Acoustic Side-Channels via Mobile Sensors

A minor project report

by

Shadab Zafar

Entry No: 2017MCS2076

Under the guidance of

Prof. Vinay Ribeiro

Department of Computer Science and Engineering,
Indian Institute of Technology Delhi.
May 2018.

Abstract

The popularity of smartphones continues to grow because of the wide variety of functionality they offer - from just being able to make calls and access the internet to recording videos and playing games. To provide a lot of this functionality, a modern smartphone comes equipped with sensors such as camera, microphone, GPS etc. Data from these sensors enables the creation of applications that offer rich and personalised user experience.

Use of these sensors also opens up the possibilities of new attacks by leaking information via side channels. In this report, we explore how data from a particular mobile sensor - the accelerometer - can be used to eavesdrop acoustic signals in the vicinity of the phone, thereby converting the accelerometer into a microphone, and since accessing the sensor doesn't require any special permission, this allows an adversary unregulated access to audio surrounding the device's environment.

Contents

Chapter 1

Introduction

As mobile phones become more and more ubiquitous, they are no longer used just for communication purposes but also as personalized computing devices, since a smartphone offers a mix of functions ranging from telephony and internet connectivity to use as a camera.

Like any technology, mobile phones have their caveats - with their widespread use, the issues of security and privacy have taken a central role in their design and usage. Since the computation platform a modern smartphone offers is akin to a general purpose computer, they suffer from similar threats like viruses and ransomwares and also more targetted attacks like leaking of sensitive information etc.

Smartphones today, come equipped with a wide variety of sensors like camera, microphone, GPS to enable applications that offer a rich user experience. However, they are not without their own caveats - recent research [**CITE**] has shown a host of different ways how these sensors leak information that can be used against the users and has far reaching privacy implications.

A particular class of sensors is used for motion detection - gyroscopes, accelerometer etc. While privacy issues associated with the use of a microphone and GPS are understood by most users, those related with motion sensors are not. So an adversary is more likely to use such senors.

1.1 Research Goal

> The goal of this project is to explore how MEMS accelerometers can be used as microphones and see if they are sensitive enough to eavesdrop audio in the environment surrounding a phone.

1.2 MEMS Accelerometer

Accelerometers in smartphones are based on Micro Electro Mechanical Systems (MEMS) design, which emulate mechanical parts through micro-machining technology. At their core, the MEMS accelerometers have a sensing mass, suspended with springs, which gets displaced due to forces (that cause acceleration.) It was shown in [1] that accelerometers are susceptible to acoustic interferences, i.e. an acoustic wave can exert a force on the sensor that is strong enough to displace the sensing mass affecting the sensor's output.

1.3 Sensors on an Android Phone

Android is a mobile operating system from Google. Launched in 2008, it's growth paralleled that of smartphones themselves. It has the majority marketshare today with around 85% of all smartphones coming with Android installed (as of 2017.) [2]

One of the reasons of Android's growth are its permissive APIs - applications on this platform enjoy a lot more freedom compared to iOS and even though new features (to both hardware and software) are added in a quest to increase the user experience, the security & privacy preserving aspects take a back seat and do not improve at the same pace.

Applications that require access to a particular hardware capability need to ask permission from the user to be able to use it. This is the core line of defense that prevents malicious applications from being able to abuse critical sources of information like camera, microphone, SMS etc.

However, sensors like accelerometer, gyroscope etc. do not require a special permission and can be accessed by any application, in both foreground and background. This allows an attacker to use data from these zero-permission sensors maliciously without any sort of consent from the user.

1.4 Recent Developments

In March 2018, Google announced the first developer preview for the privacy centered features of the next Android version - P. Among the major behaviour changes, there are some privacy centeric changes too, wherein, the applications will not be able to record motion sensors, microphone, camera in background. Applications that legitimately need to record motion sensors will have to show a persistent notification so that the user is aware of what is happening in the background. This serves use cases for apps that use motion data to perform tasks such as step counting etc.

Foreground applications will have no such restrictions and will continue to work the same way.

Chapter 2

Literature Review

There have been a lot of work in exploting zero-permission sensors found in a typical smartphone. Such security and privacy breaches via sensors have been investigated in publications like [3], [1], [4], [5], [6] etc.

In **Gyrophone** [3], the authors showed how a Gyroscope could be used as a Microphone, but since they too were limited by the sampling rate, they used machine learning techniques (like SVM, GMM, DTW) to test their accuracies on Speaker, Gender Identification tasks. Speech Recognition was done on the TIDIDIGITS dataset. They also had a section where they used multiple such phones gathering data and then used it to further improve the accuracy of the model. Our goal with this project is to perform something similar with accelerometer data.

In the **Walnut** attack [1], the authors found critical vulnerabilities in the design of MEMS accelerometers, that could potentially allow an attacker to completely control the sensor readings being output. These sensors are not exclusively used on mobile phones, but in another devices as well, such as the SensorTile kit - which we discuss in a later section. The walnut attack requires frequencies close to the resonant frequency of the sensor (which need to be found out experimentally or from a datasheet by the manufacturer.) Essentially, this work further solidified the fact that that accelerometers (like gyroscopes) are susceptible to acoustic interferences in their vicinity.

As an example of an attack that is a bit different from acoustic injections, the authors of [4] used a combination of zero permission sensors to guess a users PIN. This required modelling of the touch events and then some feature engineering and machine learning techniques (KNN, GNB, MLP, RF etc.) As their result they found that a combination of sensors (accelerometer + gyroscope) worked best.

2.1 Browsers

Exploitation of these sensors on a modern smartphone doesn't always require creating a specialized application for the platform, as the authors of [7] showed, sensor data is also readily available via any modern browser - such as Firefox, Chrome etc. This has far reaching consequences, because this means that any website could capture sensor data and upload it to their server for later offline processing - coupled with existing web fingerprinting techniques - this could be used to target user behaviour and worst of all this raises absolutely no flags, or permission popups!

Since the paper came out, browser vendors have implemented further hardening techniques to make the exploit tougher, specifically, they have restricted the sampling rates and only web pages with secure contexts can capture data in background.

To understand this explot better, and confirm that the restrictions were in place, we created a demo webpage in Javascript that uses the window.ondevicemotion API to gather data (from motion sensors.)

Figure 2.1 shows the Google Chrome browser, and the data frequency has been further reduced (from 200Hz to 60Hz.)

But, as Figure 2.2 shows this is not the case with Firefox. It still allows sampling at a rate of 200Hz. So is still vulnerable to all sorts of sensor recording exploits.

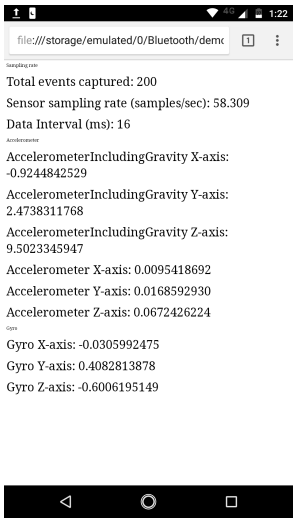

Figure 2.1: A webpage recording motion sensors on Chrome (60Hz)

Figure 2.2: A webpage recording motion sensors on Firefox (200Hz)

Chapter 3

Sensor Data

3.1 Sensor Tile Kit

In section 1.3, we explained how Android limits the sensor data sampling rate to around 200Hz, which is good enough for monitoring device movements, such as tilt, rotation etc. but is low for our goal to use it as an microphone since human audible frequencies lie in the 20Hz to 20kHz range. To explore what could be done if we had access to high frequency sensor data, we tried a specialized sensor board - the SensorTile by STMicroelectronics.

The SensorTile is a tiny, square-shaped IoT module that packs powerful processing capabilities leveraging an 80 MHz microcontroller, a Bluetooth low energy connectivity based on BlueNRG-MS network processor as well as a wide spectrum of motion, such as a triaxial accelerometer, gyroscope, magnetometer, and environmental sensors, such as pressure, humidity and temperature and even a digital microphone. [8]

Figure 3.1 shows the contents of the SensorTile kit.

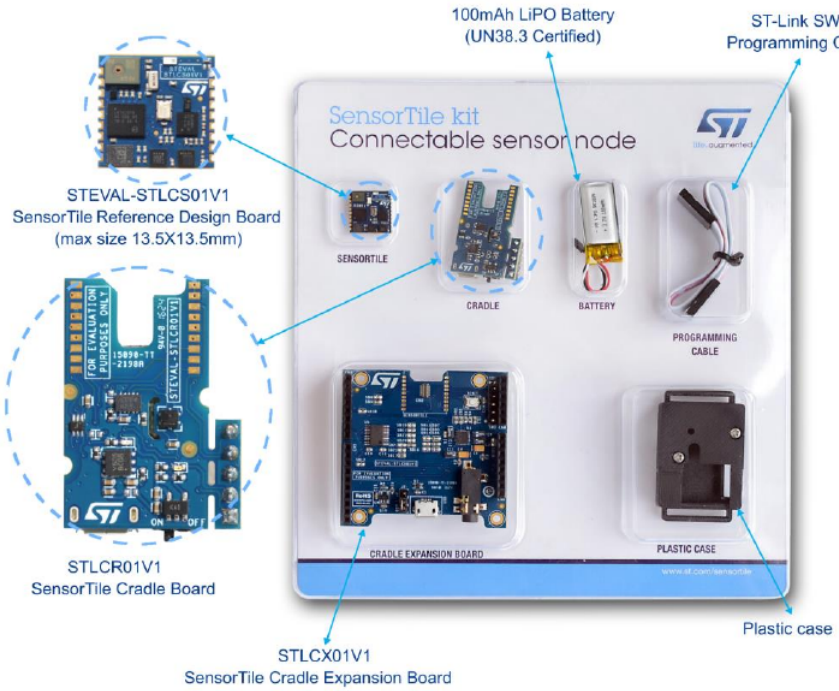

Figure 3.1: SensorTile Development Kit

Figure 3.2 shows position of main components of the SensorTile board, and Table 3.1 lists their description. [9]

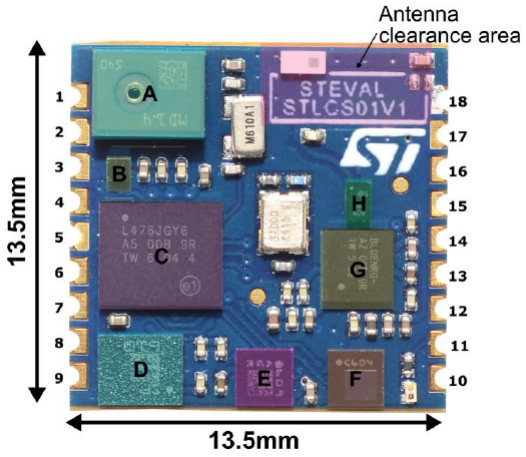

Figure 3.2: SensorTile Main Components

Reference	Device	Description
A	MP34DT04	MEMS audio sensor digital microphone
B	LD39115J18R	150 mA low quiescent current low noise LDO 1.8 V
C	STM32L476 MCU	ARM Cortex-M4 32-bit microcontroller
D	LSM6DSM	iNEMO inertial module: low-power 3D accelerometer and 3D gyroscope
E	LSM303AGR	Ultra-compact high-performance eCompass module: ultra-low power 3D accelerometer and 3D magnetometer
F	LPS22HB	MEMS nano pressure sensor: 260-1260 hPa absolute digital output barometer
G	BlueNRG-MS	Bluetooth low energy network processor
H	BALF-NRG01D3	50 balun with integrated harmonic filter

Table 3.1: SensorTile Main Components

The LSM6DSM sensor on the SensorTile board is the accelerometer and is capable of capturing data at a maximum rate of 6000Hz.

To interface with the sensor hardware, there are three available firmwares for the board:

1. DataLogger - allows recording motion sensor data
2. AudioLog - allows recording onboard microphone
3. ALLMEMS - allows SensorTile to be controlled via BlueNRG android app

Our goals with the SensorTile were two fold:

1. Gather data at the maximum rate possible.
2. Record both motion sensors and microphone at the same time (so that we could have a baseline to compare the sensor recordings to.)

We didnt succeed in achieving either of these goals as neither of the three listed stock firmwares have the capabilities that we desired. All three firmwares restricted the sensor sampling rate to around 200Hz (similar to what we would get on Android.) Another restriction was the board only allows either motion sensors or microphone to be recorded but not both simulataneously.

Even though these firmwares are open-source, the documentation is relatively scarce, making new customizations very difficult. The SensorTile kit is relatively new, so not a lot of people are working on similar things.

3.2 Android App

After not being able to get the SensorTile working as a better source for data, we turned back to Android and it's limited 200Hz sampling rate.

Our goals were again, to gather sensor data at the fastest rate possible and to also capture the microphone simulataneously. Since android has a plethora of apps for almost any purpose, we first tried to find an existing application that could fit our needs, but after trying out about a dozen of the apps, we found that all of them suffered from similar issues: they either didnt have fine grained control over sampling rate for sensors Or didnt allow both motion sensors & microphone to be recorded at the same time.

So we designed our own application that can record Accelerometer, Gyroscope & Microphone simultaneously at the maximum rate offered by Android. Data from the sensors is saved to separate files on disk (CSV for motion sensors and M4A for microphone.)

Figure 3.3 shows the design of our application.

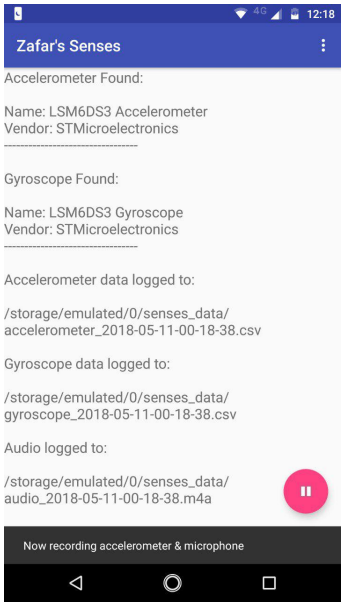

Figure 3.3: Sensor Data Recording App (main view)

Chapter 4

Analysis

Once we had a working Android application, we performed experiments on a Motorola Moto G5s Plus phone with Android version Nougat 7.1. We also used an external speaker to create acoustic signals at loud volume ($\tilde{7}$0dB).

We use Python and it's scientific suite (Numpy, Scipy, Pandas etc.) to perform analysis and create plots.

Our android application stores the audio files in M4A format because that is what the Android APIs allow, but to make processing easier we use the FFmpeg tool to convert these M4As to WAV files as there are a lot more libraries that support WAV out of the box.

The experimentation was straightforward - try playing loud audio signal in the vicinity of the phone, and see if the accelerometer data shows a response.

According to the Nyquist sampling theorem a sampling frequency f enables us to reconstruct signals at frequencies of up to f/2. Since the frequency of an Android motion sensor is limited to be 200Hz, this allows us to directly sense audio signals of up to 100 Hz.

But, there is also a separate effect of Aliasing [3] - a phenomenon where for a sinusoid of frequency F, sampled with frequency f, the resulting samples are indistinguishable from those of another sinusoid of frequency

$$|F - N * f|$$

for any integer N. The values corresponding to N != 0 are called images or aliases of frequency f. An undesirable phenomenon in general, here aliasing allows us to sense audio signals having frequencies which are higher than 100 Hz, thereby extracting more information from the gyroscope readings.

So, if we play a tone of 270Hz near an accelerometer (or a gyroscope for that matter) we expect to see peaks in the Fast Fourier Transform curve of the signal at around 70Hz.

The results of this experiment are shown in the figures that follow. The first graph is the signal itself, the second is the FFT curve of the signals, showing the dominant frequencies and the third is the spectrogram, showing when the frequencies are high (darker color in the spectrogram).

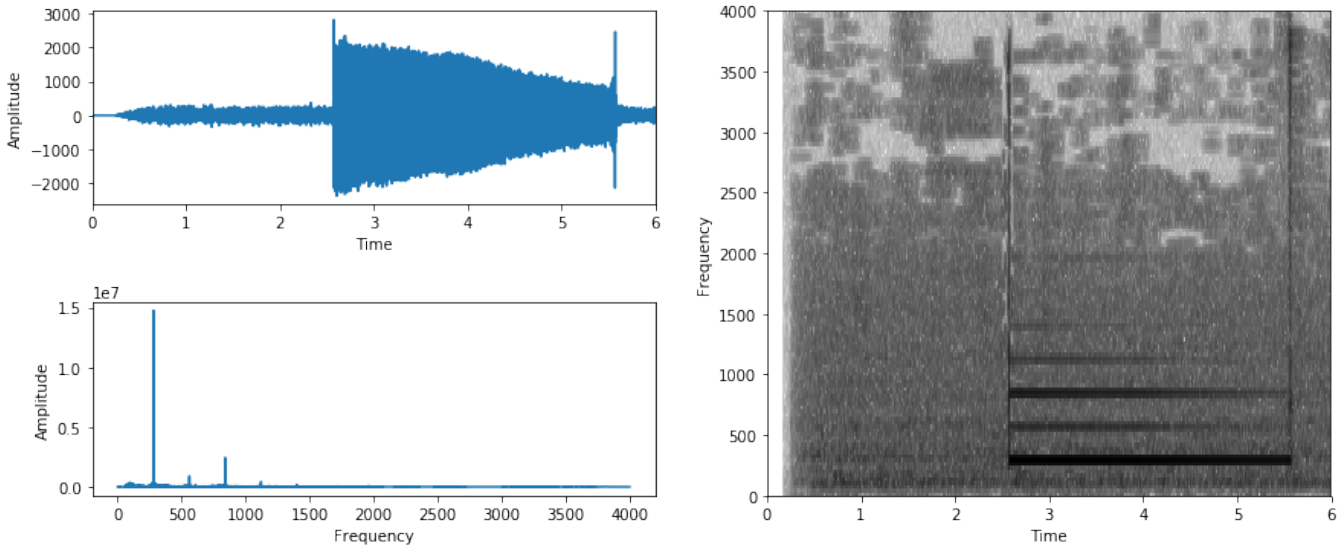

Figure 4.1: Microphone (when playing a 270 Hz audio wave)

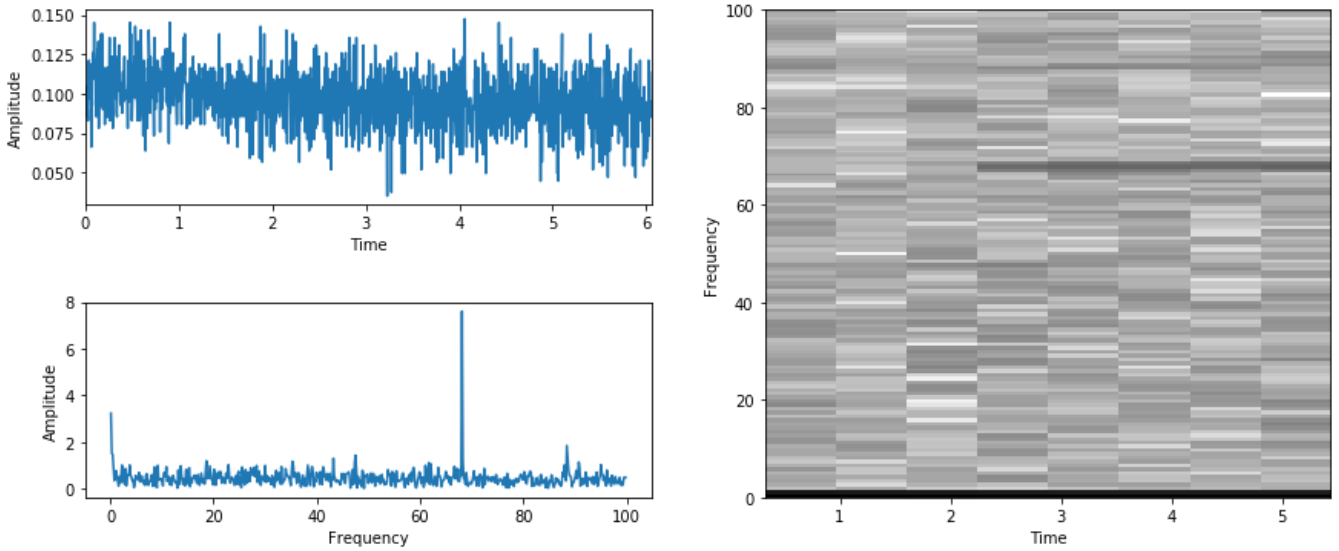

Figure 4.2: Accelerometer (when playing a 270 Hz audio wave)

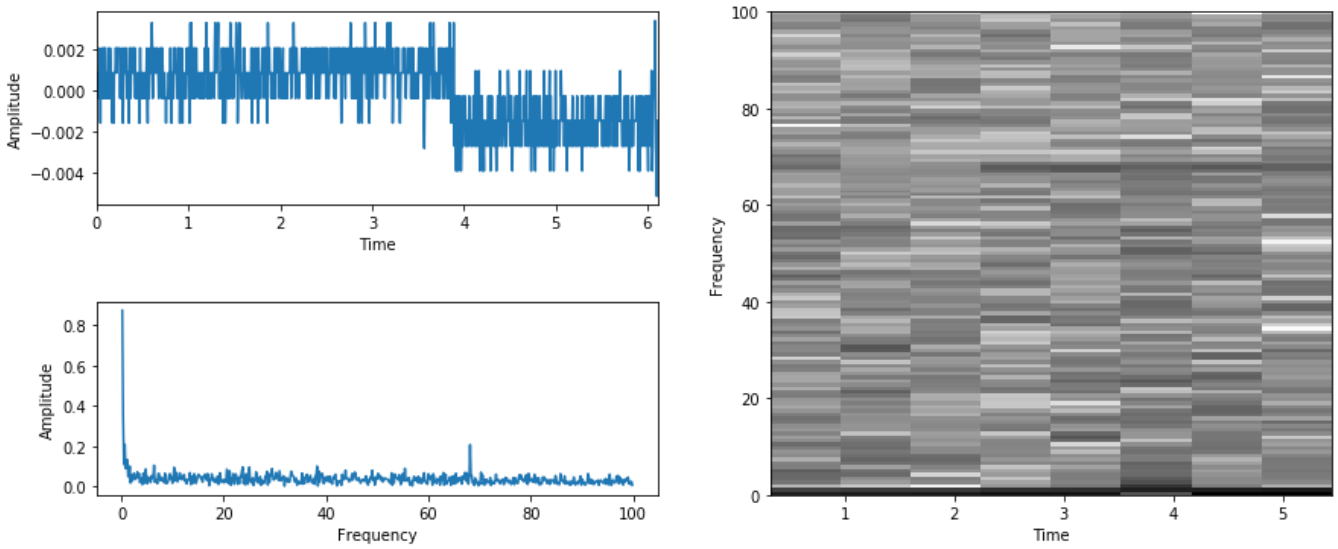

Figure 4.3: Gyroscope (when playing a 270 Hz audio wave)

As can be seen from figure 4.1, the 270Hz tone begins at around 2.5 seconds, and its FFT clearly shows a distinct peak at 270Hz. The spectrogram is dense since there's a lot of data (as microphones sample at rates close to 8000Hz) but we can clearly observe that from 2.5 seconds to 5.5 seconds, there are multiple frequency bands (which are just aliases of each other.)

What is more interesting is figure 4.2 where just from looking at the signal plot, it is not quite evident that there is any particular pattern to it, but the FFT plot again reveals a peak at around 70Hz (this is the aliased frequency of 270Hz.) Looking at the spectrogram we again find that there is a single distinct line at around 70 Hz.

Similar results can be seen in figure 4.3 but the values are fainter than the accelerometer.

Chapter 5

Conclusions

After not being able to get a high frequency sensor data source, we performed experiments on a typical Android phone and found that motion sensors found in such devices are in fact susceptible to acoustic signals (to varying degree.)

Now the question remains whether they can be used to eavesdrop human speech. It is clear that since the sampling rate on the phone is low, it is not possible to directly use the sensor signals in any way.

5.1 Future Work

As in [3] the next step is to apply signal processing techniques like noise removal, feature extraction and then machine learning algorithms to see whether the data being sensed is accurate enough to actually be of use.

Since we found that both accelerometer & gyroscopes are susceptible to audio signals, we could also look into how the data from these sensors could be combined.

During our experiments, we also did not try anechoic chambers because existing research [3] found that they didnt really affect the results, but we could still explore that.

Bibliography

[1] Timothy Trippel, Ofir Weisse, Wenyuan Xu, Peter Honeyman, and Kevin Fu. Walnut: Waging doubt on the integrity of mems accelerometers with acoustic injection attacks. In *Security and Privacy (EuroS&P), 2017 IEEE European Symposium on*, pages 3–18. IEEE, 2017.

[2] Worldwide Smartphone OS Market Share (in Unit Shipments). `https://www.idc.com/promo/smartphone-market-share/os`. Accessed: 2018-04-27.

[3] Yan Michalevsky, Dan Boneh, and Gabi Nakibly. Gyrophone: Recognizing speech from gyroscope signals. In *USENIX Security Symposium*, pages 1053–1067, 2014.

[4] David Berend, Bernhard Jungk, and Shivam Bhasin. There goes your pin - exploiting smartphone sensor fusion under single and cross user setting. 2017.

[5] Philip Marquardt, Arunabh Verma, Henry Carter, and Patrick Traynor. (sp) iphone: decoding vibrations from nearby keyboards using mobile phone accelerometers. In *Proceedings of the 18th ACM conference on Computer and communications security*, pages 551–562. ACM, 2011.

[6] Andreas Kurtz, Hugo Gascon, Tobias Becker, Konrad Rieck, and Felix Freiling. Accleprint - fingerprinting mobile devices using personalized configurations. *Proceedings on Privacy Enhancing Technologies*, 2016(1):4–19, 2016.

[7] Maryam Mehrnezhad, Ehsan Toreini, Siamak Fayyaz Shahandashti, and Feng Hao. Touchsignatures: Identification of user touch actions and pins based on mobile sensor data via javascript. *CoRR*, abs/1602.04115, 2016. URL `http://arxiv.org/abs/1602.04115`.

[8] STEVAL-STLKT01V1 - SensorTile development kit. `http://www.st.com/en/evaluation-tools/steval-stlkt01v1.html`, . Accessed: 2018-04-27.

[9] UM2101: Getting started with the STEVAL-STLKT01V1 SensorTile integrated development platform. `http://www.st.com/resource/en/user_manual/dm00320099.pdf`, . Accessed: 2018-04-27.