization requirements while maintaining authentication reliability. Our work differs from common ZIA implementations because ZIA schemes are reliant on the time domain for synchronous key exchanges [22], [25], [26]. We are aware of only one other paper [5] that implemented a non-trivial attack on ZIA systems.

VI. CONCLUSION

In this work, we presented TREVOR, a solution to critical vulnerabilities in ZIA systems. We first demonstrated SyncBleed, a novel attack that exploits synchronization messages to reconstruct approximately 50% of authentication keys in under one second, representing the first proven security breach in ZIA systems. TREVOR eliminates this vulnerability by generating keys from environmental signals without requiring synchronization messages. By applying principal component analysis to frequency-domain representations, it creates timing-invariant keys that remain consistent despite device misalignments. Our evaluation shows TREVOR generates keys in under 4 seconds with 90–95% bit agreement rates between legitimate devices while completely rejecting adversarial attempts. Beyond security improvements, TREVOR enhances efficiency through minimal network transmissions and protects privacy by eliminating environmental context sharing. These characteristics make TREVOR a practical, secure solution for IoT authentication that maintains the convenience of zero-involvement approaches while establishing a new standard for security in real-world deployments.

VII. ACKNOWLEDGEMENTS

This work was supported by National Centers of Academic Excellence in Cybersecurity(NCAE) H98230-22-1-0306; National Science Foundation(NSF) 2107020; and National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. RS-2024-00463802).

REFERENCES

[1] Cmsis dsp software library. https://arm-software.github.io/CMSIS_5/DSP/html/index.html.

[2] Raspberry pi 4. https://www.raspberrypi.com/products/raspberry-pi-4-model-b/specifications/.

[3] Sam e54 xplained pro user's guide. https://ww1.microchip.com/downloads/en/DeviceDoc/70005321A.pdf. Accessed: 2022-06-21.

[4] Solocast - usb gaming microphone. https://hyperx.com/products/hyperx-solocast-usb-microphone?variant=41031679312029. Accessed: 2022-06-14.

[5] Isaac Ahlgren, Jack West, Kyuin Lee, George Thiruvathukal, and Neil Klingensmith. A Signal Injection Attack Against Zero Involvement Pairing and Authentication for the Internet of Things . In *2024 IEEE Workshop on Design Automation for CPS and IoT (DESTION)*, pages 9–15, Los Alamitos, CA, USA, May 2024. IEEE Computer Society.

[6] Djamel Djenouri and Miloud Bagaa. Synchronization protocols and implementation issues in wireless sensor networks: A review. *IEEE Systems Journal*, 10(2):617–627, 2016.

[7] Mikhail Fomichev. fastzip, 2021.

[8] Mikhail Fomichev, Julia Hesse, Lars Almon, Tim Lippert, Jun Han, and Matthias Hollick. Fastzip: faster and more secure zero-interaction pairing. In *MobiSys '21: Proceedings of the 19th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '21, pages 440–452. Association for Computing Machinery, 2021.

[9] Mikhail Fomichev, Flor Álvarez, Daniel Steinmetzer, Paul Gardner-Stephen, and Matthias Hollick. Survey and systematization of secure device pairing. *IEEE Communications Surveys & Tutorials*, 20(1):517–550, 2018.

[10] Jun Han, Albert Jin Chung, Manal Kumar Sinha, Madhumitha Harishankar, Shijia Pan, Hae Young Noh, Pei Zhang, and Patrick 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)*, 2018 Symposium on Security and Privacy (SP), pages 836–852, New York, NY, USA, 2018. Institute of Electrical and Electronics Engineers.

[11] Ari Juels and Madhu Sudan. A fuzzy vault scheme. *Designs, Codes and Cryptography*, 38(2):237–257, 2006.

[12] Ari Juels and Martin Wattenberg. A fuzzy commitment scheme. In *Proceedings of the 6th ACM Conference on Computer and Communications Security*, CCS '99, page 28–36, New York, NY, USA, 1999. Association for Computing Machinery.

[13] Ari Juels and Martin Wattenberg. A fuzzy commitment scheme. In *Proceedings of the 6th ACM conference on Computer and communications security*, pages 28–36, 1999.

[14] Arun Kumar, Nitesh Saxena, Gene Tsudik, and Ersin Uzun. Caveat eptor: A comparative study of secure device pairing methods. In *2009 IEEE International Conference on Pervasive Computing and Communications*, pages 1–10, 2009.

[15] Kyuin Lee, Neil Klingensmith, Suman Banerjee, and Younghyun 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), September 2019.

[16] Kyuin Lee, Neil Klingensmith, Dong He, Suman Banerjee, and Younghyun 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*, WiSec '20, page 25–30, New York, NY, USA, 2020. Association for Computing Machinery.

[17] Kyuin Lee, Yucheng Yang, Omkar Prabhune, Aishwarya Lekshmi Chithra, Jack West, Kassem Fawaz, Neil Klingensmith, Suman Banerjee, and Younghyun Kim. Aerokey: Using ambient electromagnetic radiation for secure and usable wireless device authentication. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 6(1), mar 2022.

[18] Christoph Lenzen, Philipp Sommer, and Roger Wattenhofer. Optimal clock synchronization in networks. In *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*, SenSys '09, page 225–238, New York, NY, USA, 2009. Association for Computing Machinery.

[19] Qi Lin, Weitao Xu, Jun Liu, Abdelwahed Khamis, Wen Hu, Mahbub Hassan, and Aruna Seneviratne. H2b: Heartbeat-based secret key generation using piezo vibration sensors. In *Proceedings of the International Conference on Information Processing in Sensor Networks*, IPSN '19, pages 265–276, 2019.

[20] Davide Maltoni, Dario Maio, Anil K Jain, and Salil Prabhakar. *Handbook of fingerprint recognition*. Springer Science & Business Media, 2009.

[21] Suhas Mathur, Robert Miller, Alexander Varshavsky, Wade Trappe, and Narayan Mandayam. Proximate: Proximity-based secure pairing using ambient wireless signals. In *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services*, MobiSys '11, pages 211–224, New York, NY, USA, 2011. ACM.

[22] Markus Miettinen, N. Asokan, Thien Duc Nguyen, Ahmad-Reza Sadeghi, and Majid 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*, CCS '14, page 880–891, New York, NY, USA, 2014. Association for Computing Machinery.

[23] Markus Miettinen, Thien Duc Nguyen, Ahmad-Reza Sadeghi, and N Asokan. Revisiting context-based authentication in iot. In *Proceedings of the 55th Annual Design Automation Conference*, pages 1–6, 2018.

[24] RichRoll. Change your brain: Neuroscientist dr. andrew huberman. https://www.youtube.com/watch?v=SwQhKFMxmDY&ab_channel=RichRoll.

[25] Phillip Rieger, Marco Chilese, Reham Mohamed, Markus Miettinen, Hossein Fereidooni, and Ahmad-Reza Sadeghi. ARGUS: Context-Based detection of stealthy IoT infiltration attacks. In *32nd USENIX Security Symposium (USENIX Security 23)*, pages 4301–4318, Anaheim, CA, August 2023. USENIX Association.

[26] Luis Puche Rondon, Leonardo Babun, Ahmet Aris, Kemal Akkaya, and A. Selcuk Uluagac. Survey on enterprise internet-of-things systems (e-iot): A security perspective. *Ad Hoc Networks*, 125:102728, 2022.

[27] Masoud Rostami, Ari Juels, and Farinaz Koushanfar. Heart-to-heart (h2h): authentication for implanted medical devices. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, CCS '13, pages 1099–1112, New York, NY, USA, 2013. ACM.

[28] Amit Sahai and Brent Waters. Fuzzy identity-based encryption. In *Annual international conference on the theory and applications of cryptographic techniques*, pages 457–473. Springer, 2005.

[29] Dominik Schürmann and Stephan Sigg. Secure communication based on ambient audio. *IEEE Transactions on Mobile Computing*, 12(2):358–370, Feb 2013.

[30] The Ink Spots. I don't want to set the world on fire. https://youtu.be/6l6vqPUM_FE.

[31] Shruthi Sreedharan and N Rakesh. Securitization of smart home network using dynamic authentication. In *International Conference on Computer Networks and Communication Technologies*, pages 287–293. Springer, 2019.

[32] Joachim Thiemann, Nobutaka Ito, and Emmanuel Vincent. DEMAND: a collection of multi-channel recordings of acoustic noise in diverse environments, June 2013. Supported by Inria under the Associate Team Program VERSAMUS.

[33] Jack West, Kyuin Lee, Suman Banerjee, Younghyun Kim, George K Thiruvathukal, and Neil Klingensmith. Moonshine: An online randomness distiller for zero-involvement authentication. In *Proceedings of the 20th International Conference on Information Processing in Sensor Networks (co-located with CPS-IoT Week 2021)*, pages 93–105, 2021.

[34] Lin Yang, Wei Wang, and Qian Zhang. Secret from muscle: Enabling secure pairing with electromyography. In *Proceedings of the 14th ACM Conference on Embedded Network Sensor Systems CD-ROM*, SenSys '16, pages 28–41, New York, NY, USA, 2016. ACM.

[35] Mengjia Zeng and Huibin Xu. Mix-context-based pseudonym changing privacy preserving authentication in vanets. *Mobile Information Systems*, 2019, 2019.