Acoustic Keylogger

HAN Xicun

Director : Christophe. Rosenberger

Informatique M2 , Science

University of CAEN

xicun.han@gmail.com

24 February 2017

Table des matières

Résumé

When assessing the security of security systems, it is fatal to take a consideration of conceptual new attacks, as countermeasures nowadays can only be implemented against known threats. With this consideration, in this report, we are going to explore the acoustic side channel attack and evaluate each one.

In this report we are going to review the related works and bring analysis to the characteristics of each one, and at last perform a empirical study of one approach.

1 Introduction

Nowadays, password is still performing an important role in our daily authentication process. It has been used as a primary means to get access to the Internet resources, the sensitive user data or even more to the banking payment process.

As most of the authentication method, it is now facing a verity of security problems, such as brute-force attack, key logger, vibration side channel attack, timing attack, electromagnetic radiation attack etc. In this paper we will focus on the empirical implementation of the acoustic side channel attack with different strategies.

1.1 Password and Related Security Issues

The password, as we mentioned, is used with an identity to gain access to certain resources, and its security is often questioned. It has been proved to be weak against brute force attacks, keyloggers as well as side channel attacks.

A brute force attack can be easily performed by trying each possible combination of alphabets, since a user is often in the habit of putting his or her personal information into the password and moreover keeping it short so that he / she can easily remember, this classical method is sometimes very effective.

A keylogger can be a malware in the system or hardware memories the typing of user.

Side channel attack on the other hand, is harder to detect and defend, because it measures the implementation of an algorithm beyond the algorithm itself. For example, timing attack through the internet measures the reacting-time of IP package, the power monitoring attack measures the power consuming during each computation. In this project we will analyses the acoustic side channel attack.

1.2 Acoustic Side Channel Attack

The acoustic side channel attack is based on the acoustic emanations produced by electronic devices such as a US-101 standard keyboard or a specified keyboard used by Bank, the conspicuous sounds produced by typing can be used to learn information about the input data.

Recent studies utilize a variety of methods in order to capture the features of one's in order to lance this type of attack.

2 Related Works Overview

Asonov and Agrawal[1] were the first researchers to demonstrate the threat of side channel attacks using acoustic leakage from the keyboard. They used the Fast Fourier Transform (FFT) features of the extracted keystroke as an identifier and use a neural network to classify and recognize the keystrokes. This process involved a training phase that used labeled data pair consisting of a key and its corresponding feature, and a testing phase that took a feature as an input and the output consisted of the closest matching key.

Zhuang et. al[2]. extended the work of Asonov and Agrawal by using cepstrum features, in particular Mel-Frequency Cepstrum Coefficients (MFCC) as identifiers for the keystrokes and used unlabeled data in the training phase for the neural network unlike Asonov and Agrawal.

Berger et al.[3] used cross correlation between the recorded keystroke signals and euclidean distance between frequency based features to classify and recognize the keystrokes. They then used dictionary based attack to reconstruct the text from the recovered keystrokes.

Halevi and Saxena[4] combined the cross correlation information between the two keystrokes signals and the frequency distance measure from the work done by Berger et al. to create a new feature called time-frequency classification. This new feature was used for identifying different keystrokes and then used for password detection. They also studied the effect of various typing styles (hunt and peck, and touch typing) on keystroke signal similarities and found out that the signal similarity decreases with change in the typing style.

Fiona[5] presented a distance-time based triangulation attack that is able to identify a keystroke by recording the keystroke with multiple microphones. Due to fixed location of each key on the keyboard, the sound recorded by each microphone arrives at a different time and the time delay for each keystroke can be used to distinguish between the keystrokes

2.1 Asonov and Agrawal

In 2004, at the very beginning, the keyboard acoustic emanations was recognized as a threat by Dmitri Asonov and Rakesh Agrawal from IBM Almaden Research Center in their paper.

The basic idea is the assumption that the sound of clicks differ slightly from one key to another. In this way, features can be extracted from these different sound of keystrokes.

Their methodology :

- The press and release action produced different sounds. It is found that knocking at different parts of a mechanical keyboard plate produce different sounds. It is like striking at the different locations of a drum will produce different sounds.

- Within the sounds of thirty keys of a single PC keyboard they recorded, they found out that the Fast Fourier Transform values of the press peaks give significant variation for perceptually similar sounds.

- The frequency distributions were passed into a neural network for classification. The neural network was trained with pairs {key, feature}

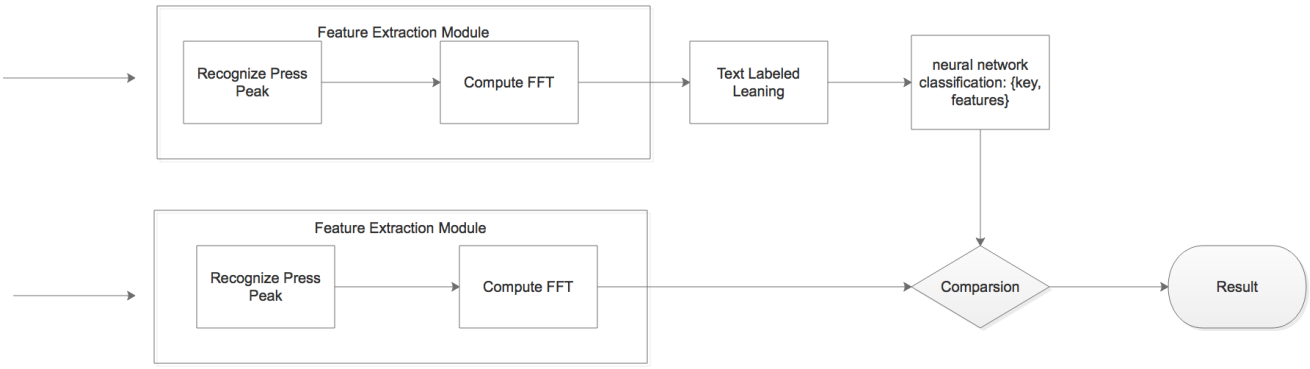

FIGURE 1 – The Asonov and Agrawal proposal

Through experimenting with several sets of data, the best recognition rate can be achieved by including the entire active interval in the feature extraction, the 300-3400 Hz interval is approved to be the most informative.

The accuracy rate for a key to be recognized correctly is approximately 80%. Besides the facts that different keyboards produce different sounds, the sound for each key is unique, theoretically, the keys can be recognized regardless of the keyboard type and the typing style, but as their experiment is held by one person, the performance would have been affected if they would have tried to recognize the typed keys on a different key board and with another tying style.

2.2 Zhuang et. al

From the previous introduction to Asonove and Agrawal's attack, we recognize immediately the limit of this model. Their attack requires neural network training with text-labeled samples, and for the most of the time in reality, the training sample of a significant length is difficult to obtain.

Zhuang Zhou and al. published another proposal which is developed by Zhuang, F.Zhou and J.D.Tygar in the year 2005. They argue that a labeled training sample requirement is unnecessary for an attacker, as they mentioned in their paper " *The key insight in our work is that the typed text is often not random, when one types English text, the limited number of English words limits the possible temporal combinations of keys, and the English grammar limits the word combinations.*" This fact makes the keyboard emanation attacks a more serious security problem.

Given the ideal assumption that each key sounds exactly the same then it is pressed, then the problem can be considered as a machine learning version of substitution ciphers. The Figure 2 captures an overview of their approach.

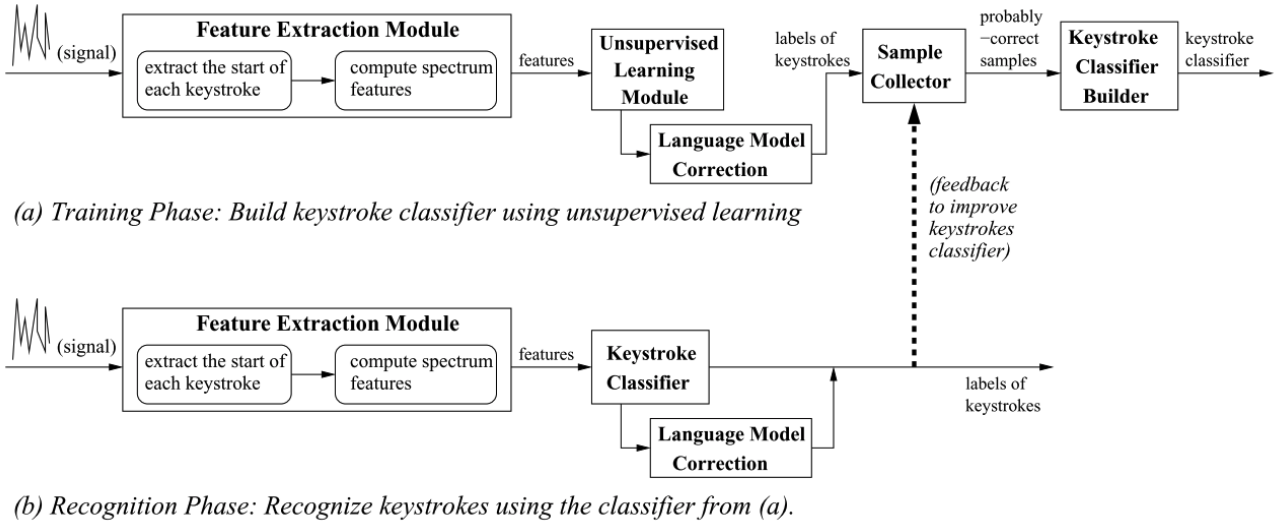

FIGURE 2 – Over View of the Zhuang Zhou Tygar Model

Keystroke Recognizing

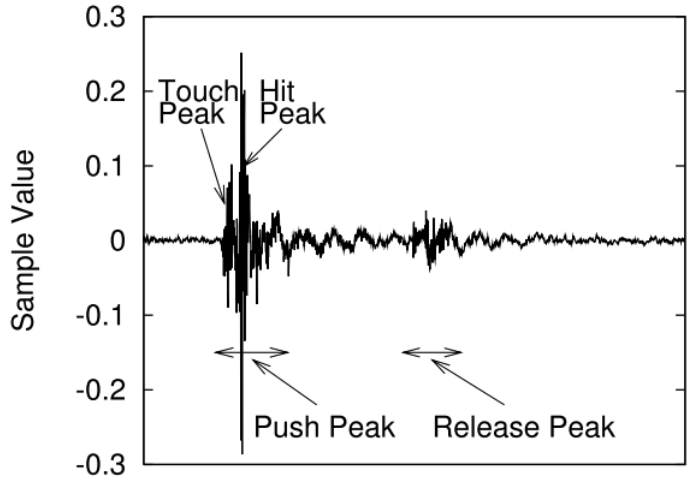

FIGURE 3 – The audio signal of a key stroke

As shown in Figure 3, a keystroke consists a push and a release actions which can be easily observed. Studies of Asonove and Agrawal have shown that the period from push to release is typically about 100 milliseconds[6]. The fact to have more than 100 milliseconds between consecutive key strokes allows us to distinguish the consecutive keystrokes and find the beginning for each window as it is shown in Figure 4. They calculate windowed

discrete Fourier transform of the signal and use the sum of all FFT coefficients as energy, then use a threshold to detect the start of each one.

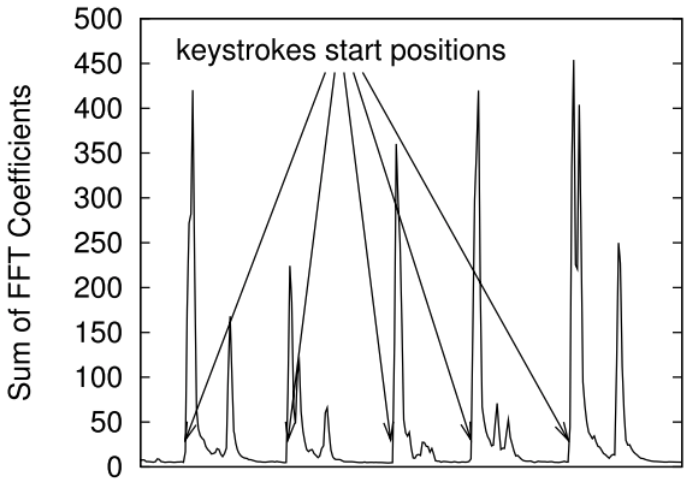

FIGURE 4 – Energy levels over the duration of 5 keystrokes

2.2.1 Cepstrum features extraction

In the part of feature extraction, in their paper, it has been proved that the Cepstrum features which is widely used in speech analysis and recognition has empirically higher performance than the plain FFT coefficients, In particular, they used Mel-Frequency Cepstral Coefficients (MFCCs) in the experiments.

2.2.2 Unsupervised key recognition

As we discussed before, our ideal aim is to build an one-to-one mapping between the keys and classes. however, as the variability of the situation, one key stroke can be in different classes, at the same time, one class can contain several keystrokes. In the end, a particular key will be in each class with a certain probability.

The next work is trying is trying to find the sequence of keystrokes based on a sequence of classes for each keystroke. A **HMM(Hidden Markov Models) algorithm** will be served for this prediction.

In a hidden Markov model, the state is not directly visible, but the output, dependent on the state, is visible. Each state has a probability distribution over the possible output tokens. Therefore, the sequence of tokens generated by an HMM gives some information about the sequence of states. The adjective 'hidden' refers to the state sequence through which the model passes, not to the parameters of the model ; the model is still referred to as a 'hidden' Markov model even if these parameters are known exactly.

For example, if the current key can be either "h" or "j" (e.g. because they are physically close on the key- board) and we know the previous key is "t", then the current key

is more likely to be “h” because “th” is more common than “tj”.

2.2.3 Spelling and Grammar Checking

In the work of Zhuang Zhou et Tygar, they implemented :

- A dictionary based spelling correction
- A simple static model of English grammar.

2.2.4 Feedback based training

Feedback training procedure will not use the English spelling and grammar model in order to detect a password. The previously obtained corrected results is used as a labeled training samples. In this way a circle is formed, and the spelling and grammar correction, on the other hand is used as a quality indicator. The same feedback procedure is done repeatedly until no significant improvement is seen.

2.3 Berger et al.

Yigael Berger, Avishai Wool and ArieYeredor (Berger, et al., 2006) present a dictionary attack based on their previous research, this attack does not require any training and works on an individual recording of typed word and is efficient for the short patterns which we always chose as password.

This attack is made up from several steps :

- Signal Processing and Feature Extraction
- Keystroke Processing
- Constraint Formulation
- Constraint Evaluation
- Outcome Prioritization

2.3.1 Signal Processing

As we mentioned before, the keystroke is consist of the action of PRESS and the action of RELEASE. The out put of this part should be two arrays of signal segments consist $PRESS_i$ and $RELEASE_i$ separately.

2.3.2 Keystroke Processing

Our goal is always seeking for a metrics $sim(K_1, K_2)$ which will allow us to distinguish two keystrokes, and particularly in this part, they aimed at finding a metrics that will represent the keys’ physical proximity on the keyboard.

FIGURE 5 – Physical Proximity Measure

That is to say, as shown in Figure 5, $sim(K_i, K_k) < sim(K_i, K_n)$ implicate the physical distance between key i and key n is further than that between key i and key k. For this propose the most adaptive metric is the cross-correlation metric who has a even better performance than fft and Cepstrum.

In this way, we calculate the $sim(K_1, K_2)$ for each $PRESS_i$ to every other press and the same with each $RELEASE_i$. After the calculation we will have two matrix the $N * N$. In the end, they found out that an unweighted average method perform good in combining the Press Similarities with the Release Similarities.

2.3.3 Constraint Formulation

With the function $sim(K_1, K_2)$ whom we calculate the distance, 4 kinds of relations between two keys are defined :

Type	Notation	Meaning
EQ	$=$	$K_i = K_j$ means that the i'th keystroke and the j'th keystroke stem from the same key on the keyboard.
ADJ	$\simeq$	$K_i \simeq K_j$ means that the j'th keystroke stems from a key that is adjacent to the key which the i'th keystroke stems from. For example, $Q \simeq W$ but not $Q \simeq E$ since E is located two positions away from Q on a QWERTY keyboard.
NEAR	$\sim$	$K_i \sim K_j$ means that K_i and K_j are at most two keys apart on the keyboard, e.g., keys *NEAR* G include R, D, N, J, etc.
DIST	$\nsim$	Distant keys are those that are not *NEAR* to each other.

FIGURE 6 – The 4 Constraint types

A word will produces a set of constraints, but a sequence of constrains corresponds to several different sets of word. For example, the word "help" in a QWER keyboard produces the following constrains : (DIST$< 1, 2 >$, DIST$< 1, 3 >$, DIST$< 1, 4 >$, DIST$< 2, 3 >$, DIST$< 2, 4 >$, ADJ$< 3, 4 >$). Meanwhile, we will find that "nose", "path" etc. have exactly the same combination.

2.3.4 Constraint Evaluation

The process of evaluation is based on the assumption that we know the length of the word N, what we will do is to go over all permutation the words of length N and output all those who match the constraints.

They also use some measures to reduce the errors, for the reason that if in the sequence of constraints, there's an error happened, we will never get the good response. For this purpose, two metrics for each constraint type are implicated :

- Precision :True positives/Total number of positives predicted = TP/(TP+FP)
- Recall : True positives/Total number of actual positives = TP/(TP + FN)

And in order to get the best combination, they came up with the idea of *BestFriendsPickPolicy* with which we can generate the possible arrangement based on the physical positions of each key on the keyboard.

2.3.5 Dealing with False Constraints

Given that all we have argued are true, after the evaluation, many combinations will be consistent with the correct word. However, this method will produce eventually the possible combinations which is explosive. So as to overcome this explosion, they empirically found that 1000 random chosen combinations are usually suffice, then for each combination c, c is evaluated against the possible words in the dictionary. As an output, we will have L_c that conform with the selected constraints of combination c.

2.3.6 Outcome Prioritization

Based on L_c lists, For each word w in the dictionary we count the number of combinations c for which $w \in L_c$ and sort the words in decreasing order and after trying enough combinations the correct word will appear near the top of the sorted unified list.

2.4 Halevi and Saxena

As we discussed, The study of Halevi and Saxena brings optimization above the Detection techniques and analyze the influence of user typing styles.

2.4.1 Time-frequency Classification

We have already discussed several technologies of detection of signal :

Technique	description
Dynamic Time Warping	an algorithm which measures the similarities between sequences (calculate the simple distance measure between two signal vectors) Usage : - Collect Letter Date Set{key, samples} - Compare to get best match
Cross-Correlation	Algorithm measures the similarity : - Signal normalized according to their energy - calculated for the press and release regions. Usage : - Construct Letter Data Set - Calculate the average of their Cross-correlation measurement - find matches with the highest similarity.
Frequency-based Distance Measure	Algorithm measures the similarity : - calculate Euclidean difference for PRESS and RELEASE parts - calculate Average as a distance measure Usage : - calculate the distance between each test sample and each letter Data Set - output the letter with the smallest distance.
Frequency Features and Neural Networks	implement the frequency-domain features, using MFCC features as input to neural networks.

With an empirical evaluation, a table of detection rate out of 26 alphabet letters is constructed as shown in the Figure 7, from their observation, the Cross-Correlation has a relatively high score, and when they combined the correlation calculation with the frequency based calculations to choose the best-matching letter, they got a even higher score and the named it "Time-Frequency Classification".

Method	Detection Rate
DTW	46.15%
Cross-Corr	73.08%
Freq-Distance	63.46%
MFCC-Neural Networks	56.73%
Time-Frequency	82.69%

FIGURE 7 – Single Character Detection

As described in Figure 8, in the time frequency classification method. Firstly we calculate the Cross-Correlation denote F and the Frequency Distance denote C from the instance. Then, for the purpose of having the same ascending order, DC is calculated from C where $DC_i = 1 - C_i$.

In order to combine two factors, we have a lot of proposals :

- Picking the minimum of each value (min(DC,F))
- Getting the average of the two values : Mean(DC,F)
- Considering (F,DC) as a point on a 2-D space and calculate the Euclidean distance from zero

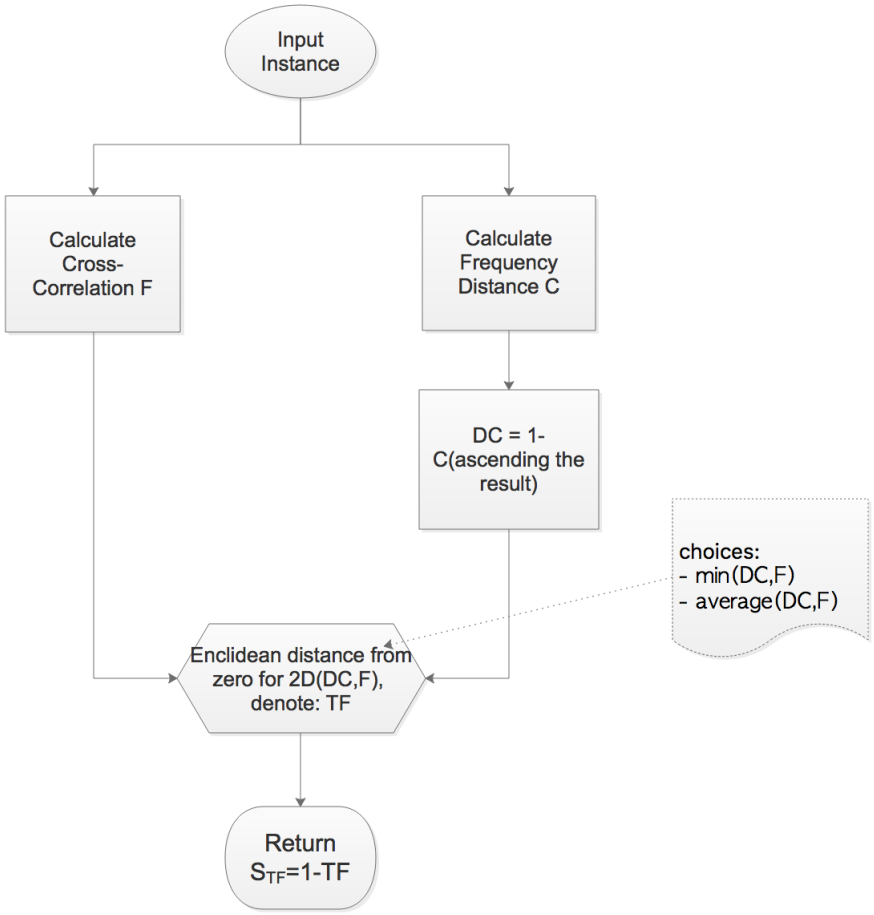

FIGURE 8 – Overview of Time Frequency Classification

At last, to get a result of decreasing order we return the $S_{TF} = 1 - TF$.

2.4.2 Typing styles

The authors categories typing styles in 3 for performance testing :

Style	Features
Straw Man Approach :	Typing each letter multiple times using the same finger. - No overlap of keyboard - Approximately the same force - Hit the key from the same angle
Hunt and Peck :	random words typed - Type the letter consecutively different - Hit the key from different angles
Touch Typing :	each key has its own designated finger and the rest fingers can possibly touch the key - The keys are hit from different angles - Hand position and finger choice are important

2.5 Fiona

A.Fiona proposed in 2006 published a empirical study of the acoustic triangulation attack based on a distance-time theory.

2.5.1 Hypothesis

Triangulation is defined as seeking the distance of a point using the concept of triangle, as shown by Figure 9 below.

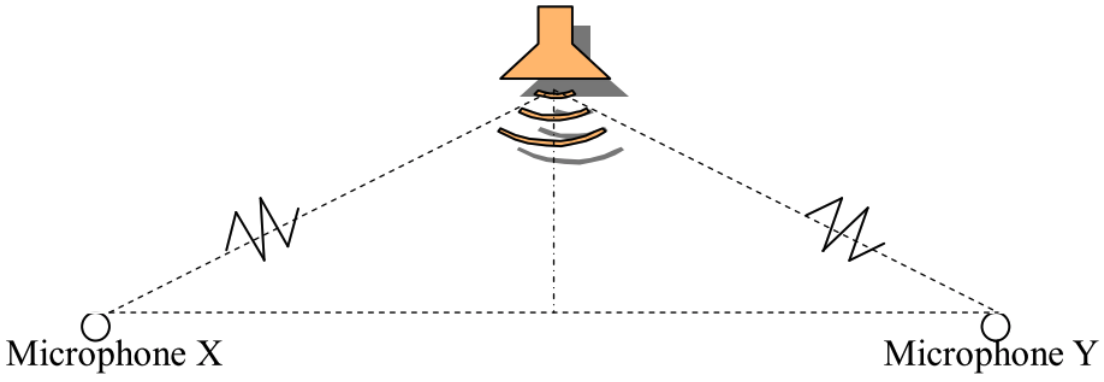

FIGURE 9 – Acoustic Triangulation Description

By detecting and measuring the differences in arrival times from two microphones, the location can be easily found.

Given that the distance between key i and microphone X be D_{ix} and the one between key i and microphone Y be D_{iy}, and the keystroke emits a sound which has a velocity in the air :V.

The difference between two microphones can be calculated as follows :

time delay of the sound of the key i is ti :

t_i = the time required for the sound wave to reach microphone X - the time required for the sound wave to reach microphone , which is :

$t_i = D_{ix}/V - D_{iy}/V = (D_{ix} - D_{iy})/V$

The hypothesis is that the t_i we calculate is unique for each key.

2.5.2 Compare Technologies

In the study of A.Fiona, 2 approaches is suggested for the comparisons arriving times between microphone X and microphone Y.

The Maximum Peak Position captures the maximum peak for each PRESS and RELEASE pair, and compare the time difference between T1-T2 as shown in Figure 10

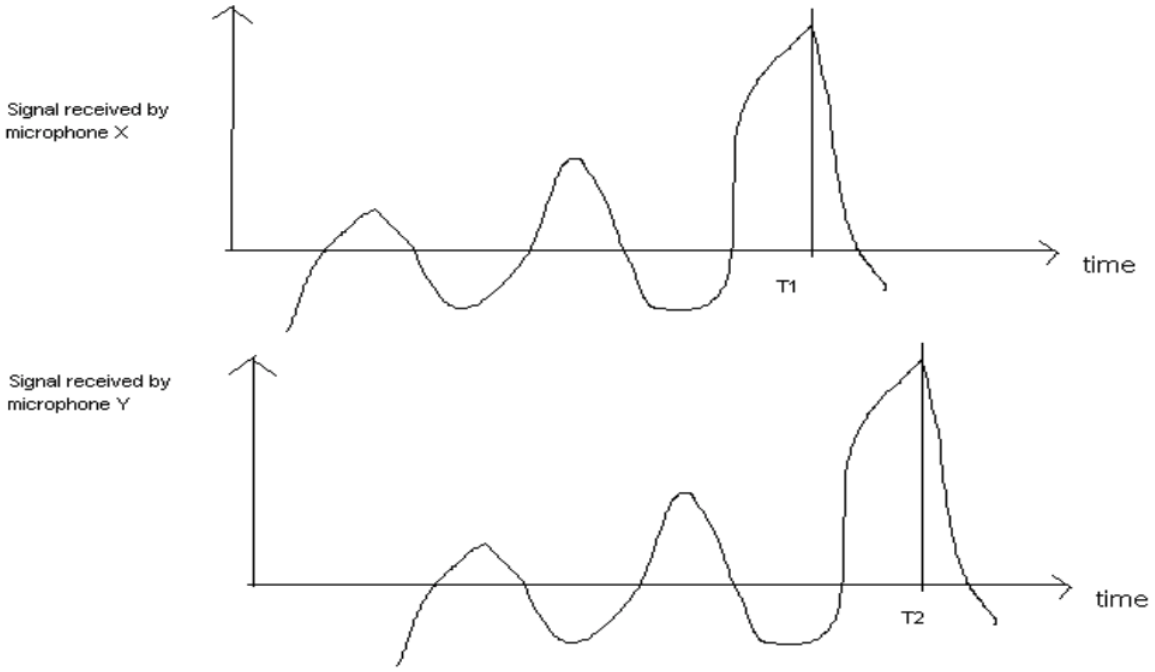

FIGURE 10 – Maximum Peak Difference

Another approach is **The Cross-Correlation** : consider x(i) and y(i) the digital signals received by Microphone X and Microphone Y respectively. The correlation at delay d is :

$$r(d) = \frac{\sum_i [(x(i)-mx)*(y(i-d)-my)]}{\sqrt{\sum_i (x(i)-mx)^2}\sqrt{\sum_i (y(i-d)-my)^2}}$$

As shown in Figure 11, r(d) will be the greatest when x(i) overlaps with y(i-d). In this way, we can find the delay d by constructing the graph r(d) against d.

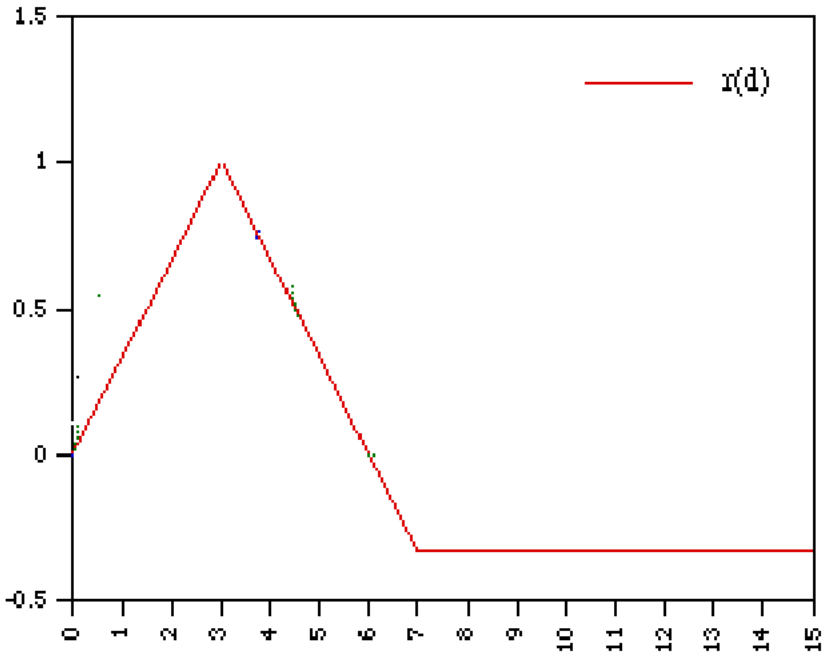

FIGURE 11 – A graph of r(d) against d

2.5.3 Attack Processes Overview

This type of attack will not require the attacker to have a relatively good understanding in the feature extraction and classification, but simply measuring the time delay between keystrokes. Further more, since no computation in frequency domain is required in the new attack, even the keyboard is designed to produce the same sound, this measurement is still effective in getting the information.

3 steps are generated for this purpose :

Step	Description
1. Sound Collection :	- Convert analogue sound waves into digital signals - Parameters set in order to keep its original form - Pre-processing making it more distinguishable
2. Signal Analysis :	- calculate the arriving time differences between two microphones - Analysis Methodologies : - Maximum Peak Position - Correlation
3. Statistical Classification :	- Neural Networks - HMM (Hidden Markov Models)

3 Experiment Design

3.1 Overview of Technologies

As we summarized in Table 1, each approach has its own advantage and defaults. One of the objective of this project is to recreate one of the attack acoustic described there.

Since our time is limited, the goal of this experiment is to generate a key press by analyzing the acoustic signal as it was described by Asonov and Agrawal in the 2004.

3.2 Environment Setup

In this experiment, we will use the HHKB keyboard as shown in Figure 12

FIGURE 12 – Keyboard HHKB Pro 2

TABLE 1 – Compare Horizontal Methodologies

Procedure	Asonov and Agrawal	Zhuang et al	Berger et al	Halevi and Saxena	Fiona
Hypothesis	different sound in different position	The input is not random	Interval decide position	Type style influence the result	Distance can be measured by delay
Equipments	1 Microphone	1 Microphone	1 Microphone	1 Microphone	2 Microphone
Collection sound pre-processing	No	Yes	No	No	Yes
supervised unsupervised	supervised	unsupervised	unsupervised	—-	supervised
Keystroke recognizing	Press Peak	start position	Press and Release	—All—-	Maximum Peak
Feature Extraction	FFT	FFT and MFCC Cepstrum features	$Sim(K_1, K_2)$	FFT and X-Corr	Delay or d from X-Corr
Classification method	Neural network classification	Neural Network Linear Classification Gaussian Mixture	Prioritization Algorithm specific	Not Specified	Neural Network HMM
Linguistic Approach	No	HMM and Grammar Model	Dictionary or Not	No	No
Evaluation	No	HMM and Grammar Model	Precision and Recall	——	No
Feeding Back Training	No	Yes	No	——	No

For the system environment we will use :

Env	Details
System :	MacOS Sierra Version 10.12.3 Model mi-2014 Retina Processor : 2.6GHz Intel Core i5 Memory : 8G
Version Mathlab :	Matlab R2016a 9.0.0.341360
Recording Software :	Sound Studio Version 4.8.3

3.3 Experiment Process

3.3.1 Record the Signals

Make sure that the computer has a microphone which functions well. When we click on the record button of Sound Studio, it will create a recorder object that records on two channel of data 44.1 KHz with 16-bit samples.

Once we stopped recording, we will capture the recording as a *.wav file, and save it locally.

In the Matlab, we can extract the raw data with the command :

[x,fs]=audioread('q1.wav') ;

Then get one channel from it by extracting one colon from the list :

s1 = x(:,1) ;

We can plot out the figure by these command :

figure(1) ;

plot(s1) ;

We will have Figure 13 with with we can observe easily the PRESS and RELEASE actions.

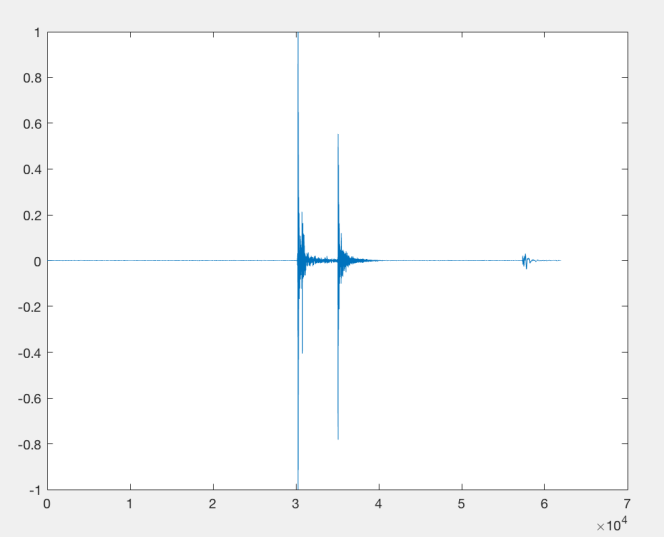

FIGURE 13 – The graph of a single keystroke

We should zoom in to get the peak of a press action then extract it from the array by using the command :

s1 = s1 (begin : end)

3.3.2 Process the Signals

Now for each array, we want to do an FFT to find the frequency spectrum for the key press :

N=512 ;

Xa = fft(a,N) ;

Xb = fft(b,N) ;

3.3.3 Classify the Signals

Using these FFTs, we can create a neural network that can be used to guess key presses. MATLAB has a neural network toolbox that can be used for pattern recognition and classification.

X = [Xa Xb] ;

T = [0 1] ;

net = newpr(X, T, 20) ;

The X array is the set of input vectors to the neural net, T is the set of target outputs. X is a 512x2 array with 2 input vectors each with 512 data points. The target array has two values that correspond to the 2 input vectors. A 0 indicates the corresponding input vector is key ‘a’ and a 1 indicates the input vector is for key ‘b’. If you collect more input data – i.e. more recordings of the two keys – we can add them as input vectors to X. The corresponding values will need to be added to the target vector as well. The more inputs we have, the more accurate the neural network.

the next step is to train the network.

net = train(net, X, T) ;

This, will bring up a new window that shows the neural network training progress as shown in Figure 14

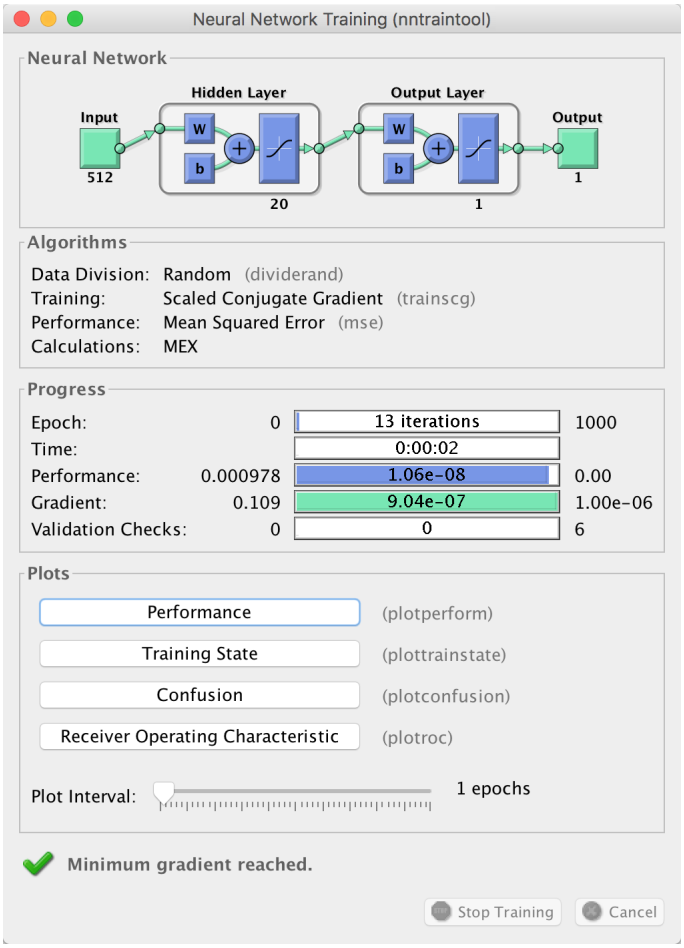

FIGURE 14 – Neutral Classification Window

Now that the network has been trained, we can use the neural network to evaluate a new input FFT with the function sim (net, x) ;

A value close to 1 indicates that the key is more probably be a ’b’.

As this goes, the rest is to enrich the dataset of this model.

4 Conclusion

We have performed an state of art computing techniques concerning the acoustic side channel attack and we have performed it on the signal press based on the work of Asonov and Agrawal in the year 2004.

But as we limited by the time, I didn't have a static study of the accuracy for the method Asonov and Agrawal, and for the same reason, the time will not allow me to go for all the technologies listed above.

Références

[1] D. Asonov and R. Agrawal, "Keyboard acoustic emanations," in *2004 IEEE Symposium on Security and Privacy (S&P 2004), 9-12 May 2004, Berkeley, CA, USA*. IEEE Computer Society, 2004, pp. 3–11. [Online]. Available : http://dx.doi.org/10.1109/SECPRI.2004.1301311

[2] L. Zhuang, F. Zhou, and J. D. Tygar, "Keyboard acoustic emanations revisited," in *Proceedings of the 12th ACM Conference on Computer and Communications Security, CCS 2005, Alexandria, VA, USA, November 7-11, 2005*, V. Atluri, C. A. Meadows, and A. Juels, Eds. ACM, 2005, pp. 373–382. [Online]. Available : http://doi.acm.org/10.1145/1102120.1102169

[3] Y. Berger, A. Wool, and A. Yeredor, "Dictionary attacks using keyboard acoustic emanations," in *Proceedings of the 13th ACM Conference on Computer and Communications Security, CCS 2006, Alexandria, VA, USA, Ioctober 30 - November 3, 2006*, A. Juels, R. N. Wright, and S. D. C. di Vimercati, Eds. ACM, 2006, pp. 245–254. [Online]. Available : http://doi.acm.org/10.1145/1180405.1180436

[4] T. Halevi and N. Saxena, "A closer look at keyboard acoustic emanations : random passwords, typing styles and decoding techniques," in *7th ACM Symposium on Information, Compuer and Communications Security, ASIACCS '12, Seoul, Korea, May 2-4, 2012*, H. Y. Youm and Y. Won, Eds. ACM, 2012, pp. 89–90. [Online]. Available : http://doi.acm.org/10.1145/2414456.2414509

[5] A. Fiona, "Keyboard Acoustic Triangulation Attack," pp. 1–40, 2006. [Online]. Available : http://personal.ie.cuhk.edu.hk/{~}kwwei/FYP/keyboard{_}acoustic{_}attack/Fiona--ERG4920CM.pdf

[6] S. Gould, "A Novel Approach to User Authentication Through Machine Learning of Keyboard Acoustic Emanations," *CS229 Machine Learning Final Project-Gould*, pp. 1–5, 2005. [Online]. Available : http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.142.7896{&}rep=rep1{&}type=pdf