MagSnoop: Listening to Sounds Induced by Magnetic Field Fluctuations to Infer Mobile Payment Tokens

Myeongwon Choi
Chung-Ang University
South Korea
auspic7@cau.ac.kr

Sangeun Oh
Ajou University
South Korea
sangeunoh@ajou.ac.kr

Insu Kim, Hyosu Kim*
Chung-Ang University
South Korea
{dlstn1121,hskimhello}@cau.ac.kr

ABSTRACT

Samsung Pay, one of the most representative mobile payment services, allows mobile users to make payment transactions almost anywhere using only their smartphone. This is thanks to MST (Magnetic Secure Transmission) that supports communication between smartphones and payment terminals for magnetic cards by transferring payment tokens via magnetic waves. Several attack methods have targeted this new technology by eavesdropping on magnetic fields to intercept the tokens, but with the use of dedicated hardware. This paper raises new security concerns for mobile payment users in a different, yet more effective way; by introducing MagSnoop, a novel framework that infers payment tokens from listening to MST sounds generated during the activation of MST payment transactions. More specifically, we first explore the principle, causing the generation of MST sounds, and the fundamental characteristics of these sounds. We then use these observations to infer payment tokens with a high degree of accuracy, robustness, applicability, and data efficiency. Our experiments with a prototype of MagSnoop demonstrate that it can support high accuracy in token inference (more than 77.8%). In addition, MagSnoop can maintain a reasonable level of accuracy regardless of the payment environments (e.g., 69.2% with a noise level of 50 dBA) and even in the real world (an inference success rate of 68.0% with 15 real-world users).

CCS CONCEPTS

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

KEYWORDS

Mobile Security; Mobile Payment Token Inference; Acoustic Side Channel Attacks; Magnetic Secure Transmission

ACM Reference Format:
Myeongwon Choi, Sangeun Oh, and Insu Kim, Hyosu Kim. 2022. MagSnoop: Listening to Sounds Induced by Magnetic Field Fluctuations to Infer Mobile Payment Tokens. In *The 20th Annual International Conference on Mobile Systems, Applications and Services (MobiSys '22), June 25–July 1, 2022, Portland, OR, USA.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3498361.3538937

*A corresponding author

MobiSys '22, June 25–July 1, 2022, Portland, OR, USA

ACM ISBN 978-1-4503-9185-6/22/06…$15.00
https://doi.org/10.1145/3498361.3538937

1 INTRODUCTION

The widespread prevalence of smartphones has profoundly changed the commercial activities of mobile users. In particular, they tend to increasingly use mobile payment services via their smartphones instead of cash or credit cards when purchasing goods. In line with this trend, various mobile payment services, such as Samsung Pay [30], Apple Pay [4], and Google Pay [16], have been commercialized, becoming primary applications for smartphones.

These wallet-less payments are basically achieved through near-field wireless communications, including Magnetic Secure Transmission (MST) and Near-Field Communication (NFC). Particularly, MST [36] used by Samsung Pay allows a smartphone to communicate with a traditional payment terminal for magnetic cards even without any hardware or software modification[1]. To this end, MST mimics a magnetic card swipe by flowing currents through a built-in conductor coil, called an MST coil, and finally by emitting a magnetic wave around the coil. The payment terminal then processes the requested payment transaction by reading the emitted magnetic wave. Note that the magnetic wave is repeatedly generated until the completion of the payment.

MST with these convenient features benefits mobile payment users but also opens new opportunities for adversaries because it carries sensitive data (i.e., payment tokens). For example, some studies [6, 8, 9] showed that it is possible to intercept an MST payment token by eavesdropping on magnetic waves. The leaked token then can be used for a fraudulent transaction. However, to extract the token accurately, the proposed methods require an attacker to be located close to a victim user and use a high-performance magnetic receiver (e.g., a pizza-box-sized coil) that is too large to carry. These points make it difficult to apply the magnetic-based attack methods in the real world.

In this paper, we explore the feasibility of introducing new and more serious security threats that infer MST payment tokens by listening to sounds. A specific pattern of sounds (called MST sounds) exposing payment tokens are generated when MST payment services are running. This is because smartphones have several hardware components made of ferromagnetic materials, such as a magnetic shield used to improve wireless charging efficiency. When a magnetic wave from an MST coil is applied to such materials, they are deformed due to the movement of their magnetic domains, i.e., the magnetostriction effect. The deformation then causes the vibration of the materials and eventually produces an MST sound. Hence, once an

[1]MST support may vary by country [31]. South Korea is a major country that uses MST.

MST payment transaction is activated, the MST sound is generated for each periodic emission of the magnetic wave.

Such a relationship between an MST sound and magnetic wave enables the following acoustic-based attack on MST-supported smartphones. A user activates an MST payment service to purchase something and an adversary's application installed on the victim's smartphone sneakily records audio using built-in microphones. The adversary then infers a payment token from the collected MST sound and uses the token for a deceitful transaction. That is, under our threat model, the attacker can easily intercept the payment token *i*) remotely and *ii*) without any special hardware equipment. To realize this attack, the most important thing is to support an accurate token inference while meeting the following requirements:

- *Noise robustness.* It should be possible to recover tokens even in the presence of ambient noise (e.g., music and conversational noise). When such noise is mixed with MST sounds, it becomes difficult to precisely detect the MST sounds from audio recordings. In addition, critical errors may occur in inferring tokens by distorting the features of the MST sounds.
- *Applicability.* We need to achieve a successful token inference on any type of MST-supported smartphone with different internal structures. For example, because the ferromagnetic components of each device have different shapes and sizes, they have unique frequency responses and eventually produce MST sounds with different characteristics. This makes it challenging to design a comprehensive way to extract tokens from MST sounds.
- *Data efficiency.* A token should be inferred correctly even with the analysis on a few numbers of MST sounds collected during the valid period of the token. For instance, in our experiments with real-world users, we observed that the average number of MST sounds emitted during the valid period (e.g., until a user completes a payment transaction) is only 4.6. This means that we have fairly limited opportunities to infer a token, making it more challenging to succeed the proposed attack.

We develop MagSnoop, an acoustic-based framework that enables the accurate, robust, applicable, and data-efficient inference of MST payment tokens. The proposed system consists of the three key techniques:

- We develop a noise-tolerant MST sound detection method. It leverages the time and frequency characteristics of MST sounds and allows to distinguish MST sounds from background noise.
- We design the ASE (Adaptively-Selected Envelope) feature which shows a significant difference depending on the transmitted bit value. In particular, we optimize the ASE feature set adaptively to a given payment environment.
- We combine the two different types of decoding techniques, which recover tokens in different viewpoints, and provide a rich set of token candidates even in the lack of available MST sounds.

We evaluate the performance of MagSnoop with our prototype implementation running on MST-supported Android smartphones. Our evaluation results demonstrate that MagSnoop can precisely recover MST payment tokens included in MST sounds. For example, a high degree of inference accuracy (77.8 % or more) is achieved on various smartphones and even with the analysis on only one MST sound. Furthermore, we show that MagSnoop supports accurate token inference in diverse environments and situations (e.g., above

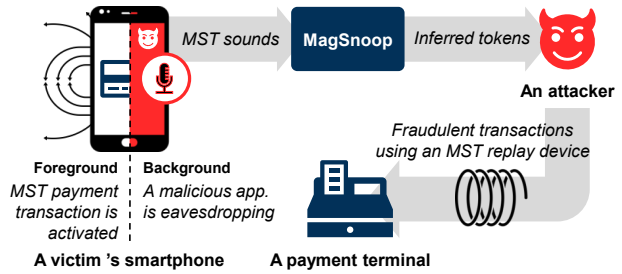


Figure 1: Attack flow for MST payment services.

69.2 % in the presence of ambient noise with a level of 50.0 dBA), thus allowing an attacker to make fraudulent transactions with a high success rate of 68.0% in the real world.

Our contributions can be summarized as follows.

- To the best of our knowledge, this work is the first attempt to explore the feasibility of leveraging MST sounds in inferring MST payment tokens.
- We deeply study the fundamental characteristics of MST sounds and payment tokens. Based on these observations, we design MagSnoop, a novel acoustic-based inference framework that accurately infers MST tokens while achieving a high degree of noise robustness, applicability, and data efficiency.
- We implement a prototype of MagSnoop[2] and reveal that it can introduce severe security threats for real-world mobile payment users.
- We suggest an unprecedented security warning for smartphone designers such that they should have concerns over putting permanent magnets within magnetic transmission-supported smartphones.

2 THREAT MODEL

An adversary, in this paper, *i*) targets a user who owns an MST-supported Android smartphone, *ii*) eavesdrops the victim's MST payment transaction, and *iii*) uses the payment token inferred by MagSnoop for a fraudulent transaction (see Figure 1). In particular, we make the following four assumptions in our threat model:

First, the adversary's malicious application should be installed on the victim's smartphone, with permission to access built-in microphones and the Internet. This is a common assumption used in many studies [3, 22, 32] to access raw sensor data from target devices. Specifically, the adversary can camouflage his/her application as a legitimate voice chatting application, which requires permissions to use both built-in microphones and the Internet. The application is then distributed through official or third-party app markets and installed on the victim's smartphone. During the installation, the victim may easily grant the benign-looking application permission to record audio and access the Internet because these permissions are one of the most common ones in Android applications[3].

Second, the malicious application should be able to collect audio samples sneakily. Android recently prevents applications from performing sensitive operations in the background without notifying a user. Therefore, the adversary's application should run as a

[2] See https://youtu.be/_uEjHtTrts0 for our demo video.

[3] 41.5% and 99.5% of Android applications among top 200 applications on the Google Play store use the `RECORD_AUDIO` and `INTERNET` permission, respectively.

foreground service that runs in the background, but with status bar notifications, including the application's icon and a microphone indicator, to the victim. However, through our field studies with real-world users, we observed that none of the users were aware of the notifications when they are using mobile payment services. In addition, the application can easily modify or even conceal the notification icon by using an empty image and run in the background furtively. The possibility to hide the application's running status can further increase by reducing the energy consumption required for audio recording. Towards this, the application can check when the smartphone's screen is on and collect data only for a while, e.g., 1 minute, after the event is detected. This is based on our observations that mobile payment services are usually activated soon after the screen turns on (see Section 6 for more implementation details).

Third, the attacker should be able to replay the payment token extracted by MagSnoop. With the help of the open-source implementation of magnetic signal transmitters, we can achieve this requirement with no effort. For instance, MagSpoof [21] proposed a coin-sized device that can emulate any magnetic stripe card. By using such MST-like devices, the attacker can easily generate magnetic signals for the inferred token and execute a deceitful transaction by transmitting the token to a magnetic payment terminal.

Fourth, the fraudulent transaction should be concluded before the victim completes his/her legitimate transaction. Samsung Pay generates a one-time token for security reasons (see more details in Section 3.3). Therefore, the attacker should minimize the time required to make a payment. One possible solution is to use always-accessible payment terminals, e.g., terminals installed in an unmanned store or his/her own terminals as proposed in [9]. This would allow the attacker to request a deceitful transaction as soon as a token is inferred. Note that through our experiments with the personal terminals, we observed that an end-to-end attack takes only 1.47 seconds, short enough to successfully conclude fraud transactions. More details will be discussed in Section 7.2.2.

Note that we can also consider another eavesdropping attack leveraging an external microphone. This attack model assumes that an adversary can sneakily place an external microphone (e.g., the adversary's smartphone) near a payment terminal in a store. In such a situation, when a victim brings his/her smartphone with an MST transaction activated close to the terminal, the adversary captures sounds emanating from the device via the external microphone and executes a fraudulent transaction with the help of MagSnoop. Additionally, at this point, the adversary can interfere with the victim's transactions using commercial jammers to increase the attack success rate, as proposed in other studies [6, 9]. We demonstrate the feasibility of using external microphones in Section 7.1.3.

3 BACKGROUND

To better understand the design of MagSnoop, we describe preliminaries, including magnetic stripe cards, magnetic secure transmission (MST), and Samsung Pay.

3.1 Magnetic Stripe Cards

A magnetic stripe card is a kind of plastic card that has a band of magnetic material, called a magnetic stripe, on its surface. The magnetic stripe holds up to three horizontally stacked tracks, called

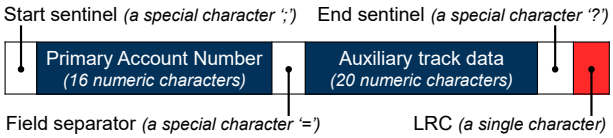
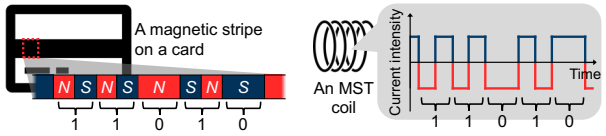


Figure 2: Typical data structure of track 2 defined in the ISO/IEC 7811-2 standard [20].



(a) On a magnetic stripe. (b) In an MST current pulse.

Figure 3: Encoded bit sequence of the start sentinel for track 2.

track 1, 2, and 3, each of which has its own data format [20][4]. For example, as illustrated in Figure 2, track 2 contains 40 alphanumeric data, consisting of the numbers (0-9) and six special characters (:, ;, <, =, >, ?). It starts with the start sentinel (character ';') followed by a primary account number, which is usually a credit card number, and a separator character '='. After that, auxiliary track data including the expiration date, service code, and discretionary data is located and followed by the end sentinel (character '?') and a longitudinal redundancy check (LRC).

The alphanumeric data in track 2 is converted into a sequence of bits using a 5-bit scheme (see Table 1) and embedded onto a magnetic stripe. It should be noted that at this time, a series of zeros are added before and after the bit sequence, each of which is called leading zeros and trailing zeros, respectively. They are used for clock synchronization with receivers. The magnetic stripe is encoded with this bit sequence by magnetizing its tiny iron-based magnetic particles in either a north or south pole direction (see Figure 3(a)). More specifically, a two-frequency (F2F) encoding technique is applied as follows; a single bit has a fixed length (approximately several hundreds of micrometers) on the stripe and magnetic flux transitions from north to south or vice versa are located at the beginning and end of each bit and also in the middle of each bit that represents "1".

The encoded information on a magnetic stripe card is transferred to a magnetic stripe reader, such as a point-of-sale (POS) device, by swiping the card through the device. When the card is swiped, the magnetic environment of the reader varies due to the flux transitions in the stripe of the card. According to Faraday's law [17], these changes in the magnetic field cause a voltage to be induced in the coils of the reader. That is, the flux transitions in the magnetic stripe card are converted into a time-varying electrical signal. The reader then decodes the received signal in the following steps. It first computes a clock period, using leading and trailing zeros. For each clock period, the reader determines its bit value as 0 or 1 based on the absence or presence of voltage fluctuations in the middle of the period, respectively. Finally, the track data is recovered from the bit sequence and used for further transaction processes.

[4]In this paper, we cover only track 2 because Samsung Pay, our target application, mainly utilizes this format of data for payment transactions [26]

Table 1: Coded character set for 5-bit numeric used for track 2. Note that the last bit (denoted as b_4) of each code is a parity bit.

	b_0	b_1	b_2	b_3	b_4		b_0	b_1	b_2	b_3	b_4
0	0	0	0	0	1	8	0	0	0	1	0
1	1	0	0	0	0	9	1	0	0	1	1
2	0	1	0	0	0	:	0	1	0	1	1
3	1	1	0	0	1	;	1	1	0	1	0
4	0	0	1	0	0	<	0	0	1	1	1
5	1	0	1	0	1	=	1	0	1	1	0
6	0	1	1	0	1	>	0	1	1	1	0
7	1	1	1	0	0	?	1	1	1	1	1

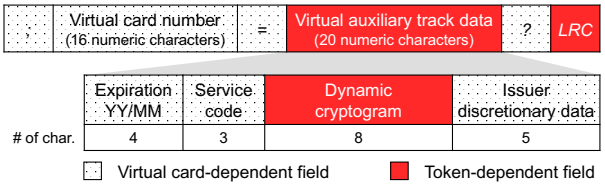


Figure 4: Typical structure of a Samsung Pay token. Different financial cards are converted into non-identical virtual cards. Plus, tokens even for the same card hold different cryptogram, making them unique.

3.2 Magnetic Secure Transmission (MST)

MST is a patented technique [36] to support near-field data transmission. According to Ampere's law [17], when electric current passes through an inductor, a magnetic wave is generated and its polarity and strength are determined by those of the current. Based on this, an MST-supported device transfers data by designing a corresponding series of current pulse and driving its built-in inductor with the pulse.

Such a magnetic-based transmission allows to communicate with existing magnetic stripe readers without any modification on them. Towards this, MST-supported devices mimic variations in a magnetic field produced when a magnetic stripe card is swiped through the readers. Figure 3(b) illustrates an example of current pulses to transmit the start sentinel for track 2, which is a bit sequence of 11010. For each clock period, each bit in the sequence is encoded by using an F2F coding scheme. In other words, a physically-encoded bit sequence on a magnetic stripe (as shown in Figure 3(a)) is turned into a pulse-like time-series signal. This current signal then passes through an inductor, leading to the emission of a magnetic wave. At this time, the generated magnetic wave has the same structure as the current. For example, each edge in the current signal causes a magnetic flux transition in the wave. Finally, a magnetic stripe reader captures a voltage introduced by these flux transitions and takes the transmitted data.

3.3 Samsung Pay

Samsung Pay [30] is the most representative mobile payment service based on MST[5]. It is available on mobile devices manufactured by Samsung Electronics and equipped with a special inductor, known as an MST coil. Samsung Pay supports secure mobile payment transactions with the following two security features:

[5]Samsung Pay also supports an NFC payment.

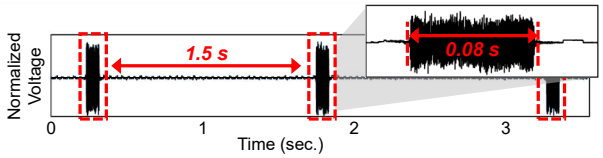


Figure 5: Voltage fluctuations observed by a magnetic stripe reader during the activation of Samsung Pay. Note that the voltage was measured by a magnetic read head.

- *Authentication.* To activate a Samsung Pay transaction, a user launches the Samsung Pay application and selects one of the registered cards. At this time, the user is authenticated with a PIN code or a fingerprint. That is, for a certain device, only a legitimate user can make a payment using Samsung Pay.
- *Tokenization.* Once the user is authorized, the financial card selected for the transaction is converted into a non-sensitive equivalent, called a payment token. As depicted in Figure 4, the token consists of virtual information, e.g., virtual card numbers, that is different depending on the financial card. In particular, even for the same card, Samsung Pay generates a unique token for each transaction request, by using special eight characters, called a dynamic cryptogram. The issued token then expires *i*) when the requested transaction is successfully completed or *ii*) within 50 seconds. It ensures that the same token can never be used more than once and thus allows more secure transactions even in the leakage of the token.

Such a secure token is then assembled in the format of track 2, broadcasted as a magnetic wave through a built-in MST coil, and received by a POS device, which supports magnetic stripe cards. More specifically, the tokenized data contains 260 bits (40 alphanumeric data × 5 bits + 30 leading zero bits + 30 trailing zero bits) and is transferred for approximately 0.08 seconds, i.e., with a clock frequency of approximately 3.2 kHz (see Figure 5)[6]. The token is repeatedly transmitted with a time interval of about 1.5 seconds until it expires. For each transfer, the POS device attempts to extract the tokenized track data from the received signal and communicates with its payment service provider to process the requested transaction.

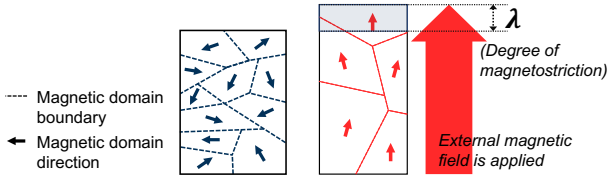
4 UNDERSTANDING MST SOUNDS

In this section, we deeply explore *i*) the reason why *MST sounds* are generated in using MST-based services, such as Samsung Pay, and *ii*) the fundamental characteristics of the MST sounds that will be a hint for MagSnoop to infer payment tokens.

4.1 Generation of MST Sounds

When a magnetic field is applied to ferromagnetic materials, such as iron, their shape and size change. This phenomenon is called *magnetostriction* [24]. As shown in Figure 6(a), a ferromagnetic material initially has a structure consisting of multiple magnetic domains. When a magnetic field is applied with an intensity of H, the boundaries between domains are shifted in a way to grow the magnetic domains in the direction of the magnetic field (see Figure 6(b)). This leads to an increase in the length of the ferromagnetic material in the magnetic field direction. At this time, the amount of an increase in the length, called the degree of magnetostriction (λ),

[6]A clock frequency can vary depending on the device



(a) In the absence of an external magnetic field. (b) In the presence of an external magnetic field.

Figure 6: Magnetostrictive phenomenon caused on ferromagnetic materials during a magnetization process.

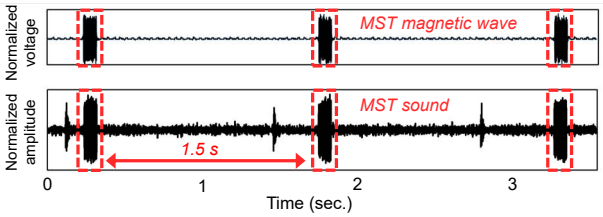


Figure 7: MST sounds generated using Samsung Pay. Note that MST sounds was collected using a Samsung Galaxy Z Flip 3 smartphone.

is determined by H. For example, as H increases, the region occupied by the dominant domains is further extended, i.e., λ grows, and finally all the domains are merged into a single one, i.e., magnetic saturation occurs.

As explained in Section 3.2, an alternating magnetic flux, we call it *an MST magnetic wave*, radiates from a smartphone's built-in MST coil when an MST payment transaction is activated. The magnetic wave is applied to nearby ferromagnetic objects, causing them to vibrate. For example, most smartphones have a thin and large ferrite plate, called a magnetic shield, to improve wireless charging efficiency. It experiences magnetostriction by the MST magnetic wave. In particular, due to the time-varying characteristic of the wave, the degree of magnetostriction (λ) also changes over time. In other words, the plate's shape and size continuously change, and it eventually vibrates. The vibration then propagates through the air in the form of a sound wave, i.e., an MST sound. Figure 7 illustrates that sounds of approximately 80 ms are generated every 1.5 seconds during the activation of a Samsung Pay transaction. This indicates that for each emission of an MST magnetic wave, a corresponding MST sound is produced because of magnetostriction.

4.2 Characteristics of MST Sounds

As mentioned above, an MST magnetic wave causes the vibration of ferromagnetic objects present in a smartphone and generates an MST sound. The characteristics of the MST sound thus vary depending on *i*) the properties of the ferromagnetic objects and *ii*) the structure of the magnetic wave.

Effect of the presence of permanent magnets. Ferromagnetic objects vibrate harder in the presence of permanent magnets placed close to them [18]. This is because of a nonlinear relationship between the degree of magnetostriction (λ) and the intensity of an applied magnetic field (H) as shown in Figure 8. In the absence

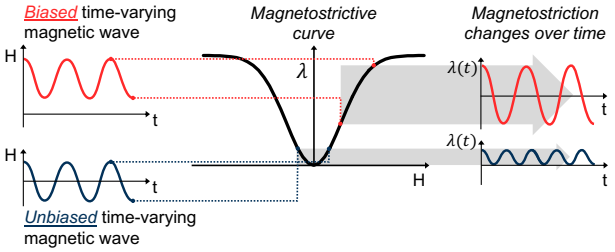


Figure 8: Typical magnetostrictive curve for ferromagnetic materials and effect of an external (bias) magnetic field on changes in magnetostriction.

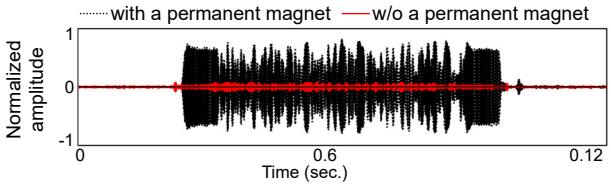


Figure 9: Effect of permanent magnets on the amplitude of an MST sound. Note that the MST sound was recorded on a Samsung Galaxy S10 smartphone using its built-in microphone. In particular, we applied a magnetic field to the smartphone by putting a magnetic case on it.

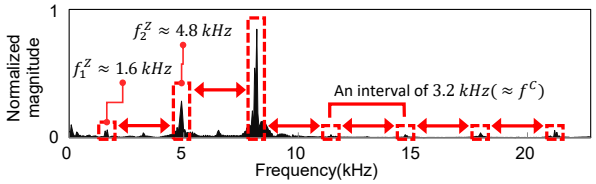


Figure 10: FFT spectrum of an MST sound for transmitting leading zeros.

of a permanent magnet, an MST magnetic wave causes λ to vary with low amplitude and a doubled frequency compared to the wave. In contrast, a permanent magnet creates a bias magnetic field and shifts the average λ point, resulting in a more linear relationship between the MST magnetic wave and λ and eventually larger fluctuation ranges in λ over time (see Figure 8). Hence, smartphones that have built-in permanent magnets, such as foldable smartphones[7], MST sounds are produced with a sufficient level of energy to be recorded even from built-in microphones. Furthermore, many users put a magnetic case on their smartphone to stably fix the device in various positions. This external permanent magnet can also lead to an increase in the amplitude of MST sounds, as shown in Figure 9.

Effect of MST magnetic waves. As explained in Section 3.2, an MST magnetic wave is encoded by using an F2F scheme. That is, the polarity of the wave is reversed every certain time interval. For example, a bit sequence of 0 (e.g., leading zeros) is transferred as a periodic wave in which a magnetic flux transition occurs every clock period (l^C). This wave then causes periodic changes in magnetostriction. Let denote the amount of changes in magnetostriction at time t as $\Delta\lambda(t)$, i.e., $\lambda(t) - \lambda(t - \Delta t)$, where Δt is unit time. During the

[7]Foldable smartphones, such as Samsung Galaxy ZFlip 3 and ZFold 3, have small neodymium magnets to support the folding function.

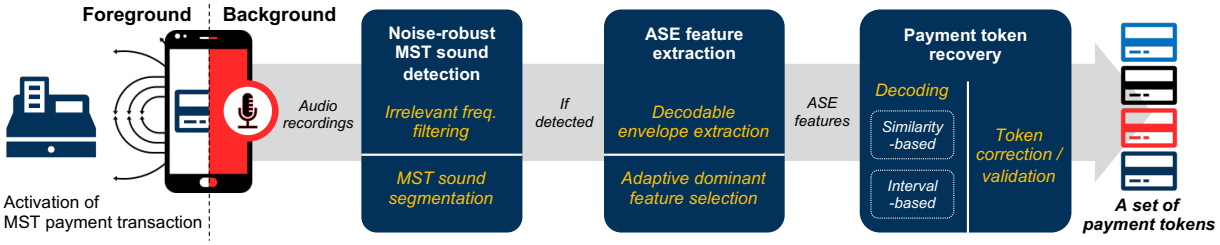


Figure 11: Overall procedure of MagSnoop to infer payment information from MST sounds.

transmission of zeros, $\Delta\lambda(t)$ has a distinctive positive or negative peak at each clock period. In other words, we can model $\Delta\lambda(t)$ as a cosine wave $cos(\omega_0 t)$ sampled at a rate of f^C Hz, where f^C is $1/l^C$ and ω_0 is πf^C. Then, the discrete-time Fourier transform of $\Delta\lambda(t)$, denoted as $X(\omega)$, is computed as follows:

$$X(\omega) = \sum_{k=-\infty}^{\infty} \pi(\delta(\omega + \omega_0 - k\omega^C) + \delta(\omega - \omega_0 - k\omega^C)), \quad (1)$$

where ω^C is an angular sampling frequency of $2\pi f^C$ rad/s. $X(\omega)$ has a peak value at an angular frequency of $k\omega^C \pm \omega_0 = (2k \pm 1) \cdot \omega_0$ rad/s, where $k \geq 1$. It causes the generation of MST sounds consisting of frequency components of $\frac{(2k-1)f^C}{2}$ Hz (see Figure 10). Similarly, in transmitting a sequence of 1s, the magnetic field polarity is reversed every $l^C/2$, radiating MST sounds with frequency components of $(2k-1)f^C$ Hz. It is noteworthy that the frequency characteristics of MST sounds are also affected by the properties of ferromagnetic materials. For example, MST sounds carry more energy at frequencies near the resonant frequency of the materials.

5 MAGSNOOP DESIGN

MagSnoop aims to accurately recover a payment token from a set of MST sounds, collected during the activation of an MST payment transaction, especially i) regardless of the payment environment (e.g., ambient noise), ii) on any mobile device of different structure, and iii) before the token expires (with a few number of MST sounds).

A key enabler of MagSnoop is to fully utilize the characteristics of MST sounds in the whole inference process, as illustrated in Figure 11. First, MagSnoop detects an MST sound precisely even from noisy recordings by filtering out irrelevant frequencies and pinpointing its starting point based on the time and frequency characteristics of MST sounds. It then extracts ASE features from the MST sound by choosing dominant frequency components adaptively to the given payment environment, including the payment device. Finally, MagSnoop decodes the features using the preliminary knowledge about the structure of magnetic stripes and Samsung Pay tokens. More specifically, MagSnoop increases the likelihood of successful inference by suggesting multiple candidate tokens from a single MST sound as a result of the combined decoding schemes.

5.1 MST Sound Detection

During the activation of an MST payment transaction, MagSnoop keeps recording sounds and attempts to detect MST sounds every l^P seconds, where l^P is a magnetic wave emission period in Samsung Pay (approximately 1.5 seconds). In other words, for each period, MagSnoop determines the existence of an MST sound in $y(t)$, audio

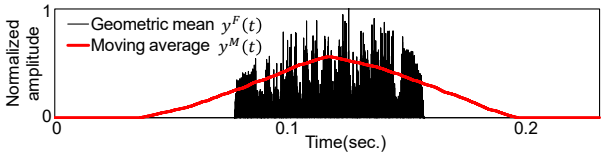


Figure 12: Geometric mean and moving average of audio recordings in effective frequencies. These features are used to pinpoint the starting point of MST sounds.

samples collected in the last l^P seconds. In particular, to avoid false positives in detection, MagSnoop fully utilizes the characteristics of MST sounds.

MagSnoop first selects effective frequency ranges in which MST sounds show a significant sound level. As explained in Section 4.2, depending on the transferred value, an MST sound consists of different frequency components. For example, the i-th dominant frequency of an MST sound to transmit a bit value of 0 and 1 are $\frac{(2i-1)f^C}{2}$ and $(2i-1)f^C$ Hz, respectively, where f^C is set to 3.2 kHz, an approximate clock frequency of Samsung Pay. Let denote such a dominant frequency for a bit 0 and 1 as f_i^Z and f_i^O, respectively. MagSnoop determines the k-th effective frequencies as a range from f_{2k-1}^Z to f_k^O and builds a band-pass Butterworth filter [7] with an order of 10 for each frequency range. It then computes $y_k^F(t)$ by applying the k-th filter to $y(t)$ and takes their geometric mean $y^F(t)$ as follows:

$$y^F(t) = (\prod_{k=1}^{n^F} |y_k^F(t)|)^{\frac{1}{n^F}}, \quad (2)$$

where n^F is set to 4 so as to exclude frequencies above 24 kHz which are rarely captured by commodity microphones. $y^F(t)$ then has high amplitude only in the existence of sound waves, such as MST sounds, which have a significant level of energy in all the effective frequency ranges, as illustrated in Figure 12.

MST sound segmentation. Given $y^F(t)$, MagSnoop detects an MST sound in the following steps. First, it computes $y^M(t)$, a moving average of $y^F(t)$, as $\frac{1}{l^S}\sum_{i=t-l^S}^{t} y^F(i)$, where l^S is the length of MST sounds (i.e., 0.08 seconds). $y^M(t)$ has a triangular shape when an MST sound, a sound of l^S seconds, exists (see Figure 12). Based on this characteristic, MagSnoop determines the presence of an MST sound at a time t^* if $y^M(t^*)$ is greater than a certain threshold ϵ^M and $y^M(t)$ in a range from $t^* - l^S$ to $t^* + l^S$ shows a high degree of Pearson correlation (> 0.95) with a triangular-shaped template data. Note that an MST sound is produced at most once for a duration of l^P seconds. Therefore, if the conditions are satisfied

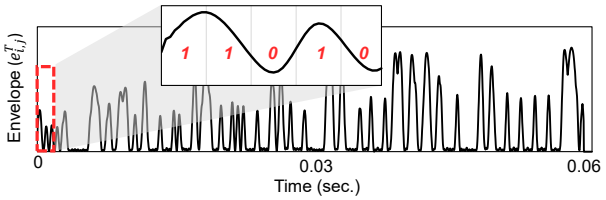


Figure 13: Envelop feature ($e_{i,j}(t)$) extracted from an MST sound with a frequency range from 9.0 to 10.2 kHz.

multiple times in this period, the time instant which has the highest correlation value is selected. Finally, MagSnoop takes the original (i.e., non-filtered) sound between $t^* - l^S$ and t^* as an MST sound.

5.2 ASE Feature Extraction

Once an MST sound $s(t)$ is detected, MagSnoop takes ASE (*Adaptively-Selected Envelope*) features, designed to easily distinguish between a bit of 0 and 1 on any mobile device. First, it extracts the frequency components of $s(t)$ each of which carries different dominant frequencies for transmitting a bit 1. Towards this, it builds a band-pass Butterworth filter ($b_{i,j}$) with an order of 16 and a frequency range in $[f_i^O - \frac{f^C}{10}j, f_i^O + \frac{f^C}{10}j]$, where i and j are in the range of 1 to 4 and 1 to 5, respectively. MagSnoop applies each filter $b_{i,j}$ to the detected sound and eventually obtains its corresponding frequency component, denoted by $s^F_{i,j}(t)$, which has a higher amplitude when transmitting a bit 1 than when transmitting a bit 0. Finally, it computes the upper envelope of the squared $s^F_{i,j}(t)$, denoted by $e_{i,j}(t)$. As illustrated in Figure 13, in $e_{i,j}(t)$, the positive and negative peaks indicate the transmission of a bit value of 1 and 0, respectively.

Adaptive feature selection. Among these envelopes, MagSnoop selects a few of them which show a significant difference depending on the transmitted bit value. Note that this selection process is performed every time when inference is requested. In other words, a set of envelope features is chosen adaptively to the given payment environment. To this end, MagSnoop computes a range of $e^L_{i,j}$ to $e^U_{i,j}$, called a maximum distinguishable range, for each $e_{i,j}(t)$. Here, $e^L_{i,j}$ and $e^U_{i,j}$ are the lower and upper limits, respectively, to differentiate a bit value of 0 and 1. As an MST payment token starts with leading zeros followed by a start sentinel of [1, 1, 0, 1, 0], two distinctive positive peaks always appear in the early part of $e_{i,j}(t)$ (see Figure 13). Based on this, $e^U_{i,j}$ is initially set to the minimum between these peaks and further optimized as the minimum value among all positive peaks greater than $0.5 \cdot e^U_{i,j}$. After that, MagSnoop calculates $e^L_{i,j}$ as the maximum among positive and negative peaks less than $e^U_{i,j}$. This heuristic method enables us to get an approximate maximum range that can distinguish the positive and negative peaks caused during the transmission of a bit 1 and 0. MagSnoop then chooses the top-n^E envelopes that have the largest difference between $e^L_{i,j}$ and $e^U_{i,j}$, where n^E is set to 3.

Token part extraction. Suppose that $e_i(t)$ and $[e^L_i, e^U_i]$ denote the i-th selected envelope and its maximum distinguishable range, respectively. MagSnoop extracts the token part of $e_i(t)$, denoted by $e^T_i(t)$, which contains the actual payment information (e.g., credit card numbers) except leading and trailing zeros. It simply detects the first and last time instants, which have a greater value than e^U_i and takes $e_i(t)$ between the two time points.

However, this simple method may entail a loss of information because $e^T_i(t)$ includes only part of the token starting and ending with a bit 1. For instance, LRC, the last character of the token, can end with n^Z consecutive zeros, where n^Z varies depending on the other characters in the token. Thus, $e^T_i(t)$ misses such zero bits, causing errors in inferring tokens. MagSnoop addresses this by concatenating a series of zeros at the end of $e^T_i(t)$. At this time, the length of the padded zeros l^Z_i is computed as $\frac{l^T_i}{(n^B - n^Z)}n^Z$, where l^T_i is the sample length of $e^T_i(t)$ and n^B is the total number of bits in the payment token, i.e., 200. Here, because e^L_i is always greater than 0, the zero sequence of length l^Z represents n^Z bits of zeros in the token. Note that as shown in Table 1, n^Z can vary from 0 to 4 depending the token. Therefore, MagSnoop repeats this zero-padding process for each $e^T_i(t)$ and each possible n^Z and generates $5 \times n^E$ features in total. The zero-padded envelope is then used as our main feature for token inference.

5.3 Payment Token Recovery

MagSnoop basically generates a number of candidate tokens by leveraging different types of decoding techniques. At this time, as the number of candidate tokens increases, the likelihood that the candidate set holds the correct token also increases. This, however, can make our attack scenario impractical because an adversary should replay all the candidate tokens one by one to make a payment. Therefore, the number of candidates should be kept small enough to perform the replay attack in a short time period (e.g., a few second). To this end, MagSnoop first focuses on generating as many candidate tokens as possible and then prunes useless ones via token correction and validation.

Possible character map construction. As observed in Section 3.3, each field of a Samsung Pay token can hold a specific set of characters. For example, the token always starts with the character ';' and ends with '?'. The 2nd to 17th fields (i.e., card numbers) are filled with numeric data. Based on this characteristic, MagSnoop determines a set of possible characters, denoted as $\mathcal{P}_i$, for the i-th token field. This mapping information is used to validate and optimize decoded tokens.

Token set generation. Given the ASE feature and its maximum distinguishable range, denoted by *ASE* and $[e^L, e^U]$, respectively, MagSnoop infers tokens using the two different kinds of decoding methods.

- *Similarity-based decoding. ASE* is separated into n^A segments, where n^A is set to 40, the total number of alphanumeric characters in a payment token. The i-th segment, denoted as w_i, is related with the i-th character of the token (c_i), encoded using a 5-bit scheme. Therefore, MagSnoop further divides w_i into 5 sub-segments, each of which is denoted as $w_{i,j}$, and computes their mean value $\mu_{i,j}$ as $\frac{1}{l^W}\sum_{t=1}^{l^W} w_{i,j}(t)$, where l^W is the sub-segment's size. It then computes the similarity between a sequence of $\mu_{i,j}$ and characters in $\mathcal{P}_i$. Let $r_{i,k}$ denote the similarity of the

k-th character in $\mathcal{P}_i$, which is computed as:

$$r_{i,k} = \sum_{j=1}^{5} (\mu_{i,j} - \frac{e^L + e^U}{2}) \times \alpha_{k,j}, \quad (3)$$

where $\alpha_{k,j}$ is a similarity coefficient which is set to 1 if the j-th bit of the character is 1, otherwise -1. MagSnoop finally decides c_i as the m-th character in $\mathcal{P}_i$, which provides the highest similarity among all possible characters.

- *Interval-based decoding.* MagSnoop first detects time instants at which a bit transition from 0 to 1 or 1 to 0 occurs. To this end, it first subtracts $\frac{e^L+e^U}{2}$ from *ASE* and finds every zero-crossing point. Let t_i^X denote the i-th zero-crossing point in *ASE*. MagSnoop computes n_i^B, the number of bit values transferred until t_i^X, as $\frac{t_i^X}{l^{ASE}} n^B$, where l^{ASE} is the length of *ASE*, and n^B is the total number of bits in a payment token. It then decides the $(n_{i-1}^B + 1)$-th to the n_i^B-th bits of the inferred token as 0 if i is even, otherwise 1. MagSnoop finally determines c_j, the j-th character of the token, using the corresponding five values in the bit sequence. Note that if the 5-bit of data is not matched with any character in $\mathcal{P}_j$, then MagSnoop applies the similarity-based method to obtain c_j.

Invalid token pruning. For the decoded token, MagSnoop checks its validity by leveraging the last character of the token (LRC). Specifically, LRC is re-calculated using other decoded characters. MagSnoop then determines the token as invalid one if *i*) its computed LRC is not a character used for track 2 or *ii*) it does not end with n^Z consecutive zeros, where n^Z is the number of padded zero bits calculated in our feature extraction process to construct *ASE*. These decoding and validation processes are performed for each ASE feature and eventually, n^V valid tokens are generated.

Further optimizations via token reuse. As explained in Section 3.3, Samsung Pay tokens derived from the same magnetic card have the same value in most fields, except dynamic cryptogram consisting of 8 characters. This might lead to an extreme increase in the possibility of accurately recovering tokens in the following token reuse scenario. Assume that a specific card was used several times for payment in Samsung Pay and MagSnoop successfully inferred the tokens. Then, MagSnoop can simply reuse some of the token values, e.g., virtual card numbers, for future token inference. In other words, to obtain the token used for the next transaction, MagSnoop is just required to recover the eight characters, i.e., dynamic cryptogram.

6 IMPLEMENTATION

We implemented a prototype of MagSnoop consisting of the following two components: *i*) a mobile application and *ii*) a server application.

Mobile application. A malicious application, disguised as a voice chatting app, was implemented on commodity MST-supported smartphones running Android 11.0 (SDK version 30). This application operates as a foreground service to utilize built-in microphones by complying with the Android security regulations. In addition, its notification icon displayed in the status bar is set as an empty image to hide its execution status, as discussed in Section 2.

The primary issue in implementing the mobile application is to detect when a victim activates an MST payment service. A naive way to do this is to keep recording all day, but it can drain the smartphone's battery quickly and expose its malicious behavior to the user. Instead, we made our mobile application perform recording only for a minute after the smartphone screen is on to make MagSnoop more stealthy. This is based on the assumption that the victim will activate the payment service soon after the screen turns on. To this end, the application employs a broadcast receiver to wait for the Android system events, such as `SCREEN_ON`. Once the event is received, the application starts recording in the background, detects an MST sound, and transmits it to the server. At this point, the recording is performed with a sampling rate of 192 kHz using the `UNPROCESSED` audio source to eliminate the effect of manufacturer-specific pre-processing techniques. Note that the energy consumed by the MagSnoop application is negligible. For example, when the application is running in the background on Samsung Galaxy Z Flip3, the battery consumption increases by 127.32 mW on average, but just for the recording duration, i.e., 1 minute.

Server application. We developed the server application as a Matlab implementation running on a desktop computer with an AMD Ryzen 5 5900 and 64GB RAM. It infers the victim's payment token by performing our proposed feature extraction and decoding algorithms.

7 EVALUATION

In this section, we first evaluate the overall performance of MagSnoop and verify how well MagSnoop can be applied in diverse payment environments and in the real world.

7.1 Micro-Benchmark Tests

We conducted several micro-benchmark experiments to evaluate the performance of MagSnoop in a quiet place (e.g., an office) using three kinds of MST-supported devices, such as Samsung Galaxy Z Flip 3 (*ZFlip3*), Z Flip (*ZFlip*), and S10 5G. In particular, we used two configurations for Galaxy S10: a device with a magnetic cover case (*S10-case*) and a device without any cover case (*S10-no-case*). Note that we did not use any cover case for other devices unless otherwise mentioned. We first activated Samsung Pay from each device placed on a desk and collected the ground truth of the payment token through a commodity magnetic stripe reader. This value is used to check whether the payment token inferred by MagSnoop is correct. During that time, we collected raw audio sensor data using our mobile application running in the background.

Metrics. We verify the performance of the proposed detection and inference methods with the following metrics:

- *Detection accuracy:* We define detection accuracy as the ratio of successfully detected MST sounds to the total number of emitted MST sounds.
- *Inference accuracy:* Suppose that n^D MST sounds were detected during the activation of one MST payment transaction. Under this assumption, we define inference accuracy as the probability of successfully inferring the payment token from the n^D MST sounds. Note that this is equivalent to the probability that one of the candidate tokens generated from the n^D MST sounds is exactly the same as the target token.

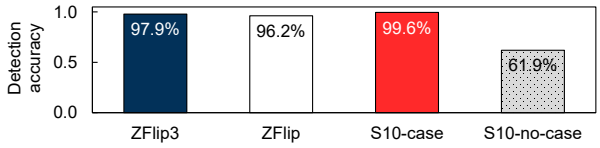


Figure 14: Overall detection accuracy of MagSnoop.

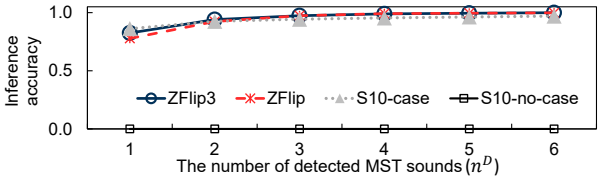


Figure 15: Overall inference accuracy of MagSnoop. Note that during the activation of an MST payment transaction, MST sounds are repeatedly emitted and detected every 1.5 seconds.

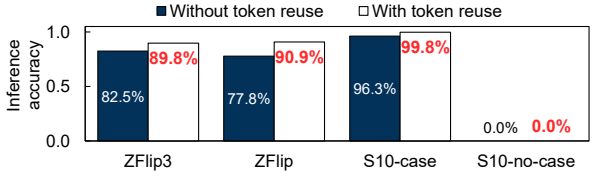


Figure 16: Impact of token reuse on inference accuracy (when n^D is equal to 1).

7.1.1 Accuracy test. We evaluate detection/inference accuracy that MagSnoop can achieve in an ideal environment with little noise.

Overall accuracy. Figure 14 shows that MagSnoop provides high detection accuracy of 96.2% or more for all devices except S10-no-case. Low accuracy of 61.9% is obtained for S10-no-case because the intensity of MST sounds is too low to detect. On the contrary, in the case of the other devices, MST sounds are emitted with sufficient amplitude thanks to their built-in permanent magnets; ZFlip3 and ZFlip internally have permanent magnets to support the folding function, and S10-case has a cover case with a permanent magnet. As described in Section 4, such magnets form bias magnetic fields, which increase the amount of changes in magnetostriction, and help MagSnoop detect MST sounds with a high degree of accuracy.

The inference accuracy of MagSnoop shows a similar trend with the detection accuracy; it achieves high inference accuracy of more than 77.8% for all devices except S10-no-case (see Figure 15). In particular, the accuracy gradually increases and converges to 100.0% as the the number of detected MST sounds (n^D) increases. This means that the longer the MST activation time is, the more likely MagSnoop will intercept the transmitted payment token. On the other hand, we observed that all interference attempts failed for S10-no-case due to the low intensity of the MST sounds. This implies that the MST sound amplification caused by permanent magnets is also an important factor in inference as well as detection.

Accuracy improvement via token reuse. As mentioned in Section 5.3, upon a payment token is successfully inferred, we can reuse some parts of the token (e.g., virtual card numbers) for the next inference attempt. Figure 16 shows the accuracy of inferring

Table 2: Performance trade-off according to decoding schemes

	Inference accuracy	No. of candidate tokens
Similarity	72.4%	3.7
Interval	68.9%	3.4
Combined	82.5%	7.1

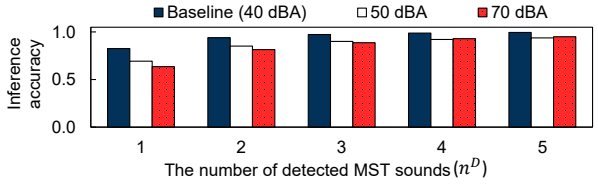


Figure 17: Inference accuracy of MagSnoop in the presence of ambient noise. The noise level was measured at the location of the smartphone.

8-digit variable characters (i.e., dynamic cryptogram), assuming that the remaining parts of the payment token are referenced from the previous inference. We confirmed that reusing historical data of the previous inference is quite effective; it improves inference accuracy by up to 13.1 percentage points. This means that the leaked token can cause severe security threats not only now, but also in the future, even though it is a one-time token.

Trade-off according to decoding schemes. MagSnoop infers payment tokens by using both similarity-based and interval-based decoding schemes together to increase inference accuracy as described in Section 5.3. However, this may generate a large number of candidate tokens, making it difficult for an attacker to perform a replay attack with them. To examine this trade-off, we compare the method using the combination of the two schemes and methods using only one decoding scheme (similarity-based or interval-based). Table 2 shows inference accuracy and the number of candidate tokens for each method when only one MST sound is detected on ZFlip3. The combined method significantly improves the inference accuracy by 10.1% to 13.6% compared to the similarity-based and interval-based methods, respectively, but, generates more candidate tokens by about two times than the single methods. However, we confirmed that the number of the generated tokens is still small enough to perform a replay attack in the real world. For example, an attacker can attempt to make a payment with each token in parallel, using multiple (e.g., two or three) replay devices and payment terminals.

7.1.2 Robustness test. We evaluate how robustly MagSnoop can extract payment tokens from MST sounds in the presence of several hindrances. Note that in this experiment, we used ZFlip3 as our primary device[8]. We divide such factors into three cases: *i*) when ambient noise exists, *ii*) when external accessories (e.g., cover cases) are equipped, and *iii*) when user behaviors incur noise. Note that we mainly focus on the results of inference accuracy in this section because we observed that there is no significant difference in detection accuracy compared to the situation with little noise.

Robustness against ambient noise. MagSnoop, an acoustic-based token inference framework, can be inherently vulnerable to ambient

[8]We observed that other devices, such as ZFlip and S10-case, show similar results with ZFlip3.

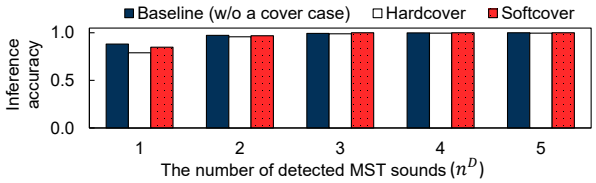


Figure 18: Impact of using external accessories (cover cases) on inference accuracy.

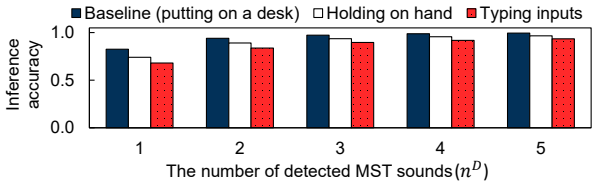


Figure 19: Robustness of MagSnoop against user behaviors.

noise, such as people's conversations. To check the effect of ambient noise, we first captured real-world noise from restaurants where MST payments are mainly made. Then, we reproduced each noise with a sound pressure level of 50.0 dBA and 70.0 dBA, on average, using the speaker of a laptop (Apple Macbook Air). Figure 17 shows that inference accuracy tends to decrease slightly as the noise level increases. With n^D of 1, the amount of accuracy drops for 50 dBA and 70 dBA noises are about 13.2 and 18.9 percentage points, respectively. However, MagSnoop still provides high accuracy more than 60.0% for each noise level. In addition, as the number of detected MST sounds increases, the inference accuracy is further improved. This is because the majority of ambient noise (e.g., conversations) appears in frequencies below 8 kHz, whereas MST sounds have dominant energy in frequencies near or even above 9 kHz. Thus, these characteristics help MagSnoop provide a reasonable degree of inference accuracy even in the presence of noise.

Robustness against using external accessories. A cover case, one of the popular smartphone accessories, may interfere with the generation and propagation of MST sounds, thereby degrading the performance of MagSnoop. To check whether this negative effect may occur in practice, we used different cover cases made of two materials, plastic (*hardcover*) and silicone (*softcover*). Note that these cover cases have no permanent magnet. Figure 18 illustrates there is no significant decrease in inference accuracy even when we put the cover cases on ZFlip3. Basically, MST sounds are generated inside a smartphone and propagate internally to built-in microphones, so external accessories, such as cover cases, rarely influence the generation and propagation of MST sounds.

Robustness against user behaviors. Some user behaviors in payment situations can adversely affect MagSnoop's performance. For example, when an MST payment service is activated on a smartphone, a grip on the smartphone (a *holding on hand* case) or touch inputs to the screen (a *typing inputs* case) may generate noisy sounds, which can affect the performance of MagSnoop. We then measured inference accuracy in such situations to evaluate MagSnoop's robustness against user behaviors. Figure 19 shows that inference accuracy slightly drops for the above two cases. In particular, when n^D is equal to 1, the accuracy drops by 8.4 and 14.5 percentage points,

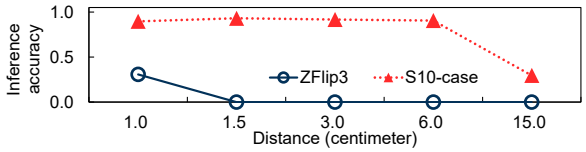


Figure 20: Inference accuracy in using external microphones.

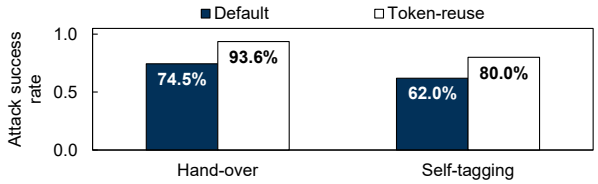


Figure 21: Inference success rate in two payment scenarios with real-world users.

in holding on hand and typing inputs cases, respectively. However, the reduced inference accuracy in both cases is still much higher than 65.0%, and the accuracy can be improved with more number of detected MST sounds. This implies that MagSnoop can provide acceptable performance even if user behaviors incur considerable noise.

7.1.3 Feasibility of leveraging external microphones. As mentioned in Section 2, MagSnoop also considers another eavesdropping attack that intercepts payment tokens using an external microphone. To verify the feasibility of this attack model, we conducted experiments that capture and infer tokens, generated by one device, via other microphone-equipped devices. Specifically, we used two devices, ZFlip3 and S10-case, for the generation of MST sounds and Google Pixel 4 for the token inference, while varying the distance between the devices. Figure 20 illustrates that the inference accuracy decreases as the external device is placed further away from the target devices. However, we observed that in the case of S10-case, the target payment token could be inferred with high accuracy of 89.7% or more up to 6.0 centimeters. This means that MagSnoop is likely to succeed in its attack just by placing the attacker's device near the victim device with Samsung Pay enabled. On the other hand, ZFlip3 shows low inference accuracy even at a distance of 1.0 centimeters. This is because the strength of the permanent magnet inside ZFlip3 is not high enough. In other words, the bias magnetic field formed by the magnet cannot amplify the MST sounds enough to catch them with the external microphone.

7.2 Real-world Deployment Tests

We evaluate *i*) the performance of MagSnoop and *ii*) the feasibility of our proposed attack model in the real world via a user study and an end-to-end attack time evaluation, respectively.

7.2.1 User study. We recruited 15 participants (11 males and 4 females, mean age 22.5), who had experience using Samsung Pay, through an online community posting. The user study was conducted at a fast-food restaurant in operation with the ambient noise of 55.0 dB on average. In the study, we asked the participants to order food from a cashier at the restaurant and make a mobile payment through Samsung Pay. At this time, the instructor took the role of the cashier,

and the participants used our experimental smartphone, ZFlip3, for the mobile payments. The device has our mobile application installed, which attempts to intercept each participant's payment token once he activates Samsung Pay. Note that we did not mention the existence of the malicious application to the participants at all.

In this experiment, we covered diverse scenarios by asking the participants to make payments in two different ways. One is a *hand-over* scenario in which a user hands his smartphone to the cashier after activating Samsung Pay. Then, the cashier puts the smartphone close to the POS terminal on behalf of the user. The other is a *self-tagging* scenario in which a user directly puts the smartphone close to the POS terminal. The participants repeated these two scenarios an average of 2.8 times each and they activated Samsung Pay at any time they wanted in each run. During this experiment, we used a commodity magnetic stripe reader as the POS terminal.

User study results. To verify the feasibility of MagSnoop, we use a metric, called *inference success rate*, which is defined as the accuracy to correctly infer a payment token using MST sounds detected before a user's payment is made. Figure 21 shows the average inference success rate of MagSnoop for the above two scenarios. Here, *default* indicates a general method of MagSnoop that performs each inference attempt independently, whereas *token-reuse* indicates an extended method of MagSnoop that reuses some parts of tokens obtained from the previous inference. As shown in Figure 21, default has high success rates of 74.5% and 68.0% for the hand-over and self-tagging scenarios, respectively, even with the ambient noise of 55.0 dB. Token-reuse further improves the success rates up to 93.6% and 80.0% for the two scenarios by reducing the number of token fields to be inferred. This means that MagSnoop can provide high accuracy and robustness even in the real world.

Also, this user study demonstrates that MagSnoop can intercept a payment token in a few seconds. In this experiment, we observed that MST sounds were emitted an average of 4.6 times, i.e., 6.9 seconds elapsed, until each participant completed the payment transaction. On the other hand, MagSnoop required only 2.4 MST sounds on average for successful inference. This implies that MagSnoop can take the payment token approximately 3.3 seconds earlier than the user's transaction is made. This time can further increase. In the post-survey of our user study, 80% of the participants (12 out of 15) responded that they usually activate an MST payment service in advance, e.g., when they wait in line or make an order, to avoid delays in payment. That is, MagSnoop can provide an attacker enough time to use an inferred token for fraudulent transactions.

7.2.2 End-to-end attack time evaluation. We verify how much time is required for an attacker to conclude his/her fraud transaction in the real world. Specifically, we assume the attack scenario consists of the following four steps: *i*) a victim user activates an MST payment transaction on her smartphone and an attacker's application installed on the device detects an MST sound (*detection*), *ii*) the application relays the MST sound to the attacker's server using LTE (*transmission*), *iii*) the server application recovers a payment token from the received sound and configures an MST replay device with the token information (*inference*), and *iv*) the attacker makes a payment transaction using the replay device and his own payment terminal (*transaction*). This attack scenario was repeated approximately 500 times with one victim user and one attacker using ZFlip3, MagSpoof [21], and a commodity magnetic reader as the victim's smartphone, the MST replay device, and the personal payment terminal, respectively. We then measured the time taken for each step. As shown in Table 3, the total time elapsed from when an MST sound is emitted to when the payment terminal receives any token is 1.47 seconds on average. In our user study, we observed that on average, users take approximately 7 seconds to complete their legitimate transaction. Conclusively, we can say that the adversary has sufficient time to attempt an attack even several times until it succeeds.

Table 3: End-to-end attack time taken for four major steps.

	Detection	Transmission	Inference	Transaction
Mean (ms)	722.2	32.2	243.0	475.3
Stdev. (ms)	396.1	7.7	25.5	36.8

8 DISCUSSION

Using other types of sensors. Motion sensors, such as accelerometers and magnetometers, have the potential to be used to infer MST payment tokens. For example, MST magnetic waves and vibrations induced by the waves can be captured using magnetometers and accelerometers, respectively. Yet, motion sensors built-in most commodity smartphones support too low sampling rate (e.g., below 500 Hz) to sense MST magnetic waves of several kHz. There is, however, an increasing need for high-performance motion sensors, especially to support advanced motion-based services, such as gesture-based input systems [23]. These demands can lead to the emergence of a new class of side-channel attacks against MST payment services.

Support for a new mobile payment. The capability of MagSnoop to infer MST payment tokens from sounds can encourage the emergence of a new class of mobile payment services where tokens are transferred from an MST-supported device to a microphone-equipped device. In other words, it allows commodity smartphones to act as magnetic stripe readers without additional hardware equipment, and introduces new interesting payment scenarios; food deliverymen can process MST payment requests through their smartphones without carrying a magnetic reader or external accessories. Thus, in our future work, we will further extend MagSnoop to support new payment methods.

Possible defenses. There are several security defenses to prevent our proposed MST token inference attack. One naive way is to generate artificial noise, which has similar frequency characteristics with MST sounds, when an MST payment transaction is requested. However, it may considerably degrade the usability of mobile payment users due to the audible, but irritating noise. Another simple, yet effective method is to disable audio recording during the activation of an MST payment transaction, but it may also compromise usability. Or, notifying recording status to users can be another solution. For instance, on devices running Android 12 or higher, a special indicator appears in the status bar when an application attempts to access the microphone. However, we observed that all participants were not aware of the recording indicator during using mobile payment services. Therefore, it might be required to give an explicit warning

message when background recording is detected in the middle of MST payment transactions. A more advanced method is to insert some context information (e.g., a POS terminal's ID, user location) into a payment token so that the token cannot be used in other places. In this case, even if a remote attacker intercepts the token, he cannot make a payment in a different place. Exploring the feasibility of these defense methods is our future work.

Limitations. There are some cases in which our method may fail to recover payment tokens. The first case is when the time from MST payment activation to completion is extremely short (e.g., about 1-2 seconds). Imagine a victim user orders food using a kiosk. She will activate an MST payment transaction after choosing a menu, i.e., right before a payment is made. Then, the attacker may fail to intercept the payment token due to the lack of chances to collect MST sounds. According to the post-survey of our user study, we observed that 60% of participants (9 out of 15) tend to run an MST payment service right before payment when using a kiosk system. They answered that it is not easy to run the service in advance because they should order some food by touching the kiosk screen. The second is when the bias magnetic fields around a smartphone are too weak or absent. In this case, MagSnoop may not detect MST sounds because they are not amplified sufficiently. However, we expect that such bias magnetic fields will get stronger in near future, considering the current boom in foldable devices and external accessories that include permanent magnets.

Report on the MagSnoop attack. We recently informed Samsung Electronics of the Samsung Pay vulnerability we discovered, along with a demonstration video. Furthermore, we plan to cooperate with Samsung Electronics to help them address the security threat by exploring the defense methods discussed above.

9 RELATED WORKS

Attacks on MST payment services. Several studies propose attack methods against MST payment services, i.e., Samsung Pay. Some works [6, 8, 9] aim to eavesdrop on the MST channel between a mobile device and a POS terminal to intercept a payment token. They utilize special magnetic receivers that can collect magnetic waves emitted from MST-supported smartphones. However, applying these methods in the real world is impractical for the following reasons; they require an attacker *i)* to be located close to a victim at all times, and *ii)* to carry too large magnetic receivers (e.g., a pizza-box-sized coil). In contrast, the capability of MagSnoop to infer payment tokens from listening to sounds makes it possible for an attacker to get the tokens remotely and without any special hardware.

In addition, another study [26] proposes a forgery attack against Samsung Pay payment tokens. It shows that it is possible to analyze the token generation pattern of Samsung Pay and counterfeit fake tokens by mimicking it. However, this attack method is currently blocked by the security enhancement of Samsung Pay, which generates tokens via a proprietary randomization algorithm [29]. In contrast, MagSnoop is a kind of side-channel attack that can completely bypass the security measure.

Eavesdropping on other kinds of payment systems. Besides MST channel, many studies focus on eavesdropping attacks on other payment systems which leverage different communication channels. One study [6] targets a POS system that employs audio channels such as Alipay [2] and ToneTag [35]. Because these payment systems transfer their tokens through already known frequency bands, attackers can easily intercept the tokens by just recording sounds for the corresponding bands without any signal processing. However, MST sounds, the attack vector of this paper, have much more complicated characteristics, such that their dominant frequencies can vary from device to device. Thus, it is required to design sophisticated signal processing techniques as proposed in this work. Moreover, there are also studies of eavesdropping on POS terminals that use NFC [1, 10, 11, 25] and RFID [12, 14]. However, because they have different physical properties from sound, it is impossible to apply the proposed techniques to our problem.

Acoustic-based inference attack techniques. Similar to MagSnoop, several studies have been proposed to infer a user private information by sensing acoustic leaks from various physical devices. Some works [27, 33] aim to infer a victim's tap inputs by analyzing tap sounds emanating from a touchscreen. Keynergy [28] infers the bitting pattern of a physical key by using audible click sounds captured during a victim's key insertion. Besides these, there are also acoustic-based attacks against several types of devices, such as keyboards [5, 37], computer screens [15], 3D printers [19, 34], and DNA synthesis machines [13]. However, unlike the above studies, MagSnoop utilizes MST sounds induced by magnetic waves. Thus, it is necessary to design its inference technique based on a deep understanding of the MST sound's physical characteristics.

10 CONCLUSION

This paper explored the feasibility of inferring payment tokens used for MST payment transactions. Specifically, we presented the design and implementation of MagSnoop that supports accurate, robust, applicable, and data-efficient token inference from listening to MST sounds. We first extensively studied the fundamental characteristics of MST sounds and MST payment services. Based on these observations, we developed noise-robust MST sound detection, ASE feature extraction, and token recovery/validation techniques. Through the evaluation with a prototype implementation of MagSnoop, we demonstrated that it can infer payment tokens from MST sounds accurately in diverse payment environments, e.g., in the presence of ambient noise, on any type of smartphone, and with an insufficient number of available MST sounds. Our experiment in the real world also showed that MagSnoop can raise severe security concerns for real-world mobile payment users. We believe that MagSnoop can also bring up a new mobile communication channel between MST-supported devices and microphone-equipped devices and this extension could be our future work.

ACKNOWLEDGMENTS

We thank the anonymous reviewers and the shepherd, Inseok Hwang, for their valuable comments. This work was supported in part by the National Research Foundation of Korea (NRF-2022R1C1C1012664 and NRF-2021R1F1A1063785), Korea Institute for Advancement of Technology (KIAT) grant funded by the Korea Government (MOTIE) (P0012724, The Competency Development Program for Industry Specialist), and the Institute of Information & Communications Technology Planning & Evaluation (IITP-2021-2018-0-01431).

REFERENCES

[1] N. Akinyokun and V. Teague. Security and privacy implications of NFC-enabled contactless payment systems. In *Proceedings of the International Conference on Availability, Reliability and Security*, 2017.

[2] Alipay. Alipay: Trust makes it simple. https://intl.alipay.com/. Retrieved: December 20, 2021.

[3] S. A. Anand, C. Wang, J. Liu, N. Saxena, and Y. Chen. Spearphone: A lightweight speech privacy exploit via accelerometer-sensed reverberations from smartphone loudspeakers. In *Proceedings of the ACM Conference on Security and Privacy in Wireless and Mobile Networks*, 2021.

[4] Apple Inc. Apple pay - apple. https://www.apple.com/apple-pay/. Retrieved: December 20, 2021.

[5] D. Asonov and R. Agrawal. Keyboard acoustic emanations. In *Proceedings of IEEE Symposium on Security and Privacy*, 2004.

[6] X. Bai, Z. Zhou, X. Wang, Z. Li, X. Mi, N. Zhang, T. Li, S.-M. Hu, and K. Zhang. Picking up my tab: Understanding and mitigating synchronized token lifting and spending in mobile payment. In *Proceedings of USENIX Security Symposium*, 2017.

[7] S. Butterworth et al. On the theory of filter amplifiers. *Wireless Engineer*, 7(6):536–541, 1930.

[8] D. Choi and Y. Lee. Eavesdropping one-time tokens over magnetic secure transmission in samsung pay. In *Proceedings of the USENIX Workshop on Offensive Technologies*, 2016.

[9] D. Choi and Y. Lee. Eavesdropping of magnetic secure transmission signals and its security implications for a mobile payment protocol. *IEEE Access*, 6:42687–42701, 2018.

[10] S. Drimer and S. J. Murdoch. Keep your enemies close: distance bounding against smartcard relay attacks. In *Proceedings of the USENIX Security Symposium*, 2007.

[11] M. Emms and A. Van Moorsel. Practical attack on contactless payment cards. In *HCI2011 Workshop-Heath, Wealth and Identity Theft*, 2011.

[12] M. Engelhardt, F. Pfeiffer, K. Finkenzeller, and E. Biebl. Extending ISO/IEC 14443 type a eavesdropping range using higher harmonics. In *Smart SysTech 2013; European Conference on Smart Objects, Systems and Technologies*, 2013.

[13] S. Faezi, S. R. Chhetri, A. V. Malawade, J. C. Chaput, W. Grover, P. Brisk, and M. A. Al Faruque. Oligo-snoop: A non-invasive side channel attack against dna synthesis machines. In *Proceedings of the Network and Distributed Systems Security Symposium*, 2019.

[14] K. Finkenzeller, F. Pfeiffer, and E. Biebl. Range extension of an ISO/IEC 14443 type A rfid system with actively emulating load modulation. In *Proceeding of the 7th RFID SysTech European Workshop on Smart Objects: Systems, Technologies and Applications*, 2011.

[15] D. Genkin, M. Pattani, R. Schuster, and E. Tromer. Synesthesia: Detecting screen content via remote acoustic side channels. In *Proceedings of the 2019 IEEE Symposium on Security and Privacy*, 2019.

[16] Google Inc. Google pay. https://pay.google.com/. Retrieved: December 20, 2021.

[17] D. J. Griffiths. Introduction to electrodynamics, 2005.

[18] M. Hirao and H. Ogi. *EMATs for science and industry: noncontacting ultrasonic measurements*. Springer Science & Business Media, 2003.

[19] A. Hojjati, A. Adhikari, K. Struckmann, E. Chou, T. N. Tho Nguyen, K. Madan, M. S. Winslett, C. A. Gunter, and W. P. King. Leave your phone at the door: Side channels that reveal factory floor secrets. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, 2016.

[20] International Organization for Standardization. ISO/IEC 7811-2:2018 Identification cards — recording technique — part 2: Magnetic stripe: Low coercivity.

[21] S. Kamka. Magspoof. https://samy.pl/magspoof/. Retrieved: December 20, 2021.

[22] H. Khan, U. Hengartner, and D. Vogel. Targeted mimicry attacks on touch input based implicit authentication schemes. In *Proceedings of the Annual International Conference on Mobile Systems, Applications, and Services*, 2016.

[23] G. Laput, R. Xiao, and C. Harrison. Viband: High-fidelity bio-acoustic sensing using commodity smartwatch accelerometers. In *Proceedings of the Annual Symposium on User Interface Software and Technology*, 2016.

[24] E. W. Lee. Magnetostriction and magnetomechanical effects. *Reports on Progress in Physics*, 18(1):184–229, 1955.

[25] K. M. Lishoy Francis, Gerhard Hancke and K. Markantonakis. Practical relay attack on contactless transactions by using nfc mobile phones. In *Radio Frequency Identification System Security*, 2012.

[26] S. Mendoza. Samsung pay: Tokenized numbers, flaws and issues. In *Proceedings of the Black Hat USA*, 2016.

[27] S. Narain, A. Sanatinia, and G. Noubir. Single-stroke language-agnostic keylogging using stereo-microphones and domain specific machine learning. In *Proceedings of the ACM Conference on Security and Privacy in Wireless & Mobile Networks*, 2014.

[28] S. Ramesh, R. Xiao, A. Maiti, J. T. Lee, H. Ramprasad, A. Kumar, M. Jadliwala, and J. Han. Acoustics to the rescue: Physical key inference attack revisited. In *Proceedings of the USENIX Security Symposium*, 2021.

[29] Samsung Electronics. Commonly asked questions about samsung pay. https://www.samsung.com/us/support/answer/ANS00080583/. Retrieved: December 20, 2021.

[30] Samsung Electronics. Samsung pay | apps - the official samsung galaxy site. https://www.samsung.com/global/galaxy/samsung-pay/. Retrieved: December 20, 2021.

[31] Samsung Electronics. What is mst? https://www.samsung.com/global/galaxy/what-is/mst/. Retrieved: May 15, 2022.

[32] R. Schlegel, K. Zhang, X. Zhou, M. Intwala, A. Kapadia, and X. Wang. Soundcomber: A stealthy and context-aware sound trojan for smartphones. In *Proceedings of the Network and Distributed System Security Symposium*, 2011.

[33] I. Shumailov, L. Simon, J. Yan, and R. Anderson. Hearing your touch: A new acoustic side channel on smartphones. *CoRR*, abs/1903.11137, 2019.

[34] C. Song, F. Lin, Z. Ba, K. Ren, C. Zhou, and W. Xu. My smartphone knows what you print: Exploring smartphone-based side-channel attacks against 3d printers. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, 2016.

[35] ToneTag. Tonetag: Data over sound and sound wave technology company. https://www.tonetag.com/. Retrieved: December 20, 2021.

[36] G. Wallner. System and method for a baseband nearfield magnetic stripe data transmitter, Aug 2014. US Patent No. 8,814,046.

[37] L. Zhuang, F. Zhou, and J. D. Tygar. Keyboard acoustic emanations revisited. *ACM transactions on information and system security*, 13(1):1–26, 2009.