Zero-Permission Acoustic Cross-Device Tracking

Nikolay Matyunin
Technical University of Darmstadt, CYSEC
Darmstadt, Germany
matyunin@seceng.informatik.tu-darmstadt.de

Jakub Szefer
Yale University
New Haven, CT, USA
jakub.szefer@yale.edu

Stefan Katzenbeisser
Technical University of Darmstadt, CYSEC
Darmstadt, Germany
katzenbeisser@seceng.informatik.tu-darmstadt.de

***Abstract*—Adversaries today can embed tracking identifiers into ultrasonic sound and covertly transmit them between devices without users realizing that this is happening. To prevent such emerging privacy risks, mobile applications now require a request for an explicit user permission, at run-time, to get access to a device's microphone. In this paper, however, we show that current defenses are not enough. We introduce a novel approach to acoustic cross-device tracking, which does not require microphone access, but instead exploits the susceptibility of MEMS gyroscopes to acoustic vibrations at specific (ultrasonic) frequencies. Currently, no permissions are needed to access the gyroscope's data, and the gyroscope can be accessed from apps or even from a web browser. In this manner, gyroscopes in modern smartphones and smartwatches can be used as zero-permission receivers of ultrasonic signals, making cross-device tracking completely unnoticeable to users. We evaluate our approach on several mobile devices using different audio hardware, achieving 10–20bit/s transmission bandwidth at distances from 35cm to 16m in realistic attack scenarios. Finally, we discuss potential countermeasures against the presented attack.**

***Index Terms*—cross-device tracking, ultrasonic communication, covert channels, MEMS gyroscope, web tracking**

I. Introduction

Commercial companies today collect an increasing amount of information about their users, to improve customer experience, but also to increase financial profits by showing targeted advertisements. Advertising components can be embedded as third-party content on hundreds of websites or TV streams, making it possible to analyze user activity on a large scale. Moreover, the widespread use of mobile and wearable devices has resulted in the demand for *cross-device* tracking technologies, which allow companies to correlate user activities even across different devices. This introduces a serious privacy threat, since such aggregated user profiles may contain sensitive information about personal interests, location, health, beliefs or sexuality, while users remain unaware of the scope and mechanisms of such tracking [1], [2].

Usually cross-device tracking is performed by linking the device to some deterministic information provided by users themselves, e.g., application or website login credentials [3], or by comparing attributes shared by all the devices, such as IP addresses or location data. Recently, ultrasonic cross-device tracking (uXDT) has emerged, based on embedding tracking identifiers into ultrasonic sounds and detecting them with a microphone on a user's smartphone. In particular, companies like Shopkick, Lisnr and Signal360 provide a way to deploy ultrasonic beacons at specific locations (e.g., shops or festivals) and detect them in a mobile application to show location-relevant content. The company Silverpush developed means of embedding tracking identifiers into TV streams. Meanwhile, researchers [4] demonstrated how the uXDT technology can be used for web-tracking purposes, e.g., for transmitting a tracking ID from the Tor browser to an application on the user's smartphone in order to de-anonymize the web session.

Due to privacy concerns, uXDT technology raised the attention of public media [5] and the security community [4], [6]. In response, the Federal Trade Commission issued warning letters to app developers who use Silverpush components, asking them to explicitly disclose the usage of ultrasonic tracking. Nevertheless, researchers recently found 234 Android applications that are listening in the background for ultrasonic beacons from TV streams [6], some of them with millions of users, proving that the technology is being actively deployed.

Starting from Android 6, mobile applications are required to ask for a user permission at run-time to access the microphone (in the past it was only done during app installation). This way, Android devices prevent stealthy audio recording. On iOS devices, a pop-up warning is additionally shown when the app accesses the microphone in the background. This way, an attempt to start detection of uXDT signals by the app will most likely raise user's attention.

In this paper, we present a new approach to perform uXDT, which does not require access to a microphone at all, and instead uses gyroscopes in smartphones or smartwatches as receivers for ultrasonic signals. It has been shown that microelectromechanical (MEMS) gyroscopes are susceptible to acoustic vibrations at specific *resonance* frequencies [7], [8], typically within ultrasonic range (19–29kHz). In our work, we show that ultrasonic signals can be emitted at these frequencies with commonly-used audio hardware, and subsequently be captured at a distance by gyroscopes of modern smartphones and smartwatches. By analyzing spectral characteristics of gyroscope's response to sound, the signal can be decoded even in the presence of device movements, e.g., when a smartphone is held in a hand, or a smartwatch is worn on a wrist.

We show that cross-device tracking can be established between commonly-used devices (e.g., a laptop and a smartphone) at distances of up to 35cm using internal laptop speakers at 75% volume level, achieving a bitrate of 10bit/s. With a more powerful speaker, distances of several meters and a bitrate of up to 20bit/s are achieved. Although distance and bandwidth are limited in comparison to existing uXDT solutions with recording using microphones, the proposed method can be run completely stealthily to users, as access to gyroscope data does not require any explicit permissions. Adding such a permission, as well as applying other possible countermeasures (as discussed in Section VI), introduces technical and usability problems and may not completely eliminate this new attack vector. Therefore, our work demonstrates that even with the hardened permission model on mobile devices, uXDT technologies still pose a significant privacy risk.

A. Paper contributions

Our contributions can be summarized as follows:

- We introduce a novel method of cross-device tracking, exploiting the sensitivity of gyroscopes in mobile devices to acoustic vibrations.
- We present an implementation of the method with an encoding scheme that is robust to background noise and natural device movements, and demonstrate its applicability to cross-device tracking scenarios, including web tracking, TV media tracking and location tracking.
- We evaluate the implementation on different hardware setups. To the best of our knowledge, our work is the first to consider and evaluate acoustic transmission using MEMS gyroscopes as receivers *at a distance*, when sensor is not directly adjacent to the source.

II. Background

In this section, we briefly describe use of MEMS gyroscopes in mobile devices, their susceptibility to acoustic vibrations, and abilities of modern audio hardware to transmit covert acoustic signals.

A. MEMS gyroscopes in mobile devices

Most modern smartphones and smartwatches are equipped with gyroscopes, which measure the rotation rate of the device in radians per second around three physical axes, to estimate its orientation in space. The acquired data is used in games, virtual reality applications, etc. In Android and iOS devices, 3-axis gyroscope values are retrieved by using the Sensor API [9] and the Core Motion [10] framework, respectively. The sampling rate of the measurements depends on the sensor, and is additionally limited by the operating system to reduce power consumption. On all the tested devices, we were able to access gyroscope data with sampling rates of 60–500 Hz.

Usually gyroscopes consist of one or several sensing masses, constantly vibrating at a specific *resonance frequency*. When the gyroscope is rotated, the Coriolis force is applied to the sensing mass orthogonally to its vibration direction and the rotation axis, with an amplitude proportional to the rotation rate. A detailed explanation of MEMS gyroscope design can be found in [11].

More importantly for our work, it is known that MEMS-based gyroscopes are susceptible to acoustic signals at frequencies close to the resonance frequency [12], [13]. The acoustic waves cause the sensing mass to additionally vibrate on the axes corresponding to the Coriolis force direction, and disturb the resulting measurements on one or all of the axes. For example, Figure 1 demonstrates gyroscope measurements from a stationary iPhone 6s placed near a speaker. Once 80dB sound is played at gyroscope's resonance frequency, a clear disturbance in gyroscope measurements is noticeable.

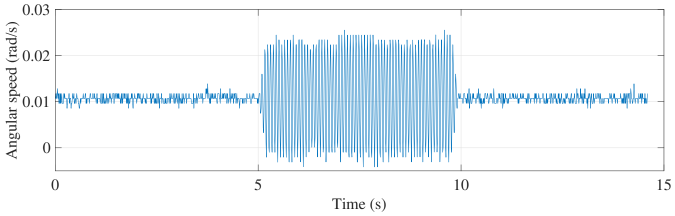

Fig. 1. Example of acoustic disturbance of gyroscope measurements over the x-axis, due to playing a 27kHz sine wave (5–10s). Measurements are recorded on an iPhone 6s in a stationary position, located 30cm from the speaker.

To limit acoustic disturbance from background noise, manufacturers design the sensors to have a resonance frequency in ultrasonic range. Most of gyroscopes tested in other works [7], [8] and in our experiments (Section IV-A), had a resonance frequency within 19–29 kHz. This fact makes gyroscopes suitable receivers for covert ultrasonic signals: humans can perceive sounds only within the 20 Hz–20 kHz range, with the upper threshold declining with age, so most people over 18 years cannot hear frequencies above 16 kHz [14].

B. Capabilities of commodity audio hardware

In order to perform uXDT by disturbing gyroscopes at resonance frequencies, the transmitter should be able to produce ultrasonic signals at 19–29 kHz. According to the Nyquist sampling theorem, the highest possible frequency of the digital signal to be reproduced without aliasing should not be more than half of the sampling rate. Sound interfaces of computers and mobile devices typically support at least 44.1 kHz, and most of them even a 48 kHz sampling rate, to comply with popular audio codecs. Moreover, many digital-to-analog converters (DAC) and speakers in modern computers, home theater systems and smartphones have 96 kHz or even 192 kHz sampling rates to support *high-resolution* audio [15], [16]. Therefore, ultrasonic signals of up to 24 kHz can natively be generated by most commodity audio hardware, and many consumer devices are able to reproduce sounds of higher frequencies, covering the aforementioned 19–29 kHz range.

III. Methodology

In this section, we describe the considered tracking scenarios, present our encoding scheme and techniques used to detect and decode the cross-device tracking signals.

A. Tracking scenarios

We consider cross-device tracking scenarios with two devices and an adversary, who is trying to link information about the victim user or their activity on one device (transmitter) with user activity or profile on another device (receiver), by transmitting a unique tracking identifier between them.

TABLE I
SETUP OF DIFFERENT TRACKING SCENARIOS.

	Web tracking	TV tracking	Location tracking
Distance	short (10–50cm)	medium (0.5–3m)	long (>1m)
Speaker quality	low	medium	high
Sound level[a]	low ($\approx$60–70dB)	medium ($\approx$70–85dB)	high ($\geq$85dB)
Transmitter device belongs to	victim	victim	attacker
Special requirements	in-browser implementation	background noise	device movements

[a] measured near the source

The transmitter is assumed to be equipped with a non-muted speaker, while any gyroscope-equipped smartphone or a smartwatch is considered as the receiver.

The transmitter encodes the tracking identifier into ultrasonic sounds and plays them through the speaker. We assume that the attacker has control over an application or a webpage on the receiver, which records gyroscope data, captures transmitted signals and decodes the ID. This malicious application or a web page does not require any user permissions, unlike ultrasonic tracking implementations which rely on access to the microphone. Therefore, in our case code can be hidden in any application which the user is likely to install, or can be embedded on any webpage. To successfully capture ultrasonic signals, the receiver device is assumed to be naturally located near the transmitter. The actual achievable distance is evaluated in Section IV-C for different use cases.

Following existing research works [4], [6] and commercial implementations (Shopkick, Lisnr, Signal360 and Silverpush) with microphones as receivers, we consider three real-world applications of uXDT, summarized in the Table I.

1) Web tracking: In this scenario, the victim visits a web page, which aims to track or de-anonymize the browsing session. We assume that the web session is protected from traditional tracking mechanisms, such as tracking cookies or browser fingerprinting (e.g., by using private browsing modes, disallowing cookies, etc.). We further assume that the victim has a malicious application on another device (a smartphone or a smartwatch), placed nearby, e.g., on the same working desk. Then the attacker can transmit the tracking ID between the devices and link it to a concrete user. Furthermore, instead of requiring an installed application, it is enough to have another attacker-controlled web page opened on user's smartphone. Both transmitting and receiving web pages either belong to the attacker, or only contain attacker-controlled components, similarly to a technique of embedding third-party advertisements or analytics components. Therefore, tracking code can potentially appear on thousands of websites, which increases the scale of the attack.

For this scenario, the transmitter is assumed to have only low-quality speakers (e.g., internal speakers of laptops and smartphones) and a comparably low volume level. In this setup, we believe that even a short distance between devices (10–50cm) is practical to make cross-device tracking approach applicable. We evaluate the transmission distance of our approach in Section IV-C, and investigate how transmitter and receiver can be implemented on web pages in Section IV-E.

2) TV tracking: In this scenario, the adversarial TV media provider embeds ultrasonic beacons with encoded tracking IDs into broadcasted TV content. By capturing these IDs with an application installed on user's mobile device placed nearby, the adversary can track what and when users watch.

In comparison to the web-tracking scenario, in this scenario we assume that TV systems contain higher-quality audio hardware, and volume level is usually higher, around 75 dB at a source [17]. However, in this case we must take into account that ultrasonic signals will not be played individually, but rather embedded into existing TV audio content. In Section IV-F, we evaluate how robust is the transmission in presence of background noise.

3) Location tracking: In this scenario, the attacker places ultrasonic beacons at specific locations (e.g., in shops) and captures tracking IDs by a malicious application on victim's smartphone. This way, the captured ID reveals the user location. Unlike other considered scenarios, in this case the transmitter is fully under attacker's control. Therefore, we assume the transmitter to have high-quality audio hardware, and the signal to be emitted at a maximum possible volume. However, in this case the signal must be captured at higher distances (at least 1–3m), and a receiver is unlikely to remain static, due to movements of the device. We evaluate robustness of our solution against natural device movements in Section IV-G.

B. Signal modulation and transmission

The easiest way to encode tracking IDs into ultrasonic signals is to apply on-off keying (OOK) modulation: generate and play a sine wave at the resonance frequency to encode a 1, and produce no sound to encode a 0. By observing resulting gyroscope disturbances within time frames, the binary data can be decoded. In particular, this encoding has been applied to use MEMS sensors as receivers for ultrasonic covert channels at zero distance [8] and within the same device [18].

In this work, we propose a more advanced modulation scheme, which allows us to transmit binary data at a larger distance, potentially at higher rates, and apply the solution to practical scenarios. Our method is based on the fact that the resonant signal causes the gyroscope sensing mass to *vibrate*, i.e., a strong signal power becomes noticeable at a specific frequency (further referred to as *resulting frequency*) in gyroscope measurements. Moreover, we discovered that sounds played at frequencies slightly different from the resonant one (±10 Hz), subsequently cause different values of resulting frequencies in gyroscope measurements. Figure 2 (a) shows recorded gyroscope measurements for a stationary smartphone located near the speaker, after playing consecutively four 2-second sine waves with 5 Hz step, starting from a resonance frequency, together with a signal spectrogram. The resulting frequency components remain noticeable even for disturbance

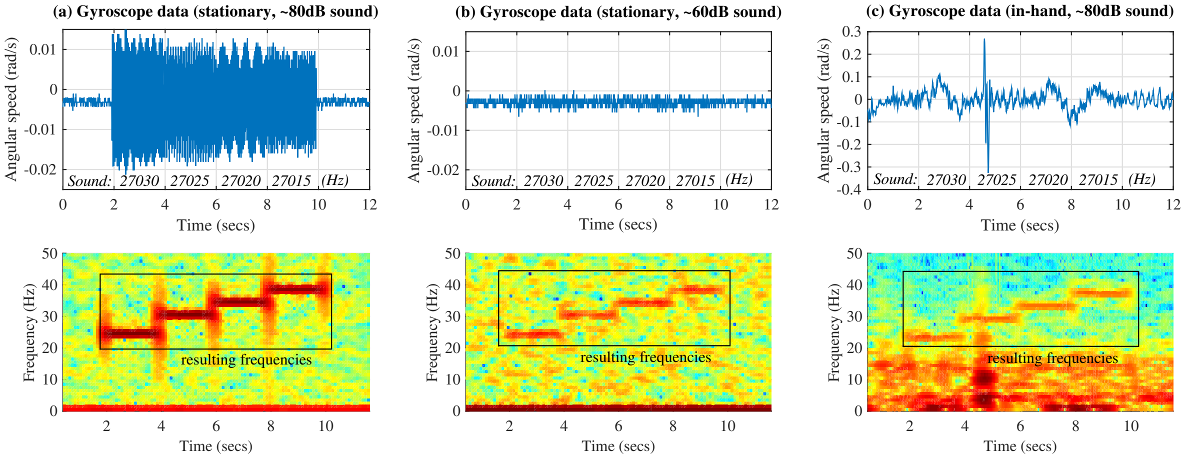

Fig. 2. Example of gyroscope disturbance: gyroscope measurements recorded while playing 4 sine waves of near-resonance frequencies to an iPhone 6s in a stationary position within 50cm from a speaker at 80dB (a), at 60dB (b), and when it is held in hand (c), together with corresponding spectrograms.

caused by signals of lower sound pressure level, when actual disturbance of measurements becomes indistinguishable from a background noise (Figure 2 (b)), or in presence of disturbances caused by device movements, e.g., when the device is held in hand (Figure 2 (c)). Therefore, in our implementation we utilize several frequencies around the resonance frequency, and observe the spectral characteristics of the signal.

More specifically, to transmit binary data, we empirically choose N frequencies close to the resonance frequency (further referred to as *transmitting frequencies*), and apply a Multiple Frequency Shift Keying (MFSK) modulation scheme: each of N frequencies is associated with a binary sequence of $log_2 N$ bits, so that a sine wave of a particular frequency, played within a time frame, encodes the represented sequence. The actual number of N is limited by the fact that the resulting disturbance of gyroscope measurements (and subsequent frequency power) gradually degrades when the frequency of sound is changing starting from the resonance frequency. One can observe such attenuation in Figure 2 (a). Moreover, for some of the transmitting frequencies we observed additional distortions and aliasing in lower frequencies of the received gyroscope signal, which reduced the range of potential frequencies to be used for transmission. In our experiments, we successfully utilized 4 frequencies, which allowed us to double the bitrate in comparison to OOK modulation.

Speakers may produce hearable audio clicks at the beginning and the end of ultrasonic transmission, due to abrupt changes of the amplitude [19]. To prevent them, a Hann [20] window is applied at the beginning and the end of each generated sine wave. Additionally, in order to help the transmitter recognize the start of the transmission, each signal is prepended with a sequence of several short sine waves played at the resonance frequency with a small pause between them. In our experiments, we used three waves of 250ms each.

To decode the signal, the receiver first applies the short-time Fourier transform (STFT) for the resulting frequency with a window size equal to the duration of a single wave in the preamble. A signal start is detected in resulting STFT values by looking for a peak/no-peak sequence corresponding to the signal preamble. Then, the signal is decoded bit by bit. Within each time frame, a Fast Fourier transform (FFT) is calculated for all N resulting frequencies (corresponding to chosen transmitting frequencies). The binary sequence corresponding to a resulting frequency with the highest FFT-magnitude is chosen as transmitted within the time frame.

In practice, the transmitter will not know the resonance frequency of the receiver's gyroscope. To target multiple devices, the binary ID must be modulated into several sound waves, corresponding to different resonance frequencies. Then these sounds can be either played subsequently, or combined into one signal with equal weights. For simplicity, we present evaluation results considering one sine wave at a time.

IV. Evaluation

In this section, we first examine various mobile devices and evaluate the resonance frequencies of their gyroscopes. Then, for two smartphones, we test how the amplitude of the signal depends on the sound pressure level (SPL). Afterwards, we evaluate the proposed encoding scheme, by showing how SPL affects the bit error rate (BER), determine a transmission bitrate, and demonstrate the achievable distance. Finally, we evaluate our approach when signals are transmitted between web pages (web tracking), how robust is it against background noise (TV tracking) and device movements (location tracking).

A. Resonance Frequencies

For our experiments, we chose 4 modern smartphones and a smartwatch. To detect their resonance frequencies, we placed the devices directly near a speaker, and generated sine waves at frequencies from 18 kHz to 30 kHz with 20 Hz increments. For each played sound, we calculated the average magnitude of the resulting frequency, and chose the sound frequency which caused the strongest signal. In all our experiments (unless stated otherwise), we used a single external speaker KRK Rokit 5 G3 connected to a Macbook Pro A1502 laptop,

TABLE II

RESONANCE FREQUENCY OF MOBILE DEVICES.

Device	Gyroscope Model	Sampling Rate	Resonance Frequency
Samsung Galaxy S7[a]	STM[b] LSM6DS3	500 Hz	20.20 kHz
Samsung Galaxy S8[a]	STM[b] LSM6DI	500 Hz	20.92 kHz
iPhone 6s	IS[b] MP67B	100 Hz	27.02 kHz
LG Nexus 4	IS[b] MPU6050	200 Hz	26.90 kHz
Sony Smartwatch SWR50	Bosch BMX055	200 Hz	25.48 kHz

[a] Two devices of the same model were tested to prove identical behavior

[b] IS: InvenSense; STM: STMicroelectronics

configured to output sounds with 96 kHz sampling rate. All measurements were taken in a typical office environment.

Table II lists the tested devices, their gyroscopes, available sampling rates, and discovered resonance frequencies. All the devices in our experiment had gyroscopes susceptible to ultrasonic sounds. Especially the STMicroelectronics sensors were susceptible to 20.9–21.4 kHz sounds, which can be generated even with 44 kHz audio hardware. Although other researchers discovered the gyroscopes which do not resonate in the ultrasonic range [7], we believe that the attack is practical, since very popular devices (e.g., modern Apple and Samsung smartphones) *are* affected. For further experiments, we chose two smartphones with the highest noticeable disturbance caused by sounds, namely Samsung Galaxy S7 and iPhone 6S.

B. Signal to Noise Ratio

In this experiment, we estimated the dependency of the signal strength on the SPL, independently from the payload and used encoding scheme. For this purpose, we gradually reduced speaker volume from the maximum (by ≈5dB) and, at each volume level, played a sine wave at the smartphone's resonance frequency. We recorded the produced sounds using a microphone and calculated the resulting SPL, while recording the gyroscope data on the smartphone. To precisely measure SPL for high-frequency sounds, we used an Earthworks QTC30 microphone connected to a Fireface UC sound card, with flat frequency responses of up to 30kHz. Then, we calculated the Signal to Noise Ratio (SNR), i.e., the ratio between the average magnitude of the resulting frequency in FFT values for gyroscope measurements when the sound is played, and without any sound produced. For comparison, we also calculated SNR as the ratio between raw disturbances, by calculating the standard deviation of the measurements.

Figure 3 shows the resulting SNR levels. One can see that the resulting SNR is stronger for high SPL (80–90dB), and the signal is slightly stronger for the iPhone 6S. For both smartphones the SNR rapidly attenuates and becomes comparable to noise at SPL lower than 60dB. Nevertheless, one can see that the signal based on spectral characteristics is stronger in comparison to the signal based on raw disturbance.

C. Transmission

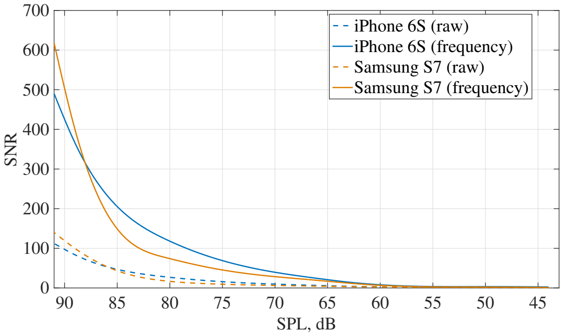

Fig. 3. SNR levels recorded for two smartphones located in 50cm from a speaker, depending on the SPL, measured at the same point. A signal based on a magnitude of the resulting frequency (solid) is compared to a signal based on raw disturbance of the measurements (dashed).

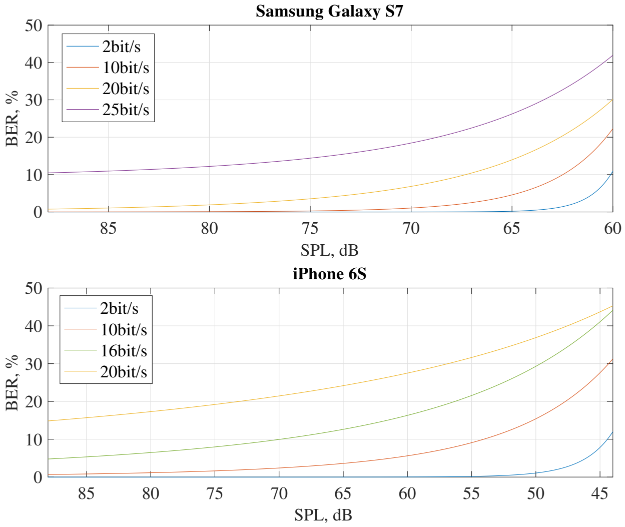

Fig. 4. Dependency of BER on the SPL. Smartphones are placed in 30cm from the speaker, the sound is recorded with a measurement microphone at the same point. Average of 50 transmissions for each bitrate and SPL.

In this experiment, we evaluated data transmission using the proposed MFSK encoding scheme. For this purpose, we used the same experimental setup as in the previous experiment, and transmitted tracking IDs with different bitrates and volume levels (50 IDs of a length of 30bits in each case), and measured how SPL and the chosen bitrate affect the resulting bit error rate (BER). We consider a BER of <10% as practically suitable, since in this case error-correcting codes can be used to ensure correct decoding with manageable overhead. For example, a primitive narrow-sense BCH(31,16) code [21] can be applied to correct 3 bits of a 31-bit code (in a channel with BER of up to 9.7%), with a payload block of 16-bit.

Figure 4 shows the resulting BERs for two smartphones. One can see that results comply with the SNR experiment, with decoding errors appearing when the raw signal becomes weak. The iPhone 6S, which has a higher SNR, performs better for lower SPL values (45–60dB). The transmission bandwidth is limited by the measurement sampling rate: at some point (20bit/s for the iPhone with 100Hz sampling rate, and 25bit/s for the Samsung with 500Hz sampling rate), the number of samples within a recording time frame becomes too small to correctly identify the resulting frequency, even with a high-level SPL. Based on the results, we consider bitrates of 10bit/s for low SPLs (>62dB) and 16–20bit/s for high SPLs (>68dB) as practically suitable for corresponding tracking scenarios.

D. Distance

To theoretically estimate the maximum possible distance for the transmission, we used the dependency between SPL and distance from the source, known as *inverse square law* [22]: given the reference SPL$_{ref}$ measured at a distance d_{ref}, the SPL depends on a distance d as

$$SPL = SPL_{ref} - 20\log(d/d_{ref}),$$
$$d = d_{ref} * 10^{|SPL_{ref} - SPL|/20}.$$

Given the last formula and boundary SPL values found in the previous experiment (62dB for 10bit/s, 68dB for 16–20bit/s), we could estimate the transmission distance depending on a reference SPL. In practice, the actual distance can be also affected by reflections, reverberations, interference, the direction of the sound wave propagation, etc. Table III shows the estimated and practically confirmed distances d for the reference SPLs of 70dB, 80dB, and 95dB (maximum for our speaker) at $d_{ref} = 50\text{cm}$. All measurements were taken on a Samsung Galaxy S7 smartphone, in a typical office room (up to 3.5m) and in the office corridor (>3.5m). As one can see from the table, our practical results comply with theoretical estimation. When the source SPL is high (≥80dB at 0.5m), a distance of 3.5–16m is achieved, which proves applicability of our approach to TV and location tracking scenarios.

To precisely evaluate the distance when the source SPL is low and the transmitter is equipped with comparably low-quality speakers (e.g., in the web-tracking scenario), we transmitted tracking IDs on a laptop using only internal speakers, and measured the area around the laptop, where the signal was successfully decoded (BER<10%), depending on a system volume level. The results are shown in Figure 5. The resulting distance is limited to 45cm, but even this area can be sufficient to transmit a tracking ID to a smartphone naturally located near the laptop on a working desk. We consider a volume level of 75% and corresponding distance of up to 35cm as realistic.

E. Web tracking: in-browser implementation

To confirm applicability of the approach to the web-tracking scenario, we implemented web-based versions of both transmitter and receiver. We used the Web Audio API to play sounds on a web page, and confirmed that it works in desktop and mobile versions of modern browsers (Chrome, Firefox, Tor Browser, and Safari). We must note that the transmission is not completely covert to users: in all *desktop* browsers, a small notification icon appears when the tab is playing sound. Moreover, some browsers prevent automatic playback of audio without an explicit user interaction. Nevertheless, the signal can be embedded into benign audio or video content, not being suspicious to users. We also discovered that Safari does not show the notification icon if sound is played for less than 1.5s, enabling completely covert transmission in a short time.

TABLE III
Maximum distance of the transmission.

Reference SPL (at 50cm)	Max. distance (10bit/s)		Max. distance (20bit/s)	
	estimated	**confirmed**	estimated	**confirmed**
70dB	1.26m	**1.2m**	0.63m	**0.6m**
80dB	3.97m	**3.5m**	1.99m	**2.0m**
95dB	22.33m	**16.0m**	11.19m	**9.0m**

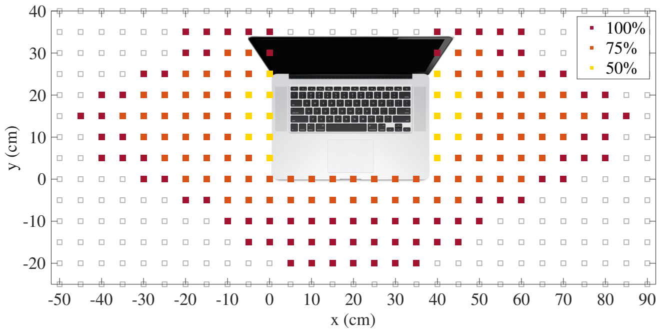

Fig. 5. Area around laptops where the signal can be decoded with BER<10%, depending on a volume level set using default system volume control (50%, 75%, and 100% of the maximum volume level).

To record the gyroscope measurements from a web page, we used the DeviceMotion API. This API does not provide *raw* gyroscope data, but instead uses a combined signal from several sensors (so-called *sensor fusion*), which reduced the resulting disturbance of the measurements. Moreover, the sampling rate is reduced to 60–100Hz, depending on the browser. As a result, we were able to achieve only 5bit/s bandwidth in mobile browsers. On the Chrome mobile browser, the amplitude of the combined signal was significantly lower in comparison to raw gyroscope data, so we were able to decode the signal only in close proximity to the source (≈80dB).

F. TV tracking: robustness against noise

To prove the robustness of the transmission against background noise, we transmitted tracking IDs with 80dB sounds in presence of background sounds, played in parallel through another speaker: background music and TV news broadcasting (≈80dB), and white noise (60, 80, and 95dB). The signals were captured on the Samsung Galaxy S7 smartphone located at 3m distance. We found that the resulting decoding was not affected at all (≤1% increase in BER), not only by music and news broadcasting (which is expected, since these sounds mostly lie within 20Hz–20kHz range), but even by white noise significantly exceeding the source signal (95dB). Therefore, gyroscopes naturally filter acoustic noise of other frequencies, making the sensor a robust receiver for tracking signals.

G. Location tracking: robustness against movements

To test how robust the proposed method is against device movements, we transmitted tracking IDs using a signal with SPL of 95dB (measured at 50cm distance), and captured the signal on the Samsung Galaxy S7 (at ≈3m distance), under the following conditions: the smartphone was held in a hand, the

TABLE IV
ROBUSTNESS AGAINST MOVEMENTS.

Experiment	BER (20bit/s)	BER (10bit/s)
Static position (no movements)	2%	1%
Smartphone in the hand	7%	4%
Smartphone is used	11%	5%
Person is walking	28%	22%

smartphone was used to play a simple game, and the person holding the smartphone was freely walking near the speaker. In each case, we transmitted 50 IDs with a bitrate of 20bit/s, and calculated the resulting BER.

The results are presented in the Table IV. The transmission remains stable in presence of slight movements: when the smartphone is held in a hand, or is being naturally used at the time of recording. When the person is actively moving, however, additional errors appear. Nevertheless, we believe that the approach is applicable to the location tracking scenario at specific locations, where the person is not *always* moving (e.g., in shops), since the comparably high bandwidth allows to quickly transmit the tracking ID.

V. RELATED WORK

A. Acoustic cross-device tracking

The idea of ultrasonic communication has been explored in research over the last years, mainly focusing on establishing covert channels between isolated computers [25]–[27] and mobile devices [19] by using their speakers and microphones. When commercial solutions of tracking TV ads and user location using ultrasonic beacons emerged on the market, researchers started to investigate the security and privacy implications of such technology. Mavrodius et al. [4] described several potential uXDT-based attacks, and designed a browser extension to filter out high frequencies from audio playback, as well as an Android permission to provide fine-grained control over microphone recordings. Arp et al. [6], [28] presented a detailed analysis of Silverpush and Lisnr implementations, and found 234 existing Silverpush Android applications that are listening in the background for ultrasonic beacons from TV streams, proving that the technology is being actively deployed in the wild. In our work, we demonstrate that described attacks may pose even more significant privacy risk, as they can be established fully unnoticeably to users, utilizing zero-permission access to gyroscope sensors on mobile devices.

B. MEMS Sensors reaction to acoustic vibrations

Table V summarizes prior works on exploiting acoustic susceptibility of MEMS sensors, and compares them with our work. Michalevsky et al. [23] showed that gyroscopes in smartphones can be used as low-frequency microphones. Son et al. [7] tested 15 kinds of MEMS gyroscopes against acoustic vibrations, demonstrating that gyroscope measurements can be disturbed by sounds at resonant frequencies, and exploited this fact to disorient drones by affecting their gyroscopes. Trippel et al. [24] confirmed that MEMS accelerometers are also susceptible to similar acoustic attacks.

To the best of our knowledge, utilizing MEMS gyroscopes as receivers for communication channels was proposed for the first time by Farshteindiker et al. [8]. A low-powered piezoelectric transducer was considered to physically touch the surface of a smartphone (≈0cm distance) and to send data with minimal possible power. Recently, Block et al. [18] proposed to exploit acoustic resonance of MEMS accelerometers to establish a covert channel between two mobile applications within one smartphone. In our work, we instead focus on transmitting data *over a distance*, with commonly-used audio hardware, and consider the practical cross-device tracking scenario with victim users naturally using their mobile devices.

VI. COUNTERMEASURES

Several ways to prevent the presented cross-device tracking are possible:

- Physical shielding is the most straightforward way to limit acoustic susceptibility of the sensor. Existing experiments show that a layer of foam, paper or aluminum reduces the disturbance of the sensor by 16–60% [7]. However, this measure mismatches with an industry trend of making consumer mobile devices thinner and lighter.

TABLE V
EXISTING ATTACKS EXPLOITING ACOUSTIC SUSCEPTIBILITY OF MEMS SENSORS.

	Work	Attack scenario	Sensor	Setup[a]	Distance
Other	Michalevsky et al. [23]	Recognizing speech with gyroscopes as microphones	gyroscope	T: consumer speaker; R: smartphone	–[b]
	Trippel et al. [24]	Disturbing and controlling accelerometer output.	accelerometer	T: consumer speaker; R: acceler.-equipped device	–[c]
	Son et al. [7]	Disorienting drones by disturbing gyroscope output	gyroscope	T: consumer speaker; R: drone flight controller	≈17cm (achieved); ≈37m (expected)[d]
Covert data transmission	Farshteindiker et al. [8]	Exfiltrating data from a surveillance implant	gyroscope	T: piezoelectric transducer; R: smartphone	≈0cm (physical touch)
	Block et al. [18]	Breaking Android application sandboxing	accelerometer	T: smartphone; R: smartphone (T=R)	≈0cm (intra-device)
	This work	Ultrasonic cross-device tracking	gyroscope	T: low/high-quality speakers; R: smartphone or smartwatch	from 35cm (low SPL) to 16m (high SPL)

[a] T: transmitter; R: receiver
[b] not evaluated; smartphone is placed "as close as possible to speakers"
[c] not evaluated; sensor is located ≈10cm from the speaker
[d] potential distance when using dedicated Long Range Acoustic Devices (LRADs)

- Designing the gyroscopes to have a resonance frequency of at least more than 25kHz prevents the attack from being feasible, at least with audio hardware that does not support high-resolution sampling.
- A low-pass filter can be applied to gyroscope measurements on the hardware or OS level to prevent detection of the transmitting frequency. However, precise gyroscope measurements can be required for some apps.
- Mobile OS vendors can forbid access to gyroscope data without an explicit permission granted by the user. However, users may not correctly estimate potential privacy threats [29]. Moreover, most of mobile devices in use run outdated operating system versions [30] and do not regularly receive security updates [31].
- Starting from version 8.0, Android will allow applications to run in a background only for a limited period of time, similarly to iOS. We advocate this measure and believe that additional restrictions can be introduced for access to sensors from background processes.

Most described countermeasures would require hardware or software changes, and may have performance or production cost drawbacks. Therefore, the presented attack remains completely feasible at present and can cause significant privacy threat to end users.

VII. Conclusion

In this work, we presented a novel approach to establish ultrasonic cross-device tracking, using acoustic susceptibility of gyroscopes in modern mobile devices. We showed that observing the reaction of gyroscopes to resonance frequencies in the frequency domain allows us to reliably capture tracking signals at a distance, making the approach applicable to web tracking, TV tracking and location tracking scenarios. The presented method does not require any explicit permissions and can be run unnoticed to end users.

VIII. Acknowledgment

This work has been partly funded by the German Academic Exchange Service (DAAD). The authors would like to thank anonymous HOST reviewers for their valuable comments and numerous suggestions for a future work.

References

[1] E. Ramirez, M. K. Ohlhausen, and T. McSweeny, "Cross-Device Tracking: An FTC Staff report," Federal Trade Commission, Tech. Rep. January, 2017.

[2] C. Calabrese, K. L. McInnis, G. Hans, and G. Norcie, "Comments for November 2015 Workshop on Cross-Device Tracking," Center for Democracy & Technology, Tech. Rep., 2015.

[3] J. Brookman, P. Rouge, A. Alva, and C. Yeung, "Cross-Device Tracking: Measurement and Disclosures," *Proceedings on Privacy Enhancing Technologies*, vol. 2017, no. 2, pp. 113–128, 2017.

[4] V. Mavroudis, S. Hao, Y. Fratantonio, F. Maggi, C. Kruegel, and G. Vigna, "On the Privacy and Security of the Ultrasound Ecosystem," *Proceedings on Privacy Enhancing Technologies*, vol. 2017, no. 2, pp. 95–112, 2017.

[6] D. Arp, E. Quiring, C. Wressnegger, and K. Rieck, "Privacy Threats through Ultrasonic Side Channels on Mobile Devices," *Proc. of IEEE European Symposium on Security and Privacy (EuroS&P)*, 2017.

[5] T. Fox-Brewster, "Meet the 'ultrasonic' tracking company privacy activists are terrified of," https://forbes.com/sites/thomasbrewster/2015/11/16/silverpush-ultrasonic-tracking/, 2015, [Accessed: 01.10.2017].

[7] Y. Son, H. Shin, D. Kim, Y. Park, J. Noh, K. Choi, J. Choi, and Y. Kim, "Rocking Drones with Intentional Sound Noise on Gyroscopic Sensors," *Usenix Security*, pp. 881–896, 2015.

[8] B. Farshteindiker, N. Hasidim, A. Grosz, and Y. Oren, "How to Phone Home with Someone Else's Phone: Information Exfiltration Using Intentional Sound Noise on Gyroscopic Sensors," *10th USENIX Workshop on Offensive Technologies (WOOT 16)*, 2016.

[9] "Android developer API guides: Sensors overview," https://developer.android.com/guide/topics/sensors/sensors_overview.html, 2017, [Accessed: 01.10.2017].

[10] "Apple developer documentation: : Core Motion framework," https://developer.apple.com/documentation/coremotion, 2017, [Accessed: 01.10.2017].

[11] V. Kaajakari *et al.*, *Practical MEMS: Design of microsystems, accelerometers, gyroscopes, RF MEMS, optical MEMS, and microfluidic systems*, 2009.

[12] S. Castro, R. Dean, G. Roth, G. T. Flowers, and B. Grantham, "Influence of acoustic noise on the dynamic performance of mems gyroscopes," in *ASME 2007 International Mechanical Engineering Congress and Exposition*. ASME, 2007, pp. 1825–1831.

[13] R. N. Dean, G. T. Flowers, A. S. Hodel, G. Roth, S. Castro, R. Zhou, A. Moreira, A. Ahmed, R. Rifki, B. E. Grantham *et al.*, "On the degradation of mems gyroscope performance in the presence of high power acoustic noise," in *Industrial Electronics, 2007. ISIE 2007. IEEE International Symposium on*. IEEE, 2007, pp. 1435–1440.

[14] P. Hallmo, A. Sundby, and I. W. Mair, "Extended high-frequency audiometry: air-and bone-conduction thresholds, age and gender variations," *Scandinavian audiology*, vol. 23, no. 3, pp. 165–170, 1994.

[15] "Sony corporation: High resolution audio," https://www.sony.com/electronics/hi-res-audio/, 2017, [Accessed: 01.10.2017].

[16] "Japan audio society: Status of hi-res audio logo," https://www.jas-audio.or.jp/english/hi-res-logo-en/use-situation-en, 2015, [Accessed: 01.10.2017].

[17] "IAC Acoustics: Comparative examples of noise levels," http://www.industrialnoisecontrol.com/comparative-noise-examples.htm, 2017, [Accessed: 01.10.2017].

[18] K. Block, S. Narain, and G. Noubir, "An autonomic and permissionless android covert channel," in *Proceedings of the 10th ACM Conference on Security and Privacy in Wireless and Mobile Networks*. ACM, 2017, pp. 184–194.

[19] L. Deshotels, "Inaudible Sound as a Covert Channel in Mobile Devices," *Proceedings of the 8th USENIX Workshop on Offensive Technologies - WOOT '14*, p. 16, 2014.

[20] A. V. Oppenheim, *Discrete-time signal processing*. Pearson Education India, 1999.

[21] I. S. Reed and X. Chen, *BCH Codes*. Boston, MA: Springer US, 1999, pp. 189–231.

[22] G. Davis and G. D. Davis, *The sound reinforcement handbook*. Hal Leonard Corporation, 1989.

[23] Y. Michalevsky, D. Boneh, and G. Nakibly, "Gyrophone: Recognizing Speech from Gyroscope Signals," *Usenix Security*, pp. 1053–1067, 2014.

[24] T. Trippel, O. Weisse, W. Xu, P. Honeyman, and K. Fu, "WALNUT : Waging Doubt on the Integrity of MEMS Accelerometers with Acoustic Injection Attacks," no. April, 2017.

[25] M. Hanspach and M. Goetz, "On covert acoustical mesh networks in air," *Journal of Communications*, vol. 8, no. 11, pp. 758–767, 2013.

[26] ——, "Recent Developments in Covert Acoustical Communications."

[27] B. Carrara and C. Adams, "On Acoustic Covert Channels Between Air-Gapped Systems," 2015, vol. 10128, pp. 3–16.

[28] D. Arp, E. Quiring, and C. Wressnegger, "Bat in the Mobile : A Study on Ultrasonic Device Tracking," 2016.

[29] M. Mehrnezhad, E. Toreini, S. F. Shahandashti, and F. Hao, "Stealing pins via mobile sensors: actual risk versus user perception," *International Journal of Information Security*, pp. 1–23, 2016.

[30] "Android developer API guides: Dashboards. platform versions," https://developer.android.com/about/dashboards/index.html, 2017, [Accessed: 01.10.2017].

[31] "CBS Interactive: Most Android users running outdated security patches," https://cnet.com/news/most-android-users-running-outdated-security-patches-report-says/, 2017, [Accessed: 01.10.2017].