Privacy Leakage in Wireless Charging

Jianwei Liu, *Student Member, IEEE*, Xiang Zou, *Student Member, IEEE*, Leqi Zhao, Yusheng Tao, Sideng Hu, *Member, IEEE*, Jinsong Han, *Senior Member, IEEE*, and Kui Ren, *Fellow, IEEE*

Abstract—Wireless charging is becoming an essential power supply pattern for electronic devices. Currently, mainstream smartphones are almost compatible with wireless charging. However, when the charging efficiency is continuously improved, its security challenge still remains open yet overlooked. In this paper, we reveal that severe security flaws exist in the wireless charging procedure of off-the-shelf commodity smartphones. Specifically, we find that an attacker can utilize the electromagnetic induction effect between the wireless charger and the smartphone to detect the activities and operations performed on the smartphone. We term such attack as *EM-Surfing* side-channel attack and build a theoretical model to show its feasibility. To explore the hazard of *EM-Surfing*, we propose a three-module attack method, with which we conduct real-world experiments over three mainstream models of smartphones. The results show that the attacker can achieve over 99%, 96%, 94%, and 97% accuracy when inferring the passcode, keystroke, App information, and speech content, respectively. We also design an App named *SecCharging* to prevent smartphones from *EM-Surfing* attacks. The defense experiment results demonstrate that *SecCharging* can mitigate the threats posed by *EM-Surfing* effectively.

Index Terms—Side-channel, wireless charging, machine learning

1 Introduction

Wireless charging, as a kind of remote power transfer technology, has been widely deployed. In particular, the wireless charging technology for smartphones becomes pervasive in recent years. Such rapid development is driven by many benefits of wireless charging. For example, using wireless charger is safer and more harmless, because it does not reproduce any sparkle or electrical shock, which often incur in wired charging scenarios [1]. Given that wireless charging is supported by more and more phone models in the electronics market, wireless charging may be a common configuration for all phones in the future [2].

Technically speaking, smartphone wireless charging is enabled by the electromagnetic induction effect [3] between two components, i.e., a transmitter coil and a receiver coil [4]. With this effect, the transmitter coil can generate a fluctuating magnetic field to produce inductive potential on the receiver coil, resulting in remote power transfer. However, due to the openness of magnetic field and the universality of the electromagnetic induction effect, wireless charging inevitably raises security concerns.

Unfortunately, prior works related to wireless charging dominantly focus on enhancing the charging efficiency [5], [6], [7], while the security aspects have not been given sufficient attention. In this paper, we reveal that there is a severe security threat in smartphone wireless charging. An attacker can infer the private data entered/shown on a smartphone by analyzing the voltage dynamics on the power line of the wireless charger. We term such attack as *Electromagnetic-Surfing* (*EM-Surfing*) attack. The root cause triggering *EM-Surfing* side-channel lies in the correlation between the user's private information and the induced voltage change on the transmitter coil of the wireless charger. In wireless charging, the electromagnetic induction effect between the transmitter coil (wireless charger) and the receiver coil (smartphone) is mutual. Thus, once the user performs operations or activities on the smartphone, the load of the receiver coil inevitably changes, which further produces extra voltage on the transmitter coil. As a result, a malicious third party could monitor the smartphone's activities by observing the voltage dynamics on the wireless charger, which is imperceptible to the smartphone user. In reality, a large number of third parties' wireless chargers deployed in public spaces further aggravate this threat.

We theoretically and experimentally demonstrate the feasibility of *EM-Surfing* attacks. Specifically, we first show the risk of privacy leakage by building a theoretical model based on the principles of electromagnetism. The model proves that the load change caused by smartphone activity is highly correlated to the voltage change on the power line of the wireless charger,

- *Jianwei Liu and Jinsong Han are with the Zhejiang University and Key Laboratory of Blockchain and Cyberspace Governance of Zhejiang Province, Hangzhou 310027, China. E-mail: liujianwei@stu.xjtu.edu.cn, hanjinsong@zju.edu.cn.*
- *Xiang Zou is with Xi'an Jiaotong University, Xi'an 710049, China. E-mail: xiang_zou@stu.xjtu.edu.cn.*
- *Leqi Zhao, Yusheng Tao, and Sideng Hu are with Zhejiang University, Hangzhou 310027, China. E-mail: {3190105730, 3190103050, husideng}@zju.edu.cn.*
- *Kui Ren is with the Zhejiang University, Key Laboratory of Blockchain and Cyberspace Governance of Zhejiang Province, Hangzhou 310027, China, and also is with the Alibaba-Zhejiang University Joint Research Institute of Frontier Technologies, Hangzhou 310058, China. E-mail: kuirentnse@163.com.*

Manuscript received 21 December 2021; revised 2 April 2022; accepted 3 May 2022. Date of publication 6 May 2022; date of current version 14 March 2024.
This work was supported in part by the National Key R&D Program of China under Grant 2021QY0703, in part by the National Natural Science Foundation of China under Grants U21A20462, 61872285, 52177199, 62032021, 61772236, and 61972348, in part by the Research Institute of Cyberspace Governance in Zhejiang University, Leading Innovative and Entrepreneur Team Introduction Program of Zhejiang under Grant 2018R01005, in part by Ant Group Funding under Grant Z51202000234, and in part by the Alibaba-Zhejiang University Joint Institute of Frontier Technologies.
This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by Ethics Committee of College of Biomedical Engineering&Instrument Science, Zhejiang University.
(Corresponding author: Leqi Zhao.)
Digital Object Identifier no. 10.1109/TDSC.2022.3173063

which means that the voltage dynamics on the wireless charger could leak user's privacy, e.g., keystroke. Then, we conduct a series of preliminary experiments. The results further confirm the feasibility of stealing private information by analyzing the voltage dynamics. We find that at least four classes of privacy would be leaked through *EM-Surfing*, including login passcode, keystroke, App information, and speech content.

To explore the hazardness of *EM-Surfing*, we propose a three-module privacy inference method to recover private information from voltage traces. In this method, we first design a template-based algorithm to identify the model of the target phone, which avoids laborious training data collection. Then, statistical features and frequency-domain features are extracted to form feature vectors with fixed number of features. We leverage machine learning and feature vectors to achieve privacy class determination under adverse conditions where the duration of voltage dynamics is variable. At last, we design a dedicated deep neural network to establish the relationship mapping voltage dynamics to privacy content, so as to achieve accurate privacy inference. It is noteworthy that our *EM-Surfing* method can be easily realized and stealthily deployed over off-the-shelf commodity wireless chargers.

As a quick response to such severe threat, we design an App named *SecCharging* to prevent smartphones from *EM-Surfing* attacks. The defensive capabilities of *SecCharging* are attributed to three mechanisms. For each privacy class, *SecCharging* generates pertinent random processes according to these mechanisms. The objective of these processes is to induce random load changes to pollute the voltage signals in the *EM-Surfing* side-channel, so as to damage the privacy information. An attacker cannot get desired privacy inference effectiveness when *SecCharging* is running on the target smartphone.

We conduct real-world attack experiments over four phones with three mainstream operation systems (Android, Harmony, and iOS). The results show that an attacker can achieve over 99%, 96%, 96%, and 97% accuracy for passcode, keystroke, App, and speech inference. The results of an cross-device experiment demonstrate the real threats of *EM-Surfing*, in which the attacker can launch effective attacks even when he does not have any prior knowledge about the victim's smartphone. Our defense experiment results show that *SecCharging* can greatly reduce the inference accuracy, and it has expected defensive capabilities.

In summary, we make the following contributions:

- We reveal a new side-channel attack in smartphone wireless charging, namely *EM-Surfing*. Our theoretical model shows that all the activities on the smartphone could be stealthily monitored via *EM-Surfing* side-channel.
- We propose a three-module privacy inference method. An attacker can acquire the private information of a target smartphone by analyzing the voltage dynamics related to smartphone activities. The results of our real-world attack experiments demonstrate the severe and real threats of *EM-Surfing*.
- We design a defensive App *SecCharging* to protect smartphones from *EM-Surfing* attacks. *SecCharging* can greatly reduce the privacy leakage risks by randomizing the voltage signals in the *EM-Surfing* side-channel. Our defense experiment results show that *SecCharging* provides effective countermeasures towards *EM-Surfing*. We expect that *SecCharging* would arouse more secure designs for wireless charging protocols and hardware.

The remainder of this paper is organized as follows. In Section 2, we introduce some related works and highlight the difference between our work and prior ones. Section 3 shows our threat model and the consequences caused by privacy leakage. In Section 4, we build a theoretical model for *EM-Surfing* side-channel and experimentally show the feasibility of *EM-Surfing* attack. In Section 5, we demonstrate how an attacker can infer private data from the collected voltage dynamics caused by phone activities. We conduct real-world attack experiments and show the high attack effectiveness of *EM-Surfing* in Section 6. Section 7 gives a defensive App, named *SecCharging*, aiming at mitigating the hazard of power side-channels on smartphone. In Section , we summarize this paper.

2 Related Work

In this section, we first briefly introduce some works that are dedicated to side-channel exploiting on smartphones. Then, we show some mitigation methods proposed towards smartphone side-channel attacks.

2.1 Side-Channel Attack

Smartphone is one of the most commonly carried devices that contain users' privacy in our daily lives. Unfortunately, even with secure authentication techniques, users' privacy could be stolen by adversaries via various side-channels [8], [9], [10], [11], [12], [13], [14], [15], [16]. Existing side-channels lie into two categories: in-device ones and out-device ones. For the former, adversaries obtain privacy-related data from the components (e.g., sensors) inside target smartphones. For example, Ba *et al.* [10] propose to eavesdrop speech played by loudspeaker by in-built accelerometers and gyroscopes. They design a deep neural network to recover speech content from sensor data, Although in-device side-channels enable accurate privacy inference, they have to collect sensitive data from the inside components of smartphones, which is often impractical.

By out-device side-channels, adversaries can get privacy-related information without direct accessibility to the target smartphone. Thus, they are more covert compared to in-device ones. For instance, Yang *et al.* [12] show that the webpage information a smartphone loaded may leak via the power line while charging even when no data communication is allowed. The reason behind is that smartphone consumes power from the charger rather than the battery during charging. By monitoring the power consumption on the power line, Cronin *et al.* [13] achieve passcode inference because tapping different locations on the touchscreen consume different power. However, in a wireless charging environment, the phone cannot directly use the power from the charger. Is there still a side-channel in wireless charging? La Cour *et al.* [17] give a 'Yes' answer. They find that when the phone battery is almost full (e.g., exceeds 80%), the power consumption of webpage loading will cause the current increase on the power line in order to maintain a stable battery voltage. The current dynamics can be manipulated as a side-channel. This is to blame the Qi charging standard [18]. They also show that there is no current variation when the battery state is low (e.g., 30%). Also, due to the openness of Qi standard, Wu *et al.* [19] eavesdrop the message transmitting between the smartphone and charger to identify notifications and Apps. Nevertheless, they have not thoroughly explored the potential side-channels in wireless charging. In this paper, through modeling the charging process, we find that the *EM-Surfing* is a *brand-new* side-channel in wireless charging. *EM-Surfing* is irrelevant to the design of Qi standard. There are privacy leakage risks during the entire charging process. Instead of using a current sensor to collect current trace to infer webpage in [17] or eavesdropping Qi message in [19], an *EM-Surfing* attacker can measure the voltage dynamics on the power line to infer login passcode, keystroke, App information, and speech content.

2.2 Side-Channel Mitigation

In view of the great security threats brought about by side-channel attacks, researchers have also proposed many defense methods accordingly [8], [9], [10], [11]. For example, to mitigate the threat caused by smartphone accelerometer measurement leakage, Owusu *et al.* [8] recommend users to pay attention to identifying suspicious applications. The key is to force all applications to declare their intention to access the motion sensors. Moreover, smartphones users can limit attackers' privacy inference ability by reducing the sampling rate of motion sensors. Xu *et al.* [9] believe that changing the password frequently and increasing the length of password can increase the difficulties of attacks. To protect smartphones from speech eavesdropping attacks achieved by motion sensor monitoring, Ba *et al.* [10] also propose to reduce the sampling rate to stop attackers' high-frequency sensor reading. In order to mitigate the effect of Charger-Surfing attack, Cronin *et al.* [13] suggest the smartphone vendor to randomly change the positions of virtual buttons on the touchscreen. Besides, they propose to use dynamic lock-screen wallpapers, so that the power caused by animations can pollute the data leaked through side-channel. Although the above-mentioned countermeasures seem to be promising, most of them have two shortcomings. That is, they either show weak defense ability or inconvenience users. For instance, reducing the sampling rate of motion sensors too much would impact benign applications, too little would not effectively prevent the attack. Unlike these solutions, we design *SecCharging*, a novel application that can effectively block the *EM-Surfing* side-channel, while not impacting the normal use of smartphones.

3 Threat Model and Consequence

In this section, we first present the threat model of *EM-Surfing* attack, and then show its consequences.

3.1 Threat Model

EM-Surfing attack aims to steal private information from smartphone through the wireless charging side-channel, i.e., the power line of the wireless charger. Taking a realistic attack scenario into consideration, we assume a victim charges his smartphone through a shared power bank [13], a public wireless charger [20] or a personal wireless charger connected to a public USB interface. Such a USB interface could be a charging station or a power bank [13]. Fig. 1a shows an example of a free wireless charger provided in a cafe. A hotel could provide free USB interface (as shown in Fig. 1b) to customers. These charging interfaces can provide power normally, and they are so common that a smartphone user would not be aware of the risk of privacy leakage.

(a) Wireless charger in cafe. (b) USB interface in hotel.

Fig. 1. Free wireless charger and USB interfaces in public environments.

Since the above-mentioned public wireless charger or USB interface is owned/controlled by a third party (could be malicious), rather than the smartphone user, the voltage trace of the wireless charger could be monitored by the third party behind the charging interface. Moreover, the victim would not be aware of such monitoring, owing to that the monitoring behavior does not impact the functionality of the charger. Meanwhile, current wireless charging protocol does not provide any security options while initiating the charging. Thus, an attacker can stealthily record the voltage trace of the charger and perform on/offline analysis for privacy mining.

Further, we assume that the attacker has no access to the victim's smartphone, i.e., the attacker has no prior knowledge about the target smartphone. This makes the attack more realistic. However, we assume that the attacker has the knowledge of the voltage profiles of popular smartphone models, so that the attacker can establish the relationship mapping voltage dynamics to privacy content to achieve *EM-SurfIng* attack.

3.2 Consequences

In above attack scenarios, we find that the voltage variation on the power line of wireless charger is related to user's operations on the smartphone. Thus, by analyzing the voltage trace, an attacker can infer these operations and their relevant private information, such as the access control passcode, keystroke, information of installed Apps, and multimedia information. The leakage of such private information could lead to severe consequences. 1) *Access control:* The revealing of access control passcode would allow the attacker to unlock the victim's smartphone, and steal more private data or reset other online passwords [13]. Even if the attacker has no accessibility to the victim's smartphone, the passcode still could imply other privacy. This is because users tend to reuse their passcodes around five times [21]. A passcode could be reused as the password of a credit card or payment system (e.g., Apple Pay) [13], [22]. 2) *Keystroke:* The inference of the keystroke enables an attacker to know the text the victim entered through the smartphone keyboard. This is dangerous as the typed text often contains sensitive information, such as the password of credit/debit card, phone number, and address [23]. 3) *Installed App:* The

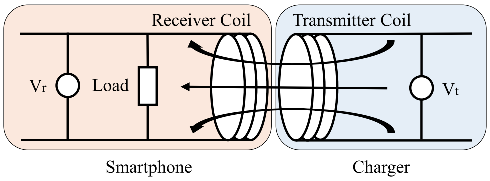


Fig. 2. Smartphone wireless charging process.

leakage of the App information gives an attacker at least two revelations. First, the attacker can speculate the victim's health condition, hobbies, habits, and location according to the App types [24]. Second, the App information could facilitate the identification of other kinds of privacy. For example, a sophisticated attacker can easily realize that the victim is likely to enter the payment password after opening Alipay [25]. 4) *Multimedia:* The multimedia information, e.g., the speech played by a smartphone, if eavesdropped by an attacker, can have potentially dangerous consequences, because the victim may reveal some sensitive information (e.g., credit card number, password, and address) when making a call [10].

4 Privacy Leakage Exploration

To investigate the effectiveness of *EM-Surfing* attacks, we theoretically model its side-channel and the privacy leakage process. We also conduct a series of experiments to demonstrate the attack feasibility.

4.1 Side-Channel Modeling

A wireless charging process is aimed to transfer power from a transmitting source to a receiving gadget. In smartphone charging, the wireless charger and the smartphone play the roles of the transmitting source and the receiving gadget, respectively. Basically, as shown in Fig. 2, the power transfer is achieved by two components: a transmitter coil in the wireless charger and a receiver coil in the smartphone.

We model the wireless charging process as follows. To ease our explanation, the direction of any variable is represented by the sign of the variable's value (positive or negative), which would not impact the conclusion we draw from our modeling. At the beginning of the power transfer, the mains voltage of the power source is changed into high-frequency alternating current (AC) flowing in the transmitter coil. According to Ohm's law [26], the AC can be formulated as

$$I_{tra}(t) = \frac{U_{tra}(t)}{R_{tra}}, \tag{1}$$

where $U_{tra}(t)$ and R_{tra} are the voltage and the resistance of the transmitter coil. Such AC can produce a time fluctuating magnetic field in the transmitter coil. According to the Biot-Savart law [26], the magnetic linkage introduced by $I_{tra}(t)$ can be formulated as

$$\Psi_{tra}(t) = L_{tra} I_{tra}(t), \tag{2}$$

where L_{tra} is the inductance of the transmitter coil. Further, the magnetic flux of the transmitter coil can be calculated by:

$$\Phi_{tra}(t) = \frac{\Psi_{tra}(t)}{N_{tra}} = \frac{L_{tra} I_{tra}(t)}{N_{tra}}, \tag{3}$$

where N_{tra} is the number of turns in the transmitter coil. Suppose that the magnetic flux harvested by the receiver coil is $\Phi_{rec}(t)$ and the harvest ratio is k_{rec}, then we have

$$\Phi_{rec}(t) = k_{rec}\Phi_{tra}(t). \tag{4}$$

According to the Faraday law of electromagnetic induction [26], the inductive voltage produced by the variation of $\Phi_{rec}(t)$ can be calculated by:

$$U_{rec}(t) = N_{rec}\frac{\Delta\Phi_{rec}(t)}{\Delta t}, \tag{5}$$

where N_{rec} is the number of turns in the receiver coil. Thus, we have

$$U_{rec}(t) = k_{rec} L_{tra}\frac{N_{rec}}{N_{tra}}\frac{\Delta I_{tra}(t)}{\Delta t}. \tag{6}$$

After substituting Eqs. (1) into (6), we have

$$U_{rec}(t) = \frac{N_{rec} k_{rec} L_{tra}}{N_{tra} R_{tra}}\frac{\Delta U_{tra}(t)}{\Delta t}. \tag{7}$$

It is worth noting that the activity on the smartphone will change the load on the receiver coil [27]. The change can be denoted as $R_e(t)$, a resistance change rate that vary with time. Further, the introduced extra current $I_e(t)$ can be represented by:

$$I_e(t) = \frac{U_{rec}(t)}{R_e(t)}. \tag{8}$$

The current $I_e(t)$ will produce an extra magnetic field, the magnetic flux of which is

$$\Phi_e(t) = \frac{L_{rec} I_e(t)}{N_{rec}}, \tag{9}$$

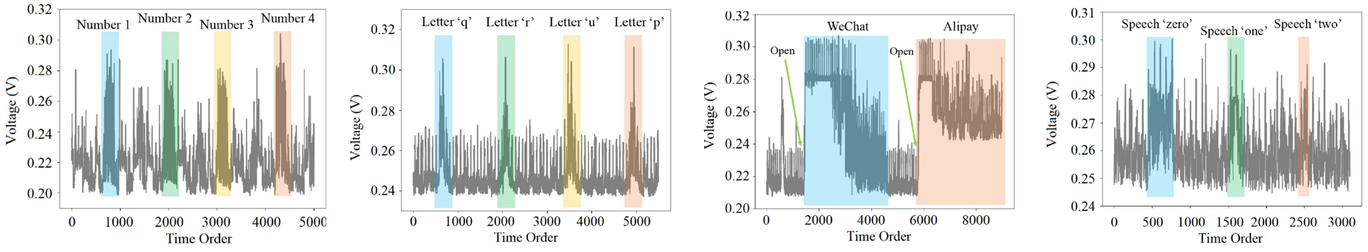
where L_{rec} is the inductance of the receiver coil. Due to that the electromagnetic induction effect is mutual for the two coils, if we denote the harvest ratio of the transmitter coil as k_{tra}, then an extra produced voltage on the transmitter coil can be formulated as

$$U_e(t) = k_{tra} L_{rec}\frac{N_{tra}}{N_{rec}}\frac{\Delta I_e(t)}{\Delta t}. \tag{10}$$

Thus, the voltage of the transmitter coil, i.e., the voltage an attacker can monitor, is

$$\begin{aligned} U'_{tra}(t) &= U_{tra}(t) + U_e(t) \\ &= U_{tra}(t) + k_{tra} L_{rec}\frac{N_{tra}}{N_{rec}}\frac{\Delta I_e(t)}{\Delta t} \\ &= U_{tra}(t) + \frac{L_{tra} L_{rec} k_{tra} k_{rec}}{R_{tra}}\frac{\Delta(\frac{1}{R_e(t)}\frac{\Delta U_{tra}(t)}{\Delta t})}{\Delta t}. \end{aligned} \tag{11}$$

In this formula, N_{tra}, N_{rec}, L_{tra}, L_{rec}, and R_{tra} are invariables. k_{tra} and k_{rec} are relatively stable during charging. Thus, the dynamics of the voltage of the wireless charger is correlated to $R_e(t)$. Due to the strong correlation between $R_e(t)$ and the smartphone activities, and different activities introduce different load [13], [28], [29], an attacker is capable to infer what a smartphone user entered or played on the smartphone by



(a) Passcode. (b) Keystroke. (c) App. (d) Speech.

Fig. 3. Voltage dynamics induced by passcode inputting, keystroke, App opening, and speech playing.

analyzing the voltage dynamics on the wireless charger. For example, entering unlock passcode will cause the load variation, and further the voltage variation. Meanwhile, touching different buttons will produce different animations on the touchscreen [13]. Since different animations consume different power (i.e., load), the voltage variations of different buttons are distinguishable. Attackers can leverage such distinguishability to infer the passcode.

4.2 Smartphone Activity Detection

In this work, we focus on four kinds of activities, i.e., passcode login, keystroke, App opening, and speech playing via speaker, which are highly possible to reveal users' private data. To explore the potential for identifying these activities, we carry out a series of preliminary experiments. We first charge a smartphone (Samsung Galaxy S8) with a wireless charger, and then observe the voltage dynamics on the power line. To obtain the voltage information, we connect a 0.33 Ohm resistor to the power line and use an analog-to-digital converter (ADC) to record the voltage variation on the resistor. The battery state of charging is between 20% and 30% during data collection.

■ *Passcode.* To validate that entering passcode indeed will cause voltage variation, we enter four numbers ‘1,’ ‘2,’ ‘3,’ and ‘4’ in the unlock interface. The time interval between any two continuous numbers is about two seconds. The voltage dynamics are shown in Fig. 3a. It can be observed that entering passcode would introduce large-scale and dense voltage variation, which can be easily distinguished from the noise beside. This phenomenon can be explained by Eq. (11). When user touches a button on the screen, the animation and CPU together produce a large load change $R_e(t)$. This makes the second term $f(R_e(t))$ of Eq. (11) larger, which further leads $U'_{tra}(t)$ to be larger. From the attacker's point of view, it is the sudden rise of voltage in Fig. 3a. Thus, an attacker can easily identify the passcode and further analyze the corresponding voltage dynamics to infer the passcode.

■ *Keystroke.* Similar to the passcode detection experiment, we enter letters via the virtual keyboard in the smartphone and record the voltage value. The time interval between any two adjacent hits is about two seconds as well. The experiment results shown in Fig. 3b demonstrate that keystroke also would introduce obvious voltage variation that is more drastic and has longer duration than surrounding noise. Although we only show the results of four letters ‘q,’ ‘r,’ ‘u’ and ‘p’. the voltage dynamics of other letters are as obvious as that of these four letters. Therefore, it is also potential for an attacker to detect the keystroke and recover the information entered through keyboard.

■ *App*. To explore the ability of *EM-Surfing* in identifying Apps loaded on smartphones, we open two payment Apps ‘Wechat’ and ‘Alipay’ and show the voltage information in Fig. 3c. It can be seen that the opening of Apps would greatly change the voltage. Meanwhile, the variation trends of different Apps are distinguishable. Hence, an attacker could be aware of the information of opened App by analyzing the voltage trace.

■ *Speech*. In this experiment, we play the speech containing different numbers via the smartphone's speaker. The voltage measurement is shown in Fig. 3d, in which one can find that the speech of ‘zero,’ ‘one,’ and ‘two’ would produce large and different voltage variations. Meanwhile, longer speech would cause longer drastic voltage change. Similarly, the voltage dynamics of other numbers (‘three’ to ‘nine’) are obvious. Thus, it is also possible to steal sensitive information from the voltage traces recorded during speech playing.

5 Private Data Inference

With the collected voltage dynamics, an attacker can infer the user's private information. To achieve this goal, we propose a three-module attack method, as shown in Fig. 4. By performing these three modules, i.e., *phone model identification*, *privacy class determination*, and *privacy content inference*, the attacker successively knows the victim smartphone's model, classes of privacy, and the content of privacy.

5.1 Phone Model Identification

To pave the way for inferring private information, identifying the model of the victim smartphone is the first step. This is because different phone models have diverse charging parameters (e.g., the parameters in Eq. (11)) or charging protocols. In this case, the voltage traces of different phone models are different even when there is no activity on the phone. Fig. 5 shows the voltage traces of three models of smartphones: Samsung Galaxy S8, HUAWEI P40 Pro, and iPhone XR. It can be easily observed that the voltage traces are significantly stable when the phone is locked or no activity is

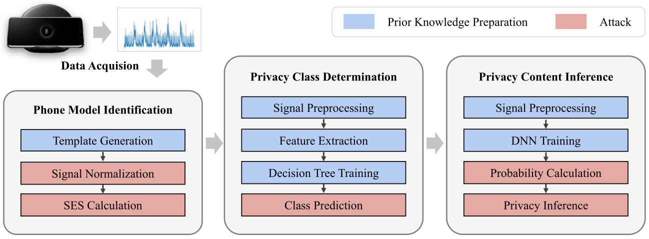


Fig. 4. Three-module privacy inference method.

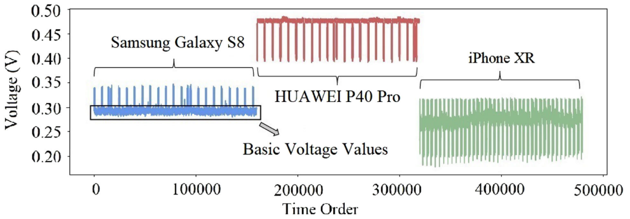


Fig. 5. Voltage dynamics of Samsung, HUAWEI, and iPhone when there are no activities on the phone.

conducted on the phone, while the patterns of voltage traces are distinguishable with different phone models.

Intuitively, we can use machine learning techniques to distinguish different phone models. However, in this way, the attacker needs to collect a large number of training samples for each model to train a classifier. A sophisticated attacker is more willing to invest the least effort to achieve the greatest benefits. Thus, we propose a similarity-based method to identify the phone model, in which only one sample is required to be collected as a template. Specifically, we first collect voltage values for T_{temp} seconds when the phone is locked. These voltage values form a signal sample. With a sampling rate of F, the signal sample contains $T_{temp} \times F$ voltage values. Since different battery states and different phone locations with respect to the charger may lead to different basic voltage values (shown in Fig. 5), we need to normalize the signal sample to benefit the following similarity calculation. We opt to use *min-max normalization* [30], which can be formulated as

$$v^i_{new} = \frac{v^i_{ori} - v_{min}}{v_{max} - v_{min}}, i \in [1, T_{temp} * F], \tag{12}$$

where $v^i_{new}, v^i_{ori}, v_{min}$, and v_{max} are the i_{th} normalized voltage value, the i_{th} raw voltage value, the minimum of the signal sample, and the maximum of the signal sample, respectively. The normalized signal sample is used as a template and the template only needs to be generated once.

Henceforth, when the model of a charging phone need to be identified, we only need to collect the voltage values for T_{simi} seconds, where $T_{simi} \leq T_{temp}$. The signal sample is normalized by Eq. (12) as well. Then, we leverage the *euclidean distance* [31] to measure the similarity between the template and the signal sample. The similarity is called sliding euclidean similarity (SES), which can be formulated as

$$SES = min \frac{1}{\sqrt{\sum_{i=1}^{N_s} \left(v^i_s - v^{i+j}_t\right)^2}}, j \in [1, N_t - N_s], \tag{13}$$

where v^i_s and v^i_t are the i_{th} value of the signal sample and the i_{th} value of the template, respectively. Meanwhile, $N_s = T_{simi} \times F$ and $N_t = T_{temp} \times F$. If the SES is larger than a predefined threshold, the signal sample and the template belong to the same phone model. The attacker can collect the templates of mainstream phone models as prior knowledge, so as to determine the models of target phones.

5.2 Privacy Class Determination

After identifying the model of the smartphone phone, *privacy class determination* module aims to determine which class of privacy the collected voltage dynamics leak, which will facilitate the following privacy content inference. In this paper, we call a segment of voltage trace that contains privacy as a signal sample. Thus, we need to determine the class of each signal sample. As aforementioned, we focus on four classes of private data: passcode, keystroke, App, and speech. As shown in Fig. 3, different classes of private data last different time lengths. With inconsistent time lengths, it is hard to establish a relationship mapping voltage dynamic to private data. Thus, we need to determine the class of each target signal sample before privacy content inference. More importantly, knowing the privacy class can help reduce our search scope for privacy content. For example, when we know that a signal sample belongs to the passcode class in advance, we only need to determine which number ('0' to '9') the signal sample represents, without considering a lot of unnecessary candidates, such as letters.

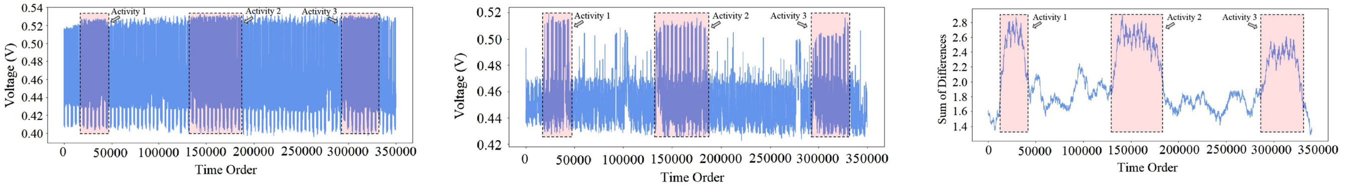
In order to achieve privacy class determination when the time lengths of signal samples are inconsistent, we extract statistical features and frequency-domain features to obtain feature vectors with fixed number of features. Afterwards, we can leverage a machine learning classifier to do determination. Specifically, the *privacy class determination* module contains three steps: signal preprocessing, feature extraction, and class prediction.

In the signal preprocessing, we perform filtering, activity detection, and segmentation. First, a low-pass filter with a cutoff frequency of 500 Hz (set empirically) is used to remove high-frequency noise components from raw signals, because the features of phone activities may be overwhelmed by noise. As shown in Figs. 6a and 6b, the filtering would uncover the voltage dynamics introduced by phone activities. Then, we leverage a sliding window to improve the distinguishability between the voltage traces of phone activities and that of surrounding noise. This will benefit the activity detection. In each window with N_w voltage values, we calculate the sum of differences between a current value and a prior value

$$S = \sum_{i-1}^{N_w - 1} |v^{i+1} - v^i|. \tag{14}$$

The voltage trace after filtering and the trace of sums are shown in Figs. 6b and 6c, respectively. It can be observed that the latter is more smooth. So, it becomes easier to find the start point of the activity. For a specific point, if its subsequent n points are continuously increased, the activity is considered to start from this specific point. The end point is determined by a similar method. We regard the voltage trace between a start point and an end point as a signal sample containing the voltage dynamics of a phone activity. As for the case that two voltage dynamics are very close to each other due to quick continuous tapping, we adopt the countermeasure introduced in [13]. Next, since in different battery states, the basic voltage values may be different, we subtract the minimum value of the sample from each value of the signal sample. In this way, the basic voltage values of each signal sample approximate zero.

After that, we extract a specific number of features to form a feature vector for each signal sample. The features include statistical ones and frequency-domain ones. For the former, we calculate the maximum, standard deviation, variance,



(a) Before filtering.

(b) After filtering.

(c) Sum of the differences in sliding windows.

Fig. 6. Voltage signal preprocessing method.

mean, entropy [32], and the length of the signal sample as features. For the latter, we calculate 13 Mel Frequency Cepstrum Coefficients (MFCCs) [33] as features. Finally, we obtain a feature vector with 19 elements, which will be used in the following classification.

To predict the privacy class of a signal sample, there are many machine learning classifiers worth considering. However, we noticed that the ranges of values of different features are very different. For example, the length of a signal sample could be 1000, while the mean of a signal sample is smaller than 1. Meanwhile, it is difficult to normalize these features, because the maximum of some features, e.g., the length of the signal sample, is agnostic. Thus, we opt to use decision tree (DT) classifier [34] due to its strong tolerance for diverse value ranges of features.

5.3 Privacy Inference

Within the scope narrowed by phone model and privacy class, we can infer the private content in a signal sample by deep learning technology. We opt to use deep neural network (DNN) because it has powerful feature extraction ability, which can help us get high inference accuracy.

In particular, we train a dedicated deep neural network for each class of private information. For example, to infer the passcode, the trained network tends to efficiently and accurately calculate the probability that an inputted signal sample belongs to each specific number. The number that has the largest probability is most likely to be the one the user entered.

Our efforts focus on the design of the architecture of the DNN. Specifically, we select one-dimensional convolutional layers [35] as the main components of our DNN. There are two reasons behind such selection. First, the voltage trace is time-series data, and hence suitable for one-dimensional convolutional operation. Second, one-dimensional kernel is capable of extracting fine-grained local features. Meanwhile, the convolutional operation along the voltage trace is helpful to extract global temporal features. However, the kernel size and sliding stride of the kernel should be cautiously considered, as large kernel size would make the extracted feature too coarse while small sliding stride might lead to massive computational overhead. Our empirical studies indicate that the proper kernel size and sliding stride are 8 and 4, respectively. As shown in Fig. 7, we totally use three convolutional layers, each of which is followed by a *batch normalization* (*BN*) function [36] and a *rectified linear unit* (*ReLU*) [37]. The *BN* and *ReLU* work together to prevent our network from distribution offset and overfitting. After the last convolutional layer, we add two fully-connected layers to map extracted features to probabilities. A *Sigmoid* function [38] is added between them to increase the nonlinearity of our DNN.

Furthermore, we utilize a keyboard-split method and a corpus to improve the keystroke inference accuracy. The challenge behind is that the area of a key (letter) on the virtual keyboard is very small, resulting in low distinguishability between two adjacent keys. Thus, if we directly distinguish 26 letters, the accuracy would be relatively low. To tackle this problem, we first split the standard virtual keyboard into five groups, as shown in Fig. 8. The splitting is based on two principles: i) Two adjacent keys are more likely to be grouped together. ii) Two keys that are less likely to appear in the same word are more likely to be grouped together, i.e., two keys that are more likely to be contained by the same word are more likely to be assigned to different groups. In this way, we can regard each group as a specific class towards the DNN, and it is more likely to construct a meaningful word. The probability that two letters appear in the word is calculated according to a Top3000 English corpus [39]. For instance, the probability for ‘A’ and ‘B’ is the ratio the number of words containing both ‘A’ and ‘B’ to the number of all words (i.e., 3000). Accordingly, the DNN is trained to map a voltage dynamic to the probability that the input belongs to each group. After that, to infer a word containing N_l signal samples (each signal sample is associated to a letter), we perform two steps: i) We use the trained neural network to predict which group each signal sample belongs to. ii) We construct all possible N_l-letter words using the letters in these groups.

6 Attack Effectiveness Evaluation

In this section, we introduce our real-world implementation and detail the effectiveness of *EM-Surfing* attacks.

■ *Experiment Setup.* As shown in Fig. 9. we cut the ground wire of the charging cable and insert a 0.33 Ohm resistor. The data collected by the *NI USB-4431* ADC [40] is the voltage across the resistor. Our default sampling frequency is 5KHz. Since the charger in our threat model can be controlled by the attacker, we only use one wireless charger (*HUAWEI SuperCharge* [41]) connected to a *HUAWEI SuperCharge* plug to conduct experiments. We use LabVIEW [42] to collect voltage measurements on an *Alienware m15* laptop with Intel(R) Core(TM) i7-9750H CPU. In addition, We invite two males and one female to collect voltage

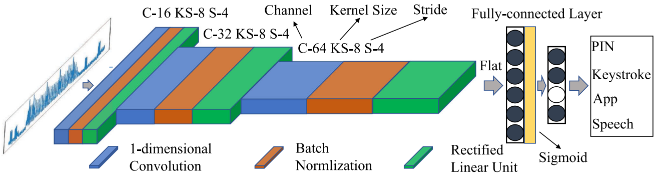


Fig. 7. Architecture of the privacy inference network.

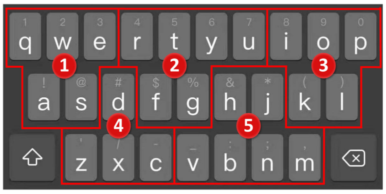


Fig. 8. Letters in the keyboard are divided into five groups.

traces over four smartphones, including two Samsung Galaxy S8, a HUAWEI P40 Pro, and an iPhone XR. We select these phone brands because: 1) Samsung, iPhone, and HUAWEI are highest-selling and wireless charging-compatible phone brands over the world [43], 2) they cover three mainstream smartphone operation systems (Samsung Galaxy S8-Android, iPhone XR-iOS, HUAWEI P40 Pro-Harmony).

■ *Data Collection.* For phone model identification, we collect a batch of signal samples when these phones are locked. For the login passcode, we first collect 50 samples for each number as training data, and then ask each volunteer to enter at least 50 4-digit PIN codes per phone. To evaluate the keystroke inference accuracy, we first collect 50 samples for each letter as training data, and then randomly select 50 words (from the top3000 English corpus) for each volunteer per phone to enter through virtual keyboard, As for the App information, we let each volunteer open each one out of ten Apps (five most downloaded Apps [44] and five mainstream payment Apps including 'Alipay,' 'WeChat,' 'Google Pay,' 'Google Wallet,' 'Samsung Pay'/ 'HUAWEI Pay') on each phone for at least 50 times. At last, we play the speech containing English numbers from zero to nine on each phone for at least 50 times. We totally collect over 30000 signal samples. All the experiments are conducted by adhering to the approval of our university's Institutional Review Board (IRB).

■ *Metrics.* We first define accuracy for phone model identification and privacy class determination. The accuracy in the former/latter is the probability that a phone model/ privacy class of a signal sample is correctly identified. We also define passcode/keystroke/App/speech inference accuracy (PIA/KIA/AIA/SIA) to quantify the attack effectiveness. PIA is the probability that at least one number is correctly inferred from a 4-digit PIN code. KIA is the probability that the correct word is contained in the 200 most likely words we inferred. AIA/SIA is the probability that an App/number-in-speech is correctly predicted.

6.1 Overall Effectiveness

We first calculate the accuracy of phone model identification and privacy class determination. For the former, we compare our method with the deep learning method ('Cha-Sur.') proposed in [13]. For the latter, we use 75% signal samples for classifier training and the rest 25% for testing. We also compare DT with two baselines, i.e., two classic classifiers: support vector machine (SVM) [45] and K-nearest neighbours (KNN) [46]. The results are shown in Figs. 10a and 10b, respectively. It can be observed that our method can respectively achieve 100%, 98.5%, and 100% accuracy in Samsung, HUAWEI, and iPhone identification, which are comparable to the performance of 'Cha.-Sur.'. However, 'Cha.-Sur.' needs to collect about 300 training signal samples for each phone model. Thus, our method is more 'lightweight'. For the privacy class determination, DT can achieve 100% accuracy for all classes, while SVM gets 0% accuracy in PIN identification and KNN only gets 42% accuracy in speech identification. Therefore, our privacy class determination method outperforms the baselines.

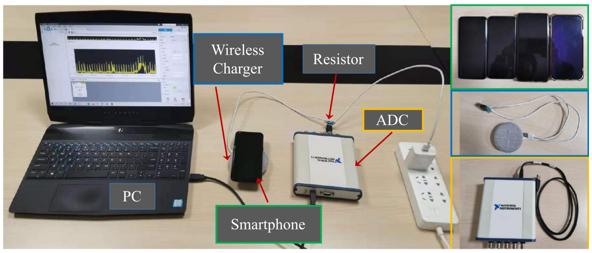
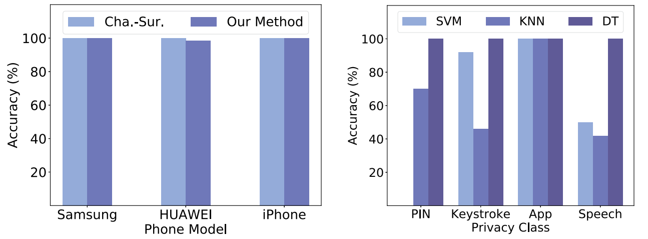


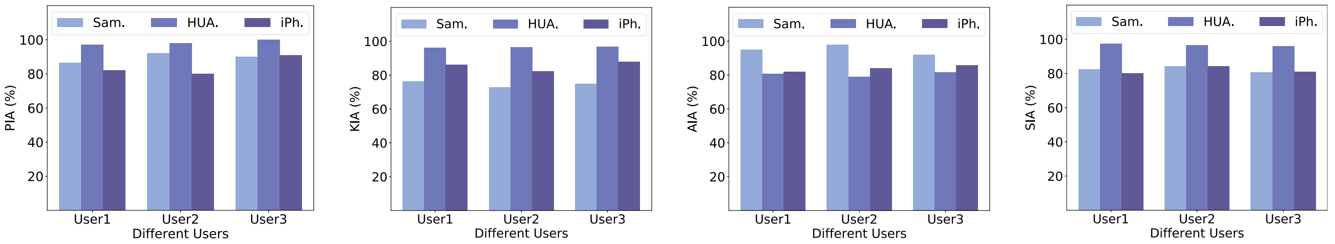
Fig. 9. Experiment setup.



(a) Phone model identification. (b) Privacy class determination.

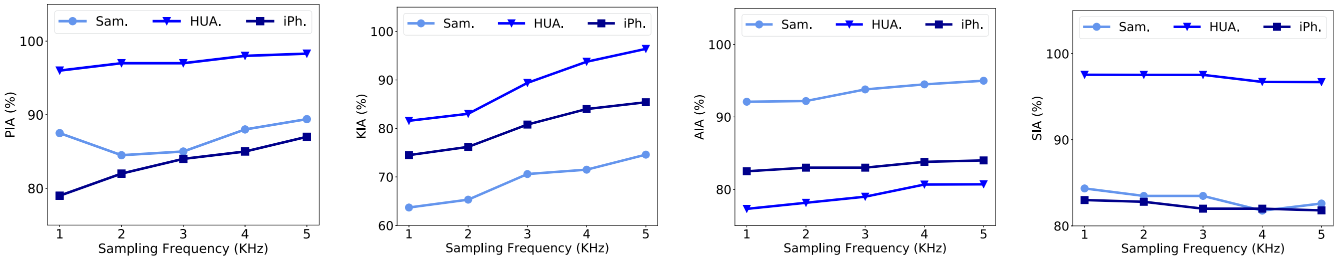
Fig. 10. Accuracy of phone model identification and privacy class determination.

Then, we calculate the PIA, KIA, AIA, and SIA. For the AIA and SIA, we use 75% signal samples as the training data and 25% ones as the testing data. The results of three volunteers are shown in Figs. 11a, 11b, 11c, and 11d, respectively. It can be found that the PIA, KIA, AIA, and SIA are significantly high over all three phone models. The average PIA of Samsung, HUAWEI, and iPhone is 89.4%, 98.3%, and 87.0%, respectively. Such high PIA demonstrates the severe security threat of *EM-Surfing* to users' login passcode. The average KIA of Samsung, HUAWEI, and iPhone is 74.6%, 96.4%, and 85.4%, respectively. Such results mean that an attacker could recover users' secret information in text inputting through analyzing the voltage dynamics. As for the App information, the AIA of Samsung, HUAWEI, and iPhone is 95.0%, 80.7%, and 84.0%, respectively. The high AIA means that *EM-Surfing* poses severe threats to users' App privacy. In the speech inference experiment, an attacker can achieve 82.6%, 96.7%, and 81.8% average SIA for Samsung, HUAWEI, and iPhone, respectively. Thus, the private speech content could be eavesdropped by an attacker through the *EM-Surfing* side-channel. Meanwhile, we find that HUAWEI phone (Harmony OS) is more vulnerable to *EM-Surfing* attacks, because the average PIA, KIA, and SIA of HUAWEI are higher than that of Samsung (Android OS) and iPhone (iOS). Besides, to show the superiority of our privacy inference network, we compare its PIA on Android with four other learning models including naive Bayes (NB) classifier, SVM, KNN, and DT. As a result, the PIA of NB, SVM, KNN, and DT is 70.9%, 81.0%, 71.45%, and 62.35%, respectively. They are all high, indicating that *EM-Surfing* is a severe threat. However, our privacy inference network outperforms the four learning models.



(a) Passcode. (b) Keystroke. (c) App. (d) Speech.

Fig. 11. Passcode/keystroke/App/speech inference accuracy.



(a) Passcode. (b) Keystroke. (c) App. (d) Speech.

Fig. 12. Passcode/keystroke/App/speech inference accuracy under different sampling frequency.

6.2 Impact of Sampling Frequency

Recalling that our default sampling frequency is 5KHz, to explore the impact of the sampling frequency, we vary the sampling frequency from 1KHz to 5KHz and calculate the inference accuracy. The PIA, KIA, AIA, and SIA are shown in Figs. 12a, 12b, 12c, and 12d, respectively. It can be found that the PIA of the three phone models increases with the increase of the sampling frequency. But the PIA does not increase a lot. Thus, an attacker can achieve a high PIA with only 1KHz sampling frequency. In Fig. 12b, we find that the KIA also increases as the sampling frequency increases. When the sampling frequency is only 1KHz, the KIA is also high. The curves of the AIA in Fig. 12c are relatively flat. The sampling frequency does not impact the AIA too much when it is in the range of $[1, 5] KHz$. The curves in Fig. 12d show that SIA decreases with the increase of the sampling frequency, which is different from the variation patterns of PIA, KIA, and AIA. A possible reason is that there is too much noise in the speech-related voltage traces, higher sampling frequency introduces more noise, resulting in lower SIA.

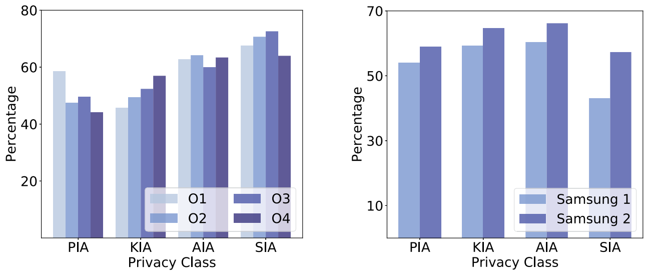
6.3 Cross-Factor Testing

In this part, we consider three crucial factors that are related to the effectiveness of real-world attack: orientation, device (phone), and user.

■ *Cross-Orientation Effectiveness.* In practice, the orientation of the phone with respect to the charger, i.e., the angle between the phone and the transmitter coil is uncertain during charging. To explore the impacts of such angle, we collect voltage signals under four angles (orientations): 0, 90°, 180°, and 270° (shown in Fig. 16). We train with the data collected under three angles and test with that collected under the rest one angle. The results are shown in Fig. 13a. It can be observed that the change of the orientation indeed impacts the effectiveness of *EM-Surfing* attacks. The highest PIA, KIA, AIA, and SIA are 58.5%, 57.0%, 64.2%, and 72.6%, respectively. Although these inference accuracy decreases when the angle varies, they can exceed 56%, which still poses severe threats to users' privacy on smartphones.

■ *Cross-Device Effectiveness.* In order to further demonstrate the real threats posed by *EM-Surfing*, we carry out a cross-device experiment, in which we train with the signal samples of a Samsung Galaxy S8 phone and test with that of another phone with the same model. The attack results are shown in Fig. 13b. The PIA, KIA, AIA, and SIA of Samsung 1 are 54.0%, 59.2%, 60.4%, and 43.1%, respectively. The PIA, KIA, AIA, and SIA of Samsung 2 are 59.0%, 64.6%, 66.2%, and 57.3%, respectively. It can be observed that the inference accuracy of Samsung 1 is similar to that of Samsung 2. Meanwhile, most inference accuracy exceeds 55%. Thus, *EM-Surfing* poses severe real threats to smartphone users, because the attacker can use his own phones to collect training data while launching effective attacks towards the victim's phone, even though the attacker does not have any prior knowledge about the victim's phone.

■ *Cross-User Effectiveness.* Since the use habit of phone vary from person to person, and the attacker and victim in practical *EM-Surfing* attack are different, we perform cross-user evaluation in this experiment. Specifically, the training data and testing data are from different users. We use the data collected by User 1 as training set and conduct testing



(a) Cross-orientation attack effectiveness. (b) Cross-device attack effectiveness.

Fig. 13. Effectiveness of cross-orientation and cross-device attacks.

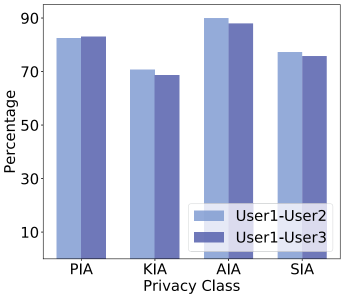


Fig. 14. Cross-user effectiveness.

with the data collected by User 2 and 3. The experiment results are shown in Fig. 14. It can be found that even the training data and testing data are from different users, the inference accuracy is still high.

6.4 Effect of Charging Distance

In our default setting, we use *HUAWEI* wireless charger to conduct experiments. The smartphone is placed on the charger. Due to the limitation of its charging efficiency, the distance between the smartphone and the charger is small (almost contact). However, with the development of charging technologies, the charging range will become very large, e.g., 5 meters [47]. To explore if *EM-Surfing* still exists when the charging distance increases, we use ADC function of ESP32 [48] to collect voltages measurements on *WENERY* wireless charger [49]. This charger supports 30 mm charging. The voltage dynamics of opening WhatsApp under 10 mm, 15 mm, 20 mm, and 25 mm are shown in Figs. 17a, 17b, 17c, and 17d, respectively. It can be found that the voltage traces are relatively flat when there is no activity on the smartphone. When we open WhatsApp, the voltages under all distances increase in large scale. This demonstrates that *EM-Surfing* remains when the charging distance increases.

6.5 Comparison With Existing Works

In this part, we compare *EM-Surfing* with two wireless charging side-channels: La Cour *et al.* [17] and Wu *et al.* [19]. Table 1 lists the differences between *EM-Surfing* and the other two works, First, the principle of *EM-Surfing* is completely different from that of [17] and [19]. The side-channels of [17] and [19] derive from the imperfection of Qi standard, but *EM-Surfing* is caused by the essence of wireless charging, i.e., electromagnetic induction effect. Thus, *EM-Surfing* is more obstinate. Second, the signals measured by [17] and [19] are current of charger and amplitude of magnetic field respectively, while that of *EM-Surfing* is voltage of charger. Third, [17] only explores the feasibility of three types of attacks. [19] confirms that four types of privacy can be eavesdropped. In this work, we show that six types of privacy could be leaked by *EM-Surfing*. Forth, [17] claims that privacy leakage could occur only when the battery is almost full, while the threats of [19] and *EM-Surfing* exist in the whole charging process. Fifth, the stealthiness of [17] and *EM-Surfing* is high because they only need to monitor the power consumption through the USB interface. But [19] needs to place an eavesdropping coil as well as necessary processing units near the wireless charger. Last but not least, we build a theoretic model to explain the scientific principle behind *EM-Surfing*, while [17] and [19] give intuitive explanations.

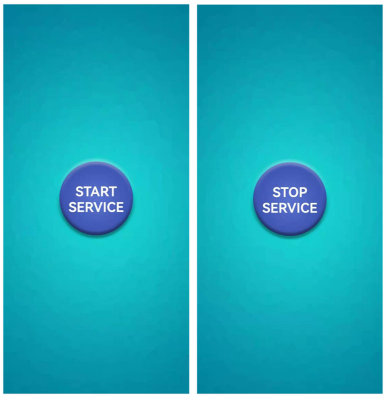


Fig. 15. Interface of *SecCharging*.

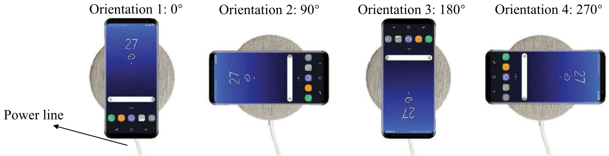
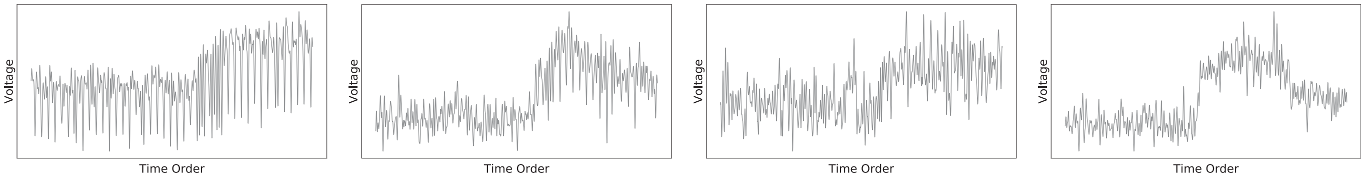


Fig. 16. Setup for cross-orientation experiments.

7 Defense

Our work shows that an attacker can infer user's private data by analyzing the voltage dynamics on the wireless charger. In order to mitigate the security threats caused by dynamic voltage leakage, prior studies [13], [17], [50] have proposed some countermeasures. These defensive methods can be divided into two categories: hardware-based and software-based. However, previous approaches have several limitations that hinder their deployment in practice. To be specific, hardware-based methods, e.g., adding a low-pass filter on the charger, would bring the overhead of hardware modification or the cost of extra hardware. Existing software-based methods either protect limited privacy classes or introduce large additional overhead. For instance, keyboard randomization is only designed for keystroke protection. Code injection requires modifying the source code for each App that needs to be protected. Performing random power consumption during the whole charging process leads to significant increase of time required to charge. In this section, taking into account the user-friendliness and the security simultaneously, we design a trigger-based defense App for power side-channels (including wired charging [13] and wireless charging), named *SecCharging*.[1] *SecCharging* monitors the user's activities on the smartphone, and generates random loads to pollute the side-channel voltage signals only when it discovers that there is a risk of privacy leakage, so that the privacy information in the voltage dynamics is damaged. Meanwhile, the introduced randomness makes it difficult for the attacker to restore the original signal. *SecCharging* only needs to be activated when the user is charging, and it does not interfere with the normal use of the smartphone, significantly reducing the risk of privacy leakage caused by passcode

[1] An Android version of *SecCharging* is available at https://github.com/403forbidden0/SecCharging.



(a) 10mm. (b) 15mm. (c) 20mm. (d) 25mm.

Fig. 17. Voltage dynamics under different charging distances.

TABLE 1
Comparison With Existing Works

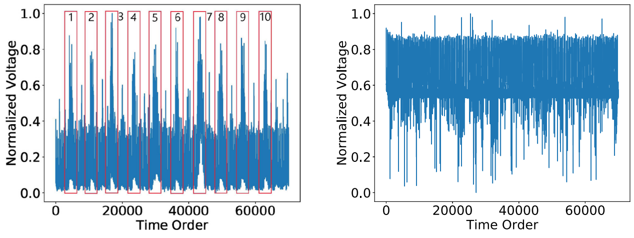
Work	Principle	Signal	Threat	Persistence	Stealthiness	Theoretical model
Cour *et al.* [17]	Charging status maintenance in Qi standard	Current of wireless charger	Webpage, PIN length, screen power consumption	Only high battery state (>80%)	High	No
Wu *et al.* [19]	Power demand embedded in Qi message	Amplitude of magnetic field	Device identifier, screen on/off, notification, App	Whole charging process	Medium	No
Ours	Electromagnetic induction effect	Voltage of wireless charger	Device identification, privacy class, PIN, keystroke, App, speech	Whole charging process	High	Yes

inputting, keystroke, App opening, and speech playing. Its interaction interfaces are shown in Fig. 15.

■ *SecCharging Design*. The defensive capabilities of *SecCharging* comes from the following mechanisms:

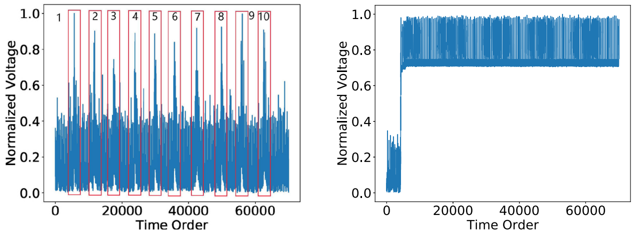
- *Passcode*. *SecCharging* continues to detect if the user attempts to unlock the phone. If so, it generates simple processes to pollute the voltage signal until the phone is found unlocked.
- *Keystroke and App*. *SecCharging* keeps on monitoring if a tapping happens. If the user taps the touchscreen, *SecCharging* would randomly generate n threads to change the load deliberately.
- *Speech*. Playing the voice will trigger *SecCharging* to generate random processes continuously until the user stops playing, greatly disrupting the voltage dynamics caused by the voice playing.

■ *Defense Performance*. To evaluate the performance of *SecCharging*, we install it on a Samsung Galaxy S8 phone and collect voltage dynamics with it running backstage. Figs. 18a and 18b respectively show two voltage traces collected when we enter PIN ‘1’ without *SecCharging* running and with *SecCharging* running. Figs. 19, 20, and 21have similar meanings for keystroke, APP, and speech. Apparently, *SecCharging* flattens the voltage dynamics, so that the privacy information is effectively erased. Moreover, in each of Figs. 18a, 19a, and 21a, there are ten times of drastic voltage



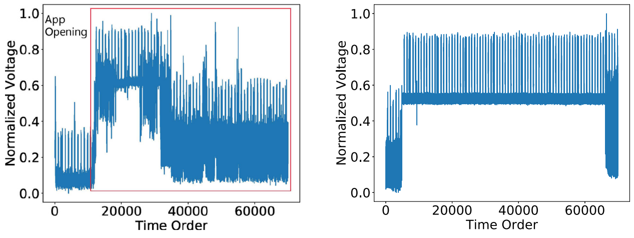
(a) Before running *SecCharging*. (b) After running *SecCharging*.

Fig. 18. Voltage signal profile of entering PIN ‘1’ for ten times before and after running *SecCharging*.



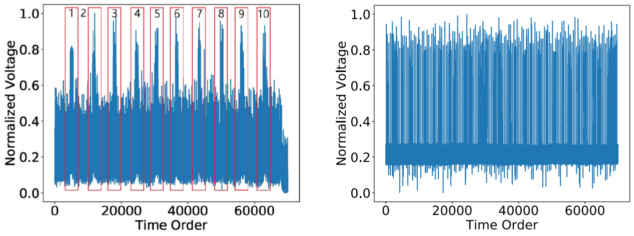
(a) Before running *SecCharging*. (b) After running *SecCharging*.

Fig. 19. Voltage signal profile of entering letter ‘a’ for ten times before and after running *SecCharging*.



(a) Before running *SecCharging*. (b) After running *SecCharging*.

Fig. 20. Voltage signal profile of App opening before and after running *SecCharging*.



(a) Before running *SecCharging*. (b) After running *SecCharging*.

Fig. 21. Voltage signal profile of playing audio ‘zero’ for ten times before and after running *SecCharging*.

TABLE 2
Defense Performance of *SecCharging*

Privacy Class	PIA	KIA	AIA	SIA
SecCharging	35.5%	29.4%	11.7%	9.6%
Decrease	53.9%	45.2%	83.3%	73.0%

TABLE 3
Overhead of *SecCharging*

Privacy Class	Passcode	Keystroke	App	Speech
Normal-Sam.	8.1 min	10.3 min	9.8 min	11.5 min
SecCharging-**Sam.**	8.8 min	12.3 min	12.7 min	12.5min
Normal-HUA.	15.7 min	9.1 min	12.2 min	19.5min
SecCharging-**HUA.**	16.3 min	10.2 min	14.0 min	20.1min

variations, which are consistent with our ten times of operations. In each of Figs. 18b, 19b, and 21b, there are also some drastic voltage variations. But they are caused by the Qi charging standard because the number of these variations is much larger than that of our operations. Then, we train our DNNs with 75% voltage samples and test with the rest 25% ones. The PIA, KIA, AIA, and SIA under protection are shown in the second row of Table 2, and the accuracy decrease caused by *SecCharging* is shown in the third row of the Table 2. It can be found that after using *SecCharging*, an attacker can only get very low PIA, KIA, AIA, and SIA. *SecCharging* helps smartphone users decrease the inference accuracy a lot. Thus, *SecCharging* has outstanding defensive capabilities and can prevent smartphones from *EM-Surfing* attacks effectively.

■ *Overhead Evaluation.* Besides the power consumed by normal operations of user, *SecCharging* will consume extra power when generating random process. Thus, it is necessary to evaluate the power consumption of *SecCharging*. To intuitively show the results, we estimate the time used for charging the battery for a certain percentage, i.e., 5%. Specifically, we separately estimate the time used for charging the battery from 45% to 50% when using or not using *SecCharging*. For testing towards unlock passcode, keystroke, App, and speech, we unlock for 50 times, enter word 'address' for 50 times, open 'Alipay' for 50 times, and play audio samples for 50 times, respectively. The time consumption of Samsung Galaxy S8 (Sam.) and HUAWEI P30 Pro (HUA.) is displayed in Table 3. It can be seen that using *SecCharging* indeed will induce the increase of charging time. However, since *SecCharging* only generates random process when there is a risk of privacy leakage, its power consumption, i.e., the increase of charging time, is acceptable. A little sacrifice of charging time can be exchanged for a high level of security, which is often worth it.

Until *SecCharging* is widely installed, it is essential for smartphone users to be aware of the threats posed by *EM-Surfing*. Besides, we focus on four privacy classes in this paper, yet there are still many other privacy classes, e.g., videos, that have not been explored. *SecCharging* may not work in protecting them. Therefore, *EM-Surfing* remains an open issue worthy of attention.

8 Conclusion

In this paper, we reveal a new side-channel attack, namely *EM-Surfing*, in smartphone wireless charging. We first build a theoretical model to show that users' activities on smartphones can be recorded by the voltage dynamics on the wireless charger. Then, we propose a three-module method to infer the privacy content. To help users prevent their privacy from *EM-Surfing*, we design an App named *SecCharging*. Our real-world experiment results show that an attacker can accurately infer the privacy of the target smartphone through *EM-Surfing*. The results of our defense experiments demonstrate the high defensive capabilities of *SecCharging*.

References

[1] T. FAQ, "Uses and benefits of wireless charging in smart phones," 2021. [Online]. Available: https://technofaq.org/posts/2016/09/uses-and-benefits-of-wireless-charging-in-smart-phones/#:~:text=Benefits%20of%20wireless%20charging.%20Benefit%20of%20wireless%20charging,can%20use%20wireless%20charging%20device%20safely%20and%20harmlessly

[2] T. N. Y. times, "A common charger for all phones? The E.U. is on the case," 2021. [Online]. Available: https://www.nytimes.com/2020/01/17/business/european-union-phone-charger.html

[3] J. R. Reitza and F. J. Milford, *Foundations of Electromagnetic Theory*. Noida, UP, India: Pearson Education India, 1967.

[4] Tutorialspoint, "What is the principle behind wireless charging?," 2021. [Online]. Available: https://www.tutorialspoint.com/what-is-the-principle-behind-wireless-charging#:~:text=The%20principle%20of%20Wireless%20Charging%201%20Mains%20voltage,loop%20of%20the%20gadget.%20...%20More%20items...%20

[5] W. Wang, C. Zhang, J. Wang, and X. Tang, "Multipurpose flexible positioning device based on electromagnetic balance for EVs wireless charging," *IEEE Trans. Ind. Electron.*, vol. 68, no. 10, pp. 10 229–10 239, Oct. 2021.

[6] W. Xu, W. Liang, J. Peng, Y. Liu, and Y. Wang, "Maximizing charging satisfaction of smartphone users via wireless energy transfer," *IEEE Trans. Mobile Comput.*, vol. 16, no. 4, pp. 990–1004, Apr. 2017.

[7] W. Xu, W. Liang, S. Hu, X. Lin, and J. Peng, "Charging your smartphones on public commuters via wireless energy transfer," in *Proc. Perform. Comput. Commun. Conf.*, 2015, pp. 1–8.

[8] E. Owusu, J. Han, S. Das, A. Perrig, and J. Zhang, "Accessory: Password inference using accelerometers on smartphones," in *Proc. Workshop Mobile Comput. Syst. Appl.*, 2012, Art. no. 9.

[9] Z. Xu, K. Bai, and S. Zhu, "TapLogger: Inferring user inputs on smartphone touchscreens using on-board motion sensors," in *Proc. Conf. Secur. Privacy Wireless Mobile Netw.*, 2012, pp. 113–124.

[10] Z. Ba *et al.*, "Learning-based practical smartphone eavesdropping with built-in accelerometer," in *Proc. Netw. Distrib. Syst. Secur. Symp.*, 2020.

[11] M. Zhou *et al.*, "PatternListener: Cracking android pattern lock using acoustic signals," in *Proc. Conf. Comput. Commun. Secur.*, 2018, pp. 1775–1787.

[12] Q. Yang, P. Gasti, G. Zhou, A. Farajidavar, and K. S. Balagani, "On inferring browsing activity on smartphones via USB power analysis side-channel," *IEEE Trans. Inf. Forensics Secur.*, vol. 12, no. 5, pp. 1056–1066, May 2017.

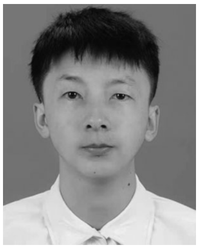
[13] P. Cronin, X. Gao, C. Yang, and H. Wang, "Charger-surfing: Exploiting a power line side-channel for smartphone information leakage," in *Proc. USENIX Secur. Symp.*, 2021, pp. 681–698.

[14] R. Spolaor, L. Abudahi, V. Moonsamy, M. Conti, and R. Poovendran, "No free charge theorem: A covert channel via USB charging cable on mobile devices," in *Proc. 15th Int. Conf. Appl. Cryptogr. Netw. Secur.*, 2017, pp. 83–102.

[15] Y. Michalevsky, A. Schulman, G. A. Veerapandian, D. Boneh, and G. Nakibly, "PowerSpy: Location tracking using mobile device power analysis," in *Proc. USENIX Secur. Symp.*, 2015, pp. 785–800.

[16] E. Miluzzo, A. Varshavsky, S. Balakrishnan, and R. R. Choudhury, "Tapprints: Your finger taps have fingerprints," in *Proc. Conf. Mobile Syst. Appl. Serv.*, 2012, pp. 323–336.

[17] A. S. La Cour, K. K. Afridi, and G. E. Suh, "Wireless charging power side-channel attacks," in *Proc. Conf. Comput. Commun. Secur.*, 2021, pp. 651–665.
[18] Qi enabled phones & compatible devices, 2021. [Online]. Available: https://qi-wireless-charging.net/qi-enabled-phones/
[19] Y. Wu, Z. Li, N. V. Nostrand, and J. Liu, "Time to rethink the design of Qi standard? Security and privacy vulnerability analysis of Qi wireless charging," in *Proc. Comput. Secur. Appl. Conf.*, 2021, pp. 916–929.
[20] S. Anand, "Cafe coffee day introduces wireless charging in more than 100 cafes in india," 2021. [Online]. Available: https://www.hungryforever.com/cafe-coffee-day-introduces-wireless-charging-in-more-than-100-cafes-in-india/
[21] D. A. F. Florêncio and C. Herley, "A large-scale study of web password habits," in *Proc. Int. Conf. World Wide Web*, 2007, pp. 657–666.
[22] Apple, "Apple pay," 2021. [Online]. Available: https://www.apple.com/apple-pay/
[23] M. Sabra, A. Maiti, and M. Jadliwala, "Zoom on the keystrokes: Exploiting video calls for keystroke inference attacks," in *Proc. Netw. Distrib. Syst. Secur. Symp.*, 2007.
[24] Y. Chen, X. Jin, J. Sun, R. Zhang, and Y. Zhang, "POWERFUL: Mobile app fingerprinting via power analysis," in *Proc. Conf. Comput. Commun.*, 2017, pp. 1–9.
[25] ALIPAY, "Alipay," 2021. [Online]. Available: https://www.alipay.com/
[26] D. Halliday, R. Resnick, and J. Walker, *Fundamentals of Physics*. Hoboken, NJ, USA: Wiley, 2002.
[27] Qi (standard), 2021. [Online]. Available: https://en.wikipedia.org/wiki/Qi_(standard)
[28] A. Pathak, Y. C. Hu, and M. Zhang, "Where is the energy spent inside my app?: Fine grained energy accounting on smartphones with Eprof," in *Proc. Eur. Conf. Comput. Syst.*, 2012, pp. 29–42.
[29] A. Pathak, Y. C. Hu, M. Zhang, P. Bahl, and Y. Wang, "Fine-grained power modeling for smartphones using system call tracing," in *Proc. Eur. Conf. Comput. Syst.*, 2011, pp. 153–168.
[30] Tutorials, "Min max normalization in data mining," 2018. [Online]. Available: https://t4tutorials.com/min-max-normalization-of-data-in-data-mining/
[31] ScienceDirect, "Euclidean distance," 2018. [Online]. Available: https://www.sciencedirect.com/topics/mathematics/euclidean-distance
[32] Y. Wang, K. Wu, and L. M. Ni, "WiFall: Device-free fall detection by wireless networks," *IEEE Trans. Mobile Comput.*, vol. 16, no. 2, pp. 581–594, Feb. 2017.
[33] A. Das, M. R. Jena, and K. K. Barik, "Mel-frequency cepstral coefficient (MFCC) - A novel method for speaker recognition," *Digit. Technol.*, vol. 1, no. 1, pp. 1–3, 2014.
[34] T. Pranckevicius and V. Marcinkevicius, "Comparison of naive bayes, random forest, decision tree, support vector machines, and logistic regression classifiers for text reviews classification," *Baltic J. Modern Comput.*, vol. 5, no. 2, 2017, Art. no. 221.
[35] D. Guidici and M. L. Clark, "One-dimensional convolutional neural network land-cover classification of multi-seasonal hyperspectral imagery in the San Francisco bay area, California," *Remote Sens.*, vol. 9, no. 6, 2017, Art. no. 629.
[36] J. Liu *et al.*, "A behavior privacy preserving method towards RF sensing," in *Proc. Symp. Qual. Serv.*, 2021, pp. 1–10.
[37] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. Conf. Artif. Intell. Statist.*, 2011, pp. 315–323.
[38] S. Singh, "Sigmoid as an activation function in neural networks," 2021. [Online]. Available: https://deeplearninguniversity.com/sigmoid-as-an-activation-function-in-neural-networks/#:∼:text=Sigmoid%20function%20also%20known%20as%20logistic%20function%20is,of%20all%20the%20inputs%20and%20the%20bias%20term
[39] Wictionary top 100,000 most frequently-used english words [for john the ripper], 2021. [Online]. Available: https://gist.github.com/h3xx/1976236
[40] NI USB-4431 ADC, 2021. [Online]. Available: https://www.ni.com/zh-cn/support/model.usb-4431.html
[41] Huawei supercharge wireless charger, 2021. [Online]. Available: https://consumer.huawei.com/en/accessories/sc-wireless-charger/
[42] Labview download, 2022. [Online]. Available: https://www.ni.com/zh-cn/support/downloads/software-products/download.labview.html#443274
[43] Top selling smartphones Q1 2021, 2021. [Online]. Available: https://dazeinfo.com/2021/05/24/top-selling-smartphones-q1–2021-iphone-12-redmi-9a-report/
[44] Startup, "A-Z of most downloaded apps in 2021," 2021. [Online]. Available: https://www.startuptalky.com/most-apps-downloaded/
[45] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.
[46] D. K. Sharma, A. AayushSharma, and J. Kumar, "KNNR: K-nearest neighbour classification based routing protocol for opportunistic networks," in *Proc. Conf. Contemporary Comput.*, 2017, pp. 1–6.
[47] NIC charger launches the world's first totally wireless charger, 2022. [Online]. Available: https://www.pr.com/press-release/766705
[48] Esp32, 2022. [Online]. Available: https://www.espressif.com/en/products/socs/esp32
[49] Wenergy wireless charger, 2022. [Online]. Available: https://m.tb.cn/h.fowypOe?tk=nQ7y26zI087
[50] R. Matovu, A. Serwadda, A. V. Bilbao, and I. Griswold-Steiner, "Defensive charging: Mitigating power side-channel attacks on charging smartphones," in *Proc. Conf. Data Appl. Secur. Privacy*, 2020, pp. 179–190.



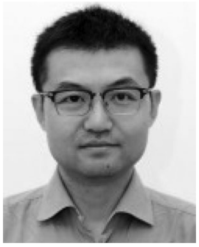
Jianwei Liu (Student Member, IEEE) received the BS degree from Northwestern Polytechnical University, in 2018, and the master's degree from Xi'an Jiaotong University, in 2021. He is currently working toward the PhD degree with Zhejiang University. His research interests include RFID, mobile computing, and smart sensing.

Xiang Zou (Student Member, IEEE) received the BS degree from the Xi'an University of Posts and Telecommunications, in 2014, and the master's degree from Chang'an University, in 2018. He is currently working toward the PhD degree with Xi'an Jiaotong University. He research interests include RFID, mobile computing, and IoT security.

Leqi Zhao is currently working toward the BS degree with Zhejiang University. Her research interests include IoT security and software security.

Yusheng Tao is currently working toward the BS degree with Zhejiang University. His research interests include IoT security and software security.

Sideng Hu (Member, IEEE) received the PhD degree from Tsinghua University, Beijing, China, in 2011. From August 2011 to August 2013, he was a postdoctoral researcher with the University of Michigan, Dearborn, MI, USA. Since September 2013, he has been with the College of Electrical Engineering, Zhejiang University, Hangzhou, China. His research interests include high power converters, flexible circuit, and vehicle.



Jinsong Han (Senior Member, IEEE) received the PhD degree from the Hong Kong University of Science and Technology, in 2007. He is currently a professor with the College of Computer Science and Technology, Zhejiang University. His research interests focus on IoT security, smart sensing, wireless, and mobile computing.

Kui Ren (Fellow, IEEE) received the PhD degree from Worcester Polytechnic Institute, Worcester, Massachusetts. He is currently a professor of computer science and technology and the director of the Institute of Cyberspace Research, Zhejiang University, Hangzhou, Zhejiang, China. His current research interests include cloud and outsourcing security, wireless and wearable system security, and artificial intelligence security. He is also a distinguished scientist and fellow of the ACM. He was a recipient of the IEEE CISTC Technical Recognition Award 2017 and the NSF CAREER Award in 2011.