Dual-Tone Multi-Frequency Assisted Acoustic Side Channel Attack to Retrieve Dialled Call Log

Abhishek Revskar, Mahendra Rathor, and Urbi Chatterjee(✉)

Department of Computer Science and Engineering, Indian Institute of Technology Kanpur, Kanpur, India
{abhishekdr,rmahendra,urbic}@cse.iitk.ac.in

Abstract. Acoustic side channel attack (SCA) is a type of SCA which exploits the sounds emitted by computers or other devices to retrieve the sensitive information, without requiring the adversary to perform any mathematical cryptanalysis. Recently, acoustic SCA has been exploited by attackers to breach the security of mobile devices. A malicious application installed in the mobile devices can access and take control of system components such as microphone, gyroscope, camera, etc. As users may not be aware of the security guarantee of the malicious applications, they can blindly trust and download such applications in their mobile phones and grant access to unnecessary permission. This security vulnerability can be exploited by an attacker to retrieve user sensitive information and compromise the user privacy. In this paper, a novel Dual-Tone Multi-Frequency (DTMF) assisted acoustic side channel attack is proposed to retrieve dialled call log from mobile devices. In this attack, an adversary can infer the call log or phone numbers dialed by the victims on their devices by gaining access to the in-built microphone. To the best of our knowledge, the proposed acoustic SCA is the first work in the literature that exploits the standards of DTMF to uniquely identify each key/digit dialed on the dialling keypad. In the proposed acoustic SCA methodology, we infer the keys/digits dialed by the victims by first analyzing the recordings of sounds produced from dialed digits and then finding the frequency distribution for each digit using Fast Fourier Transform (FFT). Further, the characteristic frequencies of the keys/digits are matched against the DTMF specifications to uniquely identify them. Further, we have trained the machine learning (ML) models to facilitate the prediction of the call log or the phone numbers dialed by the victim. The proposed attack is device-independent and is capable of predicting the phone numbers dialed in one device while training the ML models on the other. The prediction accuracy of the proposed approach is achieved to be 100% because of exploiting the standards of DTMF which are common for all the communication devices across the globe.

Keywords: Acoustic side channel attack · Dual-Tone Multi-Frequency · Fast Fourier Transform · Machine learning

L. Batina et al. (Eds.): SPACE 2022, LNCS 13783, pp. 185–203, 2022.
https://doi.org/10.1007/978-3-031-22829-2_11

1 Introduction

Side Channel Attacks (SCAs) have been analyzed rigorously and launched successfully on hardware and embedded systems to leak the secret key over the past two decades. Side channels bring forth the state information about the implementation which might not be captured by the classical adversaries [1]. For example, crypto-processors generally have variable execution time to process the data-dependent operations. By using the timing side channel, an adversary can measure the time required for the secret key operation and retrieve some important information about the secret key which can help the adversary to launch a successful SCA. In [2], it was shown that the adversary can extract the secret key of RSA [3] and Diffie-Hellman [4] key exchange by analyzing the timing information with known ciphertext. Similarly, other types of side channels can be exploited to leak some secret information (e.g. key) such as power side channel [5] and electromagnetic radiations (EM) etc. In power side channel [5], an adversary measures the power consumption of a hardware platform while running the cryptographic algorithms and try to relate it with the secret key (Simple Power Attacks) or the differential of an intermediate state between two consecutive rounds of the cipher depending upon the secret key (Differential Power Attacks). In EM side channel, an attacker tries to extract the secret key by collecting the EM radiations of an embedded system during the execution of the cipher using EM probe. Then the secret key can be extracted using the EM side channel traces.

Sometimes, the device platforms are so vulnerable that an attack can be launched to breach the privacy without even touching the crypto-module. This is possible because of some other types of side channels such as acoustic [6], thermal emission [7], magnetic [8] etc. Among these, acoustic-channels may be formed of audible or non-audible signals which are produced by a transmitter/speaker or executing specific processes on the processing unit of the computer [9]. Acoustic side channel attack leverages computer or device acoustics to get sensitive information without mathematical cryptanalysis. Recently, it has been found that the acoustic emanations produced by electronic devices can be used to infer the operations and data entered by the users in their systems and can present a serious threat to user privacy. Some existing works have highlighted that the sounds resulting from keyboard typing can be exploited to learn information about the entered data [10]. Asonov and Agrawal [11] have showed that the frequency features from the sound emanations of various keyboard clicks can be extracted to infer the different keys. Whereas in an another work, an acoustic side channel attack has been launched on additive manufacturing systems like 3D printers to infer the object that is being printed [12]. However, the above mentioned approaches of acoustic SCA did not target the security of smartphones or mobile devices. Instead, a number of techniques that leverage built-in smartphone sensors to leak users' private information through side channel attacks have been proposed in the literature [13–22]. However, some hardware, operating-system and application-level mechanisms can be employed to block this attack more effectively [22]. Moreover, these approaches also do not target the retrieval of ‘call log’ or ‘10-digit phone numbers’ unlike the our proposed approach.

To be more specific, in [22], the acoustic SCA of retrieving the PINs/characters on mobile phones is device dependent. This is because, they have used the time-difference-of-arrival as the feature to classify the entered characters. The time-difference-of-arrival is measured between the signals received at the two microphones of the device which will vary across the devices. This leads to a question: *Can we can launch an attack that is generic to all mobile devices (i.e. system independent) and less complex to apply?* In order to cater the above mentioned issue, we have proposed a novel acoustic SCA which is independent of the type of mobile device as long as it supports dialling and calling function and have at least one microphone. In addition, the complexity of retrieving the dialed digits is comparatively lesser than the existing acoustic SCA on smartphones [22]. This is because in the proposed technique, the ML model needs to be trained only once and later it can be used to launch the attack on any type of device. Whereas in [22], the model is required to be trained separately for the type of device on which the attack is intended to be launched leading to higher implementation complexity and attack time.

The role of DTMF standards and ML models in the proposed approach are briefly described as follows:

– Dual-tone multi-frequency (DTMF) is used to produce the sound that is unique to each key on the dialling keypad. DTMF is a signaling system which is used for communication through telephone systems and mobile devices. It defines certain frequencies which are used to produce unique sound upon dialling each key. These frequencies are common across all the mobile devices which support calling function. Each key is composed of a pair of low frequency and high frequency which is unique and have no relation with other frequencies. Hence, it is possible to detect the key by analyzing the sound produced by that key. Table 1 shows the low and high frequencies associated with each key.
– In the proposed approach, we trained the ML models such as support vector machine (SVM), Random Forest, Artificial Neural Network (ANN) on only one device and the trained ML models are capable of predicting the phone numbers dialed on any type of device, making our attack device independent.

1.1 Main Intuition and Contributions

Further, the major intuitions behind the proposed attacking methodology are described as follows.

– When a user dials a phone number on dial pad, each key produces a sound which is composed of the frequencies specified by DTMF standards.
– If we record the sound through in-built microphone and analyze the frequencies present in it, we can predict the key which generated this sound.

- To do so, the recorded sound is converted from its time-domain to frequency-domain representation to find the frequencies present in it. We used a Discrete Fourier Transform (DFT) method to decompose a signal into its frequency components.
- Fast Fourier Transform (FFT) is an efficient algorithm to compute the discrete Fourier transform of a signal. Thus we can find the magnitude of each frequency present in the recorded signal.
- By finding the two frequencies which have the highest magnitudes, we can predict the key.

The top two frequencies will not be always exactly equal to what is shown in the Table 1, because of the noise present in the surrounding and limitations of the microphone hardware. However, the frequencies will certainly be somewhere around the characteristic frequencies with slight deviations. The frequencies corresponding to one key are unique and have no relation with the frequencies of the other keys. Now the problem statement boils down to a traditional classification problem wherein we classify the keys based on the frequencies present in them. In the proposed work, we trained several machine learning models to learn the mapping from frequencies to keys which accounts for the deviations in the frequencies as well. These models are later used to predict the keys by analyzing their sound recordings.

We performed this attack successfully on various devices like Samsung M31, Realme GT, Motorola, Lenovo Tablet and IPhone. The operating systems of all of the above devices are Android except for IPhone which uses iOS operating system. We used python libraries *librosa* and *numpy* to work with the recorded sound signals. The Numpy provides a function which computes the FFT of the given signal in order to find the magnitude of each frequency present in the recorded signal.

Table 1. DTMF keypad frequencies for each key

	1209 Hz	1336 Hz	1477 Hz	1633 Hz
697 Hz	1	2	3	A
770 Hz	4	5	6	B
852 Hz	7	8	9	C
941 Hz	*	0	#	D

In summary, the major contributions of our work are as follows:

- We have proposed a novel acoustic side channel attack methodology to retrieve the dialled call log by exploiting the in-built components of the victim's device such as microphone.
- The proposed approach leveraged the DTMF standards, to uniquely identify the phone number that is being dialed by the victim. The DTMF standards

being common across all the devices makes our proposed attack device independent, which we have also shown in the experimental results section.
- We have shown the use of ML models such as SVM, Random Forest and ANN for training with a number of samples of sounds corresponding to dialed digits and predicting the 10 digit phone number dialed by the victim with 100% accuracy.

The rest of the paper is organized as follows. In Sect. 2, we provide the background of the working principles of DTMF and FFT. We describe our attack methodology in Sect. 3 and provide the experimental setup and results of our work in Sect. 4. Finally, we conclude our paper in Sect. 5.

2 Background

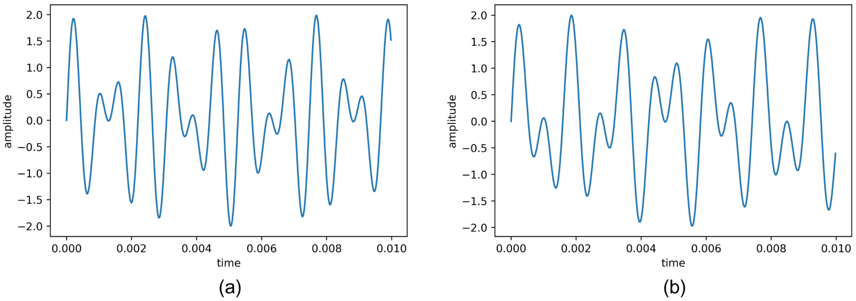
This section briefly discusses the important terminologies used in the proposed methodology of acoustic SCA, such as DTMF, FFT and different ML models. First we discuss some background about DTMF and how it defines the frequencies for each key. Next, we discuss the FFT technique followed by the ML models such as SVM, Random Forest and ANN.

2.1 Dual-Tone Multi-Frequency (DTMF) Signals

DTMF is a signaling system which is used in communicating devices like telephone systems and mobile devices. The standards for DTMF signals have been developed by the Bell System Inc., US. These standards have been specified in the International Telecommunication Union ITU-T Recommendation Q.23 [27]. DTMF tones are produced by adding two sinusoidal signals having frequencies among the 8 defined frequencies. Each key is composed of a pair of low and high frequencies as shown in Table 1. The mathematical function to generate a pure DTMF tone for a particular key is given below.

$$x(t) = Acos(2\pi f_L T + \phi) + Acos(2\pi f_H T + \phi) \tag{1}$$

Where, A is the amplitude of the signal, f_L and f_H are the low and the high frequencies respectively from which the key signal is formed, $1/T$ is the sampling rate of the signal and ϕ is the phase of the signal. Figure 1 shows the time-domain representations of the DTMF tones of digit ‘0’ and digit ‘1’. The DTMF tone corresponding to digit ‘0’ is formed by combining signals of two distinct frequencies viz. 941Hz and 1336 Hz whereas the DTMF tone corresponding to digit ‘1’ is composed of the 697 Hz and 1209 Hz. As shown in Fig. 1, the combination of different frequencies results into distinct dial tones (signals) for different digits.



(a)

(b)

Fig. 1. Combination of two sine waves to produce (a) DTMF ‘0’ and (b) DTMF ‘1’.

2.2 Fast Fourier Transform (FFT)

In this paper, we have employed the FFT to obtain the corresponding frequency-domain representation of the recorded dial tone in order to facilitate the features extraction for performing the attack. The FFT is an efficient algorithm to compute the discrete Fourier transform (DFT) of a signal. DFT is a method to decompose a signal into its frequency components. It is one of the easiest and commonly applied methods to get the frequency-domain representation of a given signal from its time-domain representation. The formula to compute the DFT of the sequence $x[n]$, corresponding to the continuous time signal $x(t)$, is given below.

$$\hat{x}(k) = \sum_{n=0}^{N-1} x(n) e^{\frac{-2\pi i n k}{N}}, \quad k = 0, ..., N-1 \tag{2}$$

Where, $\hat{x}(k)$ is a complex number in the form of $(a + ib)$ which represents the magnitude and the phase of the frequency $F(k)$ in the original signal $x(t)$. N is the number of samples in $x[n]$ and n is the sample number. The different frequencies given by $F(k)$ can be derived using the following equation.

$$F(k) = \frac{k.S_r}{N} \tag{3}$$

where, S_r is the sampling rate of the signal. The time complexity of finding Fourier transform using Eq. (2) is $O(N^2)$. However, it reduces to $O(NlogN)$ because of applying the FFT. We employ the FFT in our approach to translate the recorded dial tone into the corresponding frequency representation.

2.3 Machine Learning Models

The proposed work employs machine learning (ML) models to facilitate the prediction of the phone number digits dialed by the victim. Here, the objective of using an ML model is to classify the given sample into one of the 10 classes (10 digits from 0 to 9) which is a supervised learning task. Therefore, we have selected the classifiers namely, SVM, Random Forest and ANN which are widely used to solve classification problems.

The objective of SVM technique is to establish the best line or decision boundary that can divide n-dimensional space into classes, allowing us to quickly classify fresh data points in the future. A hyperplane is the name given to this optimal decision boundary. SVM selects the extreme vectors and points that aid in the creation of the hyperplane. Support vectors, which are used to represent these extreme instances, form the basis for the SVM method [23]. Further, random forest is also another supervised machine learning algorithm which is employed in classification problems. On various samples, it constructs decision trees and uses their majority vote to decide the class of the data point [24]. Additionally, we have also employed ANN based supervised learning model. A computational network based on biological neural networks, which create the structure of the human brain, is typically referred to as an Artificial Neural Network (ANN) [25]. It learns the weights for the edges connecting the neurons from one layer to the next layer to minimize the prediction loss/error at the output layer. We have used the softmax activation function in the output layer to predict the class of the sample. The formula for softmax activation function in given below.

$$\sigma(\overrightarrow{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_i}} \tag{4}$$

Here, σ is softmax, $\overrightarrow{z}$ is the input vector, e^{z_i} is standard exponential function for input vector, K is the number of classes in multi-class classifier (10 in our case) and e^{z_j} is standard exponential function for output vector. This softmax function outputs the probability distribution for all K classes.

Having this background on the core terminologies viz. DTMF, FFT and ML models used in our work, we present the proposed acoustic SCA methodology in the next section.

3 Proposed Acoustic Side Channel Attack Methodology

In this section, we present the acoustic SCA methodology of retrieving the ‘call log’ or inferring a ‘10-digit phone number’ while being dialed by the victim on his mobile device. The main intuition behind the attacking methodology is

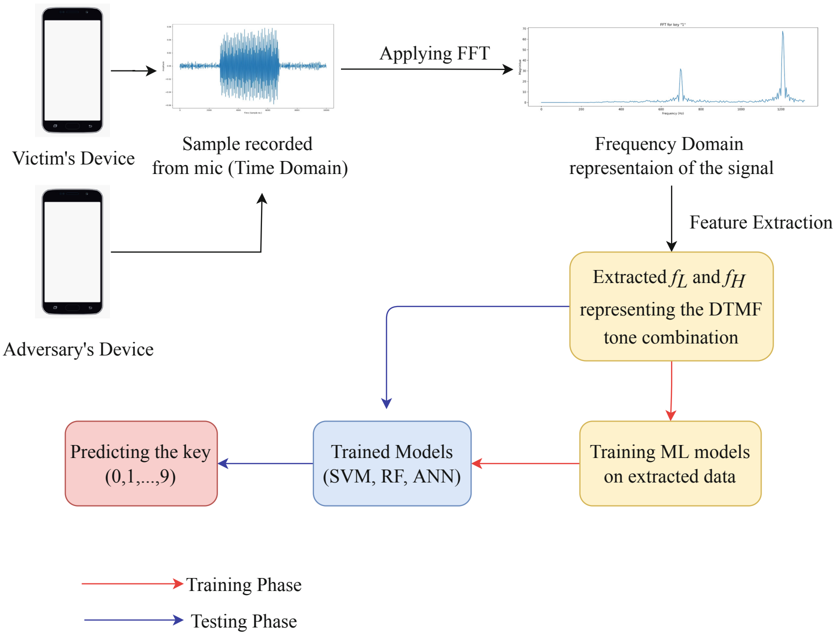

Victim's Device
Sample recorded from mic (Time Domain)
Applying FFT
FFT for key "1"
Frequency Domain representaion of the signal
Adversary's Device
Feature Extraction
Extracted f_L and f_H representing the DTMF tone combination
Training ML models on extracted data
Trained Models (SVM, RF, ANN)
Predicting the key (0,1,...,9)
Training Phase
Testing Phase

Fig. 2. The execution flow of the proposed attack methodology

based on the fact that the dial tone corresponding to each digit of a phone number is composed of the distinct frequencies specified by DTMF standards. Hence, if the frequencies present in the recorded dial tones are analyzed then the corresponding digit or key can be predicted by an attacker.

Adversarial Model: The adversarial model of our approach assumes that the victim downloads the malicious applications in his mobile phone and grants access to unnecessary permissions. For example, the victim can grant access to the device's microphone that in turn leads to recording of digits sound. We also assume that the sound of dialing digits on the device is enabled to get the tones. The malicious application installed in the mobile devices can access and take control of system's microphone. We assume that the malicious application which has been given the access to the microphone of victim's device sends the recording of dialling of a phone number by the victim to the adversary over the internet. This adversarial model is practical and complies with the standard adversarial models assumed for the state of the art attacks on smartphones [26].

The basic flow of our attack methodology is depicted in Fig. 2. This methodology is discussed in three major phases viz. (i) data collection and feature extraction (ii) training of ML models (ii) inference of dialed digits by attacker. In the *data collection and features extraction* phase, recording of the dialed digits are subjected to FFT technique followed by the features extraction with the help of DTMF standards. Further in the *training phase*, we use the features extracted in previous phase to train the ML models. Finally, in *prediction phase*, the recordings from the victim's device are processed and fed to the models to predict or infer the phone number dialed by the victim. The methodology is discussed in detail below.

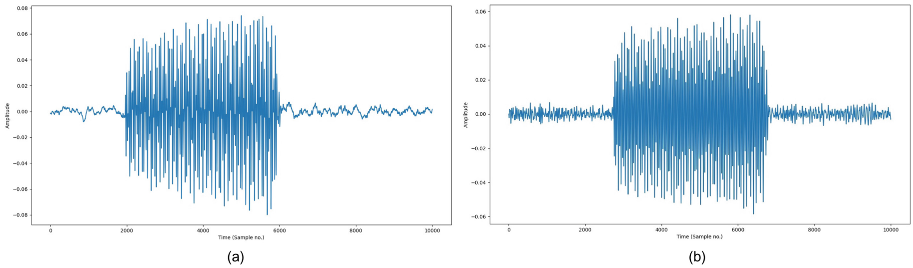
3.1 Data Collection and Feature Extraction

We have collected 210 samples of recordings corresponding to 10 digits on the keypad with 21 samples for each digit from a mobile device which acts as the adversary's device. All these samples represent the signals in time-domain where we have time on the x-axis and amplitude on the y-axis as shown in Fig. 3.

Application of FFT: Since the attack exploits the features of recorded sounds in the from of fundamental frequency components, therefore we first need to perform translation of recorded signals from time-domain to frequency-domain. To do so, we apply the FFT technique which computes the DFT of the given signal. Applying FFT on the signal from time-domain gives us the frequency distribution in the signal. This frequency distribution tells us the magnitude of each frequency that is present in the original signal. The frequency-domain representation has frequencies on the x-axis and magnitude on the y-axis. Figure 3 (a) and (b) represents the time-domain representation of a sample recording when a user dials the key '0' and key '1' on the dial pad respectively. When we apply the FFT on this signal, we obtain its corresponding frequency-domain representation which is shown in Fig. 4 (a) and (b).

Role of DTMF: Each key will have different frequency-domain representation where the two frequencies that define the DTMF tone of the key will have higher magnitude as compared to any other frequency. Figure 4 (b) shows the frequency-domain representation of a sample recording when a user dials the key '1'. As shown, the two frequencies with the higher magnitudes lie somewhere around 700 Hz and 1200 Hz and the frequencies which represent the DTMF '1' are 697 Hz and 1209 Hz. If we calculate the exact values from the above representation, they come out to be 696 Hz and 1208 Hz which is very close to the DTMF standards. This property holds for every key on the keypad and is common across all the mobile devices. Hence, this property enables an attacker to detect the keys by training the models on only one device. This makes the attack device-independent.

Feature Extraction: Once we have the frequency-domain representation of the recorded samples, the next step is to extract the two characteristic frequencies based on the DTMF standards for each of the 210 samples. To do so, we find the frequency positions/indices which have the highest magnitudes and then find the frequencies which are present at these positions/indices. After extracting these two frequencies, we created a dataset with these frequencies as the features. In

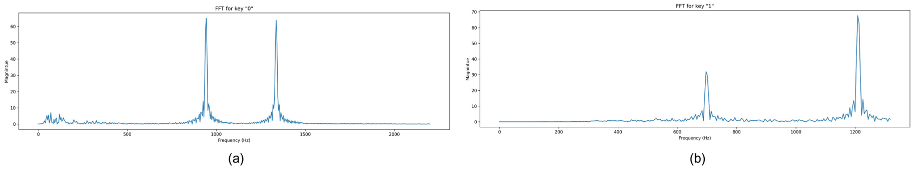


(a)

(b)

Fig. 3. Time-domain representation of (a) DTMF ‘0’ and (b) DTMF ‘1’

this dataset, the key/digit which corresponds to these frequencies acts as the label for classification. Thus, we have 10 classes (one for each key) to classify the collected samples. A row in our dataset has 3 values (f_L,f_H,label) representing the low and high frequencies and the actual label of the sample.



(a)

(b)

Fig. 4. Frequency-domain representation of (a) DTMF ‘0’ and (b) DTMF ‘1’

3.2 Training ML Models with Extracted Features

Post obtaining the features of the recorded samples, we have trained our ML models viz. SVM, Random forest and ANN using the corresponding dataset. For SVM, we have used linear kernel for classification because our data is linearly separable. In other words, the data can be separated using a single line. Training an SVM with linear kernel is much faster than any other kernel. In Random forest regression, we have used 10 decision trees to classify the samples and then the majority prediction from these 10 decision trees is used as the final label for the sample. For ANN model, we have created a neural network with 3 hidden layers with 2-dimensional input layer and 10-dimensional output layer. ANN is widely used when the testing data is not much different than the training data and same is the case for our attack methodology, hence we decide to incorporate this model into our attack strategy.

3.3 Inference of Dialed Digits by Attacker

In the final phase of the acoustic SCA methodology, the trained models are used to predict the 10-digit phone number dialed by the victim. The prediction is accomplished in the following steps.

- The malicious application that has been granted the permission to access the microphone of victim's device can record while the victim is dialling a phone number.
- This recording can be sent to the adversary over the internet and then the adversary can perform the required operations on this recording to depict the phone number. Specifically, the recording contains a series of 10 DTMF tones representing the 10-digit phone number.
- Now, we need to separate the 10 DTMF tones from this recording and predict the digit for each tone. To do so, we have plotted the recorded signal in time-domain and observed the time instance at which a particular tone starts and ends.
- Further, we extract the signal containing only one tone from this original signal on the basis of these start and end times. We perform this process for all the 10-digits by observing their respective starting and ending times.
- Once we have the signal for one DTMF tone representing a digit from the phone number, we perform the steps viz. conversion into frequency-domain representation and feature extraction on this signal. These steps are similar to what we performed in the training phase discussed in the Sect. 3.2.
- Once we have created a feature vector from the test sample, we feed it as an input to the ML model to predict the respective key/digit.

Thus, the proposed SCA attack methodology is capable to infer a digit of the phone number in this phase. Similarly, we infer all the remaining digits of the phone number by following the same process.

4 Experimental Setup and Results

This section first discusses the required experimental setup to perform the proposed acoustic SCA for retrieving phone number from mobile devices. As discussed earlier, our attack methodology is performed in the different phases viz. data collection and feature extraction, training and inference. The necessary experimental setup in these phases has been discussed in this section. Later, we illustrate the results of our approach in terms of the accuracy of prediction of the dialed digits and implementation complexity or estimated attack time of the proposed acoustic SCA methodology using different ML models.

4.1 Experimental Setup for Data Collection, Feature Extraction and Training

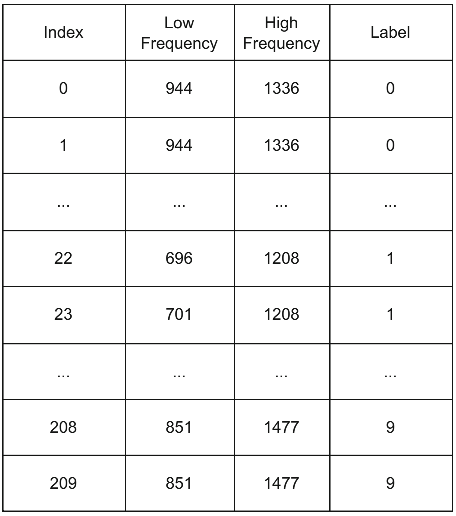
We have recorded 210 samples of dialling a digit (21 samples for each digit) in one mobile device which will act as the adversary's device. In our case, we have used Samsung Galaxy M31 as the adversary's device. We have used the usual sound recorder application which comes pre-installed in almost all the devices. The recordings are sampled at a sampling rate of 44.1 kHz, which is very common in recent mobile applications. We recorded the samples in mono mode of recording which records from the main or default microphone. We have worked with these samples using python libraries (i) *librosa* which is very famous library for music and audio analysis, (ii) *numpy* to generate the frequency-domain representation from the time-domain representation using FFT method, (iii) *matplotlib* to plot various signal representations and results of our work.

Post obtaining the samples, we have transferred them to the machine on which all the processing will happen. Each acoustic signal (sample) has a duration of nearly one second which is long enough to capture the behavior of the signal. We then follow the below mentioned steps to create the dataset from the recorded samples.

- The time-domain signal is converted to frequency-domain by the FFT method of *numpy*.
- The frequency-domain representation of the signal has an array of frequencies present in the signal along with their magnitudes. We choose the two frequencies having the highest magnitudes.
- We have created a row in the dataset containing these two frequencies as the features and the key/digit to which these frequencies belong as the label.

We have performed the above steps for all the recorded samples and obtained a dataset in a comma separated values (csv) format which will be used to train the ML models. The dataset is of size 210×3 representing 210 samples, each having 3 values namely, low frequency, high frequency and its label. A few samples from the dataset have been shown in Fig. 5.

Once the dataset is obtained, it is used to train our ML models i.e. SVM, Random Forest and ANN. The SVM and Random forest models have training and testing data in 80 : 20 ratio. For ANN model, we have used 3 hidden layers with 100 neurons in each layer. The input layer is 2-dimensional and the output layer is 10-dimensional, representing 10 classes. We have used rectified linear activation function or ReLU for the input and the hidden layers while softmax function for the output layer. The softmax function outputs the probability distribution for each of the 10 classes. The class with the highest probability is chosen as the label for the given sample. We have split the dataset into 9 : 1 ratio representing the training and the testing set to train the ANN model. The number of epochs are taken as 100 which means that we feed the training data 100 times to the neural network and each time the weights are updated such that the loss will be minimized. We have saved all these models in .sav format using python library pickle so that we can directly use them in the future for prediction without having to train all of them again.



Index	Low Frequency	High Frequency	Label
0	944	1336	0
1	944	1336	0
...	...	...	...
22	696	1208	1
23	701	1208	1
...	...	...	...
208	851	1477	9
209	851	1477	9

Fig. 5. An excerpt of our Dataset used for training

4.2 Experimental Set-up for Inferring the Digits of a Phone Number

We assume that the malicious application which has been given the access to the microphone of victim's device sends the recording of dialling of a phone number by the victim to the adversary over the internet. We have recorded the dialling of phone numbers on a number of devices like Realme GT, Motorola, Iphone and Lenove tablet to validate the attack. These recordings contain a sequence of 10-digit DTMF tones for the 10 digits in a phone number. A typical recording of dialling a 10-digit phone number looks like that in Fig. 6. We can see multiple peaks in the signal, each of which represents a certain digit in the dialed phone number.

Further, we separate these 10 peaks by estimating their start and end time. We then obtain the frequency-domain representation of each peak/digit using FFT followed by finding the two characteristic frequencies that represent this digit. We also load the models which were saved after the training phase. When we obtain the two characteristic frequencies for each digit, we treat them as the test data for our models and feed them to the models to get their predicted digit as the output. When this process is repeated for each of the 10 digits, we retrieve the full 10-digit phone number which was dialed by the victim. The Fig. 7 shows the flow of proposed attack of inferring the phone digits that are being dialed in the victim's device. As shown, the recording of phone number retrieved by the malicious app from the victim's device is sent to the adversary's machine over the internet where the adversary executes the attack and predicts the phone number from the received recording.

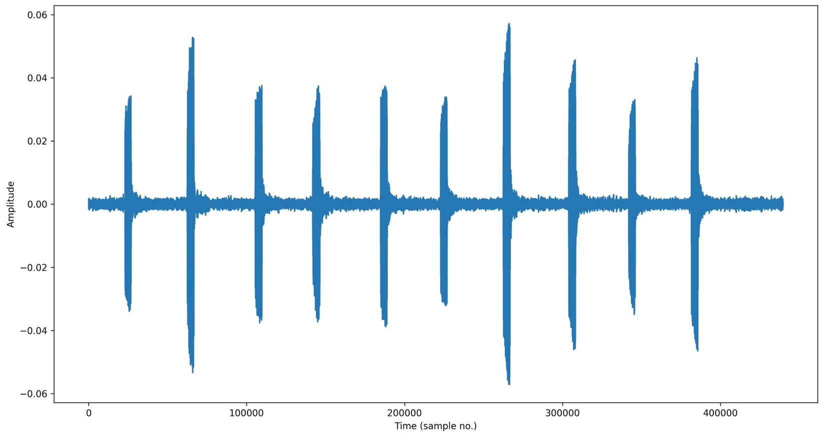


Fig. 6. Recording of a typical phone number

4.3 Accuracy of Prediction of the Dialed Digits Using the Proposed Acoustic SCA Methodology

The accuracy is measured as the ratio of the number of samples predicted correctly to the total number of samples in the testing set.

$$Accuracy = \frac{No.\ of\ samples\ predicted\ correctly}{Total\ no.\ of\ samples} \tag{5}$$

Figure 8 shows the loss and accuracy for all the 100 epochs in case of the ANN model. We have used the cross entropy loss as the loss function in our model and the optimizer we used is the adam optimizer. As shown in Fig. 8, the accuracy for this model reaches to 100% and the loss decreases to 0 when the training phase ends. The reported accuracy for all three models is 100%. For the accuracy analysis, we have varied the number of samples per digit from 3 to 21. However for each case, we are achieving 100% accuracy. The underlying reason is as follows. The two characteristic frequencies in the dial tone of corresponding digits differ by a large value. More explicitly, low frequencies are 70 Hz apart and high frequencies are 120 Hz apart. Therefore, it is highly unlikely that a digit would be predicted incorrectly. For example, the characteristic frequencies '770 Hz and 1209 Hz' corresponding to the digit '4' cannot be predicted to be any other digit by the model as its corresponding frequencies are far away from that of other digits. However, we still need to train the classifiers to capture the small difference between the frequencies. For example, in case of the digit '1' the low frequency might vary 697 Hz to 771 Hz due to the noise present in the recorded signal or the limitations of the device's microphone which might result in predicting it incorrectly. Hence, the goal of the classifiers is to capture these small differences using decision boundary and these differences will be captured better if we train with large number of samples.

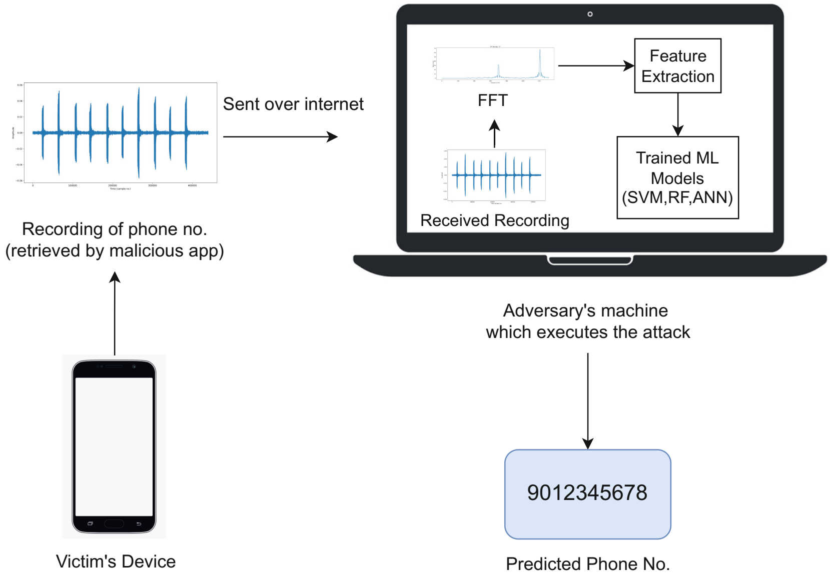


Fig. 7. The proposed attack flow of inferring the phone number from recording

Since, the DTMF tones and their corresponding fundamental frequencies remain the same irrespective of the type of mobile device. Hence, all the 10 digits predicted by the models represent the exact phone number dialed by the victim. We have successfully retrieved the dialed phone numbers on all the devices mentioned earlier by following this method. Hence, we propose that our attack methodology is device independent.

4.4 Implementation Complexity (Estimated Attack Time) of the Proposed Methodology

We have executed this attack on a system having 8 GB of RAM, AMD PRO A4-3350B APU 2 GHz Processor. The overall implementation complexity of the proposed attack methodology is divided in the following three time slices:

- implementation run time of finding FFT and feature extraction (T_f).
- implementation run time of training ML models (T_t).
- implementation run time of predicting dialed digits (T_p).

Hence, the total implementation complexity or overall attack time (T_A) is given using the following equation:

$$T_A = T_f + T_t + T_p \tag{6}$$

The implementation run time of finding FFT and feature extraction (T_f) is **12.49** s. This process is performed only once and is common for all the ML

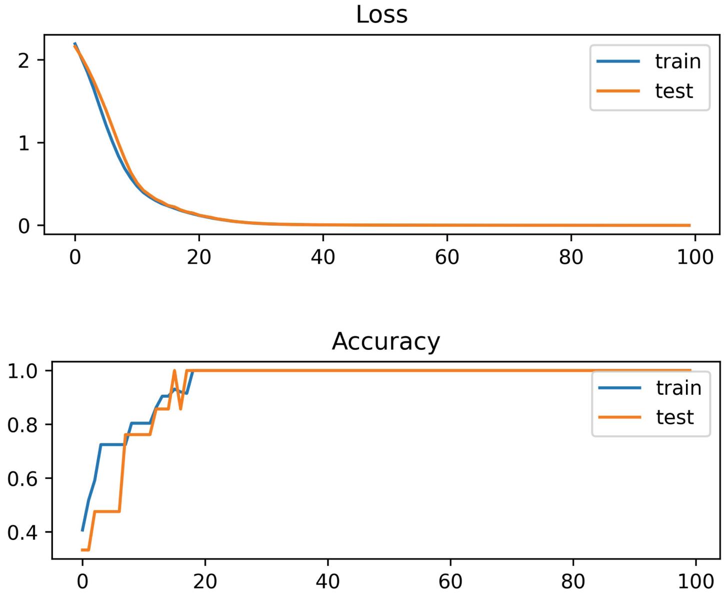


Fig. 8. Training loss and accuracy for ANN model

models. Further, the time required (in seconds) for training all the three ML models after performing the feature extraction and then testing them to predict the dialed digits is given in Table 2. Finally, the Table 2 also presents the total attack time computed using Eq. 6. The implementation complexity of retrieving the dialed digits is achieved to be very less. This is because, the ML model needs to be trained only once and later it can be used to launch the attack on any type of device. It does not required to be trained separately for the type of device on which the attack is intended to be launched. Hence, it leads to lower implementation complexity or attack time.

Table 2. Implementation run times in seconds (s) of the proposed attack for three different ML models

Model	Training time (T_t)	Testing time (T_p)	Overall attack time (T_A)
SVM	0.12 s	0.20 s	12.81 s
Random forest	0.08 s	0.28 s	12.85 s
ANN	9.17 s	0.75 s	22.41 s

5 Conclusion

Acoustic side channel attack has recently come up as a potential threat as it can breach the security of mobile devices and leak the user's sensitive data. In this work, we have shown how a malicious application in the victim's device can retrieve the phone numbers that are being dialed by him/her by recording the sounds through in-built microphone of the device. Since the DTMF tones of the keys/digits are common across all mobile devices, the adversary can exploit this property to launch a successful attack on the mobile devices making this attack device independent. The adversary just needs to train a machine learning model on his/her device to make this attack successful. The experimental results implied that an adversary can retrieve the phone number digits using the ML models with 100% accuracy. Moreover, the implementation complexity or the overall attack time of the proposed acoustic SCA methodology is very less. For a user to be less likely to fall victim to this kind of attack, he or she needs to pay close attention to the hardware and software requirements of the applications he or she wants to install. Demanding access by an application to an irrelevant component of the mobile device can be a big giveaway of such an attack.

References

1. Naik, P., Chatterjee, U.: Network data remanence side channel attack on SPREAD, H-SPREAD and reverse AODV. In: Batina, L., Picek, S., Mondal, M. (eds.) SPACE 2021. LNCS, vol. 13162, pp. 129–147. Springer, Cham (2022). https://doi.org/10.1007/978-3-030-95085-9_7
2. Kocher, P.C.: Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems. In: Koblitz, N. (ed.) CRYPTO 1996. LNCS, vol. 1109, pp. 104–113. Springer, Heidelberg (1996). https://doi.org/10.1007/3-540-68697-5_9
3. Rivest, R.L., Shamir, A., Adleman, L.M.: A method for obtaining digital signatures and public-key cryptosystems (reprint). Commun. ACM **26**(1), 96–99 (1983)
4. Diffie, W., Hellman, M.E.: New directions in cryptography. IEEE Trans. Inf. Theory **22**(6), 644–654 (1976)
5. Kocher, P., Jaffe, J., Jun, B.: Differential power analysis. In: Wiener, M. (ed.) CRYPTO 1999. LNCS, vol. 1666, pp. 388–397. Springer, Heidelberg (1999). https://doi.org/10.1007/3-540-48405-1_25
6. Carrara B.: Air-gap covert channels (2016)
7. Huang, H., Wang, X., Jiang, Y., Singh, A.K., Yang, M., Huang, L.: Detection of and countermeasure against thermal covert channel in many-core systems. IEEE Trans. Comput. Aided. Des. Integr. Circ. Syst. **42**, 252–265 (2021)
8. Ji, X., et al.: No seeing is also believing: electromagnetic-emission-based application guessing attacks via smartphones. IEEE Trans. Mob. Comput. (2021)
9. Carrara, B., Adams, C.: On acoustic covert channels between air-gapped systems. In: Cuppens, F., Garcia-Alfaro, J., Zincir Heywood, N., Fong, P.W.L. (eds.) FPS 2014. LNCS, vol. 8930, pp. 3–16. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-17040-4_1
10. Halevi, T., Saxena, N.: Keyboard acoustic side channel attacks: exploring realistic and security-sensitive scenarios. Int. J. Inf. Secur. **14**, 443–456 (2015)

11. Asonov, D., Agrawal, R.: Keyboard acoustic emanations. In: Proceedings of IEEE Symposium on Security and Privacy, pp. 3–11 (2004). https://doi.org/10.1109/SECPRI.2004.1301311
12. Al Faruque M. A., Chhetri S. R., Canedo A. and Wan J.: Acoustic side-channel attacks on additive manufacturing systems. In: 2016 ACM/IEEE 7th International Conference on Cyber-Physical Systems (ICCPS), pp. 1–10 (2016). https://doi.org/10.1109/ICCPS.2016.7479068
13. Al-Haiqi, A., Ismail, M., Nordin, R.: On the best sensor for keystrokes inference attack on android. Procedia Technol. **11**, 989–995 (2013). (4th International Conference on Electrical Engineering and Informatics, ICEEI 2013)
14. Cai, L., Chen, H.: TouchLogger: inferring keystrokes on touch screen from smartphone motion. In: Proceedings of the 6th USENIX Conference on Hot Topics in Security, HotSec 2011, p. 9. USENIX Association, Berkeley, CA, USA (2011)
15. Goller, G., Sigl, G.: Side channel attacks on smartphones and embedded devices using standard radio equipment. In: Mangard, S., Poschmann, A.Y. (eds.) Constructive Side-Channel Analysis and Secure Design: 6th International Workshop, COSADE 2015, Berlin, Germany, 13–14 April 2015. Revised Selected Papers, pp. 255–270 (2015)
16. Li, M., et al.: When CSI meets public WiFi: inferring your mobile phone password via WiFi signals. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, CCS 2016, pp. 1068–1079, New York, NY, USA (2016)
17. Negulescu, M., McGrenere, J.: Grip change as an information side channel for mobile touch interaction. In: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, CHI 2015, pp. 1519–1522, New York, NY, USA (2015)
18. Sarkisyan, A., Debbiny, R., Nahapetian, A.: WristSnoop: smartphone pins prediction using smartwatch motion sensors. In: 2015 IEEE International Workshop on Information Forensics and Security (WIFS), pp. 1–6, November 2015
19. Simon, L., Anderson, R.: Pin skimmer: inferring pins through the camera and microphone. In: Proceedings of the Third ACM Workshop on Security and Privacy in Smartphones, vol. 38; Mobile Devices, SPSM 2013, pp. 67–78, New York, NY, USA (2013)
20. Xu, Z., Bai, K., Zhu, S.: TapLogger: inferring user inputs on smartphone touchscreens using on-board motion sensors. In: Proceedings of the Fifth ACM Conference on Security and Privacy in Wireless and Mobile Networks, WISEC 2012, pp. 113–124, New York, NY, USA (2012)
21. Yan, L., Guo, Y., Chen, X., Mei, H.: A study on power side channels on mobile devices. In: Proceedings of the 7th Asia-Pacific Symposium on Internetware, Internetware 2015, pp. 30–38, New York, NY, USA (2015)
22. Shumailov, I., Simon, L., Yan, J., Anderson, R.: Hearing your touch: A new acoustic side channel on smartphones (2019)
23. Schuldt, C., Laptev, I., Caputo, B.: Recognizing human actions: a local SVM approach. In: Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004, vol. 3, pp. 32–36. IEEE, 26 August 2004
24. Sarica, A., Cerasa, A., Quattrone, A.: Random forest algorithm for the classification of neuroimaging data in Alzheimer's disease: a systematic review. Front. Aging Neurosci. **6**(9), 329 (2017)
25. Jain, A.K., Mao, J., Mohiuddin, K.M.: Artificial neural networks: a tutorial. Computer **29**(3), 31–44 (1996)

26. Narain, S., Sanatinia, A., Noubir, G.: Single-stroke language-agnostic keylogging using stereo-microphones and domain specific machine learning. In: Proceedings of the 2014 ACM conference on Security and privacy in wireless and mobile networks (WiSec 2014). Association for Computing Machinery, New York, NY, USA, pp. 201–212 (2014). https://doi.org/10.1145/2627393.2627417
27. International Telecommunication Union: ITU-T Recommendation Q.23, Technical features of pushbutton telephone sets, Fascicle VI.1, Blue Book (1993)