RadKey: An LLM-Guided RF Backscatter System for Through-Wall Keystroke Inference

Qijun Wang*, Chunqi Qian†, and Huacheng Zeng*
*Department of Computer Science and Engineering, Michigan State University
†Department of Radiology, Michigan State University

***Abstract*—In today's digitally connected world, keyboards remain the primary interface for inputting sensitive information, making them a persistent target for eavesdropping attacks. While prior keystroke inference techniques have exploited side-channel signals such as acoustics and vibrations, they typically rely on conspicuous, short-range sensors and require victim-specific data for model training, limiting their practicality, scalability, and stealth. In this paper, we present RadKey, an RF backscatter system for covert, long-range, through-wall keystroke eavesdropping. RadKey comprises two components: a compact batteryless backscatter tag and an RF reader. The tag captures keystroke-induced vibrations and acoustic signals, modulating them onto the frequency shift of its backscattered RF signal using two magnetically-coupled LC resonators. This design also enables spectral separation between the excitation and backscatter signals, mitigating self-interference for the RF reader and thus extending eavesdropping range. The RF reader demodulates the backscattered RF signal to infer typed content. It employs a dedicated signal processing pipeline that extracts user- and keyboard-independent keystroke features across time and frequency domains, enabling strong generalizability. To further enhance adaptability, RadKey integrates an LLM for online adaptation, leveraging LLM outputs as pseudo ground-truth labels to refine the classifier during runtime. We have built a prototype of the full RadKey system and evaluated it through extensive over-the-air experiments. Results show that RadKey achieves accurate and robust keystroke inference across diverse users in real-world settings. A demo video is available at: https://radkey-submission.github.io/RadKey/**

1. Introduction

In our digitally interconnected world, the keyboard remains the primary interface for inputting our most sensitive information, including passwords, financial details, confidential communications, and proprietary data. Consequently, keystroke privacy has emerged as a critical aspect of cybersecurity, with eavesdropping attacks presenting a persistent and evolving threat. Traditionally, adversaries exploit various side-channel emanations (e.g., acoustic signals [1], [2], [3], [4], [5], [6], [7], electromagnetic radiation [8], [9], [10], [11], radar-based sensing [12], [13], and mechanical vibrations generated during keystroke events [14], [15], [16], [17]) to infer user inputs. However, despite recent advancements, existing approaches suffer from some fundamental limitations that restrict their practicality for covert and long-term keystroke surveillance.

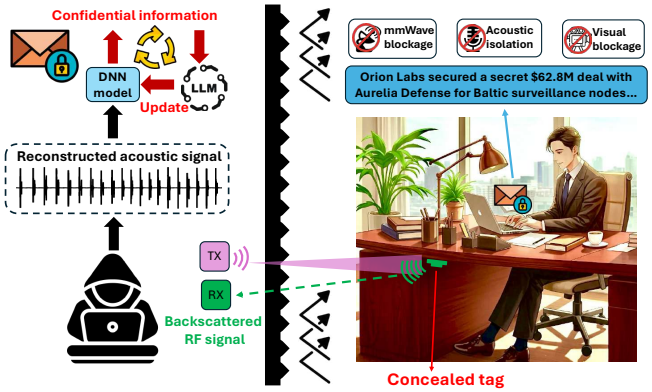


Figure 1: Threat model and system configuration.

Most existing keystroke eavesdropping attacks depend on a strategic placement of conspicuous sensing devices (e.g., microphones or smartphones) in close proximity to the target keyboard [18], [19], [20], [21], [22], [23], [24], [25], [26]. Such setups are not only easily discoverable to victims but also impractical for continuous surveillance due to their dependency on batteries or regular maintenance. Moreover, the effectiveness of attacks typically hinges on extensive user-specific training data collection, requiring attackers to gather large sets of labeled keystroke data from the specific victim and keyboard [18], [26], [27], [28], [29], [30], [31], [32]. This requirement fundamentally restricts the attack's feasibility, adaptability, and scalability. Some attacks additionally suffer from closed-vocabulary constraints (e.g., [2], [18], [33], [34], [35]), where high accuracy is only achieved under the assumption that user input belongs to a small predefined word set, limiting their generalizability in realistic scenarios.

In this paper, we consider a threat model as illustrated in Fig. 1, where an adversary seeks to eavesdrop on keystrokes generated by a victim typing on a physical keyboard or a laptop. Based on the captured keystroke signals, the adversary aims to reconstruct the typed content and infer sensitive information such as messages, search queries, or passwords. To perform this attack, we propose RadKey, an

RF backscatter system composed of two key components: *a passive RF backscatter tag* and *an RF reader*. The tag is covertly placed within the target environment, mounted underneath the table or desk supporting the keyboard. It captures keystroke-induced vibrations and modulates them onto a backscattered RF signal, which is then received by the RF reader. The RF reader, positioned either inside or outside the target space, performs signal processing to demodulate the keystroke-related signal and infer the typed content. In the design of RadKey, we face two challenges.

Challenge #1: RF Tag Design. To enable reliable eavesdropping, the RF backscatter tag has the following requirements. *First*, it must be compact (e.g., under one inch) and easily concealable. *Second*, it must support *continuous* keystroke eavesdropping using only harvested energy. While prior batteryless tags (e.g., WISP [36], Battery-Free Phone [37], MARS [38]) demonstrate RF communication, they operate intermittently with low duty cycles due to limited energy availability. *Third*, the tag must reliably modulate RF signals to ensure long-range and through-wall performance. A common limitation in existing systems (e.g., RFID) is self-interference: when the backscattered signal shares the same frequency as the excitation signal, it overwhelms the reader and severely reduces detection performance.

To meet these requirements, we introduce a novel dual-resonator architecture for the RF backscatter tag. Our proposed tag comprises four elements: an off-the-shelf piezoelectric sensor, a voltage sensing resonator (VSR), a parametric enhancement resonator (PER), and a dipole antenna. The piezoelectric sensor transduces sound and vibration-induced fluctuations into a voltage signal, which changes the capacitance of a diode on the VSR and therefore changes the VSR's resonance frequency. The design of the VSR establishes an approximate linear relationship between the piezoelectric sensor's output voltage and the VSR's resonance frequency change, achieving a direct frequency modulation of the sound/vibration signal. The VSR is magnetically coupled to the PER, which serves as an energy pump and thus amplifies the VSR's resonance frequency for radiation. More importantly, the PER introduces the spectral separation of the excitation and backscatter signals via voltage-tunable dual-mode resonance. The modulated signal is then radiated through the dipole antenna coupled to the PER, significantly enhancing the quality of the backscatter signal and therefore boosting the detection range of the RF reader.

The advantages of this tag design are multifaceted. First, it achieves a substantial spectral separation between excitation and backscatter signals, mitigating the self-interference issue for the RF reader. Second, the tag supports frequency modulation by establishing a linear relationship between shifts in its resonance frequency and the amplitude of the keystroke signal. This simplifies keystroke demodulation at the RF reader. Third, the design is compact enough to fit on a one-inch PCB, enabling easy concealment within the target environment. Finally, the tag can operate at a low frequency (e.g., 915 MHz), which offers strong wall penetration and favorable propagation characteristics for through-wall keystroke detection.

Challenge #2: RF Reader Design. Designing an effective RF reader poses several challenges. First, the backscattered signal is inherently weak due to the passive nature of the tag and the lack of a line-of-sight (LoS) path between the tag and the reader. Second, the keystroke classification task involves a large vocabulary. Unlike small-scale gesture recognition, keystroke inference must distinguish among dozens of fine-grained key events that often exhibit subtle differences in their signal signatures. Third, generalization is a grand challenge. Typing patterns vary widely across users due to differences in finger strength, typing speed, and hand posture. Environmental factors such as desk material, ambient vibrations, and tag placement further compound this variability.

To address these challenges, we design a dedicated signal processing pipeline that extracts robust keystroke features from weak backscatter signals across both temporal and spectral domains, at both coarse and fine granularities. In particular, the fine-grained feature design leverages the fact that two distinct signal paths exist from the keyboard to the RF tag: (i) solid-borne typing vibrations, and (ii) airborne acoustic sounds. Leveraging their different propagation speeds, we compute the Time Difference of Arrival (TDoA) between these two signals as a discriminative, fine-grained feature for each keystroke. This feature is largely invariant to both user and keyboard, supporting strong generalization across different scenarios. The coarse- and fine-grained features are fused via a cross-attention mechanism and passed to an offline-trained deep neural network (DNN) classifier to predict the typed content.

To further enhance RadKey's adaptability in new attack scenarios, we introduce an LLM-guided online adaptation method. While prior studies [39], [40], [41], [42] use LLMs for post hoc text correction, RadKey uses an LLM as a source of pseudo ground-truth labels to update the keystroke classifier. Prompting ensures that the online adaptation is triggered only for coherent, semantic text. This mechanism effectively transfers the LLM's language knowledge into the keystroke classifier, significantly improving RadKey's accuracy and adaptability even in challenging contexts such as non-semantic typing (e.g., passwords).

We have built a prototype of the RF tag and reader and evaluated RadKey in various scenarios. Extensive experiments show that (i) RadKey achieves robust and accurate keystroke inference across diverse keyboard types and user profiles, (ii) the system remains effective under challenging conditions, such as through-wall and long-distance settings, and (iii) the integration of LLM-guided online adaptation significantly improves generalization and reduces the character error rate without requiring victim-specific keystroke data for training.

This paper makes the following key contributions:

- We introduce a dual-resonator RF backscatter tag that can convert keystroke signals to the frequency shift of backscattered RF signals. More importantly, it separates the frequencies of excitation and backscatter signals and thus mitigates the self-interference issue.

- We design a novel RF reader that can extract reliable keystroke features from the backscatter signals. It features an LLM-guided online adaptation to improve its keystroke detection accuracy and adaptability.
- Extensive experiments confirm the effectiveness of this attack in diverse realistic scenarios.

2. Threat Model

Attack Description. We consider the attack scenario as illustrated in Fig. 1, where an adversary seeks to eavesdrop on keystrokes generated by a victim typing on a physical keyboard or a laptop. Based on the captured keystroke signals, the adversary aims to reconstruct the typed content and infer sensitive information such as messages, search queries, or passwords. To perform this attack, the adversary discreetly places a passive RF tag beneath the surface of the desk or table supporting the keyboard. This setup leverages mechanical coupling, where keystroke-induced vibrations propagate through the rigid structure and are picked up by the tag, without requiring any direct contact with the keyboard itself. The attack requires no modification to the keyboard or host device and remains effective as long as both the keyboard and tag are placed on the same rigid surface. This threat model is applicable to various environments, including offices, libraries, conference rooms, and co-working spaces.

Assumptions. For this threat model, we make the following assumptions. *First*, we assume that a passive RF tag can be unobtrusively placed underneath the target desk or table. This is feasible in both public and private settings. In shared spaces such as offices, conference rooms, libraries, or cafes, one-time temporary physical access is often sufficient for discreet tag placement, allowing it to remain unnoticed for extended periods. In private spaces such as personal offices, this assumption also holds due to the risk of insider threats, such as cleaning staff, visitors, or disgruntled employees. *Second*, we assume the adversary knows the make and model of the victim's keyboard or laptop. This information can often be inferred through casual observation, public device usage patterns, or inventory information. *Third*, while the passive tag must be placed inside the target space, the RF reader may be located either within or outside the victim's physical environment, enabling both proximal and remote eavesdropping scenarios. This flexibility accounts for diverse real-world attack surfaces, including through-wall keystroke eavesdroppers or compromised on-site infrastructure. We note that this attack is not undetectable. The passive backscatter tag may be discovered through visual inspection, and the RF eavesdropping activity can also be detected by spectrum monitoring.

3. Understanding Keystrokes

This section introduces the fundamental principles underlying our proposed keystroke eavesdropping attack. Specifically, we examine how mechanical vibrations are generated by keystrokes, how these vibrations propagate through various media, and how they are detected by a piezoelectric sensor.

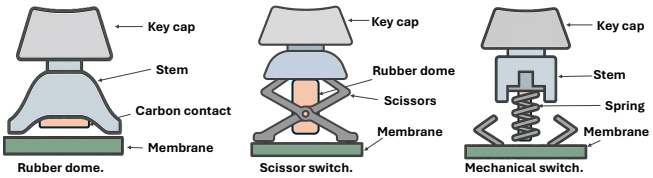


Figure 2: Three common keyboard switch types.

3.1. Keyboard Switch Structure

Modern computer keyboards, despite their varied designs, primarily operate using three types of mechanical switch actuations: rubber dome switches, scissor-switch mechanisms, and mechanical switches [35]. Each switch type exhibits distinct tactile feedback, actuation force, and acoustic profile, yet all produce impulse-like mechanical vibrations and acoustic emissions during keypresses due to the rapid contact and deformation events involved. Fig. 2 illustrates the structural differences among these three common keyboard switch types.

- *Rubber dome switches*, commonly found in commodity keyboards, use a flexible rubber membrane that collapses under pressure to complete an electrical circuit. The deformation and subsequent rebound of the dome generate both acoustic sound and mechanical vibration.
- *Scissor-switch keys*, typically used in laptops and low-profile keyboards, utilize a scissor-like structure to guide vertical key movement. Although designed for shorter travel and quieter operation, scissor switches still produce distinct vibrational and acoustic signatures due to their mechanical contacts.
- *Mechanical switches* incorporate individual spring-loaded components beneath each key. Variants such as Cherry MX and Alps switches differ in actuation force and tactile response, but all share a common trait: physical component interactions (e.g., stem movement and contact closure) that produce both audible clicks and strong mechanical vibrations.

Regardless of switch types, pressing and releasing a key involves multiple distinct mechanical events, including finger contact with the keycap (touch peak), actuation at the bottom of the keystroke (hit peak), and return motion (release peak). Each of these events contributes to a complex waveform comprising both air-borne acoustic energy and mechanical vibrations that propagate through the keyboard and its supporting surface. These signals serve as the initial stimuli for our dual-path vibration sensing approach.

3.2. Keystroke-induced Object Vibration Model

When a key is pressed, the applied force initiates a mechanical impulse that excites vibrations in both the keyboard and its supporting surface, such as a table. These

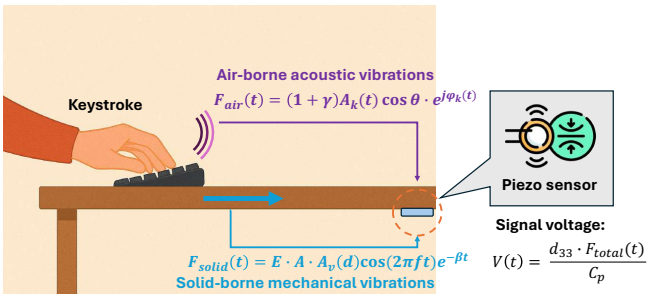


Figure 3: Keystroke-induced signal propagation model.

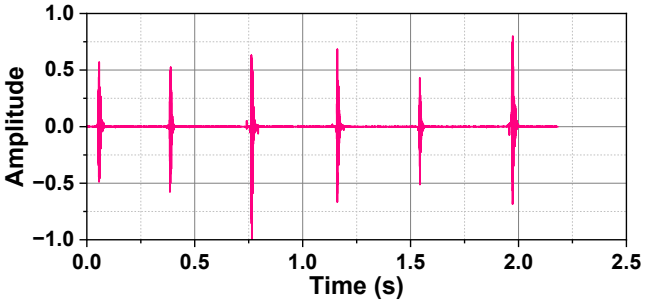


Figure 4: Demodulated keystroke signal captured when the keyboard and RF tag are placed on separate tables.

vibrations propagate outward from the impact point through two distinct transmission paths: solid-borne and air-borne propagation. Fig. 3 illustrates this dual-path propagation model.

Solid-Borne Mechanical Vibration. In the solid-borne path, the mechanical energy of a keystroke generates damped longitudinal vibrations in the keyboard body and table surface. These vibrations decay in amplitude as they propagate through the solid material due to internal damping and scattering. The amplitude $A_v(d)$ at a distance d from the source can be expressed as [43]:

$$A_v(d) = A_{v0}e^{-\alpha d}, \tag{1}$$

where A_{v0} is the initial amplitude, and α is the attenuation coefficient determined by the physical properties of the medium and vibration frequency.

To estimate the force experienced by the sensor, we assume the local displacement follows a damped sinusoidal form [44]:

$$u(t) = A_v(d)\sin(2\pi ft)e^{-\beta t}, \tag{2}$$

where f represents the dominant resonant frequency excited by the keystroke, and β is the temporal damping coefficient. This waveform reflects the transient nature of keypresses.

The strain induced by this vibration is the spatial derivative $\partial u(t)/\partial d$, which produces a time-varying stress $\sigma(t)$ according to Hooke's law for linear elastic solids [45]:

$$\sigma(t) = E\frac{\partial u(t)}{\partial d}, \tag{3}$$

where E is the Young's modulus of the surface material. The total dynamic force applied to the piezoelectric sensor area A is [46]:

$$F_{solid}(t) = \sigma(t)A = -\alpha \cdot E \cdot A \cdot A_v(d) \cdot \sin(2\pi ft)e^{-\beta t}. \tag{4}$$

This force serves as the sensor input, leading to voltage generation as described in Section 3.3.

Air-Borne Acoustic Sound. Concurrently, the keystroke produces air-borne acoustic waves that travel through the surrounding air. These waves, characterized by amplitude $A_k(t)$ and phase $\phi_k(t)$, induce surface pressure. The total acoustic pressure $F_{air}(t)$ at the surface, incorporating both direct and reflected components, is given by [47]:

$$F_{air}(t) = (1+\Gamma)A_k(t)\cos\theta \cdot e^{j\phi_k(t)}, \tag{5}$$

where Γ is the reflection coefficient based on surface impedance, and θ is the angle of incidence of the wave.

3.3. Piezoelectric Sensor for Keystroke Detection

Piezoelectric sensors are widely used for detecting mechanical vibrations due to their ability to convert dynamic stress into electrical voltage. This property arises from the *piezoelectric effect*, wherein certain crystalline materials generate an electric charge when subjected to mechanical deformation.

When a time-varying force $F(t)$ acts on a piezoelectric material, it induces an electrical potential difference across the sensor terminals. In our keystroke-induced vibration model, the total force acting on the sensor is the superposition of forces from solid-borne and air-borne vibrations, i.e.,

$$F(t) = F_{\text{solid}}(t) + F_{\text{air}}(t). \tag{6}$$

This combined mechanical input drives the sensor response and determines the resulting voltage output. The corresponding output voltage $V(t)$ is modeled as [48]:

$$V(t) = \frac{d_{33}}{C_p} \cdot F(t), \tag{7}$$

where d_{33} is the charge coefficient characterizing the material's response, C_p is the internal capacitance of the sensor, and $F(t)$ is the mechanical force applied due to surface vibrations.

Due to their high sensitivity and broad frequency response, piezoelectric sensors can respond to both solid-borne and air-borne vibrations. To validate this capability, we perform an isolation experiment where the keyboard and the RF tag are placed on separate tables with no mechanical coupling. Despite the separation, the tag reliably captures keystroke-induced air-borne signals, as shown in Fig. 4. This confirms that the piezoelectric sensor is sufficiently sensitive to air-borne acoustic energy in realistic typing scenarios. It therefore enables robust dual-path signal acquisition in our system.

Since the mechanical input force is directly related to the vibration dynamics outlined in Section 3.2, the resulting voltage signal $V(t)$ contains temporal and spectral information about the original keystroke event. For small-amplitude,

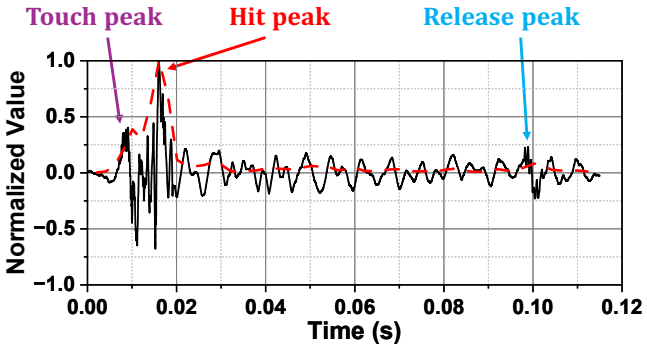


Figure 5: Example of keystroke signal captured by a piezoelectric sensor.

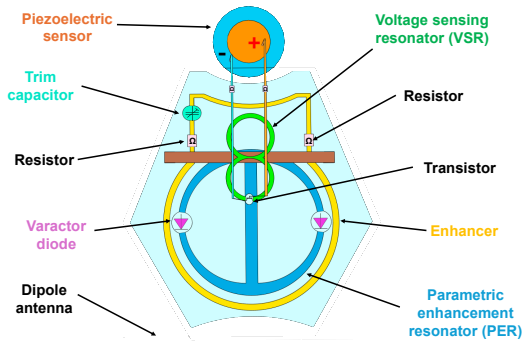


Figure 6: The diagram of our proposed backscatter tag.

linear deformations, the sensor's response remains proportional to the local stress field or surface acceleration.

Fig. 5 shows a signal sample captured by a piezoelectric sensor during a single keystroke event. The waveform reveals three prominent peaks corresponding to distinct mechanical interactions: the initial finger contact with the keycap, the bottom-out impact at full depression, and the elastic rebound as the key returns to its rest position. These peaks differ in amplitude and spacing depending on typing style and key structure, but consistently follow a characteristic temporal order. The sensor output reflects the superposition of solid-borne and air-borne components induced by the keystroke, resulting in a rich, non-stationary waveform containing both high- and low-frequency content critical for downstream inference.

4. RF Tag Design

4.1. Tag Design

The RF backscatter tag plays a critical role in enabling reliable keystroke eavesdropping. First, the tag must be compact (e.g., less than one inch in diameter) and easily concealable within typical target environments. Second, it must support continuous operation powered solely by energy harvesting. Most prior RF backscatter systems, such as WISP [36] and MARS [38], cannot sustain continuous communication and instead require prolonged energy harvesting periods, resulting in low-duty-cycle and intermittent operation. Third, it is highly desirable for the tag to decouple its excitation and reflection frequencies, so as to avoid self-interference at the RF reader.

Tag Structure. To meet these requirements, we propose a dual-resonator tag as shown in Fig. 6, which is composed of four key components: (i) a piezoelectric sensor, (ii) a voltage sensing resonator (VSR), (iii) a parametric enhancement resonator (PER), and (iv) a dipole antenna. An off-the-shelf piezoelectric sensor that is sensitive to both solid-borne mechanical vibrations and air-borne acoustic sound is selected for this tag. The VSR is an LC circuit composed of an 8-shaped coil terminated by a bipolar junction transistor (BJT) whose base and emitter connect to the piezoelectric sensor. The PER is a circular planar resonator on a fiberglass-epoxy substrate incorporating two varactor diodes across split gaps to support dual-mode resonance (circular and butterfly modes).

Operation Principles. The piezo sensor captures the keystroke-induced vibrations and acoustic fluctuations and converts them into a voltage signal, which modulates the capacitance of a diode in the VSR. Since the resonance frequency of VSR depends on this capacitance, the VSR directly translates the voltage signal into frequency shifts, enabling frequency modulation of the captured sound and vibration. The PER is magnetically coupled to the VSR. Together with the dipole antenna, the PER serves two purposes: (i) harvest energy from the RF reader to sustain the resonance, and (ii) enhance radiation efficiency to strengthen the backscattered signal. More importantly, the PER separates its excitation and reflection frequencies by adopting a dual-mode resonance structure. This design mitigates self-interference at the RF reader, thereby extending the effective eavesdropping range.

4.2. Signal Analysis

VSR Operation. For notational simplicity, denote v_s as the piezoelectric sensor's instantaneous voltage output, which is proportional to the mechanical force induced by keystrokes. This voltage is applied to the varactor diodes of the VSR, whose capacitance varies with v_s. Denote f_1 as the resulting resonance frequency of the VSR. Then, we have:

$$f_1 = \frac{1}{2\pi}\sqrt{\frac{2}{L_1 C_1}} = \frac{1}{2\pi}\sqrt{\frac{2}{L_1 C_{10}\left(1-\frac{v_s}{\Phi_1}\right)^{-\lambda_1}}}, \tag{8}$$

where L_1 and C_1 are the VSR's inductance and capacitance, respectively; C_{10} is its capacitance when the bias voltage is 0. Φ_1 is the varactor diode's junction potential. λ_1 is a constant related to the property of the varactor diode.

In practice, we have $v_s \ll \Phi_1$. Based on the first-order Taylor approximation, we have:

$$f_1 \approx f_{10}\left(1-\frac{\lambda_1 v_s}{\Phi_1}\right), \tag{9}$$

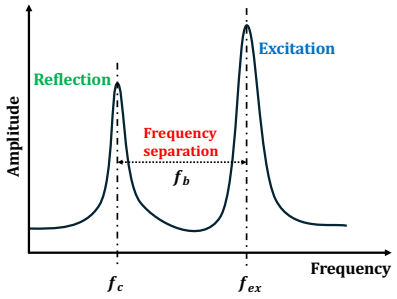


Figure 7: Illustrating spectral separation of the RF tag's excitation and reflection/backscatter signals.

where $f_{10} \equiv \frac{1}{2\pi}\sqrt{\frac{2}{L_1 C_{10}}}$ is the unperturbed resonance frequency of the VSR when no bias voltage is applied. This establishes a linear mapping between the keystroke-induced voltage and the carrier frequency shift, enabling frequency modulation in the analog domain without digitization.

PER Operation. The PER is introduced to enhance the backscatter signal strength so as to extend the eavesdropping range. The PER is designed to operate in two resonance modes simultaneously: *circular mode* and *butterfly mode*. This is achieved via a symmetric half-circle layout with dual varactor diodes as shown in Fig. 6. The circular mode couples magnetically to the VSR and thus captures frequency shifts caused by keystrokes, while the butterfly mode supports excitation by the RF reader.

Denote f_c as the resonance frequency of its circular mode. Denote f_b as the resonance frequency of its butterfly mode. When the PER is excited by a signal at frequency $f_{\text{ex}} = f_c + f_b$, it will radiate the modulated backscatter signal at frequency f_c, which we call either the reflection frequency or the backscatter frequency. Fig. 7 illustrates the separation of reflection and excitation frequencies. This fundamentally mitigates the self-interference issue for the RF reader and thus increases the eavesdropping range.

Tag Modulation Analysis. Recall that v_c is the voltage generated by the piezo sensor and f_c is the tag's backscatter signal frequency. Then, we have the following theorem:

Theorem 1. *The proposed tag achieves a frequency modulation for the voltage signal generated by the piezo sensor, i.e.,*

$$\frac{\partial f_c}{\partial v_s} = \underbrace{\left(-\frac{\lambda_1 f_{10}}{2\Phi_1}\right)\left(\frac{\frac{L_c}{R_c}}{\frac{L_c}{R_c}+\frac{L_b}{R_b}}\right)\left(\frac{f_L^3}{2f_1^3}\right)\left(1+\frac{\frac{f_2^2}{f_1^2}-1+2\left(\frac{f_1^2}{f_L^2}-1\right)\left(\frac{f_2^2}{f_L^2}-1\right)}{\left|2\frac{f_2^2}{f_L^2}-\frac{f_2^2}{f_1^2}-1\right|}\right)}_{\textit{Constant}}, \quad (10)$$

where f_1 and f_2 are the stand-alone resonance frequencies of VSR and PER, respectively. f_L is the lower resonance frequency of the coupled VSR and PER resonators. R_c and L_c are the effective resistance and inductance of the PER in its circular mode. R_b and L_b are the effective resistance and inductance of the PER in its butterfly mode.

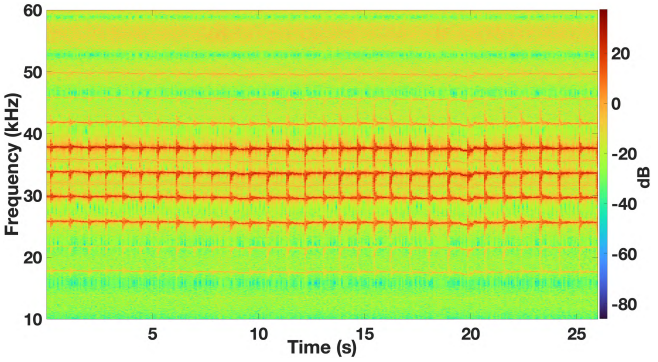


Figure 8: Spectrogram of the received multi-carrier backscatter signal at the RF reader.

The proof is provided in Appendix 9. Theorem 1 shows that the frequency of the tag's backscattered radio signal varies linearly with the output voltage of the piezoelectric sensor, confirming the tag's *frequency modulation* (FM) behavior. We note that FM is widely regarded as a more reliable modulation scheme than alternatives such as amplitude modulation (AM).

5. RF Reader Design

To demodulate the keystroke signal and estimate the victim's typing content, we propose an RF reader design composed of three key components: (i) backscatter signal demodulation, (ii) keystroke feature extraction, and (iii) LLM-guided online adaptation. We explain them below.

5.1. Backscatter Signal Demodulation

Backscatter RF Signal. Due to imperfections in tag fabrication, the keystroke voltage signal is modulated onto multiple backscatter carriers rather than a single backscatter carrier. Specifically, the received radio signal at the RF reader can be modeled as:

$$r(t) = \alpha \sum_{k=1}^{K} \cdot \exp\left(2\pi f_k t + 2\pi \Delta_k \int_0^t v_s(\tau) d\tau\right) + w(t), \quad (11)$$

where f_k represents the center frequency of the k-th carrier, K is the total number of carriers, Δ_k is the frequency modulation coefficient of carrier k, α is the signal attenuation coefficient shared by all carriers, and $w(t)$ denotes additive noise and imperfection artifacts. We note that this model does not account for the multi-path effect of RF signal propagation. This is because the frequency-modulated keystroke signal on the RF backscatter carrier is very narrowband (less than 40 kHz) and thus the multi-path effect is insignificant, especially for frequency modulation.

Fig. 8 shows an example of the received signal at the RF reader. Clearly, the received signal contains multiple

carriers with different strengths. Each carrier is frequency-modulated by the keystroke voltage signal. The carriers are equally spaced. To demodulate the keystroke voltage signal, we simply keep the strongest carrier and remove all other carriers. This method is not optimal, but our experiments show that its performance loss is marginal.

CFO Correction. The frequency-modulated backscatter signal is first down-converted to an intermediate frequency (IF) (e.g., 40 kHz) to avoid DC interference and preserve low-frequency keystroke content. However, due to tag hardware imperfections and environmental factors (e.g., temperature drift), the center frequencies of reflection signals are not static but drift slowly over time. These variations, referred to as carrier frequency offsets (CFO), degrade FM demodulation performance if uncorrected.

To address this, we implement a CFO estimation and compensation module in the frequency domain. The incoming signal is segmented into N-point frames, and each segment undergoes Fast Fourier Transform (FFT) to locate the center frequency of the strongest carrier. Let f_{c_0} be the estimated center frequency of the dominant carrier used for demodulation. Then, the offset Δf is calculated for each segment and applied to shift the IF signal to baseband (zero-IF), i.e., $\hat{x}(t) = x(t) \cdot e^{-j2\pi\Delta f t}$, where $x(t)$ is the sampled IF signal, and $\hat{x}(t)$ is the CFO-compensated baseband signal.

Signal Demodulation. After the CFO has been corrected, the RF reader is ready to demodulate the keystroke signal using phase differentiation. Let $\hat{x}(t)$ denote the complex-valued baseband signal after low-pass filtering. Then, the instantaneous phase is given by: $\phi(t) = \angle\hat{x}(t) = 2\pi\Delta \int_0^t v_s(\tau)d\tau$. The keystroke signal $v_s(t)$ can be estimated by differentiating $\phi(t)$ over time:

$$v_s(t) \approx \frac{f_s}{2\pi\Delta} \left[\phi(t + \Delta t) - \phi(t)\right], \tag{12}$$

where f_s is the sampling rate, Δ is the frequency modulation coefficient of the selected backscatter carrier used for demodulation, and Δt is a small step size (one sample interval). After demodulation, the demodulated keystroke signal is resampled to a uniform rate (e.g., 44.1 kHz) for feature extraction.

5.2. Keystroke Signal Segmentation

To enable reliable inference, RadKey requires a lightweight yet robust keystroke segmentation mechanism that can isolate individual keystroke events from continuous backscattered signals. We adopt a self-adaptive, energy-based segmentation strategy that avoids dependency on fixed thresholds or complex model-based post-alignment.

Short-Time Energy Envelope. Denote $s[n]$ as the digitalized samples of the demodulated keystroke signal $v_s(t)$ in Eqn (12). Then, we compute its short-time energy envelope $A[n]$ using a sliding window of length L:

$$A[n] = \sqrt{\frac{1}{L} \sum_{k=0}^{L-1} s^2[n-k]}. \tag{13}$$

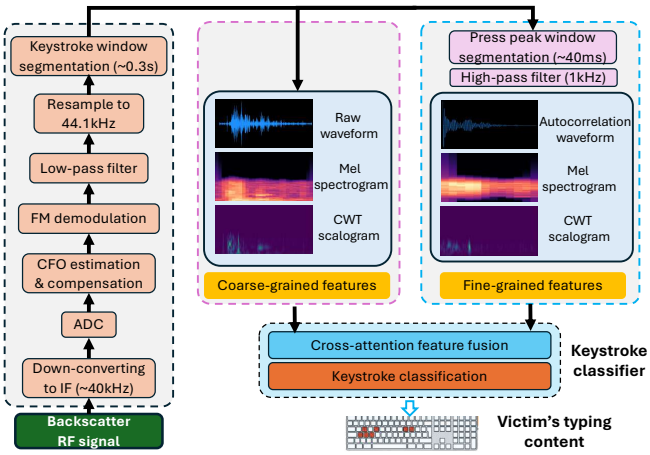


Figure 9: Overview of RF reader's signal processing and feature extraction.

This operation captures transient energy bursts caused by keystroke events while suppressing background variations.

Self-Adaptive Thresholding. Instead of using a fixed threshold, we compute a dynamic threshold using local signal statistics. A local threshold is computed at each time index based on the surrounding energy distribution. Specifically, for each point n, we compute a local mean $\mu[n]$ and standard deviation $\rho[n]$ of $A[n]$ over a sliding window, and mark n as a keystroke onset if: $A[n] > \mu[n] + \lambda \cdot \rho[n]$, where λ is a tunable sensitivity factor. This formulation enables the segmentation to adapt to different signal amplitudes, background levels, and device placements.

Onset Filtering and Segment Extraction. To eliminate spurious detections caused by reverberation or hand tremors, we enforce a minimum temporal spacing $\Delta_{\min}$ between adjacent onsets. Within each window of length $\Delta_{\min}$, only the peak with the highest energy is retained. Around each remaining onset time n_i, we extract a fixed-length segment $s_i[n]$ from the original signal, centered at n_i with asymmetric margins.

5.3. Two-Time-Scale Feature Extraction

To effectively capture the rich information embedded in keystroke acoustic signals, we propose a hierarchical time-frequency feature extraction framework. This approach extracts and processes features at two complementary levels: (i) a *coarse-grained* level that characterizes the entire keystroke event, and (ii) a *fine-grained* level that focuses on the detailed properties of keystroke peaks and their inter-channel timing relationships. Features from both levels are then fused and jointly processed to produce the final keystroke classification.

Coarse-Grained Feature Extraction. We define a coarse-grained feature extraction stage that operates over the entire keystroke event, using a time window of approximately 330 ms. This duration is not fundamental to the method; rather, it is an empirically chosen upper bound

that is sufficient to capture the full acoustic signature of a keystroke, including the initial touch, the primary impact (hit peak), and the subsequent release and reverberations.

From this time window, we extract the following three representative features.

- *Raw Time-Domain Signal:* It preserves the temporal waveform morphology, which may encode unique patterns per key and user.
- *Mel Spectrogram:* It converts the signal into a time-frequency representation emphasizing perceptual frequency resolution, mimicking human auditory processing.
- *Wavelet Transform:* It provides excellent time-frequency localization, adept at capturing transient events and varying frequency components for the keystroke.

These three representations are then fed into a dedicated coarse-grained feature extraction sub-network implemented using a lightweight CoAtNet [49] architecture. CoAtNet combines the strengths of convolutional operations for local feature extraction and attention mechanisms for modeling long-range dependencies, offering both efficiency and representational power. This sub-network learns abstract, high-level features that characterize the global structure of each keystroke event across input modalities. The outputs from processing each representation are subsequently concatenated for downstream fusion and classification.

Fine-Grained Feature Extraction. The fine-grained level focuses on extracting precise details from the critical peak events within the keystroke (solid-borne and air-borne peaks) and robustly encoding the Time Difference of Arrival (TDoA) between the two channels. Instead of using the entire 330 ms window, we define a shorter analysis window centered around the detected solid-borne hit peak. This window is extended to reliably encompass the later-arriving air-borne hit peak, considering the physical dimensions of the setup. Let D_{table} be the longest dimension of the table (or relevant surface) supporting the keyboard. Then, the time difference for sound propagation in air across this dimension is D_{table}/c_{sound}, where c_{sound} is the sound speed (approx. 340 m/s). The fine-grained window is therefore defined to capture the solid-borne hit peak and this subsequent air-borne acoustic information, plus a small margin for variability (e.g., $duration\_solid\_peak + D_{table}/c_{sound} + padding$).

A crucial fine-grained feature is TDoA. Direct TDoA calculation using cross-correlation on raw signals can be highly sensitive to noise, multipath interference, and environmental factors. To create a more robust representation, we propose transforming the TDoA information into an image-like feature. This is achieved by applying time-frequency transformations (e.g., Mel spectrograms or CWTs) to the segmented audio within this fine-grained window. The resulting image-based time-frequency representation inherently captures the phase and arrival time differences. It serves as an "image" where the TDoA is encoded in the subtle misalignments and structural differences between them. This distributed representation is expected to be more resilient to noise and other imperfections than a single scalar TDoA value. These time-frequency representations (TDoA images) are then processed by a separate fine-grained feature extraction sub-network. This network learns to extract discriminative features from these "TDoA images" that are indicative of the precise inter-channel timing differences and specific peak characteristics.

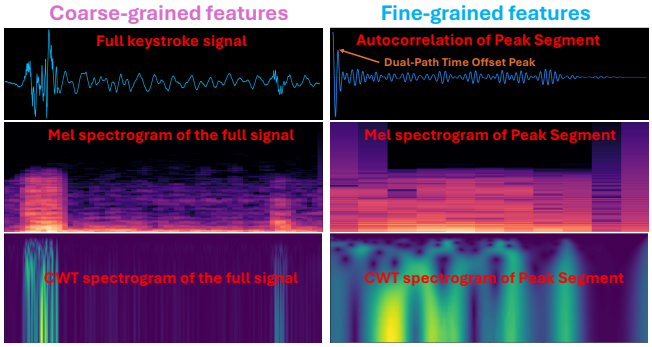


Figure 10: A sample of coarse-grained and fine-grained features.

Fig. 10 shows an example of the coarse-grained and fine-grained features used in our two-scale feature extraction pipeline. These features serve as the foundational inputs for downstream keystroke classification, capturing both global event characteristics and transient peak-level details.

Feature Fusion and Keystroke Classification. As illustrated in Fig. 9, the features extracted from the coarse-grained and fine-grained pathways provide complementary information about each keystroke. The high-level vectors from the coarse-grained analysis (capturing the overall event) and the fine-grained analysis (capturing peak details and TDoA) are concatenated to form a unified feature representation. This fused vector encodes both the long-term spectral–temporal structure of the full keystroke and the transient characteristics around the peak regions. The concatenated features are then processed by a cross-attention mechanism, which allows the model to dynamically assess the relative importance of the global event characteristics and the detailed temporal features. By learning to selectively emphasize the most discriminative elements within the combined feature space, the cross-attention layer produces a refined and contextually weighted representation. Finally, this output is passed to a classification head to predict the keystroke class.

Offline Training. We collect keystroke data from multiple participants who are instructed to type predefined content consisting of both isolated letters and complete sentences. This mixture allows us to capture the natural variability in keystroke dynamics, as single-letter inputs and sentence-level typing often exhibit different temporal and spectral characteristics due to variations in inter-key intervals, finger transitions, and contextual hand movements. The recorded signals are processed using the pipeline as shown in Fig. 9, which includes acoustic signal recovery, keystroke segmentation, and feature extraction. The known ground-truth labels from the predefined input are then used to supervise the training of the keystroke inference model.

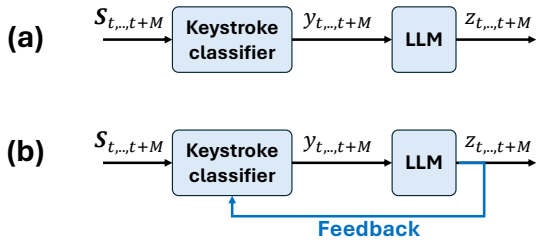


Figure 11: (a) LLM is used only for post hoc correction; and (b) LLM is used for online adaptation.

5.4. LLM-Guided Online Adaptation

Main Ideas. LLMs are powerful tools for contextual text correction, and recent work [39], [40], [41], [42], [50], [51], [52], [53] has explored integrating them into keystroke inference pipelines. However, existing approaches use LLMs only as post-processors, as illustrated in Fig. 11(a). Specifically, the raw or top-k outputs from a keystroke classifier are fed into an LLM, which is prompted to refine them using its internal language knowledge. While this can improve readability and lexical plausibility, it restricts the LLM's role to surface-level correction and leaves the underlying classifier unchanged. As a result, such approaches may enhance eavesdropping performance for coherent text but offer little benefit for non-semantic inputs such as passwords.

To overcome this limitation, we propose an LLM-guided online adaptation framework for RadKey, as shown in Fig. 11(b). In our design, the LLM is not a standalone post-processor. Instead, when the victim types coherent text, the LLM corrects the classifier's output and treats the corrected text as pseudo ground-truth labels to supervise online updates of the keystroke classifier. In this way, language knowledge from the LLM is transferred into the classifier itself, enabling accuracy improvements even when the LLM is no longer involved.

Crucially, the keystroke classifier and the LLM operate at different levels: the classifier predicts individual keystrokes or characters, whereas the LLM models relationships across entire sentences. Through online adaptation, the classifier's predictions improve even for non-semantic inputs such as passwords. To ensure safe and effective adaptation, the LLM must trigger online training only when the input is coherent. A natural question is how the system determines coherence. RadKey leverages the LLM's inherent language modeling ability: by prompting the LLM to evaluate the semantic plausibility of the classifier's output, the system initiates online adaptation only when the input is contextually meaningful. This safeguard prevents misadaptation on random, password-like, or otherwise incoherent inputs, ensuring that updates remain grounded in valid language patterns.

Our Design. Fig. 11(b) shows our online adaptation framework. Let $f_\theta(\cdot)$ be the keystroke classifier parameterized by θ. Let $\mathcal{S}_t$ be the set of coarse-grained and fine-grained features corresponding to one keystroke. Then, we define a loss function as:

$$\mathcal{L}_{\text{LLM-Align}} = \sum_t \mathcal{L}_{\text{CE}}\Big(f_\theta(\mathcal{S}_t),\ \text{LLM}(f_\theta(\mathcal{S}_t))\Big), \tag{14}$$

where $\mathcal{L}_{\text{CE}}(\cdot,\cdot)$ is the cross-entropy loss function, $\text{LLM}(\cdot)$ is the output of the LLM. This loss function uses the LLM to supervise the training of the keystroke classifier, leveraging the LLM's semantic refinement to improve the overall keystroke estimation accuracy.

Since the output of LLM is not always correct, we use a hybrid loss function, instead of solely cross-entropy loss $\mathcal{L}_{\text{LLM-Align}}$, as the loss function to update the keystroke classifier. Specifically, the hybrid loss function is defined as:

$$\mathcal{L}_{\text{total}} = \lambda_1 \mathcal{L}_{\text{LLM-Align}} + \lambda_2 \mathcal{L}_{\text{Conf}}(f_\theta(\mathcal{S}_t)) + \lambda_3 \mathcal{L}_{\text{Smooth}}(\theta), \tag{15}$$

where $\mathcal{L}_{\text{Conf}}$ is the confidence penalty to discourage overconfident incorrect predictions (entropy regularization), $\mathcal{L}_{\text{Smooth}}$ is parameter smoothing to avoid catastrophic forgetting, λ_1, λ_2, λ_3 are empirical weights balancing supervision, regularization, and stability. This hybrid loss function promotes both the alignment with LLM-inferred labels and stabilizes the convergence under noisy updates.

6. Experimental Evaluation

6.1. Implementation

RF Tag. Fig. 12 illustrates our fabricated tag, comprising a piezoelectric sensor, a VSR, and a PER. The PER features a circular inductor etched on a 0.8 mm thick G10 fiberglass-epoxy substrate, with an inner diameter of 13.5 mm and an outer diameter of 14.5 mm. Its upper/right and lower/left semi-circular sections are interrupted by split gaps, each filled with varactor diodes providing a capacitance of 9.1 pF. This configuration supports a *circular* resonance mode. A horizontal conductor bridges the two virtual voltage nulls of this mode, enabling a *butterfly* resonance mode.

The VSR is constructed by winding 32-gauge enameled copper wire around two rods, each 1.5 mm in diameter and spaced 1.8 mm apart. The coil consists of five counterclockwise turns on the first rod, followed by a single clockwise turn on the second. The two wire ends are connected to a pair of head-to-head varactor diodes with a capacitance of 3 pF. The piezoelectric transducer connects one sensing electrode to the common cathode and the other to the common anode.

RF Reader. Fig. 12 also shows our RF reader, which consists of a USRP N310, a power amplifier (PA), two directional antennas, and a laptop. The RF reader transmits a 30 dBm excitation signal at 915 MHz. The tag generates backscattered signals at 515 MHz. All signal processing is performed on the laptop using Python. For LLM-guided online adaptation, we deploy a locally hosted `LLaMA-3.2-3B` model [54], which interfaces with the keystroke classifier to provide online supervision.

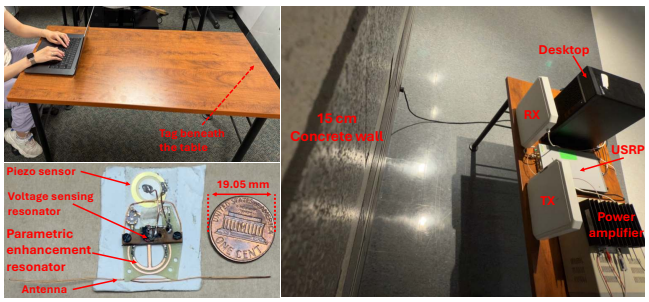


Figure 12: Experimental settings.

Figure 13: Keyboards and laptops.

6.2. Experimental Setup and Performance Metrics

Experimental Setup. All experiments were conducted in an indoor office environment using five representative input devices: a mechanical keyboard, a scissor-switch keyboard, a rubber dome keyboard, a Windows laptop, and an Apple laptop, as shown in Fig. 13. Unless otherwise noted, the experiments were performed on a wooden table; the backscatter tag was affixed to its underside; the horizontal distance between tag and keyboard was about 0.9 m; and the RF reader was positioned 3 m from the tag.

Our experiments involved 10 participants with varied typing styles and speeds. Every participant was asked to type both predefined and freeform inputs. The input corpus was designed to ensure full keyboard coverage, including all alphabetic keys ('a'–'z'), numeric digits ('0'–'9'), control keys (e.g., space, enter, and delete/backspace), common punctuation marks (e.g., '.', ',', '?', and '!'), and special characters and symbols (e.g., `@`, `$`, `%`, and `&`).

We consider two classes of typing content with distinct semantic structures in our experiments. *(i) Natural Language Input:* These are sentences and paragraphs from general-purpose corpora (e.g., Wikipedia articles, email replies, and news excerpts). This type of content reflects conversational or document-style typing where LLM's linguistic priors and contextual reasoning are most effective. *(ii) Non-linguistic Input:* This type of content includes random character sequences, passwords, email addresses, and command-line strings that lack grammatical structure or semantic context.

Performance Metrics. We evaluated the performance of RadKey using four standard metrics. *(i) Precision:* The ratio of correctly-predicted keystrokes to all predicted keystrokes. *(ii) Recall:* The ratio of correctly-detected keystrokes to all ground-truth keystrokes. *(iii) F1-Score:* The harmonic mean of precision and recall, reflecting the balance between them. *(iv) Character Error Rate (CER):* the normalized Levenshtein distance between the predicted string and the ground truth, defined as: $\text{CER} = \frac{S+D+I}{N}$, where S, D, and I denote the number of substitutions, deletions, and insertions, respectively, and N is the total number of characters in the ground-truth string.

TABLE 1: Performance of RadKey when the typing input is linguistic and non-linguistic text.

Input Type	W/o Online Updating	W/ Online Updating
Linguistic input	84.8%	97.7%
Non-linguistic input	80.1%	88.3%

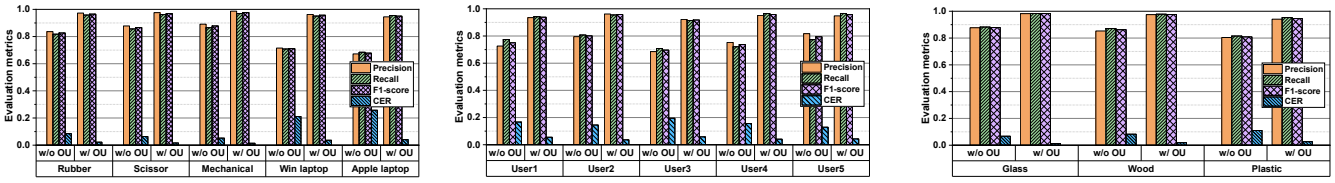
6.3. Main Results

Fig. 14 reports the overall performance of RadKey across three practical dimensions: keyboard type, user variability, and supporting surface. Overall, RadKey achieves strong performance, exceeding 90% detection precision across all conditions. For each setting, we compare the performance of RadKey with and without LLM-guided online adaptation. The results show that LLM-guided adaptation provides substantial improvements. It typically boosts the detection accuracy by more than 10%. This demonstrates the effectiveness of incorporating LLM-guided contextual correction and online refinement into our design. In the following, we present the detailed results for each condition.

Linguistic versus Non-linguistic keystroke Inputs. To understand how semantic structure affects RadKey, we evaluate the system on both linguistic and non-linguistic typing inputs. Table 1 summarizes the experimental results. Without online updating, RadKey achieves 84.8% accuracy on linguistic inputs and 80.1% on non-linguistic inputs. With LLM-guided online adaptation, the accuracy improves to 97.7% and 88.3%, respectively. The larger gain for linguistic inputs is expected, as natural-language context provides strong semantic constraints that help correct classification errors.

The improvement on non-linguistic inputs arises from the mixed-input nature of real typing streams. In our dataset, linguistic entries constitute the majority (86.7%), while non-linguistic entries account for the minority (13.3%). When users type linguistic content, the LLM-guided online adaptation module generates pseudo-labels to refine the keystroke classifier, enabling it to adapt to the user's typing patterns and keyboard characteristics. The refined classifier then yields more accurate predictions for subsequent non-linguistic inputs as well.

Impact of Keyboard Types. We evaluate RadKey across the five representative keyboard types as shown in Fig. 13. Fig. 14a presents our measurement results. External keyboards generate stronger and more consistent vibrations due to longer key travel and higher actuation force, resulting



(a) Performance for different keyboards. (b) Performance across different users. (c) Performance with different table surfaces.

Figure 14: Keystroke detection performance under different conditions. [**OU**: LLM-guided **O**nline **U**pdating.]

in clearer signal imprints and higher detection accuracy. In contrast, the Apple laptop exhibits the lowest detection accuracy. This can be attributed to its short key travel distance, low-profile mechanism, and rigid chassis, all of which attenuate vibrations and reduce SNR at the tag. As shown in Fig. 14a, LLM-guided online adaptation enables RadKey to achieve consistently high accuracy across all keyboard types, demonstrating strong robustness to the variance of keyboard types.

Performance Across Different Users. Generalizing across users is a key challenge, as typing style, finger force, posture, and keypress dynamics vary substantially between individuals. Even when using identical keyboards, subtle factors such as key aging, manufacturing tolerances, and acoustic differences introduce user-specific variations in the signal. To assess robustness, we conduct cross-user evaluations in which the model is trained on one user group and tested on entirely unseen users. Fig. 14b presents our results. The performance without LLM-guided adaptation degrades noticeably due to user-dependent signal shifts. In contrast, RadKey maintains high accuracy through its LLM component, which enforces linguistic consistency and provides pseudo-label supervision for online refinement. This allows the classifier to adapt to new users, yielding strong generalization without requiring user-specific training data.

Impact of Keyboard-Supporting Surfaces. We further evaluate performance across three common surface materials on which the keyboard and tag may be placed: glass, wood, and plastic. Fig. 14c reports our results. The non-adaptive model performs best on glass due to its rigid and uniform structure, which preserves vibrations accurately. Wood provides moderate vibration transmission and yields comparable accuracy. Plastic performs the worst because its higher damping and irregular internal structure attenuate vibrations more significantly. Despite these disparities, RadKey achieves uniformly high performance across all surfaces. The combination of dual-path sensing and LLM-guided online adaptation mitigates surface-induced variability, enabling stable performance in diverse real-world environments.

6.4. Impact of Tag Distance and Orientation

Impact of Tag-to-Keyboard Distance. To assess how proximity between the keyboard and the RF tag affects keystroke inference, we vary their horizontal distance from 10 cm to 120 cm. This range captures both tightly coupled and loosely coupled setups, reflecting practical considerations such as space constraints or concealment requirements.

TABLE 2: Keystroke detection performance at different tag-to-keyboard distances. [**OU**: LLM-guided **O**nline **U**pdating.]

	Precision		Recall		F1-score		CER	
	w/o OU	w/ OU	w/o OU	w/ OU	w/o OU	w/ OU	w/o OU	w/ OU
10 cm	90.5%	98.5%	88.2%	97.9%	89.3%	98.2%	5.1%	1.3%
20 cm	91.3%	98.1%	87.8%	97.4%	89.5%	97.7%	4.9%	1.5%
30 cm	90.1%	97.3%	89.5%	98.1%	89.8%	97.7%	5.4%	2.1%
40 cm	89.6%	98.3%	89.1%	97.5%	89.3%	97.9%	5.5%	1.7%
50 cm	87.3%	98.2%	86.7%	97.3%	87.0%	97.7%	6.4%	1.6%
60 cm	88.7%	97.4%	88.1%	96.8%	88.4%	97.1%	6.2%	1.9%
70 cm	87.1%	97.6%	85.7%	96.5%	86.4%	97.0%	6.9%	1.8%
80 cm	86.5%	97.8%	87.1%	97.4%	86.8%	97.6%	7.1%	1.9%
90 cm	85.8%	97.5%	85.7%	97.8%	85.7%	97.6%	7.5%	2.1%
100 cm	82.2%	96.7%	83.6%	96.5%	82.9%	96.6%	8.2%	2.3%
110 cm	82.7%	95.4%	82.3%	95.1%	82.5%	95.2%	8.4%	2.8%
120 cm	81.5%	95.5%	80.3%	95.7%	80.9%	95.6%	9.3%	2.7%

Table 2 reports the experimental results. As expected, performance degrades gradually as the distance increases due to vibration attenuation and weakened mechanical coupling. Larger distances reduce the SNR at the tag, making keystroke-induced vibrations more susceptible to noise. This effect becomes particularly evident beyond 100 cm, where the character error rate rises sharply without LLM-guided adaptation. However, with LLM-guided online adaptation, RadKey maintains high accuracy across all tested distances. The LLM leverages linguistic context to refine predictions and compensate for distance-induced signal degradation. These results demonstrate that RadKey remains effective even when the keyboard is placed more than one meter away from the tag, enabling flexible and covert deployment.

Impact of Tag-to-Reader Distance. Another critical factor for the key-typing eavesdropping attack is the tag-to-reader distance, which determines the effective reading range of the RF system. We evaluate this by placing the RF reader at distances ranging from 1 m to 8 m. Beyond 6 m, line-of-sight (LoS) is obstructed by furniture, representing realistic indoor obstructions. Table 3 shows the results. RadKey consistently achieves high accuracy across all distances, maintaining a 90.5% F1-score and only a 5.1% character error rate at 8 m. In contrast, removing the LLM component leads to significant degradation beyond 5 m, where lower SNR and increased distortion cause the error rate to rise sharply. These findings highlight the importance of combining physical-layer robustness with semantic-level correction to enable long-range keystroke inference. Overall, RadKey remains effective even under remote or partially obstructed surveillance conditions.

TABLE 3: Keystroke detection performance at different tag-to-reader distances. [**OU**: LLM-guided **O**nline **U**pdating.]

	Precision		Recall		F1-score		CER	
	w/o OU	w/ OU	w/o OU	w/ OU	w/o OU	w/ OU	w/o OU	w/ OU
1 m	88.5%	98.3%	87.8%	98.2%	88.1%	98.3%	5.9%	1.2%
2 m	86.5%	97.7%	86.2%	97.9%	86.4%	97.8%	7.1%	1.9%
3 m	86.1%	97.8%	85.3%	97.1%	85.7%	97.4%	7.4%	2.0%
4 m	83.5%	96.3%	82.1%	95.8%	82.8%	96.0%	8.8%	2.8%
5 m	82.8%	96.1%	82.5%	96.0%	82.7%	96.0%	9.1%	2.9%
6 m	78.7%	94.8%	77.8%	95.1%	78.2%	95.0%	14.6%	3.1%
7 m	70.5%	91.7%	71.3%	91.5%	70.9%	91.6%	20.1%	4.9%
8 m	66.3%	90.7%	67.8%	90.3%	67.0%	90.5%	23.7%	5.1%

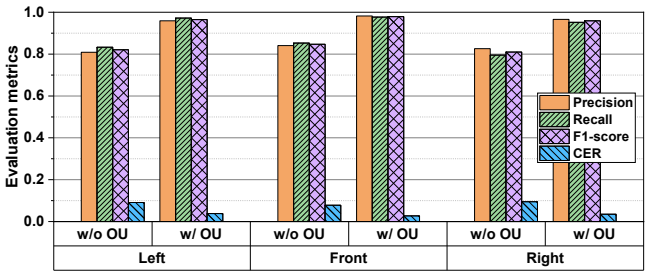


Figure 15: Impact of the RF tag's placement. [**OU**: LLM-guided **O**nline **U**pdating.]

Impact of Tag Orientation. We further evaluate the performance of RadKey when the tag is positioned in three orientations relative to the keyboard: facing the front, left, or right. Fig. 15 shows the experimental results. RadKey maintains strong performance across all orientations when equipped with LLM-guided adaptation. Among the three, the front-facing orientation yields the highest accuracy due to more symmetrical vibration propagation along the table surface. Lateral placements (left or right) experience asymmetric energy distribution and higher damping, resulting in weaker received signals and slightly lower accuracy. Nonetheless, RadKey remains robust across all configurations, demonstrating resilience to variations in tag placement.

6.5. Ablation Study

To quantify the contribution of each component in the RF reader system, we conduct an ablation study by selectively enabling or disabling key modules. Our experiments evaluate four configurations: *(i) Coarse baseline:* This baseline uses only coarse-grained features extracted directly from raw RF signals. A CoAtNet [49] classifier is trained without hierarchical fusion or language-model assistance. *(ii) Fine-grained features:* We enhance the baseline with fine-grained representations extracted via a high-resolution branch, fused with coarse features using a cross-attention module. *(iii) LLM post-processing:* An LLM-guided decoder is added for context-aware refinement, correcting classification outputs using language semantics. *(iv) LLM in the training loop:* Finally, we integrate the LLM into the training loop through a feedback mechanism that penalizes semantically inconsistent predictions. This is the final RF reader used in RadKey.

TABLE 4: Ablation study.

Configuration	Choice			
Coarse-Grained Features	✔	✔	✔	✔
Fine-Grained Features	✘	✔	✔	✔
LLM (Post-Processing)	✘	✘	✔	✘
LLM (Online Updating)	✘	✘	✘	✔
Precision	64.7%	82.6%	87.9%	98.1%
Recall	63.1%	82.3%	85.5%	97.6%
F1-Score	63.9%	82.4%	86.7%	97.8%
Character Error Rate	19.6%	10.8%	6.7%	1.5%

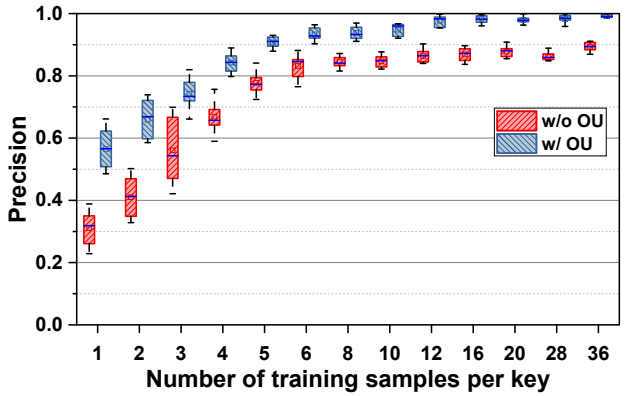


Figure 16: Impact of pre-training dataset size on the performance of RadKey. [**OU**: LLM-guided **O**nline **U**pdating.]

Table 4 summarizes the results, demonstrating that each component provides a meaningful performance gain. Incorporating fine-grained features improves the F1-score from 63.9% to 82.4%, and LLM post-processing further increases it to 86.7%. Enabling LLM-guided online adaptation yields the best performance, boosting the F1-score to 97.8% and reducing the character error rate to 1.5%.

6.6. Impact of Pre-training Dataset Size

To assess how pre-training dataset size influences RadKey's performance when training data comes exclusively from non-target users and keyboards, we vary the number of samples per key from 1 to 36 and evaluate the model on the target test set. Fig. 16 presents our experimental results. It can be seen that increasing the number of samples consistently improves performance without LLM adaptation, but the gain plateaus beyond 20 samples per key, indicating diminishing returns under non-target-only training. In contrast, with LLM-guided online adaptation, RadKey achieves high accuracy with as few as 4–8 samples per key. The LLM facilitates effective domain adaptation by injecting linguistic priors into the classifier, even in the absence of victim-specific data. These results demonstrate that RadKey is robust in low-data regimes and can be deployed without requiring any target-specific training data.

6.7. Through-Wall Keystroke Detection

To evaluate the covert sensing capability of RadKey, we examine its performance in through-wall scenarios where

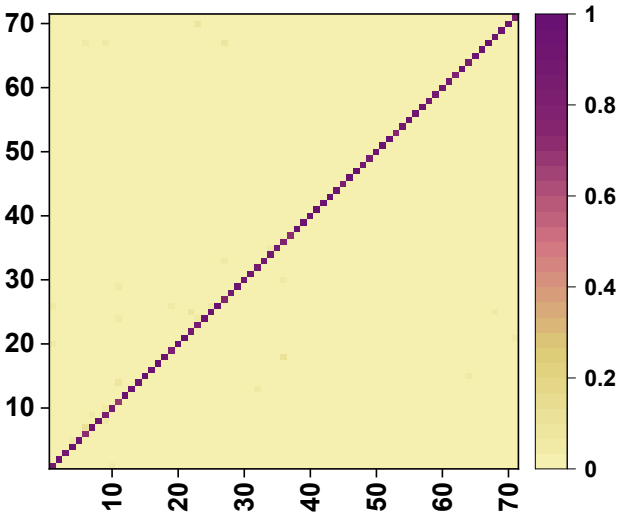


Figure 17: Through-wall keystroke detection accuracy.

the RF reader is placed outside the room, while the victim and passive tag remain inside, separated by a 15 cm concrete wall. This setup creates a fully non-line-of-sight condition, with the direct RF path completely blocked. Unlike long-distance attenuation, through-wall propagation introduces additional challenges such as multipath distortion and dielectric-induced phase shifts, which degrade signal quality and complicate demodulation.

Fig. 17 presents our experimental results, where both x-axis and y-axis represent keystroke indices ('a'-'z', 'A'-'Z', '0'-'9', and special symbols). RadKey maintains a high inference accuracy even under this challenging condition. This robustness can be attributed to: (i) the inherent resilience of the dual-resonator tag's frequency modulation to noise and fading; (ii) dual-path sensing that captures both solid-borne and air-borne signal components, mitigating blockage effects; and (iii) LLM-guided online adaptation that corrects channel-induced distortions.

7. Countermeasure Strategies

As keystroke inference attacks via passive RF backscatter become increasingly practical and covert, defending against such threats is critical to safeguarding user privacy. In this section, we discuss potential defense strategies across four levels: physical-layer obfuscation, signal-space mitigation, semantic-level hardening, and proactive defense.

Physical and Environmental Obfuscation. One natural line of defense is to disrupt the physical transduction pathway between keystrokes and RF signal perturbations. Since our system relies on subtle mechanical and acoustic vibrations transmitted through surfaces, introducing damping materials (e.g., acoustic foam, rubber mats, or vibration-absorbing desk layers) can significantly attenuate the signal before it reaches the backscatter tag.

Signal-Level Countermeasures. From the perspective of RF signal processing, defenders can employ *signal obfuscation* techniques such as artificial vibration injection or ambient RF noise generation. For instance, actuators placed near the tag could emit controlled low-level vibrations during typing events to corrupt keystroke-induced patterns. Moreover, defenders could deploy *RF fingerprinting traps*, which monitor the backscattered signal and flag suspicious passive tags based on abnormal reflection characteristics or unexpected resonance patterns.

Semantic and Language-Level Defenses. Given that our system leverages LLM to refine noisy outputs using linguistic priors, a promising direction is to *minimize the utility of such priors*. One way is to introduce randomness into typing behavior, such as inserting decoy keystrokes or using randomized input sequences, which disrupts regularity and hinders language modeling [5], [55]. Users can also adopt *linguistic-resistant input schemes* when entering sensitive information (e.g., using password managers or on-screen keyboards that randomize key positions). Since these schemes break natural language syntax, the LLM-guided correction becomes less effective, especially when domain adaptation is not feasible.

Toward Proactive Defense Frameworks. Long-term defense against RF-based keystroke attacks must be embedded into *multi-layered system architectures*. This includes (i) trusted computing modules that detect unauthorized sensing hardware, (ii) sensor access control policies that flag anomalous passive tags, and (iii) adversarial learning models that identify and degrade LLM-guided inference in hostile environments [56], [57]. Our attack shows that high-accuracy keystroke inference is feasible using passive RF signals and LLM supervision, underscoring the need for defenses that anticipate both new sensing modalities and intelligent post-processing models in adversarial pipelines.

8. Related Work

Keystroke eavesdropping attacks can generally be classified based on their sensing modalities, sensor placements, training requirements, and generalizability [5], [56], [57], [67]. Table 5 summarizes existing keystroke eavesdropping work. We detail them in the following two categories.

8.1. Microphone-Based Keystroke Eavesdropping

Research has been conducted over decades to study keystroke eavesdropping using microphones. Asonov et al. [65] first demonstrated that acoustic signals from keyboards differ among keys, achieving a 79% detection accuracy rate at 1 m. However, their method requires extensive labeled data and lacks robustness to environmental variability.

To address these limitations, Zhuang et al. [55] introduced an unsupervised method that utilized cepstrum features, Hidden Markov Models (HMM), and iterative feedback-based incremental learning. Their approach does not need labeled data and achieved over 96% character-level accuracy. Zhu et al. [25] further improved unsupervised acoustic attacks by introducing a context-free model using microphones placed 25 cm away. Their method achieved 72.2% accuracy without labeled training data.

TABLE 5: Overview and comparison of keystroke eavesdropping attack methodologies. Key to understanding this comparison: **Non-Invasive:** Attack does not require visible or suspicious sensor placement near the victim's device. **Closed Vocabulary:** Method is restricted to recognizing only a predefined, finite set of words or characters. **Symbols:** ● High; ◐ Moderate; ○ Low generalization/performance. -=Unknown/Not Provided.

Previous Work	Sensor & Signal Type	Attack Distance	Non-Invasive	Through Wall	Training/ML Dependency	Closed Vocabulary?	Target key Set	Generalization ability		Accuracy
								User	Environment	
Zhu *et al.* [25]	Smartphone microphones, Acoustic	25 cm	✗	✗	✗ Train, ✗ ML	✗	'a-z' + 3 Special keys	●	◐	72.2%
Liu *et al.* [19]	Smartphone microphones, Acoustic	5 cm	✗	✗	✗ Train, ✓ ML	✗	'a-z'	◐	◐	97.7%
KeystrokeSniffer [18]	Smartphone microphones, Acoustic	15 cm	✗	✗	✓ Train, ✓ ML	✓	'a-z' + 6 Special keys	◐	●	79.5%
Auditory Eyesight [58]	Smartphone microphones, Acoustic	50 cm	✗	✗	✗ Train, ✗ ML	✗	54 Commonly used keys	◐	◐	90.8%
UbiK [26]	Smartphone microphones, Acoustic	20 cm	✗	✗	✓ Train, ✓ ML	✗	56 Commonly used keys	◐	◐	95.0%
Giallanza *et al.* [59]	Smartphone microphones, Acoustic	5 cm	✗	✗	✓ Train, ✓ ML	✗	'a-z' + 8 Special keys	◐	○	70.6%
SIA [60]	Smartwatch microphone, Acoustic	5 cm	✗	✗	✓ Train, ✓ ML	✗	'a-z' + '0-9'	◐	◐	85%
Bai *et al.* [20]	Smartphone microphones, Acoustic	5 cm	✗	✗	✓ Train, ✓ ML	✓	'a-z' + 6 Special keys	◐	◐	91.5%
Slater *et al.* [61]	Single microphone, Acoustic	< 1 m	✗	✗	✓ Train, ✓ ML	✗	Entire keyboard (68 keys)	◐	○	84.6%
Cecconello *et al.* [62]	Single microphone, Acoustic	5 cm	✗	✗	✓ Train, ✓ ML	✗	'a-z'	◐	◐	91.7%
S&T [63]	Single microphone, Acoustic	5 cm	✗	✗	✓ Train, ✓ ML	✗	'a-z'	◐	◐	83.2%
Zhuang *et al.* [55]	Single microphone, Acoustic	-	✗	✗	✓ Train, ✓ ML	✗	'a-z' + 4 Special key	○	○	92.0%
Berger *et al.* [64]	Single microphone, Acoustic	-	✗	✗	✗ Train, ✗ ML	✓	'a-z'	◐	◐	90.0%
Asonov *et al.* [65]	Single microphone, Acoustic	1 m	✓	✗	✓ Train, ✓ ML	✓	'a-z' + 4 Special key	◐	○	79.0%
OverHear [23]	Mic & Accelerometers, Acoustic + Motion	5 cm	✗	✗	✓ Train, ✓ ML	✓	'a-z'	◐	◐	77.0%
Fang *et al.* [66]	1Tx & 1Rx, WiFi CSI	50 cm	✗	✗	✗ Train, ✗ ML	✗	'a-z'	○	◐	95.3%
WiKey [27]	2 Tx ants & 3 Rx ants, WiFi CSI	30 cm	✗	✗	✓ Train, ✓ ML	✗	'a-z' + '0-9' + 1 Special key	○	○	96.4%
Chen *et al.* [34]	SDR + FPGAs + 5 Rx ants, 2.4GHz RF signal	5 m	✓	✗	✓ Train, ✓ ML	✓	'a-z' + 1 Special key	○	○	91.8%
Marquardt *et al.* [33]	Accelerometer	5 cm	✗	✗	✓ Train, ✓ ML	✓	'a-z'	○	○	80.0%
This work	**Piezo-based RFID tag, Acoustic**	**8 m**	✓	✓	**✓ Train, ✓ ML**	✗	**Entire keyboard**	●	●	**98.1%**

In contrast, Berger et al. [64] introduced dictionary-constrained attacks, leveraging known language models to map acoustic signals to likely words. It claimed a 90% detection accuracy. Some work [62], [63] has investigated keystrokes' acoustic leakage in VoIP applications. These studies demonstrated that keystroke sounds remain distinguishable through compressed audio channels, achieving character detection accuracies exceeding 90%. Later efforts enhanced the robustness of this approach under realistic settings. Bai et al. [20] achieved 91.5% accuracy in noisy environments using refined inference techniques. Giallanza et al. [59] adopted DNN ensembles on audio captured by smartphone arrays, achieving 70.6% accuracy.

KeystrokeSniffer [18] demonstrated inference from audio recorded 15 cm away using commodity smartphones, achieving 79.5% accuracy. Auditory Eyesight [58] utilized microsecond-level timing to reach 90.8% accuracy at 50 cm. Liu et al. [19] further pushed the limits by using millimeter-scale acoustic ranging to reach 97.7% accuracy with minimal training and only one microphone. Slater et al. [61] introduced an end-to-end deep learning pipeline for keystroke inference under overlapping waveforms and fast typing. Wearable-based approaches, such as SIA [60], exploited smartwatch microphones to maintain discretion and achieved 85% recognition accuracy.

While these microphone-based works demonstrated the feasibility of acoustic side-channel attacks, their assumptions and training requirements remain questionable in realistic settings. Some require extensive supervised training data [1], [65], task- or setup-specific modeling [20], [61], or carefully controlled recording conditions [1], [19], [65]; others reduce deployment-specific training by relying on strong prior knowledge such as dictionary or language constraints [55], [64]. In general, their performance often degrades across different keyboards, users, and environments, limiting robust generalization in real deployments. RadKey addresses these limitations by replacing conspicuous nearby microphones with a covert RF backscatter tag, enabling through-wall operation, and combining dual-path feature design with LLM-guided online adaptation to achieve more robust keystroke inference across diverse real-world deployments.

8.2. Keystroke Eavesdropping Using Other Sensors

Alternative sensor modalities have also been explored in different ways for keystroke inference. Marquardt et al. [33] utilized accelerometers to sense keyboard vibrations, achieving 80% accuracy at close range. OverHear [23] enhanced inference accuracy by combining accelerometer and microphone data, effectively maintaining stealth. Wireless sensing methods have also proven feasible for this attack. WiKey [27] and Chen et al. [34] leveraged WiFi Channel State Information (CSI) and custom RF front-ends, respectively, achieving high accuracy (90-96%) but requiring complex multi-antenna setups. Fang et al. [66] developed a training-free CSI-based inference approach, enhancing deployment flexibility while achieving 95.3% accuracy. These alternative approaches demonstrate that keystroke leakage is not confined solely to acoustic channels. Methods exploiting vibrations and wireless signals often match or exceed acoustic methods under ideal conditions but generally require specialized hardware setups and calibration, limiting their practical generalization and stealth.

9. Conclusion

In this paper, we presented RadKey, an RF backscatter system composed of a compact backscatter tag and an RF reader for keystroke eavesdropping attacks. The tag features two magnetically-coupled LC resonators, allowing it to convert keystroke-induced emanations into the frequency shifts of its backscatter signal. It also achieves spectral separation between excitation and backscatter signals to mitigate self-interference. The RF reader is equipped with a sophisticated signal processing pipeline that extracts reliable coarse-grained and fine-grained keystroke features for accurate typing recognition. Additionally, it leverages an LLM-guided

online adaptation mechanism to enhance its generalizability. Experimental results demonstrate the effectiveness of our attack in realistic settings. This work highlights both the potential and risks of combining RF sensing with LLMs for input inference, underscoring the need for future research on attack resilience and effective countermeasures.

Proof of Theorem 1

As shown in Fig. 6, this backscatter tag is composed of two LC resonators: a Parametric Enhancement Resonator (PER) and a Voltage Sensing Resonator (VSR). The PER is overlaid by VSR, and they are magnetically coupled.

The PER consists of a circular-shaped conductor symmetrically divided by two gaps, each filled with a varactor (nonlinear capacitor) in a head-to-head configuration. This structure supports a circular resonance mode at frequency f_{cr}, where current flows around the ring. By connecting a horizontal conductor to the virtual voltage nodes of the circular mode, a second mode emerges at frequency f_{br}, enabling butterfly-shaped current flow in the corresponding circuit mesh. Fig. 18 illustrates these two modes of the PER. When the PER is excited by an external signal at the sum of the two resonance frequencies, i.e., $f_p \approx f_{cr} + f_{br}$, its nonlinear capacitance enables conversion of pumping power into enhanced backscattered signals near its resonant modes.

The VSR employs an 8-shaped conductor, with its ends connected by a bipolar junction transistor forming two head-to-head PN junctions. This implementation minimizes the number of solder joints, thereby decreasing parasitic resistance and increasing the quality factor. The base and emitter terminals are connected to a piezoelectric sensor, such that the VSR's resonance frequency can be modulated by pressure-induced voltages. Additionally, the 8-shaped layout helps suppress environmental electromagnetic artifacts.

When a bias voltage v_s is applied to the sensing electrodes, it alters the varactor capacitance, thereby shifting the resonance frequency f_1 as follows:

$$\begin{aligned} f_1 &= \frac{1}{2\pi}\sqrt{\frac{2}{L_1 C_1}} = \frac{1}{2\pi}\sqrt{\frac{2}{L_1 C_{10}(1 - v_s/\phi_1)^{-\lambda_1}}} \\ &\approx \frac{1}{2\pi}\sqrt{\frac{2}{L_1 C_{10}}}\left(1 - \frac{\lambda_1 v_s}{2\phi_1}\right) \equiv f_{10}\left(1 - \frac{\lambda_1 v_s}{2\phi_1}\right), \end{aligned} \tag{16}$$

where f_{10} denotes the zero-bias resonance frequency. Eqn (16) shows that when the sensing voltage v_s is much smaller than the junction potential ϕ_1, the resonance frequency f_1 is approximately linearly related to v_s. This standalone resonator can be characterized using loop antennas connected to a network analyzer; however, this direct approach suffers from limited sensitivity, especially when the resonator is located far from the measurement antennas.

To enhance the remote detectability of the VSR, it is physically overlaid onto the perimeter of the PER, which functions as a local signal enhancer. To understand the coupling mechanism between these resonators, we analyze the

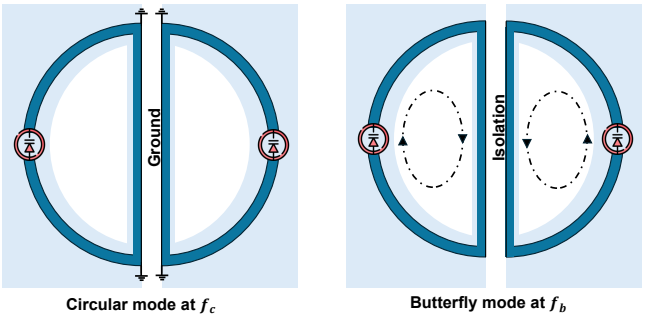


Figure 18: The PER operates in the circular resonance mode (left), where the circuit has zero voltages in the center plane, making the circular mode equivalent to two half-loops sharing the same center ground. The PER operates in the butterfly resonance mode (right), where two separate current flows are confined within their individual meshes, making the butterfly mode equivalent to two half-loops that are electrically isolated in the center plane.

relation between the electromotive force ξ and the induced current I in each resonator:

$$\begin{aligned} \xi_1 &= I_1 Z_1 + j2\pi f M I_2 \\ &= I_1\left(R_1 + j2\pi f L_1 - \frac{j}{2\pi f C_1}\right) + j2\pi f M I_2, \end{aligned} \tag{17}$$

$$\begin{aligned} \xi_2 &= I_2 Z_2 + j2\pi f M I_1 \\ &= I_2\left(R_2 + j2\pi f L_2 - \frac{j}{2\pi f C_2}\right) + j2\pi f M I_1, \end{aligned} \tag{18}$$

where ξ_1 and ξ_2 are the electromotive force for VSR and PER, respectively; I_1 and I_2 are their current. R, L, and C represent their effective resistance, inductance, and capacitance, respectively; M is the mutual inductance.

Solving for I_2 yields the induced current in the PER:

$$\begin{aligned} I_2 &= \frac{Z_1\xi_2 - jM2\pi f\,\xi_1}{Z_1 Z_2 + M^2(2\pi f)^2} \\ &= \frac{\left(R_1 + j2\pi f L_1 - j\frac{(2\pi f_1)^2 L_1}{2\pi f}\right)\xi_2 - j2\pi f M \xi_1}{\left[\begin{array}{l} R_1 R_2 + (2\pi f)^2 M^2 - \dfrac{L_1 L_2 (f^2 - f_1^2)(f^2 - f_2^2)}{f^2} \\ + \dfrac{j}{2\pi f}\left(L_1 R_2\left((2\pi f)^2 - (2\pi f_1)^2\right)\right. \\ \qquad \left. + L_2 R_1\left((2\pi f)^2 - (2\pi f_2)^2\right)\right), \end{array}\right]}, \end{aligned} \tag{19}$$

where f_1 and f_2 denote the stand-alone resonance frequencies of the VSR and PER, respectively.

When the coupled-mode frequency significantly deviates from the isolated-mode frequencies, both the term R_1 in the numerator and $R_1 R_2$ in the denominator can be neglected. To ensure I_2 is real-valued at resonance, the following condition must hold:

$$M^2 f^2 - L_1 L_2 \frac{(f^2 - f_1^2)(f^2 - f_2^2)}{f^2} = 0. \tag{20}$$

Solving Eqn (20) yields two coupled-mode frequencies:

$$f_L^2 = \frac{2f_2^2}{f_2^2/f_1^2 + 1 + \sqrt{\left(f_2^2/f_1^2 - 1\right)^2 + 4\kappa^2 f_2^2/f_1^2}}, \quad (21)$$

$$f_H^2 = \frac{2f_2^2}{f_2^2/f_1^2 + 1 - \sqrt{\left(f_2^2/f_1^2 - 1\right)^2 + 4\kappa^2 f_2^2/f_1^2}}, \quad (22)$$

where $\kappa^2 = M^2/(L_1 L_2)$ characterizes the coupling strength.

Defining $u_L = f_2^2/f_L^2$ and $u = f_2^2/f_1^2$, we can rewrite Eqn (21) as:

$$u + 1 + \sqrt{(u-1)^2 + 4\kappa^2 u} = 2u_L. \quad (23)$$

Taking the derivative of both sides of Eqn (23) with respect to u, we have

$$1 + \frac{(u-1) + 2\kappa^2}{\sqrt{(u-1)^2 + 4\kappa^2 u}} = \frac{2du_L}{du} = \frac{2f_1^3\, df_L}{f_L^3\, df_1}, \quad (24)$$

where the final equality follows from the chain rule: $du = -\frac{2f_2^2}{f_1^3} df_1$ and $du_L = -\frac{2f_2^2}{f_L^3} df_L$. Eqn (24) can be rearranged to be:

$$\begin{aligned}\frac{df_L}{df_1} &= \frac{f_L^3}{2f_1^3}\left(1 + \frac{(u-1) + 2\kappa^2}{\sqrt{(u-1)^2 + 4\kappa^2 u}}\right) \\ &= \left(\frac{f_L^3}{2f_1^3}\right)\left(1 + \frac{\frac{f_2^2}{f_1^2} - 1 + 2\left(\frac{f_1^2}{f_L^2} - 1\right)\left(\frac{f_2^2}{f_L^2} - 1\right)}{\left|2\frac{f_2^2}{f_L^2} - \frac{f_2^2}{f_1^2} - 1\right|}\right),\end{aligned} \quad (25)$$

where the second equation holds because the coupling efficiency can be solved from Eqn (21) to be $\kappa^2 = \left(\frac{f_1^2}{f_L^2} - 1\right)\left(\frac{f_2^2}{f_L^2} - 1\right)$.

Eqn (25) captures how the lower resonance frequency f_L of the coupled system varies with the standalone frequency f_1 of the VSR. This modulation ratio is also affected by the stand-alone resonance frequency f_2 of the PER as well as their coupled resonance frequency f_L.

When the pump frequency f_p deviates slightly from $f_{cr} + f_{br}$, signal regeneration occurs most efficiently when the reactance-to-resistance ratios match in both circular and butterfly resonance paths, i.e.,

$$\frac{X_c}{R_c} = \frac{2(f_c - f_{cr})2\pi L_d}{R_c} = \frac{X_b}{R_b} = \frac{2(f_b - f_{br})2\pi L_b}{R_b}, \quad (26)$$

where f_c is the actual oscillation frequency, which is slightly different from its resonance frequency f_{cr}. Substituting $f_p = f_c + f_b$, we have

$$f_c = \frac{f_{cr}L_c/R_c - f_{br}L_b/R_b + f_p L_b/R_b}{L_c/R_c + L_b/R_b}. \quad (27)$$

Since the VSR modulates f_{cr} without affecting f_{br}, we have $f_{cr} = f_L$, where F_L is defined in Eqn (21). Since the voltage sensing resonator can somehow modulate the circular mode resonance frequency f_{cr} without affecting the butterfly mode resonance frequency f_{br}, the value of f_c in Eqn (27) can also be effectively modulated. The oscillation frequency shift ∂f_c and the resonance frequency shift ∂f_L are correlated by the partial derivative relation obtained from Eqn (27), i.e.,

$$\frac{\partial f_c}{\partial f_L} = \frac{L_c/R_c}{L_c/R_c + L_b/R_b}. \quad (28)$$

By plugging the derivative of Eqn (16) and plugging this derivative along with Eqn (25) into Eqn (28), we obtain the complete derivative with respect to v_s:

$$\begin{aligned}\frac{\partial f_c}{\partial v_s} &= \frac{\partial f_c}{\partial f_L} \cdot \frac{\partial f_L}{\partial f_1} \cdot \frac{\partial f_1}{\partial v_s} \\ &= \left(\frac{L_c/R_c}{L_c/R_c + L_b/R_b}\right) \cdot \left(\frac{f_L^3}{2f_1^3}\right) \\ &\quad \cdot \left[1 + \frac{f_2^2/f_1^2 - 1}{|2f_2^2/f_L^2 - f_2^2/f_1^2 - 1|} \right. \\ &\quad \left. + \frac{2(f_1^2/f_L^2 - 1)(f_2^2/f_L^2 - 1)}{|2f_2^2/f_L^2 - f_2^2/f_1^2 - 1|}\right] \cdot \left(-\frac{\lambda_1 f_{10}}{2\varphi_1}\right).\end{aligned} \quad (29)$$

Eqn (29) reveals a closed-form expression for how the oscillation frequency f_c is linearly modulated by the piezoelectric bias voltage v_s. This completes the proof.

Acknowledgments

The authors sincerely thank the anonymous reviewers and the shepherd for their valuable comments and constructive feedback. The work of Q. Wang and H. Zeng was supported by the NSF under Grants ECCS-2225337 and ECCS-2434001. The work of C. Qian was supported by the NSF under Grant ECCS-2144138.

Ethics Considerations

All user studies in this work were conducted in accordance with established ethical guidelines for research involving human participants. The study protocol was reviewed by the Institutional Review Board (IRB) of the authors' institution and was determined to be exempt from IRB oversight. Participants were informed of the study procedures and provided informed consent prior to participation. Participation was voluntary, and participants could withdraw at any time without penalty. All collected data were anonymized to protect participant privacy. No experiments were conducted on unaware individuals or in unauthorized real-world deployments.

LLM Usage Considerations

LLMs are used in this work as part of the proposed methodology. In particular, RadKey employs an LLM during online adaptation to generate pseudo ground-truth labels for semantically coherent typing inputs, which are then used to update the keystroke classifier at runtime. The LLM was run

locally on a research workstation, and no participant typing data were sent to external services. All technical claims, system design choices, implementations, and experimental results were developed, verified, and validated by the authors. Because LLM outputs may be imperfect and may vary across models and prompting settings, the LLM is used only as an auxiliary signal in the adaptation loop rather than as definitive ground truth.

References

[1] J. Harrison, E. Toreini, and M. Mehrnezhad, "A practical deep learning-based acoustic side channel attack on keyboards," in *2023 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*. IEEE, 2023, pp. 270–280.

[2] Y. Feng, D. Liu, W. Jin, and L. Gong, "Keyprint: Practical black-box keystroke inference attacks to mobile devices," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 9, no. 2, pp. 1–30, 2025.

[3] S. Luo, A. Nguyen, H. Farooq, K. Sun, and Z. Yan, "Eavesdropping on controller acoustic emanation for keystroke inference attack in virtual reality," in *The Network and Distributed System Security Symposium (NDSS)*, vol. 1, no. 2, 2024, p. 3.

[4] P. Wang, J. Hu, C. Liu, and J. Luo, "Reflexnoop: Passwords snooping on nlos laptops leveraging screen-induced sound reflection," in *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, 2024, pp. 3361–3375.

[5] A. Taheritajar, Z. M. Harris, and R. Rahaeimehr, "A survey on acoustic side channel attacks on keyboards," in *International Conference on Information and Communications Security*. Springer, 2024, pp. 99–121.

[6] Q. Wang, C. Qian, P. Yan, S. Zhang, and H. Zeng, "A batteryless wireless microphone using rf backscatter," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 9, no. 4, pp. 1–18, 2025.

[7] Q. Wang, P. Yan, C. Qian, and H. Zeng, "Radear: A self-supervised rf backscatter system for voice eavesdropping and separation," *arXiv preprint arXiv:2603.12446*, 2026.

[8] M. Vuagnoux and S. Pasini, "Compromising electromagnetic emanations of wired and wireless keyboards." in *USENIX security symposium*, vol. 8, 2009, pp. 1–16.

[9] Q. Jiang, Y. Ren, Y. Long, C. Yan, Y. Sun, X. Ji, K. Fu, and W. Xu, "Ghosttype: The limits of using contactless electromagnetic interference to inject phantom keys into analog circuits of keyboards," in *Network and Distributed Systems Security (NDSS) Symposium*, 2024.

[10] W. Jin, S. Murali, H. Zhu, and M. Li, "Periscope: A keystroke inference attack using human coupled electromagnetic emanations," in *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, 2021, pp. 700–714.

[11] Z. Zhan, Z. Zhang, S. Liang, F. Yao, and X. Koutsoukos, "Graphics peeping unit: Exploiting em side-channel information of gpus to eavesdrop on your neighbors," in *2022 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2022, pp. 1440–1457.

[12] S. Zhang, Q. Wang, M. Gan, Z. Cao, and H. Zeng, "Radsee: See your handwriting through walls using fmcw radar," in *Proceedings of Network and Distributed System Security Symposium (NDSS)*, 2025.

[13] S. Zhang, Q. Wang, K. Song, Q. Yan, and H. Zeng, "Radeye: Tracking eye motion using fmcw radar," in *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, 2025, pp. 1–13.

[14] E. Owusu, J. Han, S. Das, A. Perrig, and J. Zhang, "Accessory: password inference using accelerometers on smartphones," in *proceedings of the twelfth workshop on mobile computing systems & applications*, 2012, pp. 1–6.

[15] L. Cai and H. Chen, "{TouchLogger}: Inferring keystrokes on touch screen from smartphone motion," in *6th USENIX Workshop on Hot Topics in Security (HotSec 11)*, 2011.

[16] X. Liu, Z. Zhou, W. Diao, Z. Li, and K. Zhang, "When good becomes evil: Keystroke inference with smartwatch," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, 2015, pp. 1273–1285.

[17] J. Liu, Y. Chen, M. Gruteser, and Y. Wang, "Vibsense: Sensing touches on ubiquitous surfaces through vibration," in *2017 14th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. IEEE, 2017, pp. 1–9.

[18] J. Huang, J.-X. Bai, X. Zhang, Z. Liu, Y. Feng, J. Liu, X. Sun, M. Dong, and M. Li, "Keystrokesniffer: An off-the-shelf smartphone can eavesdrop on your privacy from anywhere," *IEEE Transactions on Information Forensics and Security*, 2024.

[19] J. Liu, Y. Wang, G. Kar, Y. Chen, J. Yang, and M. Gruteser, "Snooping keystrokes with mm-level audio ranging on a single phone," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, 2015, pp. 142–154.

[20] J.-X. Bai, B. Liu, and L. Song, "I know your keyboard input: a robust keystroke eavesdropper based-on acoustic signals," in *Proceedings of the 29th ACM International Conference on Multimedia*, 2021, pp. 1239–1247.

[21] L. Lu, J. Yu, Y. Chen, Y. Zhu, X. Xu, G. Xue, and M. Li, "Keylistener: Inferring keystrokes on qwerty keyboard of touch screen through acoustic signals," in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*. IEEE, 2019, pp. 775–783.

[22] M. Chen, J. Lin, W. Liu, and K. Wu, "Behavicker: Eavesdropping computer-usage activities through acoustic side channel," *Wireless Communications and Mobile Computing*, vol. 2022, no. 1, p. 8090652, 2022.

[23] R. Wijewickrama, M. Abbasihafshejani, A. Maiti, and M. Jadliwala, "Overhear: headphone based multi-sensor keystroke inference," *arXiv preprint arXiv:2311.02288*, 2023.

[24] Y. Zhao, Y. Zhao, S. Li, H. Han, and L. Xie, "Ultrasnoop: Placement-agnostic keystroke snooping via smartphone-based ultrasonic sonar," *ACM Transactions on Internet of Things*, vol. 4, no. 4, pp. 1–24, 2023.

[25] T. Zhu, Q. Ma, S. Zhang, and Y. Liu, "Context-free attacks using keyboard acoustic emanations," in *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*, 2014, pp. 453–464.

[26] J. Wang, K. Zhao, X. Zhang, and C. Peng, "Ubiquitous keyboard for small mobile devices: harnessing multipath fading for fine-grained keystroke localization," in *Proceedings of the 12th annual international conference on Mobile systems, applications, and services*, 2014, pp. 14–27.

[27] K. Ali, A. X. Liu, W. Wang, and M. Shahzad, "Keystroke recognition using wifi signals," in *Proceedings of the 21st annual international conference on mobile computing and networking*, 2015, pp. 90–102.

[28] Y. Meng, J. Li, H. Zhu, X. Liang, Y. Liu, and N. Ruan, "Revealing your mobile password via wifi signals: Attacks and countermeasures," *IEEE Transactions on Mobile Computing*, vol. 19, no. 2, pp. 432–449, 2019.

[29] Z. Yang, Y. Chen, Z. Sarwar, H. Schwartz, B. Y. Zhao, and H. Zheng, "Towards a general video-based keystroke inference attack," in *32nd USENIX Security Symposium (USENIX Security 23)*, 2023, pp. 141–158.

[30] A. Maiti, M. Jadliwala, J. He, and I. Bilogrevic, "Side-channel inference attacks on mobile keypads using smartwatches," *IEEE Transactions on Mobile Computing*, vol. 17, no. 9, pp. 2180–2194, 2018.

[31] M. Sabra, A. Maiti, and M. Jadliwala, "Zoom on the keystrokes: Exploiting video calls for keystroke inference attacks," *arXiv preprint arXiv:2010.12078*, 2020.

[32] A. Taheritajar and R. Rahaeimehr, "Acoustic side channel attack on keyboards based on typing patterns," *arXiv preprint arXiv:2403.08740*, 2024.

[33] P. Marquardt, A. Verma, H. Carter, and P. Traynor, "(sp) iphone: Decoding vibrations from nearby keyboards using mobile phone accelerometers," in *Proceedings of the 18th ACM conference on Computer and communications security*, 2011, pp. 551–562.

[34] B. Chen, V. Yenamandra, and K. Srinivasan, "Tracking keystrokes using wireless signals," in *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services*, 2015, pp. 31–44.

[35] J. V. Monaco, "Sok: Keylogging side channels," in *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2018, pp. 211–228.

[36] R. Menon, R. Gujarathi, A. Saffari, and J. R. Smith, "Wireless identification and sensing platform version 6.0," in *Proceedings of the 20th ACM Conference on Embedded Networked Sensor Systems*, 2022, pp. 899–905.

[37] V. Talla, B. Kellogg, S. Gollakota, and J. R. Smith, "Battery-free cellphone," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, no. 2, pp. 1–20, 2017.

[38] N. Arora, A. Mirzazadeh, I. Moon, C. Ramey, Y. Zhao, D. C. Rodriguez, G. D. Abowd, and T. Starner, "Mars: Nano-power battery-free wireless interfaces for touch, swipe and speech input," in *The 34th Annual ACM Symposium on User Interface Software and Technology*, 2021, pp. 1305–1325.

[39] S. A. Ayati, J. H. Park, Y. Cai, and M. Botacin, "Making acoustic {Side-Channel} attacks on noisy keyboards viable with {LLM-Assisted} spectrograms'" typo" correction," in *19th USENIX WOOT Conference on Offensive Technologies (WOOT 25)*, 2025, pp. 87–101.

[40] T. Ni, Y. Du, Q. Zhao, and C. Wang, "Non-intrusive and unconstrained keystroke inference in vr platforms via infrared side channel," *arXiv preprint arXiv:2412.14815*, 2024.

[41] J. H. Park, S. A. Ayati, and Y. Cai, "Improving acoustic side-channel attacks on keyboards using transformers and large language models," *arXiv preprint arXiv:2502.09782*, 2025.

[42] D. H. Roh, R. Kumar, and A. Ngo, "Llm-assisted cheating detection in korean language via keystrokes," *arXiv preprint arXiv:2507.22956*, 2025.

[43] K. F. Graff, *Wave motion in elastic solids*. Courier Corporation, 2012.

[44] W. Thomson, *Theory of vibration with applications*. CrC Press, 2018.

[45] L. D. Landau, L. Pitaevskii, A. M. Kosevich, and E. M. Lifshitz, *Theory of elasticity: volume 7*. Elsevier, 2012, vol. 7.

[46] S. P. Timoshenko and J. M. Gere, *Theory of elastic stability*. Courier Corporation, 2012.

[47] L. E. Kinsler, A. R. Frey, A. B. Coppens, and J. V. Sanders, *Fundamentals of acoustics*. John wiley & sons, 2000.

[48] K. Uchino, *Piezoelectric actuators and ultrasonic motors*. Springer Science & Business Media, 1996, vol. 1.

[49] Z. Dai, H. Liu, Q. V. Le, and M. Tan, "Coatnet: Marrying convolution and attention for all data sizes," *Advances in neural information processing systems*, vol. 34, pp. 3965–3977, 2021.

[50] R. Lin, P. Yan, J. Lu, Q. Wang, and H. Zeng, "Integrating health sensing into cellular networks: Human sleep monitoring using 5g signals," *arXiv preprint arXiv:2603.02558*, 2026.

[51] J. Lu, P. Yan, and H. Zeng, "Eexapp: Gnn-based reinforcement learning for radio unit energy optimization in 5g o-ran," *arXiv preprint arXiv:2602.09206*, 2026.

[52] P. Yan, J. Lu, H. Zeng, and Y. T. Hou, "Near-real-time resource slicing for qos optimization in 5g o-ran using deep reinforcement learning," *IEEE Transactions on Networking*, vol. 34, pp. 1596–1611, 2025.

[53] P. Yan, H. Zeng, and Y. T. Hou, "xdiff: Online diffusion model for collaborative inter-cell interference management in 5g o-ran," *IEEE Transactions on Networking*, 2025.

[54] A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Yang, A. Fan *et al.*, "The llama 3 herd of models," *arXiv e-prints*, pp. arXiv–2407, 2024.

[55] L. Zhuang, F. Zhou, and J. D. Tygar, "Keyboard acoustic emanations revisited," *ACM Transactions on Information and System Security (TISSEC)*, vol. 13, no. 1, pp. 1–26, 2009.

[56] M. Han, H. Yang, W. Li, W. Xu, X. Cheng, P. Mohapatra, and P. Hu, "Rf sensing security and malicious exploitation: A comprehensive survey," *arXiv preprint arXiv:2504.10969*, 2025.

[57] Z. Yu, Z. Kaplan, Q. Yan, and N. Zhang, "Security and privacy in the emerging cyber-physical world: A survey," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 3, pp. 1879–1919, 2021.

[58] Y. Tu, L. Shan, M. I. Hossen, S. Rampazzi, K. Butler, and X. Hei, "Auditory eyesight: Demystifying {μs-Precision} keystroke tracking attacks on unconstrained keyboard inputs," in *32nd USENIX Security Symposium (USENIX Security 23)*, 2023, pp. 175–192.

[59] T. Giallanza, T. Siems, E. Smith, E. Gabrielsen, I. Johnson, M. A. Thornton, and E. C. Larson, "Keyboard snooping from mobile phone arrays with mixed convolutional and recurrent neural networks," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 3, no. 2, pp. 1–22, 2019.

[60] Ü. Meteriz-Yıldıran, N. F. Yildiran, and D. Mohaisen, "Sia: Smartwatch-enabled inference attacks on physical keyboards using acoustic signals," in *Proceedings of the 20th Workshop on Workshop on Privacy in the Electronic Society*, 2021, pp. 209–221.

[61] D. Slater, S. Novotney, J. Moore, S. Morgan, and S. Tenaglia, "Robust keystroke transcription from the acoustic side-channel," in *Proceedings of the 35th Annual Computer Security Applications Conference*, 2019, pp. 776–787.

[62] S. Cecconello, A. Compagno, M. Conti, D. Lain, and G. Tsudik, "Skype & type: Keyboard eavesdropping in voice-over-ip," *ACM Transactions on Privacy and Security (TOPS)*, vol. 22, no. 4, pp. 1–34, 2019.

[63] A. Compagno, M. Conti, D. Lain, and G. Tsudik, "Don't skype & type! acoustic eavesdropping in voice-over-ip," in *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, 2017, pp. 703–715.

[64] Y. Berger, A. Wool, and A. Yeredor, "Dictionary attacks using keyboard acoustic emanations," in *Proceedings of the 13th ACM conference on Computer and communications security*, 2006, pp. 245–254.

[65] D. Asonov and R. Agrawal, "Keyboard acoustic emanations," in *IEEE Symposium on Security and Privacy, 2004. Proceedings. 2004*. IEEE, 2004, pp. 3–11.

[66] S. Fang, I. Markwood, Y. Liu, S. Zhao, Z. Lu, and H. Zhu, "No training hurdles: Fast training-agnostic attacks to infer your typing," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, 2018, pp. 1747–1760.

[67] A. K. Sikder, G. Petracca, H. Aksu, T. Jaeger, and A. S. Uluagac, "A survey on sensor-based threats and attacks to smart devices and applications," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 2, pp. 1125–1159, 2021.

Appendix A.
Meta-Review

The following meta-review was prepared by the program committee for the 2026 IEEE Symposium on Security and Privacy (S&P) as part of the review process as detailed in the call for papers.

A.1. Summary

This paper introduces RadKey, a covert keystroke eavesdropping attack that leverages a concealed, battery-less RF backscatter tag and a remote RF reader to infer typed text by translating keystroke vibrations into RF signals. By combining signal processing with an LLM-enabled online adaptation loop, the system achieves robust performance across diverse keyboards and environments, attaining over 90% F1-scores and under 10% character error rates.

A.2. Scientific Contributions

- Identifies an Impactful Vulnerability.
- Provides a Valuable Step Forward in an Established Field.

A.3. Reasons for Acceptance

1) The paper identifies and evaluates an impactful keystroke inference vulnerability which employs RF backscatter signals by employing an RF backscatter tag and remote RF reader in a practical attack setting.
2) The experimental evaluation is thorough, encompassing a wide range of conditions, including multiple table materials, keyboard types, and input modalities (spanning both linguistic and non-linguistic inputs), as well as a through-wall scenario in which the RF reader is concealed behind a wall.

A.4. Noteworthy Concerns

1) As the attack primarily relies on RF backscatter signals, the tag (and the attack itself) is easily detectable through RF spectrum monitoring and visual inspection of the target area.