Fast or Secure? Push the Limit of Privacy Leakage Threat via Charging Side-Channel Attacks

Jiaxin Jiang
Nanjing University of Posts and Telecommunications, Nanjing, China
jiaxin.jiang@njupt.edu.cn

Xutong Zhang
Nanjing University of Posts and Telecommunications, Nanjing, China
b23040317@njupt.edu.cn

Jiahao Li
Nanjing University of Posts and Telecommunications, Nanjing, China
b23090521@njupt.edu.cn

Leqi Zhao
Zhejiang University
Hangzhou, China
zhaoleqi@zju.edu.cn

Zhengxin Guo
Nanjing University of Posts and Telecommunications, Nanjing, China
guozx@njupt.edu.cn

Kaiyan Cui
Nanjing University of Posts and Telecommunications, Nanjing, China
kaiyan.cui@njupt.edu.cn

Ming Gao
Nanjing University of Posts and Telecommunications, Nanjing, China
gaomingppm@njupt.edu.cn

Jinsong Han
Zhejiang University
Hangzhou, China
hanjinsong@zju.edu.cn

Fu Xiao*
Nanjing University of Posts and Telecommunications, Nanjing, China
xiaof@njupt.edu.cn

Abstract

Mobile devices have become ubiquitous, along with charging technology evolving rapidly. However, recent research has revealed a privacy vulnerability. Power traces in USB cables during charging can be maliciously exploited to eavesdrop on users' private information. Existing attacks typically measure either voltage or current to characterize power traces in low-power USB charging (5V⎓1A). In this paper, we present a novel charging side-channel attack that broadens the attack surface to include fast charging. Our approach works for both wired and wireless charging scenarios. It combines both voltage and current traces for a more comprehensive threat analysis. We systematically explore the risk of information leakage during different operations, including screen inputs (password and keystroke inference), screen outputs (application/display detection), acoustic outputs (loudspeaker eavesdropping), and acoustic inputs (microphone eavesdropping). In particular, such duplex acoustic eavesdropping via charging side channels has not been studied before. Extensive experiments demonstrate that our attacks pose real-world threats. As a countermeasure, we propose design suggestions for future fast charging systems to prevent privacy leaks.

CCS Concepts

• **Security and privacy → Side-channel analysis and countermeasures**; *Mobile and wireless security*.

Keywords

Side-channel attack, charging, privacy leakage.

*Fu Xiao is the corresponding author.

WWW '26, Dubai, United Arab Emirates.

ACM ISBN 979-8-4007-2307-0/2026/04
https://doi.org/10.1145/3774904.3792629

ACM Reference Format:
Jiaxin Jiang, Xutong Zhang, Jiahao Li, Leqi Zhao, Zhengxin Guo, Kaiyan Cui, Ming Gao, Jinsong Han, and Fu Xiao. 2026. Fast or Secure? Push the Limit of Privacy Leakage Threat via Charging Side-Channel Attacks. In *Proceedings of the ACM Web Conference 2026 (WWW '26), April 13–17, 2026, Dubai, United Arab Emirates.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3774904.3792629

Resource Availability:
The source code of this paper has been made publicly available at https://zenodo.org/records/18296115.

1 Introduction

Battery-powered mobile devices, such as smartphones, have become integral to daily life. Manufacturers have incorporated larger batteries and fast-charging technologies into modern smartphones. Nevertheless, low-battery anxiety remains widespread [1, 2]. As battery levels decline, users may behave irrationally, abruptly abandoning ongoing activities for charging opportunities. They are likely to borrow chargers from strangers, rent a public power bank from widely-deployed charging stations [3], or seek wireless chargers that are now common in public areas [4].

Users often operate their smartphones while charging, a behavior that introduces a privacy vulnerability known as charging side-channel attacks [3–7]. The phone's operations are powered partially or entirely by the charging cable, rather than exclusively by the battery. Attackers can maliciously collect power trace data from USB cables to eavesdrop on private information. Such attacks can infer passwords/keystrokes [3–10], detect application/website activity [4, 6–8, 11–13], and eavesdrop on loudspeaker playback [4, 8, 14].

The upper limits of charging side channels have yet to be fully investigated. In traditional belief, it has been assumed that either voltage or current characteristics alone are sufficient to represent power traces for privacy leakage, as previous smartphones primarily utilized low-power USB charging protocols with fixed parameters (5V⎓1A). However, modern smartphones are increasingly equipped with fast-charging technologies. They can achieve higher charging

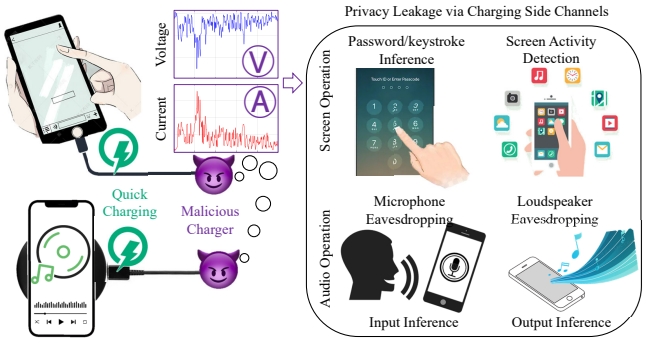


Figure 1: Illustration of our proposed charging side-channel attack, which exploits both voltage and current traces to eavesdrop on user data during fast charging. This attack poses a practical threat to both wired and wireless chargers in terms of screen- and audio-oriented privacy inference.

power by increasing voltage, current, or both. In this case, simple voltage or current characteristics can no longer capture the full complexity of power traces. The impact of fast charging on side-channel vulnerabilities thus remains unclear.

With this background in mind, we propose a novel charging side-channel attack, as illustrated in Fig. 1. Our attack comprehensively leverages both voltage and current characteristics to characterize the impact of victim operations on charging side channels. This presents two fundamental opportunities:

(1) Compatibility with fast-charging protocols. Existing attacks primarily target conventional low-power charging (5V⎓1A) [3, 9, 11, 14] or on a fixed 5 V direct-current (DC) voltage [5]. However, a variety of fast-charging technologies keep emerging, including USB-PD (Power Delivery) [15], Qualcomm QC (Quick Charge) [16] and wireless protocols, like Qi [17]. Their charging power levels can reach up to 240 W by increasing voltage, current, or both. In this case, traditional approaches that focus solely on voltage or current are insufficient to capture the full scope of possible leakage. Intuitively, a higher charging power amplifies this leakage. This perception, however, leads to several unanswered questions:

- Does higher power actually increase privacy leakage?
- Is current or voltage the dominant cause to this leakage?
- How can fast charging systems be redesigned to suppress or eliminate this side-channel vulnerability?

We systematically investigate these open questions by analyzing the combined effects of voltage and current in prevalent fast-charging systems and propose secure design guidelines for future chargers.

(2) Expansion of attack tasks. During charging, the most common operations are screen operations and phone calls. Existing attacks primarily focus on information leakage from screen interactions, such as touch input (e.g., passwords and keystrokes) [3–10] and screen output (e.g., application activities) [4, 6–8, 11, 12]. Recent studies have demonstrated charging side-channel attacks via loudspeaker eavesdropping [4, 8, 14] and false voice injection [5, 14]. However, the risk of microphone-based eavesdropping remains largely unexplored. Not only loudspeakers but also microphones draw additional power during use, altering the charging cable's power profile. This change can facilitate eavesdropping on live human speeches recorded by the smartphone, a threat that has not been previously investigated. Guided by these observations, we systematically analyze privacy leakage from the input/output (I/O) of the screen and audio. This analysis provides a comprehensive research framework for privacy leakage.

We investigate the potential risks posed by charging side channels, where we achieve four-fold privacy inference via a comprehensive analysis of both voltage and current traces. We model charging channels under both wired and wireless modes, taking into account the influence of screen and audio I/O activities. Accordingly, we detect the charging protocol and states, and subsequently employ deep learning techniques to identify private information. Table 1 compares our proposed approach with existing attacks. In particular, from the perspective of audio privacy, we eavesdrop on not only 'what you hear' through loudspeakers but also 'what you say' into microphones. In other words, our approach enables duplex speech eavesdropping. To implement this attack, we design a malicious, compact power trace capture module. This malicious module seamlessly supports both wired and wireless chargers, as well as battery banks, with plug-and-play capabilities in real-world scenarios. This exposes the threat of public charging and appeals for necessary countermeasures. Our open-source codes are available in [18].

In summary, our contributions are listed as follows.

- We propose a novel side-channel attack on fast charging in both wired and wireless modes. Our non-invasive PCB module enables practical deployment by eliminating the cable modifications required by previous research.
- A mathematical side-channel model demonstrates that voltage and current contribute to leakage distinctly and complementarily. Accordingly, our dual-channel (V+I) design maximizes leakage extraction and reveals the side-channel threat more comprehensively, which single-modality analysis cannot achieve. Against such vulnerabilities, we propose defensive strategies for future fast-charging systems.
- We implement duplex audio eavesdropping via charging channels. This audio I/O attack has never been studied before. This reveals fine-grained information leaked via charging well beyond what was previously believed.

2 Background and Related Work

Before the attack design, we briefly introduce fast charging technologies and survey the charging side-channel attacks.

2.1 Fast Charging Technology

Fast charging technologies have become ubiquitous among commercial off-the-shelf (COTS) mobile devices. These technologies operate through a suite of advanced power-delivery protocols, such as USB-PD and Qualcomm QC. By elevating the charging voltage to as high as 20 V and boosting the current, these protocols can deliver up to 240 W of power, far exceeding the legacy 5V⎓1A limit. The result is extraordinary: a battery can recover hours of runtime in just minutes. Its availability has expanded into our shared public spaces. In airports, malls, buses, subways, and cafes, shared power banks and wireless charging pads have made the rapid top-up a routine, almost unconscious, part of daily life.

Although widely adopted, fast charging technologies suffer from a prevalent privacy concern regarding the side-channel leakage.

Table 1: Comparison of Charging Side-channel Attacks.

Attack Means	Features	Smartphone OS	Charging Mode		Charging Power	Screen Task Inference		Audio Eavesdropping	
			Wired	Wireless		OUT (App. Disp.)	IN (Keystroke)	OUT (Spe.)	IN (Mic.)
[3]	Ⓥ	🤖 🍎	✔		5V⎓1A	✔			
[11]	Ⓥ	🤖	✔		5V⎓1A		✔		
[10]	Ⓐ	🤖	✔		5V⎓1A	✔	✔		
[5]	Ⓥ	🤖 🍎	✔		5V⎓1A/2A	✔	✔	✔	
[9]	Ⓥ	🤖 🍎		✔	5V⎓1A/2A	✔	✔		
[8]	Ⓐ	🤖 🍎		✔	5V⎓1A/2A	✔	✔	✔	
[4]	Ⓥ	🤖 🍎		✔	Not mentioned	✔	✔	✔	
Ours	Ⓥ+Ⓐ	🤖 🍎	✔	✔	Fast-charging	✔	✔	✔	✔

*: 🤖 represents Android, 🍎 represents Apple iOS, and represents HUAWEI HarmonyOS.

Intuitively, elevated charging power appears to amplify this leakage phenomenon. However, this assumption raises several critical questions that remain unexplored: whether increased power levels truly exacerbate the leakage, which factor, current or voltage, serves as the primary contributor to this vulnerability, and what design modifications could be implemented in fast charging systems to mitigate or eliminate this security risk entirely. This paper seeks to comprehensively examine and resolve these fundamental questions through systematic investigation.

2.2 Charging Side-Channel Attacks

Existing side-channel attacks are mainly focused on low-power USB charging (5V⎓1A) [3, 5, 9, 11, 14]. The potential threat of this side channel has not been fully explored.

Wired-charging side channels. Voltage or current ripples on charging lines leak user activity. Cronin et al. [3] localize touch points and recover passwords, while Yang et al. [11] fingerprint webpages. GhostTalk [14] turns the power line into an acoustic covert channel that injects commands into a voice assistant while quietly recording its answers. XPorter [5] leverages line-to-line crosstalk inside multi-port chargers to monitor a neighbor smartphone's screen and inject inaudible commands.

Wireless-charging side channels. Wireless charging also leaks user activity. EM-surfing [4] demonstrates the privacy leakage via wireless charging in inferring keystroke, application information, and loudspeaker playback. Surf-Snooping [9] exploits USB-crosstalk on ordinary wireless phone charging pads to recover device IDs, PINs and keystrokes, while Cour et al. [8] infer passwords, keystrokes, application activity and loudspeaker audio by measuring the electrical signals from the charging pad. BankSnoop [6] further adapts the channel to mobile wireless power-banks with few-shot learning. VoltSchemer [19] shows a malicious pad can manipulate voltage noise to inject inaudible voice commands.

3 Threat Analysis

3.1 Threat Model

Our threat model, consistent with prior research on charging side-channel attacks [3, 4], is defined by the attack scenario, the attacker's capabilities, and the attack vectors.

Attack Scenario. We consider a practical scenario where victims connect their smartphones to public chargers, such as shared wired power banks [3], wireless charging pads [4], or their own chargers plugged into public USB ports (e.g., in airports or cafes) [3, 4]. The charging equipment functions normally as expected, making the victim unaware of the underlying security risks.

Attacker Capabilities. The threat model assumes a non-intrusive attacker with the following limitations:

- No Physical Access: The attacker cannot access, tamper with, or directly operate the victim's smartphone.
- No Prior Knowledge: The attacker requires no prior information about the victim or the specific target device.

Attack Vector. We make the common assumption [3, 4] where the attack is executed by embedding a malicious monitoring system within the charger or its power interface. This system hardware, including a low-power microcontroller (detailed in Section 5.1), measures and records the smartphone's power trace throughout the charging process. Since this monitoring is passive with little impact on the charging functionality, it remains completely transparent to the user. The recorded power trace is leveraged to infer sensitive user information.

3.2 Security Risks from Leakage

Our attack poses a comprehensive threat to smartphones' screen and audio interfaces by inferring or eavesdropping on both user input and device output, as illustrated in Fig. 1 and Tab. 1.

3.2.1 Screen-oriented Inference. By analyzing a device's power trace, an attacker can infer a user's on-screen interactions, leading to two distinct types of information leakage:

- Input Leakage (Password/Keystroke): The attack can capture sensitive user inputs such as PINs, passwords, and keystrokes. A leaked PIN can grant an attacker with physical access complete control over the device (e.g., via an evil maid attack [3]). Even without physical access, a stolen PIN is dangerous due to widespread password reuse [20], which can potentially expose linked financial accounts.
- Output Leakage (Application Activities): Our attack also identifies which applications are displayed. This allows an attacker to profile the user's habits, health, and interests.

3.2.2 Audio-oriented Eavesdropping. Smartphone audio channels are also vulnerable. We extend the attack surface onto duplex eavesdropping on both audio I/O:

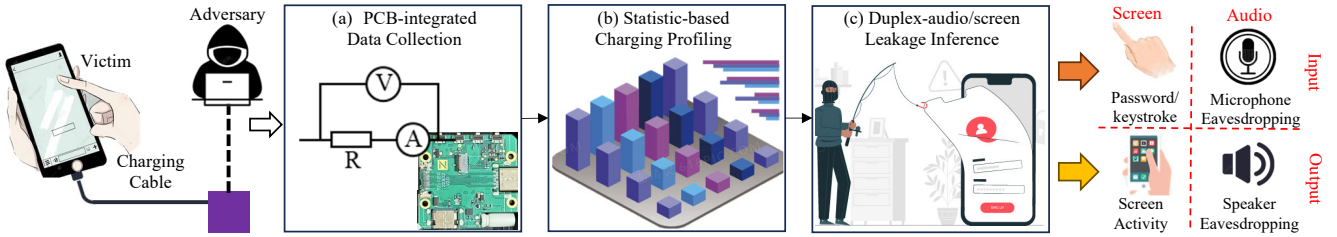


Figure 2: Overview of our proposed side-channel attack targeting fast charging. It is composed of three procedures for four-fold privacy inference, i.e., I/O operations of the smartphone screen and audio modules.

- Audio Output (Loudspeaker): The attacker can eavesdrop on sounds played through the loudspeaker. Attackers can infer private information via media playback and voice-assistant responses [4, 8, 14, 21–27].
- Audio Input (Microphone): In particular, we also capture the user's own voice as received by the microphone. These input signals introduce characteristic perturbations in the charger's power trace, which has never been explored in existing charging side-channel attacks [3–10].

Together, these capabilities enable comprehensive eavesdropping on both sides of a conversation in phone calls, online meetings, and voice interactions. This greatly expands the adversary's capabilities.

4 Charging Side-channel Modeling

We conduct a comprehensive analysis of charging-side channels. It describes how user inputs and device outputs from the screen and audio components affect the charging power in both wired and wireless modes.

For the screen, when the user taps on it, the touched area is activated at the moment of this touch. During application activities, the whole screen is always activated. This behavior directly influences power consumption. We model the dynamic power consumption of the screen P_S as

$$P_S = \sum\nolimits_{i\in R,G,B} A_S \cdot Pixel_i \cdot D_S(t) \cdot e^{-\frac{t^2}{2\sigma}} + P_{S0} \tag{1}$$

where A_S is the activated screen area, corresponding to a localized region for a tap or the entire screen for displays; $Pixel_i$ $(i \in R, G, B)$ is the power consumption per unit area for each respective RGB pixel layer; $D_S(t)$ represents the temporal pattern of the operation; σ is a constant parameter; and P_{S0} is an approximate constant denoting the power consumed by the miscellaneous components.

To properly model the duration, we must distinguish between these different actions. For user input (taps), the operation is finished nearly in an instant, so its duration can be described using a pulse $\delta(t)$ [28]. In contrast, for application activities displayed on the screen, the process exists for a period, whose impact can be approximately recognized as a gate signal [28]. Therefore, the temporal function $D_S(t)$ distinguishes between instantaneous and sustained events as

$$D_S(t) = \begin{cases} \delta(t) & \text{Input} \\ u(t) - u(t - t_0) & \text{Output} \end{cases} \tag{2}$$

where $\delta(t)$ is the Dirac delta function representing an instantaneous event, $u(t)$ is the unit step function, and t_0 is the duration of application activity display.

The screen behaves as an RC circuit [29], whose total impedance Z_S depends on the activated screen area A_S as

$$Z_S = R_S - j \cdot X_C(A_S) = R_S - j \cdot (\omega \cdot \sum\nolimits_i A_S \cdot C_{0i})^{-1} \tag{3}$$

where R_S is the equivalent series resistance; $X_C(A_S)$ is the capacitive reactance; j is the imaginary unit; ω is the angular frequency, a constant in the screen RC circuit; and C_{0i} is the capacitance per unit area for each pixel layer. According to [30], we have

$$I_S = \sqrt{P_S / R_S} \qquad U_S = I_S \cdot |Z_S| \tag{4}$$

Audio operations such as microphone recording (inputs) or speaker playback (outputs) cause dynamic power fluctuations. The power consumption P_A can be modeled [28, 31] as

$$P_A = k_A \cdot \sum\nolimits_{f_A} D_A[P_{SPL}] \cdot f_A + P_{A0} \tag{5}$$

where f_A is the acoustic frequency; P_{SPL} is the acoustic intensity; k_A is a scaling parameter; P_{A0} is the baseline power consumed by audio subsystems, including analog-to-digital/digital-to-analog converters (ADC/DACs), filters, and amplifiers; and $D_A(P_{SPL})$ is the audio power function [30], determined by the audio event as

$$D_A(P_{SPL}) = \begin{cases} \frac{\delta P_{SPL}}{\delta t}^2 & \text{Input} \\ P_{SPL}^2 & \text{Output} \end{cases} \tag{6}$$

Electrically, audio modules are modeled as RL circuits [31]. The total impedance Z_A is affected by the acoustic frequency f_A as

$$Z_A = R_A + j \cdot 2\pi f_A \cdot L_A \tag{7}$$

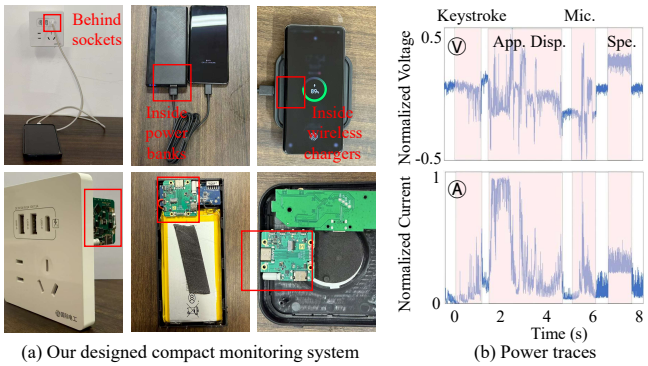
where R_A and L_A are the equivalent resistance and inductance of the module (loudspeakers or microphones). Its voltage and current traces follow Eq. 4 by replacing the subscript S by A.

In wireless charging mode, the charging pad operates as a transformer. It scales the voltage and current according to a transformation ratio without any distortion, with only a slight, constant power consumption due to internal losses.

According to the established side-channel model, both voltage and current traces are modulated by user inputs and device outputs. Consequently, joint utilization of these traces is essential to fully characterize the privacy risks posed by charging side channels.

5 Attack Design

We propose a novel side-channel attack that exploits privacy leakage from fast charging. Guided by our side-channel model, we leverage both voltage and current traces to infer passwords/keystrokes, detect application activity, and eavesdrop on microphones and loudspeakers, as illustrated in Fig. 2.



(a) Our designed compact monitoring system (b) Power traces

Figure 3: Our designed compact monitoring system and its implementation in three real-world attack scenarios, with the normalized power traces of the four private activities.

To execute this attack, we design a compact, plug-and-play malicious hardware module to collect and preprocess the raw power data. This module, compatible with both USB Type-A and Type-C standards, continuously records voltage and current traces during charging. The collected data is then subjected to preliminary analysis to perform phone model identification, fast-charging protocol recognition, and classification of potential privacy leakage types based on statistical characteristics. Finally, by correlating the power traces with screen and audio operations, we demonstrate the feasibility of significant information leakage.

5.1 PCB-integrated Data Collection

To execute the side-channel attack, we engineer a dedicated data acquisition system for the stealthy and high-fidelity monitoring of charging lines. The entire system is integrated onto a printed circuit board (PCB), which balances performance, size, and concealability.

5.1.1 Circuit Design. At the core of our module is a Huada HDSC HC32F460 microcontroller, which features a 32-bit ARM Cortex-M4 CPU. Once deployed, the module is bus-powered, drawing its operational energy directly from the Voltage Bus (VBUS) line it is monitoring. It samples both voltage and current at a synchronized frequency of 1 kHz. This sampling rate is empirically determined to be optimal, as it is sufficient to capture the high-frequency power signatures associated with user interactions while keeping the volume of generated data manageable. Figure 3(b) presents the collected power traces of the four activities, which are quite distinct from one another and from the idle state. Data is then relayed to a host system. A Bluetooth Low Energy (BLE) module allows for the covert, real-time streaming of data to a nearby attacker-controlled device. Alternatively, data can be stored on an onboard flash memory chip for later retrieval via a concealed wired interface.

5.1.2 Parameter Measurement. To measure current, we incorporate a high-precision shunt resistor in series with the VBUS power line. The minute voltage drop across this resistor is then amplified by a dedicated current-sense amplifier before being fed into the MCU's ADC. This configuration allows for sensitive and non-intrusive measurement of the current flow. The VBUS voltage, which can vary significantly during fast-charging negotiations (e.g., from 5V up to 20V), is monitored using a simple but effective voltage divider circuit that scales the voltage down to the ADC's safe input range.

5.1.3 Physical Form Factor. A fundamental design objective is to achieve a minimal physical footprint for real-world attack scenarios. With a total size of only 4.0×3.8 cm^2, our module is substantially more compact than prior solutions [3, 4, 8]. This compactness enables two primary covert deployment models:

- **Implant Model**: The PCB can be embedded inside the casing of a standard wall charger, replacing or piggybacking on the original circuitry. Therefore, it can be easily hidden in various real-world attack scenarios, as shown in Fig. 3(a).
- **Interposer Model**: The module can be housed in a small, innocuous-looking dongle that acts as an intermediary between the charger and the charging cable, or between the cable and the target device, under the same assumption as the existing methods [3, 4, 8].

5.2 Statistic-based Charging Profiling

Once captured, the raw power data is systematically profiled to form basic information for the subsequent privacy analysis.

5.2.1 Charging Protocols Identification. COTS devices support a variety of charging protocols. Our process involves a two-tiered identification: first, determining whether a device is using a low-power protocol or a fast-charging one, and second, specifying the exact fast-charging protocol being used. Common fast-charging protocols include standard protocols like various versions of USB-PD [15] and Qualcomm QC [16], as well as proprietary solutions such as Samsung's AFC (Adaptive Fast Charging), Huawei's FCP/SCP (Fast/Super Charge Protocol), and OPPO's VOOC (Voltage Open Loop Multi-step Constant-Current Charging) [32]. For compatibility with modern USB Type-C chargers, our hardware includes circuitry to monitor the Configuration Channel (CC) lines. This enables the module not only to capture raw power data but also to be aware of the protocol negotiations, providing valuable context for the attack. This is achieved by matching the detected patterns against the known ranges defined by each protocol's specification.

5.2.2 Smartphone Model Recognition. Variations in charging parameters (e.g., Eq. 1) cause different smartphone models to produce distinct power traces, even under otherwise identical conditions. Therefore, it is fundamental to recognize the model for subsequent privacy inference. To achieve this, we employ a method based on signal similarity matching. The core of this approach is the creation of a unique voltage-current template for each device model. The identification process involves comparing a power sample from a target device against a library of pre-recorded templates. We compute the Sliding Euclidean Similarity to resolve potential differences in signal length. The device model is considered a match if this similarity is above a set threshold.

5.2.3 Privacy Leakage Detection and Classification. Our attack detects the presence of private activities within power traces and classifies their specific types. This preliminary classification narrows the scope for subsequent analysis. We denoise using two filters: a high-pass filter with a 5 Hz cutoff to eliminate low-frequency bias, and a 10 Hz bandwidth notch filter centered at the mains frequency (60 Hz in the Americas or 50 Hz in Europe and East Asia)

to remove electrical grid interference. After smoothing the voltage and current traces with a sliding-window mean filter, we employ a rising/falling edge detection method [33]. It identifies the start/end points based on consecutive points that are continuously increasing or decreasing. Accordingly, we detect and segment the portions of the signal corresponding to potential private activities. The signal is zero-padded to a fixed length of 3000 samples for screen display tasks (1000 otherwise) to facilitate network processing.

For each segmented event, we extract a 20-dimensional feature vector composed of key statistical metrics from both the voltage and current signals, including their mean, median, standard deviation, maximum, minimum, and root mean square (RMS), among others. For the classification task, we utilize a random forest model, due to its computational efficiency and robust performance. The model is trained on the extracted feature vectors and their corresponding labels. In a real-world attack scenario, this classifier identifies the type of activity, which in turn determines the selection of a specialized deep learning model for the final privacy inference tasks.

5.3 Duplex-audio/screen Leakage Inference

We leverage deep learning techniques to infer the screen and audio privacy during charging. In particular, we demonstrate the feasibility of eavesdropping on the smartphone's microphone via the charging side channel, which has not been explored in prior work. Along with the capability of loudspeaker eavesdropping, our attack converts a charging smartphone into a duplex eavesdropper via deep learning, capable of monitoring both microphone-received and loudspeaker-generated audio, besides typing and display inference.

5.3.1 Network Input Selection. As modeled in Sec. 4, both voltage and current traces reveal private characteristics. For example, Eq. 3 shows that screen activation shapes voltage ripple patterns, while Eq. 4 characterizes $\delta(t)$-like touch operations that induce sharp current spikes. This theoretical orthogonality demonstrates that voltage and current provide complementary leakage insights. This motivates our dual-channel design to capture both modalities.

To leverage this, we treat the voltage and current traces as a two-channel input to our network, rather than processing the power trace as a single unit. These two channels are independently mean-filtered and normalized to eliminate the influence of the battery's current capacity, which determines the baseline voltage and current values [6]. Although spectrograms could offer richer information across time and frequency domains, we use the temporal signals directly to avoid the computational overhead of transformations like the Fast Fourier Transform (FFT). This approach ensures the real-time inference required for future on-chip integration.

5.3.2 Network Architecture. I/O operations exhibit distinct power consumption patterns. Equation 2 models input operations as transient $\delta(t)$ pulses, whereas Eqs. 5&6 describe output operations as sustained envelopes. This difference in temporal complexity necessitates distinct network architectures for input vs. output inference.

Input Operation Inference. Input operations like typing and audio-recording generate more dynamic and complex power traces. Consequently, we employ a residual network (ResNet) [34] to effectively extract the features corresponding to these dynamic input-related traces. As illustrated in Fig. 4, we utilize an attention-based

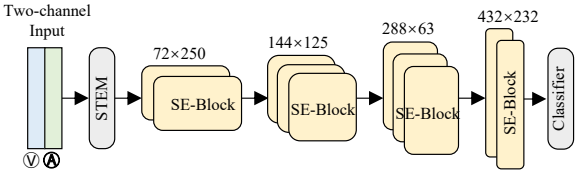


Figure 4: Network architecture for input privacy inference.

ResNet to process dynamic privacy features. Initially, the two-channel input undergoes preliminary feature extraction through a STEM module, which comprises convolutional layers, pooling layers, and Batch Normalization. Following this, the core of our network consists of four stages built from multiple Squeeze-and-Excitation (SE) blocks. These blocks adaptively recalibrate channel-wise feature responses based on a residual block backbone. Finally, a classifier, consisting of a pooling layer, a flattening layer, linear layers, and ReLU activation, outputs the privacy recognition results.

Output Operation Inference. For the relatively simpler patterns of output operations, we utilize a three-layer convolutional neural network (CNN) [35] to recognize screen content and loudspeaker playback. The CNN is composed of three sequential convolutional blocks for feature extraction, which feed into a final classifier to classify the victim device's screen display and audio playback.

6 Evaluation

6.1 Setup

We deploy our designed PCB monitoring system on various charging devices, including wired chargers, wireless chargers, and power banks, as partially shown in Fig. 3(a). Our experiments involve five smartphones that support fast charging, including a Samsung Galaxy S23 Ultra, a Samsung Galaxy S10, a Google Pixel 8 Pro (Android), an iPhone 15 (iOS), and a HUAWEI P60 (HarmonyOS). An ASUS TUF Gaming 5 Pro laptop equipped with an NVIDIA GeForce RTX 4070 Laptop and Intel Core i9-14900HX processor serves as the host system for data processing, model training, and evaluation.

Following IRB approval, we recruit six volunteers (three males and three females) to perform a series of predefined operations while charging. Participants are instructed to complete these tasks in their own preferred manner to simulate natural usage. For screen input tasks, we gather a training set of 100 samples for each digit (0-9) and each letter of the 26-key English alphabet, while the testing set comprises 50 4-digit PINs and 50 random words collected per phone. For screen output recognition, volunteers open ten popular applications and ten websites (listed in Appx. A) 100 times each. For audio eavesdropping, we record 100 samples of each English digit under two conditions: played back by the phone at maximum volume, and spoken by the volunteers at a natural volume (~62 dB). All data follows the 80/20 training/testing split.

6.2 Overall Performance

As shown in Fig. 5, our proposed attack achieves high performance. For the charging profiling task, it obtains nearly 100% accuracy across five different smartphones in charging protocol identification, model recognition, and privacy leakage detection.

For the screen inference task, our attack achieves an accuracy of over 98% in wired charging mode and over 94.5% in wireless

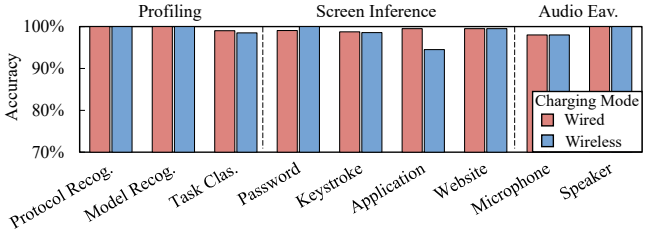


Figure 5: Overall performance.

charging mode on a Samsung Galaxy S23 Ultra charging at 45 W. The decrease in performance for wireless charging can be attributed to the additional energy loss inherent in the process, which introduces noise and lowers the signal-to-noise ratio (SNR).

Furthermore, we implement duplex eavesdropping. Our attack intercepts not only loudspeaker playback but also user speech, which has never been investigated before. This effectively turns the charger into a comprehensive surveillance device that monitors what the user hears and what they say.

6.3 Impact of Power Parameter

Given the demonstrated vulnerability of fast charging, we explore how privacy leakage depends on charging-power parameters.

6.3.1 Impact of Fast vs. Slow Charging. We compare fast charging and conventional 5W charging that follows the dedicated charging port (DCP) protocol, to analyze the impact of power delivery on privacy leakage. As shown in Fig. 6, privacy leakage increases with charging power in wired mode but decreases in wireless mode for an iPhone 15. However, we also observe that even at 15 W, the negotiated power profiles differ: for example, 5V⎓3A in the wired case in Fig.6(a) versus 9V⎓1.67A in the wireless case in Fig.6(b). This raises the question of whether voltage and current contribute differently to the observed leakage, i.e., whether their effects can be disentangled beyond total power.

6.3.2 Impact of Voltage vs. Current. To isolate the individual impacts of voltage and current, we conduct experiments on a Samsung Galaxy S23 Ultra, chosen for its support of diverse charging power levels. As shown in Tab. 2, we compare common configurations at 5 V and 9 V across a range of corresponding currents. The results reveal two clear and opposing trends: at a constant voltage, privacy leakage intensifies as the current rises, whereas for a given current, leakage diminishes as the voltage increases.

6.3.3 Root Cause Analysis. According to Eq. 3~4, the current increase amplifies the instantaneous fluctuations, thermal noise ($P_{\text{heat}} \propto I^2R$), and ground bounce $V_{\text{bounce}} = L\frac{di}{dt}$, all of which contribute to

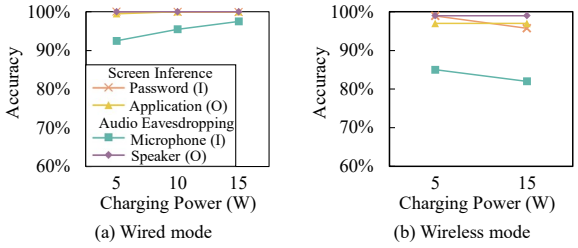


Figure 6: Impact of charging power on an iPhone 15.

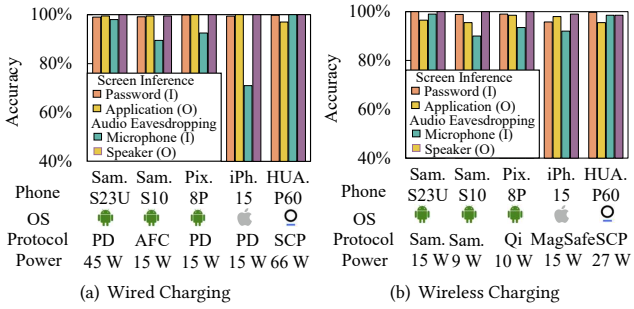


Figure 7: Performance across devices.

observable leakage. Conversely, voltage increases introduce more noise artifacts, such as those caused by grid load variations[5]. These artifacts degrade the SNR of privacy-related signals. In addition, high-voltage operating modes activate advanced power management circuits (e.g., multi-phase buck converters). These circuits act as an isolator that distorts privacy-related signals.

6.4 Scalability Study

We further analyze the scalability of our proposed attacks across diverse devices, charging modes, and users.

6.4.1 Across Smartphone Models. We evaluate our charging side-channel attack on five smartphones. They run different OSes and support multiple charging protocols. As shown in Fig. 7, all five devices, including a Samsung Galaxy S23 Ultra, a Samsung Galaxy S10, a Google Pixel 8 Pro (Android), an iPhone 15 (iOS), and a HUAWEI P60 (HarmonyOS), are vulnerable across all four kinds of privacy-inference tasks, regardless of OS and protocol.

6.4.2 Across Charging Modes & Chargers. As shown in Fig. 7, our attack is effective under both wired and wireless charging modes. It supports various chargers from different brands, protocols, and power levels. We also evaluate portable power banks, which are likewise vulnerable. The average inference accuracies are 99.5%, 100%, 88.5%, and 100% for PIN entry, application recognition, microphone (speech), and loudspeaker (audio playback), respectively.

6.4.3 Across Users. Since phone usage patterns vary across individuals, we perform cross-user evaluation to demonstrate the effectiveness of our attacks against real-world victims. The model is trained on data from a single, randomly selected participant and evaluated on testing data from the remaining users. Our approach maintains high cross-user accuracy, averaging 88.1% for keyword inference, 92.5% for application inference, 36.7% for eavesdropping on microphones, and 99.7% for speakers. This substantial degradation in microphone eavesdropping is mainly attributed to the pronounced vocal differences among users. A meta-learning strategy will be adopted in future work to improve cross-user microphone eavesdropping performance. In general, our results surpass the cross-user performance in existing work [4].

7 Countermeasure and Suggestion

To address the threat of charging side-channel leakage, we design `Charging Shield`, a deployable mobile application for smartphones, with an evaluation on its effectiveness and efficiency. Its open-source codes and `.apk` file are available in [18].

Table 2: Performance under different charging configurations for a Samsung Galaxy S23 Ultra

Mode	Charging configuration				Screen Inference		Audio Eavesdropping	
	Voltage	Current	Power	Protocol	IN (Password)	OUT (App. Disp.)	IN (Mic.)	OUT (Spe.)
Wired	5V	1 A	5 W	DCP	100%	97.0%	100%	99.5%
		2 A	10 W	DCP	100%	99.5%	100%	100%
	9V	1.67 A	15 W	PD 2.0	93.3%	99.5%	72.0%	100%
		2 A	18 W	PD 3.0	98.9%	99.5%	96.5%	100%
		2.77 A	25 W	PD 3.0	98.5%	100%	98.0%	100%
		5 A	45 W	PD 3.0	99.1%	99.5%	98.0%	100%
Wireless	5 V	1 A	5 W	Qi	100.0%	94.5%	98.0%	100.0%
	9 V	1 A	9 W	Samsung Fast Wireless	99.9%	94.0%	98.0%	100.0%
		1.67 A	15 W	Samsung Fast Wireless	100.0%	96.5%	99.0%	100.0%

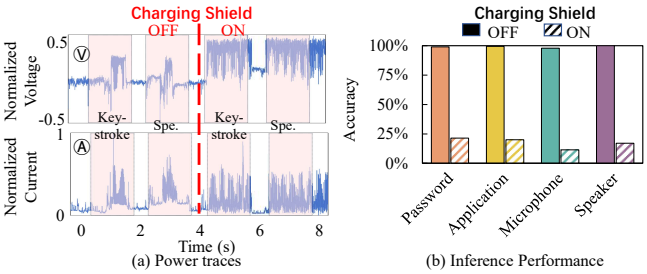


Figure 8: `Charging Shield` On vs. Off Comparison.

Existing software-based defense mechanisms suffer from significant practical limitations. Some methods offer only narrow protection, such as keyboard randomization, which solely protects keystrokes. Code injection techniques [36] obscure power trace features by embedding non-functional code. However, this method is difficult to scale, as it requires modifying each application's code, resulting in an enormous workload. A widely-recognized method [4, 6, 8] jams the side channel by inducing random power consumption during charging. This approach, while effective at jamming, leads to prolonged charging times, directly conflicting with the primary objective of fast charging.

As a countermeasure, we develop `Charging Shield` via Android Studio. This application balances the jamming effectiveness against side-channel attacks and the charging efficiency. Upon detecting private activity, it activates a random number of random computational loads to pollute charging side channels. These loads last for randomized durations to mimic continuous user actions (e.g., typing) and effectively mask the underlying signal. This on-demand detection mechanism, using the network in Sec. 5.2.3, avoids the power consumption of an always-on defense. Moreover, we design an adaptive mechanism to adjust the loads according to the phone's battery level. When the battery is low (e.g., below 50%), the quantity and duration of the loads are decreased to prioritize charging speed. Conversely, when the battery level is high (e.g., above 90%), their quantity and duration are increased to enhance protection. Users can also manually select the desired load level. More information and the interface of `Charging Shield` are present in Appx. B.

Our proposed application effectively mitigates side-channel leakage. As compared in Fig. 8(a), the user activities in charging side channels are completely jammed by `Charging Shield`. As shown in Fig. 8(b), it experimentally reduces the average attack accuracy across the four types of privacy leakage below 20% on a Samsung Galaxy S23 Ultra. This robust protection is achieved at a minimal cost to charging efficiency. During a five-minute charging period, `Charging Shield` consumes only an additional 10 mAh. In this interval, the smartphone charges from 50% to 55% with the application active while a user performs 60 random activities, compared to 56% with it turned off. These results demonstrate that Charging Shield successfully provides secure privacy protection and positive user experience simultaneously for fast charging.

Though our application works effectively, a fundamental solution to this vulnerability necessitates a secure charger design. Simple hardware defenses, such as adding a low-pass filter [3], are fundamentally flawed. They disrupt the handshake of fast-charging protocols, causing negotiation failures and forcing a fallback to slower charging speeds. A more robust strategy integrates security into the charger's design. According to our analysis in Sec. 6.3, we suggest that future protocols should favor voltage-based power negotiation, a method naturally resilient to noise that minimizes information leakage. Moreover, employing multi-phase buck converters not only improves efficiency but also inherently obfuscates the side-channel signature by distorting power draw patterns, decoupling user activity from the power line.

8 Conclusion

We present a practical charging side-channel attack that generalizes across charging modalities. By leveraging synchronized voltage and current traces, our method operates under fast charging and demonstrates vulnerabilities across four privacy-inference tasks covering both screen and audio I/O. In particular, we expose a previously unstudied surface, i.e., duplex acoustic eavesdropping via the charging interface, capturing both loudspeaker output and microphone input. By analyzing the respective roles of voltage and current in this side channel, we develop countermeasures and offer design recommendations for securing future fast-charging systems.

Acknowledgments

This paper is partially supported by National Science Fund for Distinguished Young Scholars of China (Grant No. 62125203), National Natural Science Foundation of China (Grant No. 62502235, No. 62372400, No. 62402238, NO. 62402237), Natural Science Foundation of Jiangsu Province of China (Grant No. BK20240615).

References

[1] Wired. 2019. Diagnosing (and Dealing With) Your Low-Battery Anxiety. https://www.wired.com/story/diagnosing-and-dealing-with-your-low-battery-anxiety/.

[2] Battery Hacker. 2023. How To Overcome The Fear Of A Low Battery? A Step-by-Step Guide. https://batteryhacker.com/how-to-overcome-the-fear-of-a-low-battery/.

[3] Patrick Cronin, Xing Gao, Chengmo Yang, and Haining Wang. 2021. Charger-Surfing: Exploiting a Power Line Side-Channel for Smartphone Information Leakage. In *Proceedings of USENIX Security*. 681–698.

[4] Jianwei Liu, Xiang Zou, Leqi Zhao, Yusheng Tao, Sideng Hu, Jinsong Han, and Kui Ren. 2024. Privacy Leakage in Wireless Charging. *IEEE Transactions on Dependable and Secure Computing* 21, 2 (2024), 501–514. doi:10.1109/TDSC.2022.3173063

[5] Tao Ni, Yongliang Chen, Weitao Xu, Lei Xue, and Qingchuan Zhao. 2023. XPorter: A Study of the Multi-Port Charger Security on Privacy Leakage and Voice Injection. In *Proceedings of ACM MobiCom*. 78:1–78:15.

[6] Tao Ni, Jianfeng Li, Xiaokuan Zhang, Chaoshun Zuo, Wubing Wang, Weitao Xu, Xiapu Luo, and Qingchuan Zhao. 2023. Exploiting Contactless Side Channels in Wireless Charging Power Banks for User Privacy Inference via Few-shot Learning. In *Proceedings of ACM MobiCom*. 73:1–73:15.

[7] Tao Ni, Xiaokuan Zhang, Chaoshun Zuo, Jianfeng Li, Zhenyu Yan, Wubing Wang, Weitao Xu, Xiapu Luo, and Qingchuan Zhao. 2023. Uncovering User Interactions on Smartphones via Contactless Wireless Charging Side Channels. In *Proceedings of IEEE SP*. 3399–3415.

[8] Alexander S. La Cour, Khurram K. Afridi, and G. Edward Suh. 2021. Wireless Charging Power Side-Channel Attacks. In *Proceedings of ACM CCS*. 651–665.

[9] Yue Hou, Xinyan Zhou, Huakang Xia, Jian Wang, and Haiming Chen. 2025. Surf-Snooping: USB crosstalk leakage attacks on wireless charging. *Computers & Security* 154 (2025), 104412.

[10] Richard Matovu, Abdul Serwadda, Argenis V. Bilbao, and Isaac Griswold-Steiner. 2020. Defensive Charging: Mitigating Power Side-Channel Attacks on Charging Smartphones. In *Proceedings of ACM CODASPY*. 179–190.

[11] Qing Yang, Paolo Gasti, Gang Zhou, Aydin Farajidavar, and Kiran S. Balagani. 2017. On Inferring Browsing Activity on Smartphones via USB Power Analysis Side-Channel. *IEEE Transactions on Information Forensics and Security* 12, 5 (2017), 1056–1066.

[12] Yi Wu, Zhuohang Li, Nicholas Van Nostrand, and Jian Liu. 2021. Time to Rethink the Design of Qi Standard? Security and Privacy Vulnerability Analysis of Qi Wireless Charging. In *Proceedings of ACM ACSAC*. 916–929.

[13] Riccardo Spolaor, Ning Feng, Yi Liu, Xiuzhen Cheng, and Pengfei Hu. 2025. PowerApp: Mobile Apps and User Actions Identification via USB Power Channel Analysis. In *Proceedings of IEEE ICC*. 01–06.

[14] Yuanda Wang, Hanqing Guo, and Qiben Yan. 2022. GhostTalk: Interactive Attack on Smartphone Voice System Through Power Line. In *Proceedings of NDSS*.

[15] Bob Dunstan and Richard Petrie. 2019. USB Power Delivery. In *Proceedings of USB Developer Days*.

[16] Qualcomm Inc. 2017. Qualcomm Quick Charge 5. https://www.qualcomm.com/products/features/quick-charge.

[17] Wireless Power Consortium. 2019. Taking charge of wireless power. https://www.wirelesspowerconsortium.com/cn/.

[18] Jijoxve. 2026. Fast-or-Secure-Push-the-Limit-of-Privacy-Leakage-Threat-via-Charging-Side-Channel-Attacks: Source Code for WWW 2026 Paper. https://zenodo.org/records/18296115.

[19] Zihao Zhan, Yirui Yang, Haoqi Shan, Hanqiu Wang, Yier Jin, and Shuo Wang. 2024. VoltSchemer: Use Voltage Noise to Manipulate Your Wireless Charger. In *Proceedings of USENIX Security*. 3979–3995.

[20] Dinei Florencio and Cormac Herley. 2007. A large-scale study of web password habits. In *Proceedings of ACM WWW*. 657–666.

[21] Qingsong Yao, Yuming Liu, Xiongjia Sun, Xuewen Dong, Xiaoyu Ji, and Jianfeng Ma. 2024. Watch the Rhythm: Breaking Privacy with Accelerometer at the Extremely-Low Sampling Rate of 5Hz. In *Proceedings of ACM CCS*. 1776–1790.

[22] Lei Wang, Meng Chen, Li Lu, Zhongjie Ba, Feng Lin, and Kui Ren. 2023. VoiceListener: A Training-free and Universal Eavesdropping Attack on Built-in Speakers of Mobile Devices. *Proceedings of IMWUT/Ubicomp* 7, 1 (2023), 32:1–32:22.

[23] Ke Sun, Chunyu Xia, Songlin Xu, and Xinyu Zhang. 2023. StealthyIMU: Extracting Permission-protected Private Information from Smartphone Voice Assistant using Zero-Permission Sensors. In *Proceedings of NDSS*.

[24] Ming Gao, Yajie Liu, Yike Chen, Yimin Li, Zhongjie Ba, Xian Xu, and Jinsong Han. 2022. InertiEAR: Automatic and Device-independent IMU-based Eavesdropping on Smartphones. In *Proceedings of IEEE INFOCOM*. 1129–1138.

[25] Yan Michalevsky, Dan Boneh, and Gabi Nakibly. 2014. Gyrophone: Recognizing Speech from Gyroscope Signals. In *Proceedings of USENIX Security*. 1053–1067.

[26] Zhongjie Ba, Tianhang Zheng, Xinyu Zhang, Zhan Qin, Baochun Li, Xue Liu, and Kaili Ren. 2020. Learning-based Practical Smartphone Eavesdropping with Built-in Accelerometer. In *Proceedings of NDSS*.

[27] S. Abhishek Anand and Nitesh Saxena. 2018. Speechless: Analyzing the Threat to Speech Privacy from Smartphone Motion Sensors. In *Proceedings of IEEE S&P*. 1000–1017.

[28] Pijush Kanti Dutta Pramanik, Nilanjan Sinhababu, Bulbul Mukherjee, Sanjeevikumar Padmanaban, Aranyak Maity, Bijoy Kumar Upadhyaya, Jens Bo Holm-Nielsen, and Prasenjit Choudhury. 2019. Power Consumption Analysis, Measurement, Management, and Issues: A State-of-the-Art Review of Smartphone Battery and Energy Usage. *IEEE Access* 7 (2019), 182113–182172.

[29] Kai Wang, Richard Mitev, Chen Yan, Xiaoyu Ji, Ahmad-Reza Sadeghi, and Wenyuan Xu. 2022. GhostTouch: Targeted Attacks on Touchscreens without Physical Touch. In *Proceedings of USENIX Security*. https://www.usenix.org/conference/usenixsecurity22/presentation/wang-kai

[30] Charles K. Alexander and Matthew N. O. Sadiku. 2016. *Fundamentals of Electric Circuits* (6th ed.). McGraw-Hill Education.

[31] Lide Zhang, Birjodh Tiwana, Robert P. Dick, Zhiyun Qian, Z. Morley Mao, Zhaoguang Wang, and Lei Yang. 2010. Accurate online power estimation and automatic battery behavior based power model generation for smartphones. In *Proceedings of IEEE/ACM/IFIP CODES+ISSS*. 105–114.

[32] Wired. 2025. How to Fast-Charge Your Smartphone. https://www.wired.com/story/how-to-fast-charge-your-phone/.

[33] Shanshan Yu, Ju Liu, Jing Wang, and Inam Ullah. 2020. Adaptive Double-Threshold Cooperative Spectrum Sensing Algorithm Based on History Energy Detection. *Wireless Communication and Mobile Computing* 2020 (2020), 4794136:1–4794136:12.

[34] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of CVPR*. 770–778.

[35] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Proceedings of NeurIPS*. 1097–1105.

[36] Lin Yan, Yao Guo, Xiangqun Chen, and Hong Mei. 2015. A Study on Power Side Channels on Mobile Devices. In *Proceedings of ACM Internetware*. doi:10.1145/2875913.2875934

A Targets of Display Inference

In the task of inferring the screen outputs, the ten tested applications include Amazon, Alipay, Chrome, Instagram, Messenger, Taobao, TikTok, WeChat, WhatsApp, and YouTube. The ten tested websites include Amazon, Facebook, Google, Instagram, LinkedIn, TikTok, Twitter, WhatsApp, Wikipedia, and YouTube.

B Our Developed Defending Application `Charging Shield`

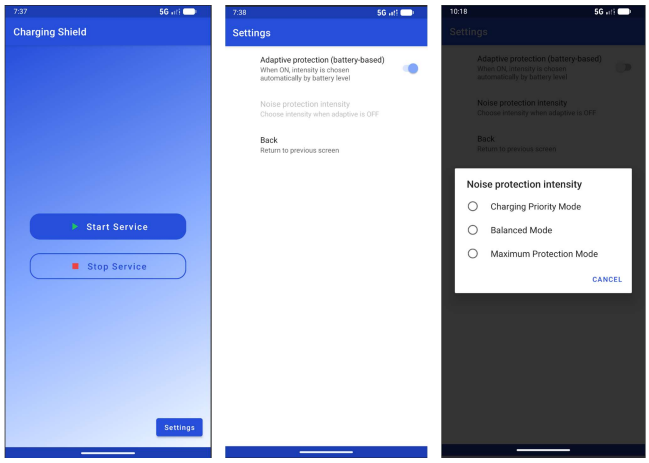


Figure 9: User interfaces and settings of `Charging Shield`.

Algorithm 1: ChargingShield: Randomized Noise Generation During Charging

Procedure *ChargingShieldMain*
while *true* **do**
  while *not IsCharging()* **do**
    Sleep(jitter);
  while *IsCharging()* **do**
    if *DetectUserActivity()* **then**
      $k \leftarrow \text{UniformInt}(K_{\min}, K_{\max})$;
      for $j = 1 \rightarrow k$ **do**
        $type \leftarrow$ WeightedChoice{Arithmetic, MatMul, Hash};
        $duration \leftarrow$ RandomDuration();
        $offset \leftarrow \text{Uniform}(-\Delta_{pre}, \Delta_{post})$;
        $\rho \leftarrow \text{Uniform}(\rho_{\min}, \rho_{\max})$;
        $t_{\text{start}} \leftarrow \max(\text{Now}(), \text{Now}() + offset)$;
        spawn NoisyTask($type, duration, \rho, t_{\text{start}}$);
    Sleep(jitter);
  CancelAllNoiseTasks();
Procedure *NoisyTask*
  $type, duration, \rho, t_{\text{start}}$
if $t_{start} > Now()$ **then**
  Sleep($t_{\text{start}} - \text{Now}()$);
$t_{\text{end}} \leftarrow \text{Now}() + duration$;
while $Now() < t_{end}$ ***and*** *IsCharging()* **do**
  $t_{\text{busy}} \leftarrow \rho \cdot \text{TimeSlice}$;
  Execute($type, t_{\text{busy}}$);
  Sleep($(1-\rho) \cdot \text{TimeSlice} + \text{jitter}$);
Procedure *Execute*
  $type, t_{\text{busy}}$
$t_0 \leftarrow \text{Now}()$;
if $type = Arithmetic$ **then**
  $i \leftarrow 0$;
  while $Now() - t_0 < t_{busy}$ **do**
    $i \leftarrow i + 1$;
else if $type = MatMul$ **then**
  $N \leftarrow \text{UniformInt}(N_{\min}, N_{\max})$;
  $A, B, C \leftarrow$ random $N \times N$;
  while $Now() - t_0 < t_{busy}$ **do**
    $C \leftarrow A \times B$;
else // Hash
  $S \leftarrow \text{UniformInt}(S_{\min}, S_{\max})$;
  $buf \leftarrow$ random bytes of size S;
  while $Now() - t_0 < t_{busy}$ **do**
    $h \leftarrow \text{SHA256}(buf)$;

`Charging Shield` enhances user privacy by operating exclusively during charging, thereby preventing interference with normal device functionality and significantly reducing the risk of data leakage.

`Charging Shield` activates a random number of computational loads for randomized durations to obfuscate the side-channel signals upon detecting the user activities. These loads are intentionally diverse, ranging from simple arithmetic (e.g., $i = i + 1$) to complex, CPU-intensive tasks like matrix multiplication and cryptographic hashing. The randomized-duration strategy is designed to mimic private activities, such as continuous typing, which typically occur over a period rather than as a single operation. In particular, it deliberately creates ambiguity around the exact start and end times of those activities. By not attempting to perfectly align the noise with the sensitive operation, our approach makes it computationally difficult for an adversary to determine the true temporal boundaries of the event. Consequently, this thwarts attempts to isolate the privacy-related power signal, even with advanced signal extraction techniques. Its basic idea is presented in Algo. 1.

`Charging Shield` balances robust side-channel protection with practical charging efficiency. We implement a dynamic, adaptive mechanism that modulates the intensity of the defensive loads based on the device's real-time battery level, operating in several distinct modes:

- **Charging Priority Mode**: When the battery is low (e.g., <50% battery) and charging speed is most fundamental, our designed `Charging Shield` reduces the quantity and duration of the computational loads. This minimizes their impact on the charging cycle, ensuring the user can power up their device quickly.
- **Maximum Protection Mode**: When the battery is nearly full (e.g., >90% battery), charging efficiency becomes a secondary concern. In this state, the system escalates the defensive measures, activating a higher number of loads for longer durations. This provides the strongest possible obfuscation of side-channel signals when the user's need for fast charging is minimal.
- **Balanced Mode**: In the intermediate battery range (for example, 60%∼90% Battery), the system offers a compromise between the two extremes, providing a moderate level of protection without significantly impacting charging time.

The user interfaces of `Charging Shield` are illustrated in Fig. 9. Users can choose the battery-based adaptive protection mode or select one of the three predefined modes according to their needs. Its open-source code and `.apk` file are available in [18].