Newcastle University

COMPUTING SCIENCE

An Acoustic Side Channel Attack on Enigma

Ehsan Toreini, Brian Randell and Feng Hao

TECHNICAL REPORT SERIES

No. CS-TR-1455 March 2015

TECHNICAL REPORT SERIES

No. CS-TR-1455 March, 2015

An Acoustic Side Channel Attack on Enigma

E. Toreini, B. Randell, F. Hao

Abstract

Breaking the encrypted message traffic from the German Enigma cipher machine was one of the key allied achievements of World War II, performed at Bletchley Park by a team led by Alan Turing. The work described in this paper was motivated by the historic significance of the Enigma, and the fact that we had the possibility of gaining access to one. This led to the realisation that it would be intellectually interesting to investigate the possible effectiveness of a "side-channel" attack on the machine that exploited the noise made by the act of typing the source plaintext. Much has been written on the cryptologic aspects and the historic impact of the Enigma, but we are unaware of any previous account of an investigation into its susceptibility to an acoustic side-channel attack. The Enigma keyboard differs greatly from a modern keyboard, and is particularly noisy, due to movements of one or more of the internal rotors and associated machinery in response to each key press. We applied state-of-the-art signal processing and machine learning techniques to investigate the possibility of identifying the individual Enigma keys from the noises they made; the outcome was a demonstration that such identification could be reliably achieved with a success rate of 84% (as opposed to 3.8% by random guess), using a simple microphone and a personal computer.

© 2015 Newcastle University.
Printed and published by Newcastle University,
Computing Science, Claremont Tower, Claremont Road,
Newcastle upon Tyne, NE1 7RU, England.

Bibliographical details

TOREINI, E., RANDELL, B., HAO, F.
An Acoustic Side Channel Attack on Enigma
[By] E. Toreini, B. Randell, F. Hao
Newcastle upon Tyne: Newcastle University: Computing Science, 2015.

(Newcastle University, Computing Science, Technical Report Series, No. CS-TR-1455)

Added entries

NEWCASTLE UNIVERSITY
Computing Science. Technical Report Series. CS-TR-1455

Abstract

Breaking the encrypted message traffic from the German Enigma cipher machine was one of the key allied achievements of World War II, performed at Bletchley Park by a team led by Alan Turing. The work described in this paper was motivated by the historic significance of the Enigma, and the fact that we had the possibility of gaining access to one. This led to the realisation that it would be intellectually interesting to investigate the possible effectiveness of a "side-channel" attack on the machine that exploited the noise made by the act of typing the source plaintext. Much has been written on the cryptologic aspects and the historic impact of the Enigma, but we are unaware of any previous account of an investigation into its susceptibility to an acoustic side-channel attack. The Enigma keyboard differs greatly from a modern keyboard, and is particularly noisy, due to movements of one or more of the internal rotors and associated machinery in response to each key press. We applied state-of-the-art signal processing and machine learning techniques to investigate the possibility of identifying the individual Enigma keys from the noises they made; the outcome was a demonstration that such identification could be reliably achieved with a success rate of 84% (as opposed to 3.8% by random guess), using a simple microphone and a personal computer.

About the authors

Ehsan Toreini is a PhD student in Newcastle University. In addition, Ehsan works as a project technician in the ERC project, which is concerned with developing self-enforcing e-voting protocols for future elections. His research interests are: Applied Cryptography, Side Channel Attacks and Biometric Authentication.

Brian Randell graduated in Mathematics from Imperial College, London in 1957 and joined the English Electric Company where he led a team that implemented a number of compilers, including the Whetstone KDF9 Algol compiler. From 1964 to 1969 he was with IBM in the United States, mainly at the IBM T.J. Watson Research Center, working on operating systems, the design of ultra-high speed computers and computing system design methodology. He then became Professor of Computing Science at the University of Newcastle upon Tyne, where in 1971 he set up the project that initiated research into the possibility of software fault tolerance, and introduced the "recovery block" concept. Subsequent major developments included the Newcastle Connection, and the prototype Distributed Secure System. He has been Principal Investigator on a succession of research projects in reliability and security funded by the Science Research Council (now Engineering and Physical Sciences Research Council), the Ministry of Defence, and the European Strategic Programme of Research in Information Technology (ESPRIT), and now the European Information Society Technologies (IST) Programme. Most recently he has had the role of Project Director of CaberNet (the IST Network of Excellence on Distributed Computing Systems Architectures), and of two IST Research Projects, MAFTIA (Malicious- and Accidental-Fault Tolerance for Internet Applications) and DSoS (Dependable Systems of Systems). He has published nearly two hundred technical papers and reports, and is co-author or editor of seven books. He is now Emeritus Professor of Computing Science, and Senior Research Investigator, at the University of Newcastle upon Tyne.

Feng Hao received a BEng (2001) and a MEng (2003) in electrical and electronic engineering from Nanyang Technological University, Singapore, and a Ph.D (2007) in computer science from the University of Cambridge, England. His research interests include biometrics, cryptography, fuzzy search algorithms, information coding, and error correction codes.

Suggested keywords

ACOUSTIC SIDE CHANNEL ATTACKS
ENIGMA
KEYBOARDS ACOUSTIC EMANATIONS
MACHINE LEARNING
SPEECH RECOGNITION

An Acoustic Side Channel Attack on Enigma

Ehsan Toreini*[1], Brian Randell†[1], and Feng Hao‡[1]

[1]Centre for Software Reliability , School of Computing Science,
Newcastle University, United Kingdom

March 10, 2015

Abstract

Breaking the encrypted message traffic from the German Enigma cipher machine was one of the key allied achievements of World War II, performed at Bletchley Park by a team led by Alan Turing. The work described in this paper was motivated by the historic significance of the Enigma, and the fact that we had the possibility of gaining access to one. This led to the realisation that it would be intellectually interesting to investigate the possible effectiveness of a "side-channel" attack on the machine that exploited the noise made by the act of typing the source plaintext. Much has been written on the cryptologic aspects and the historic impact of the Enigma, but we are unaware of any previous account of an investigation into its susceptibility to an acoustic side-channel attack. The Enigma keyboard differs greatly from a modern keyboard, and is particularly noisy, due to movements of one or more of the internal rotors and associated machinery in response to each key press. We applied state-of-the-art signal processing and machine learning techniques to investigate the possibility of identifying the individual Enigma keys from the noises they made; the outcome was a demonstration that such identification could be reliably achieved with a success rate of 84% (as opposed to 3.8% by random guess), using a simple microphone and a personal computer.

keywords Acoustic Side Channel Attacks, Enigma, Keyboards Acoustic Emanations, Machine Learning, Speech Recognition

*ehsan.toreini@ncl.ac.uk
†brian.randell@ncl.ac.uk
‡feng.hao@ncl.ac.uk

1 Introduction

1.1 Motivation

The breaking of encrypted message traffic from the German Enigma cipher machine, first achieved by the Polish cryptographers before the war, but then on a huge scale at Bletchley Park during World War II by a team led by Alan Turing, was of great strategic significance. Motivated by the historic nature of the Enigma, and the fact that we had the possibility of gaining access to an operational original machine, we decided it would be interesting to investigate the possible effectiveness of a “side-channel” attack on the machine that exploited the noise made by the act of typing the source plaintext[1]. (The issue of the likely practicality, or rather impracticality, of successfully eavesdropping on Enigmas in use during wartime did not concern us.)

In recent years there has been much work on acoustic side-channel attacks on various types of modern keyboard [1, 23, 4]. There is now a very extensive literature on the cryptologic aspects and the historic impact of the Enigma, but we are unaware of any previous account of an investigation into its susceptibility to an acoustic side-channel attack.

The Enigma keyboard is at first sight rather similar to that of an old mechanical typewriter, though typing on its keys requires much more force, due to the amount of machinery involved. It differs greatly from a modern keyboard, and is much more noisy, due to movements of one, or on occasion two or more, of the internal rotors and associated machinery in response to each key press. Nevertheless we felt it appropriate to try applying state-of-the-art acoustic side channel attacks, as used against modern keyboards, to investigate the possibility of identifying the individual Enigma keys from the noises they made.

The main goal of our project was, therefore, to build a systematic means of recognizing each keystroke of the Enigma machine, a task that involved using machine learning algorithms. Our aims were:

- To extract a characteristic feature from the sounds of keystrokes, i.e., one which differed across the various keys but was the same in all keystrokes performed using a given key.

- To find the most efficient and effective recognition method. By such means, we believed it would be possible to obtain the clear text of Enigma messages without having to perform any cryptanalysis, indeed

[1] The Enigma machine used in our research was borrowed for the experiments. All sound recordings made in the experiments are publicly available at the address given at the end of the paper. Our making these digital recordings and performing this acoustic analysis is our modest contribution to reminding others of the history of, and also to paying tribute to the code-breaking heroes at, Bletchley Park during World War II.

without needing to find out anything about the current set-up of the machine. (Discovering or determining the setting was a vital step in Enigma decryption methods.)

The possibility of learning whether and how well current acoustic side-channel attacks would work on a historic cipher machine was of course intriguing, even if of little practical utility, and allowed us to take advantage of the fact that we were in the lucky position of having access to a working original three-rotor WWII Enigma Machine. The Enigma was we hoped a good choice to test our proposed methods. Firstly, unlike ordinary keyboards which come from many different manufacturers, in various types and shapes, the form and physical specifications of the Enigma machine's keyboard and rotor machinery were, we understood, essentially the same in all the main models, so the fact that we had only a single machine for our experiments was acceptable. Secondly, the keyboard of the Enigma requires considerable pressure to operate, so we felt there might be a smaller impact on our results of individual typists' typing characteristics. (However, we involved different types of people in our data collection scenario in order to try to validate this belief.)

1.2 Organization

In this paper we document an investigation into the possibility of performing successful acoustic side channel attacks on an Enigma. This investigation aimed to apply the techniques previously applied to modern keyboard sounds to Enigma sound samples. However, we found it necessary to develop some new procedures to pre-process the sound samples and perform feature extraction.

The organization of this paper is as follows: In Section 2, we provide some background information about Enigma and acoustic side channel attacks. Section 3 discusses Enigma keystroke sound samples, and the differences between Enigma keystroke sounds and those of modern keyboards. Section 4 explains the data collection techniques we used. In Section 5 we cover the analysis procedures we developed for preprocessing and recognizing the samples. Section 6 summarises our experimental results. Finally, conclusions are provided in Section 7.

2 Background

2.1 The Enigma Machine

The Enigma is a cipher machine that was used during World War II by numerous German military and government organizations. The successful breaking of many coded messages produced by this machine was one of the key Allied successes of World War II. Many military historians have in recent years claimed that the breaking of Enigma shortened the war by at least two years [13]. This code breaking was done by a team of cryptographers at Bletchley Park, one of whose leaders was Alan Turing. However, the code-breaking techniques that were developed at Bletchley Park are beyond the scope of this paper - their details can be found in [14] and [17].

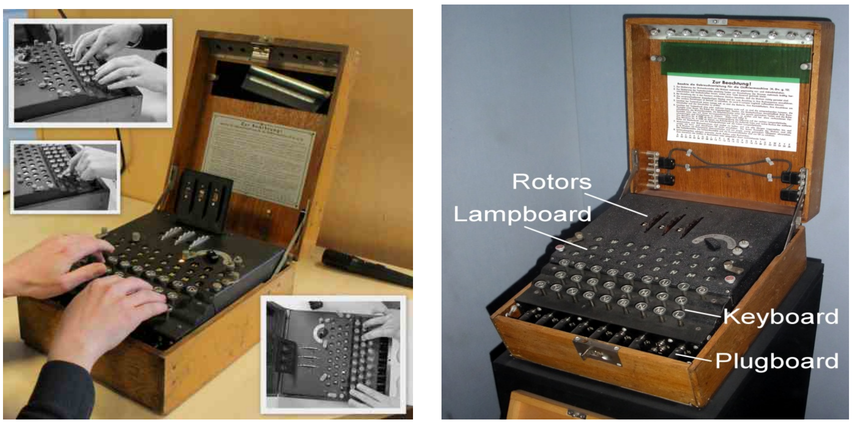
A top view of an Enigma machine is shown in Figure 1 (a) and its inner mechanism in Figure 1 (b). Each of the rotors in an Enigma embodies a mono alphabetic substitution cipher. (The machine shown is a three-rotor Enigma, the most common type.) The sender and receiver of messages need have exactly the same settings in their respective Enigma machines in order that they can encode and decode messages successfully. An Enigma's setting comprises (i) the choice of rotors (generally three out of a set of five), (ii) the relative position in the Enigma of the chosen three rotors, (iii) the position of the notch on each rotor, adjustable when the rotor is taken out of the Enigma, which controls when it will cause the rotor to its left to be forced to move (denoted by a number in the range 1 to 26), (iv) the individual starting angular position of each rotor (again denoted by a number in the range 1 to 26, visible through a window in the rotor cover), and finally (v) the way in which the front plug-board panel is plugged.

The rotors are shown to the top left of the keypad. For each keystroke, the rightmost rotor moves by one angular position. Every 26 moves of the first rotor, the second rotor also moves; and every 26 moves of the second rotor the third rotor also moves.

The operators were provided with an instruction book giving the settings to be used each day - some of the settings were changed more frequently than others. Before each typing session the operator had to set up the Enigma and its rotors as laid down in the instruction book. Then, as the plaintext message was typed, light bulbs were illuminated in the lamp board showing the encrypted character that corresponds to the typed character. These encrypted characters were then transmitted by radio, using morse code, to the distant operator to be entered into his Enigma so as to retrieve the plaintext.

2.2 Acoustic Side Channel Attacks

Acoustic Side Channel attacks on keyboard emanations were first proposed by Asonov et al. in [1]. In their research, normalized Fast Fourier Transform (FFT) features of the press segment ("down stroke") of the keystroke sound were identified and classified using a simple back-propagation Neural Network. They obtained promising results which showed that Acoustic Side Channel attacks could be a serious threat. Providing one has physical access, eavesdropping can be performed easily by using a simple microphone and can reveal a lot of information about what is being typed. Needless



(a) Enigma typing from different angles (b) Major components of an Enigma

Figure 1: Enigma Structure

to say, they also showed that their results depended on the typist and the physical specifications of the keyboard.

Zhang et al. in [23] used the Mel-Frequency Cepstrum Coefficients (MFCC) [8] of the press phase using a Linear Classifier which improved Asonov's results. We will explain more about the MFCC feature in Section5. They also adapted a Hidden Markov Model (HMM) [3] to correct the typo mistakes and misclassified characters to boost the recognition rate.

Berger et al. in [4] considered the similarity metrics of the acoustic signals (they used Cross-Correlation of the signals) and recognized words based on a pre-defined dictionary. They considered the correlation between keystrokes to determine the key presses.

There have also been examples of acoustic side channel attacks on other computer peripherals. For example, Backes et al. in [2] tried to recognize printed words based on the sounds ordinary printers produce during printing.

Tromer et al. in [21] managed to perform various timing attacks on a CPU executing cryptographic operations. They did this by recording the emanated human-audible sound of cooling fans and other ultrasound vibrations of motherboard elements. In 2013, Genkin et al. even pushed the boundaries further in [12]. They successfully extracted the RSA key from the CPU emanated sonic vibrations. They recorded the sound by using common mobile phone put aside the laptop running a RSA implementation.

In summary, as a result of recent improvements in signal processing and machine learning techniques, acoustic emanations are becoming a dangerous threat to digital systems with some mechanically moving items. Considering the new type of attacks and new technologies integrated into mobile phones, we have to expect more of such serious attacks in the near future.

3 First Observations

Evidently, modern keyboards differ greatly from that of the Enigma, and not just in appearance. The sounds produced by pressing modern keys and Enigma keys are noticeably different, and indeed the Enigma keys are very noisy. We needed to investigate two issues in order to establish the feasibility of the project:

- How do the sounds of Enigma keystrokes differ from those of ordinary modern keyboards?
- Do the sounds made by the twenty-six Enigma keys differ from each other?

In the initial stage of the project we used a small set of recordings of an Enigma that had been provided to us in order to address these questions.

3.1 Comparison with modern keyboards

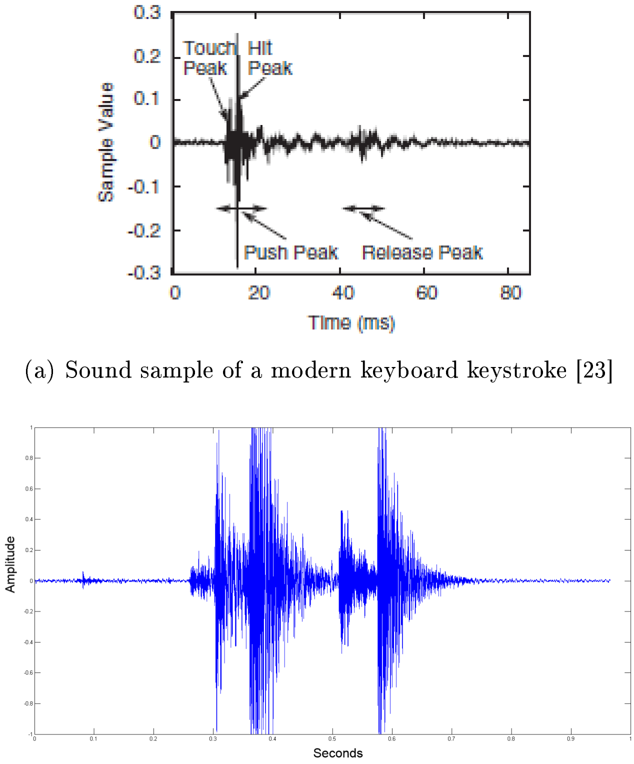
In Figure 2 (a), the amplitude of an ordinary keystroke is shown (based on [23]). The amplitude of an Enigma keystroke is shown in Figure 2 (b). A significant difference between the amplitudes is evident. The amplitude of an Enigma keystroke is greater than that from ordinary keyboards. This difference between an ordinary keystroke and an Enigma keystroke is caused by the different physical mechanisms of the devices. With Enigma, unlike ordinary keyboards, a relatively-heavy metal rotor turns through one angular position at each keystroke. Indeed on occasion two or even three rotors move in response to a single keystroke. Moreover, typing on the Enigma keyboard involves other mechanical movements, which leads to additional noises.

3.2 Comparing the different keystrokes in an Enigma

We tried to find whether there were visible dissimilarities between different keystrokes and similarities among multiple recordings of the same key. Figure 3 shows the amplitude of the sound recordings of six key strokes (“QWERTY”). They are visually different, showing distinct characteristics in the waveform shapes.

Following this, we checked whether there was any visual similarity between different strokes of the same key. In Figure 4, the amplitude of 4 strokes key “E” is illustrated. The black arrows show the areas that look alike.

These visually-identified differences and similarities were the starting point of our project. The recognition methods we used and implemented are described in the following sections.



(a) Sound sample of a modern keyboard keystroke [23]

(b) Sound sample of an Enigma keystroke

Figure 2: Sound Samples of Enigma in comparison to Modern Keyboard

4 Data Collection

We needed to record a number of real Enigma keystrokes from a reasonable variety of users. We were allowed to use a genuine Enigma from a private collection for a day. So we had to come up with a procedure to collect data in a way that we believed would fulfil the goals of the project.

We first defined what we expected from the recordings that we planned to make:

- First, the recordings had to be from different typists, differing with regard to age, gender and skill.

- Second, the recordings had to contain the effects of some double rotor movements. (We decided to ignore the much less frequent triple rotor movements.)

- Third, the recording must cover different speeds of typing.

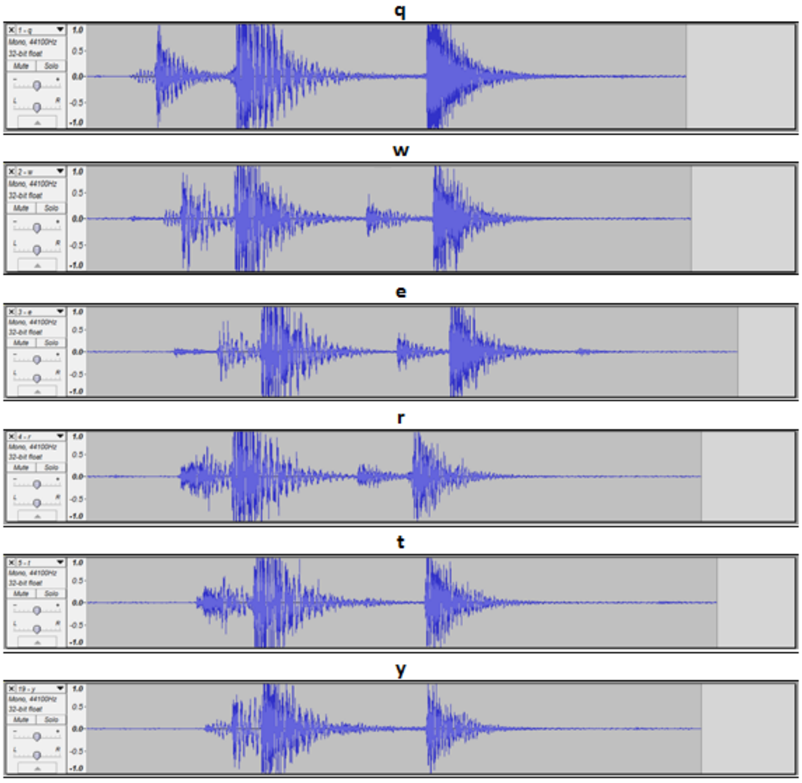


Figure 3: Sound Samples of different Enigma keys

- Fourth, each recording had to involve a single participant, who typed exactly the same sequence of letters on the Enigma machine.
- Fifth, all recordings should involve the same number of double rotor movements.
- Sixth, we should have enough recording sessions to have an adequate number of samples for each key.

Our plan for meeting these requirements was the following. Each participant was asked to proceed as follows. The three rotors were to be set to the same initial positions, denoted as $(1, 1, 1)$, and the participant had to type a sequence following the order in which they appeared on the keyboard - more specifically, starting with Q in the left hand top corner, through to L in the bottom right hand corner, and then repeating the letter L once more, hence 27 key strokes in total. Other than the rotor settings, which had to be re-initialised to $(1, 1, 1)$ after each set of 27 keystrokes, all other aspects of the Enigma's setting were left unchanged for the duration of the experiments.

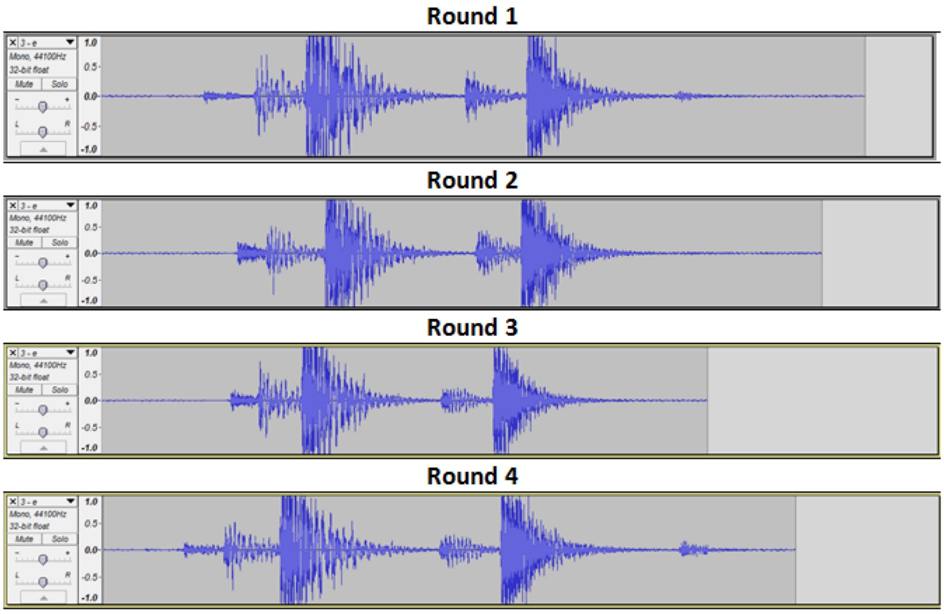


Figure 4: Sound samples from pressing the same key 4 times (key “E”)

The procedure, as it was conveyed to each participant, is given in Figure 7. During recording, a supervisor was responsible for controlling the recording, setting the rotor starting positions, and observing that the participant followed the procedure correctly, finally checking that the Enigma rotor positions had reaching the defined final setting.

The recordings were made in a conference room. We used an ordinary hand-held microphone, and held the microphone roughly 20 centimetres from the Enigma while recording. We used the open source Audio Editor and Recorder Audacity [7] to control the recording. All recordings were set to the sample rate of 44100, 32 bit per sample in mono mode. Given the conditions the final recordings contain occasional background noises as well as the keystroke noises.

We recorded 32 participants in total. Their ages ranged from the 20s to the 70s. They were of different genders, body strengths, and typing skill levels. Each participant typed the 27 letters five times in a row. We found that, since the Enigma keyboard is hard to press, the participants were usually quite tired after the third round or so, and from then on typed more slowly, thus helping to fulfil our third requirement.

We asked the participants to type the final key “L” twice. This ensured that there was one double rotor movement. In fact, the way in which the Enigma was set up, before the first pressing of “L”, when the Enigma rotor sequence had reached $(1, 1, 26)$, only the rightmost rotor had moved. Then the next keystroke, i.e., first depression of the letter L, caused both the

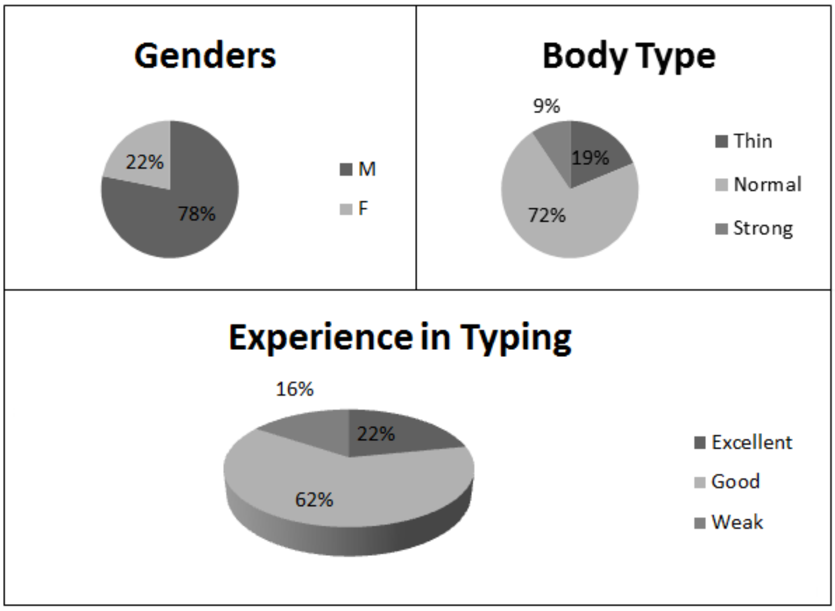


Figure 5: The Participants in the Data Collection Session

middle and the rightmost rotor to move, so leading to position $(1, 2, 1)$. On the second pressing of "L" only the rightmost rotor moved. Thus we captured the basic sounds of all the different keys, and one instance of a key-press that caused a double movement, from each typing sequence. This enabled us to satisfy, at least minimally, the fifth expectation, and to assess the recognition of double rotor movements as well as all the different keys when they caused just a single movement.

Each participant filled in a debriefing form at the end of their recording session. They were asked two questions. The questions were as follows:

1. How different was typing with Enigma comparing to the modern Keyboards? (Answers: Totally, Not Much, Same)

2. Do you think that it is possible to identify the keys stroke just from the sounds produced by typing? (Answers: Yes, No, Maybe)

Most of the participants found the Enigma typing experience totally different compared to modern keyboards. Nearly 40% of the participants rejected the possibility of performing Acoustic Side Channel Attack on the Enigma.

The distribution of participants with regard to the ages, genders and experience in typing are shown in figure 5. Participants' answers to the debriefing questions are illustrated in figure 6.

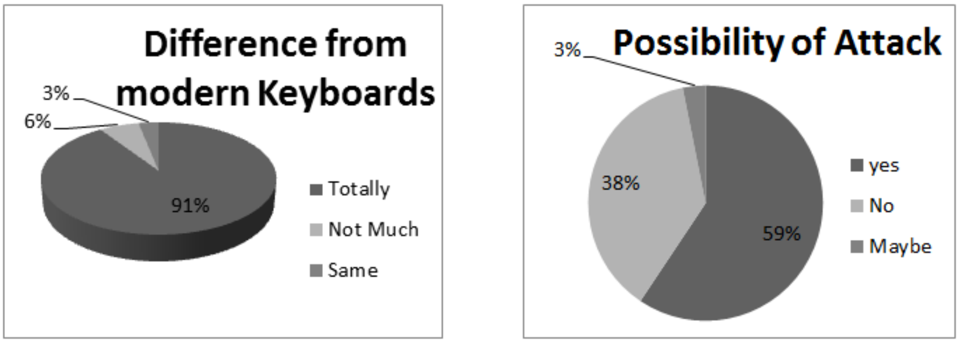


Figure 6: The Answers of the participants to debriefing questions

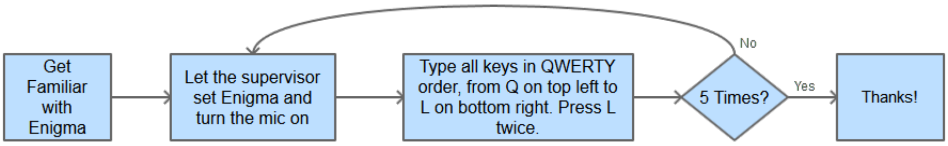

Get Familiar with Enigma
Let the supervisor set Enigma and turn the mic on
Type all keys in QWERTY order, from Q on top left to L on bottom right. Press L twice.
5 Times?
No
Yes
Thanks!

Figure 7: Data Collection Procedure

5 Data Analysis

We carried out experiments in two phases: training and recognition. An overview of the procedure is shown in Figure 8. The data analysis consists of four steps. We describe each step in the following sections.

5.1 Normalization

In this step, we imported the recorded samples for processing. After importing, we normalized the samples. In early stages of our work, we processed the unmodified samples but normalization affected the experimental results. We have normalized the samples with squared 2-norm method. In this method, first the signal norm is computed by normal Euclidean normalization represented in equation 1, in which n is the signal sample size and $s(i)$ is the signal value in the jth position.

$$N = \left(\sum_{i=1}^{n} s^2(i) \right)^{\frac{1}{2}} \tag{1}$$

Then, the signal data is divided by the squared norm plus a bias constant number b which in our case was 1e-10. The final normalized signal is computed by equation 2.

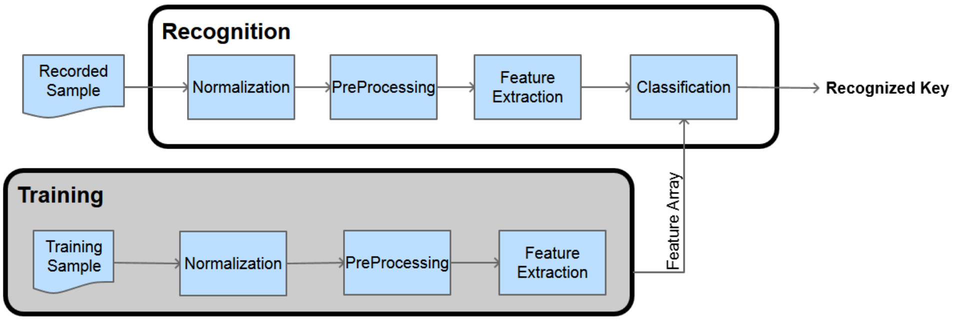


Figure 8: The Analysis Procedure for recognizing the keys

$$S(i) = \frac{s(i)}{N^2 + b} \tag{2}$$

5.2 Pre-Processing

Pre-processing is an important step in recognition. The aim of this step was to find the most important parts of each sample. Asonov et al. [1] had identified two important stages in the sound of each keystroke on a modern keyboard. For our initial investigation, we were provided with samples of Enigma sound recording. After examining these samples, we found there was a distinct difference between key pressing and releasing. Furthermore, there were some distinguishable features before the main press sound and the release sound. The physical make-up of the Enigma plus its rotor movements are the source of these extra peak sounds. Based on these observations, we decided to take these peak sounds into consideration in addition to the main press and release sounds. We term these extra peak sounds “Pre-Press” and “Pre-Release” in this paper. The four peaks of a keystroke with their corresponding names are illustrated in Figure 9.

We chose the Linear Prediction Coefficients (LPC) based algorithm to identify these peaks. The LPC is a commonly used technique in speech recognition to identify peaks in a speech signal [19].

This method was originally suggested by Gunnar Fant [10] for linearly estimating the speech production. In this approach, the speech signal is calculated as equation 3 in which the speech sample is modelled as a weighted sum of p previous samples plus some excitation parameter.

$$y[n] = x[n] + \sum_{k=1}^{p} a_k y[n-k] \tag{3}$$

In the linear model, $x[n]$ is often considered as error (or residual) and written as $e[n]$, as in equation 4. The optimum Prediction Coefficient a_i minimizes the error (residual) rate [9].

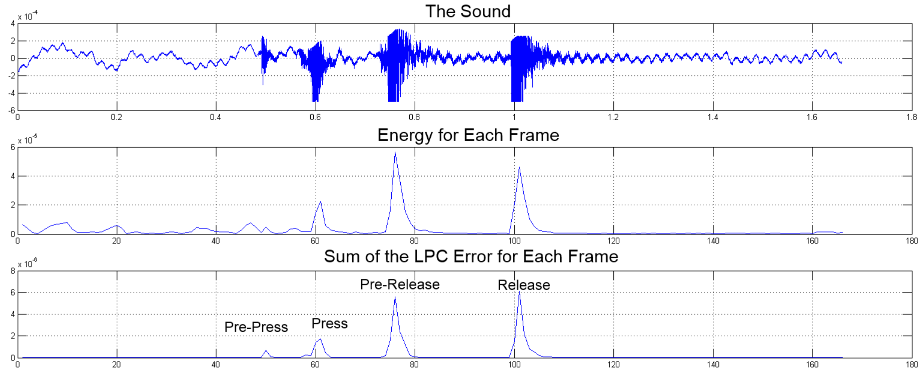


Figure 9: The Peaks recognized in each sample using the LPC error procedure

$$e[n] = y[n] - \sum_{k=1}^{p} a_k y[n-k] \tag{4}$$

where $y[n]$ is the estimated predicted signal based on $x[n]$. Therefore, the LPC indicates an estimate of the excitation signal in the current sound. Parameters a_i are the ones that minimize the energy in the signal. In other words the sum of $e[n]$ for all samples in a frame is considered as the total LP error of the frame. The frame with the maximum error rate indicates a potential peak.

Using the average energy of each frame is another common approach to finding the press and release peaks in acoustic side channel attacks on modern keyboards. In Figure 9, the analysis of an Enigma keystroke is shown in Energy-based and LPC-based approaches. Comparing the two methods, we found LPC more consistent and accurate in recognizing the main parts of the keystroke.

In the pre-processing step, we prepared the samples as follows:

1. The samples were framed using rectangular windows. The length of each window was 10 ms and there are no overlaps between windows.

2. We calculated the LPC error for each frame in our sample. We did this for all the frames in each sample.

3. We defined a threshold that we used to identify the significant frames, i.e. the frames in which the keystroke happened.

4. We found the top two significant frames from the largest LPC error. The first such frame is the Peak and the second is the Release Peak.

5. The frame with the biggest LPC error which happened before the Press Peak is the Pre-Press Peak.

6. The frame with the biggest LPC error which happened in between Press and Release Peak is the Pre-Release Peak.

5.3 Feature Extraction

Asonov et al. [1] used FFT coefficients as the feature to implement their acoustic side channel attack on modern keyboards. Zhaung et al. [23] compared different feature choices and determined that MFCC (Mel-frequency cepstral coefficients) [8, 11, 16] provided the best recognition results.

The word "cepstral" needs more clarification in this context. A cepstrum is the result of taking the inverse fourier transform (IFT) of the logarithm of the estimated spectrum of a signal. (The word cepstrum is the result of reversing the first four letters of the word "spectrum".) Investigating the cepstra is called cepstral analysis.

5.3.1 MFCC Feature

MFCC in speech recognition is the representation of the short-term power spectrum of the sound. Linear cosine transform of a log power spectrum (cepstrum) is the base of this feature. The difference between cepstrum and mel-frequency cepstrum is that MFC is equally spaced on the mel scale, which approximates the human auditory systems‘ response in a more accurate way. In other words, Mel-frequency is proportional to the logarithm of the linear frequency that has the same effect on the humans' aural perception[15]. This feature has shown more efficiency [8] in comparison with other speech recognition features.

Hence, we decided to use MFCC features for our experiments. Normally, in this feature the first 13 coefficients are considered as the important ones and the rest are not influential in the final results. So, we took account of the first 13 MFCC coefficients for each frame we selected for feature extraction in our data.

5.3.2 Feature Extraction Procedure

Asonov et al. [1] discussed that peaks are not sufficient for recognition. They used 40ms (or four 10ms frames) after the press peak to create the feature array. We could not follow this approach because the average duration of an Enigma keystroke is greater than that of ordinary keyboards. So we used four peaks in each sample, rather than two. We considered a number of different scenarios in our experiments in order to find the best combination of frames. The combination that showed the most promising results includes 2 frames before pre-Press Peak, Pre-Press Peak frame, 3 frames after pre-Press Peak, 3 frames before Press Peak, Press Peak frame, 7 Press after Press peak, 2 frames before pre-Release Peak, Pre-Release Peak frame, 3 frames after pre-Release Peak, 10 frames before Release Peak, Release Peak frame and 3 frames after Release Peak.

5.4 Recognition

The selection of a proper classifier is vital. We have implemented a variety of classifiers in our experiments. The input of each classifier is a set of features extracted in the previous step. Firstly, we randomly chose 60% of the samples for training the classifiers. Then, the remaining samples are used for recognition.

In our experiments, statistical classifiers showed better performance. In this category, each row is assigned to a class based on the membership likelihood.

Among the different statistical classification methods we assessed, the discriminative analysis classifier was the most efficient. The Discriminative analysis classifier assumes the input data has a normal multi-variant distribution and all classes share the same covariance[20].

The Naive Bayes classifier is another well-known statistical classification algorithm which is based on Bayes theory in probability. Similar to Discriminant Analysis Classification, this method assigns a membership degree to all classes for each input. Accordingly, the input belongs to the class with the maximum membership degree.

We have applied Artificial Neural Networks(ANN) classifier as well. Our ANN classifier has 2 hidden layers of 100 nodes, with Scaled conjugate gradient back propagation for training.

Furthermore, we wanted to assess the possibility of recognizing rotor movement. In our data collection, we asked the participants to type “L” twice. We consider these recordings of “L” as two classes. The first class contains the samples in which one rotor moved. The second is of “L” samples including the two rotor movements. We have trained some classifiers based on these two classes. We decided to assess SVM classifier in addition to other ones because it is widely accepted as an efficient classifier for binary classification [6]. We have applied the same strategy for training and testing the classifier.

6 Results

In this section, we discuss our experimental results. The data analysis was done using Matlab. We implemented the algorithms using an Intel Core$^{\text{TM}}$ i7-2600S CPU @ 2.80 GHz with 8.00 GB of memory. The Operating System was Windows 7 Enterprise, 64 bit and we used Matlab V. R2012a (7.14.0.739), 64 bit.

We divided our experiments into two categories: firstly, the recognition of keys based on their recording samples and secondly the recognition of

Table 1: Key Recognition results for three classifiers. Success Rate means the percentage of correct recognition

Classifier	Success Rate
Artificial Neural Networks	67.14%
Naive Bayes	75.72%
Discriminant Analysis	84.31%

rotor movement in recordings. We will discuss these two categories in the following sections.

6.1 Keyboard Recognition

Our recognition algorithm was based on the procedure described in Figure 5. The input is an array of MFCC coefficients and the output is one of the 26 letters on the Enigma keyboard. We trained the classifier with the training set of recordings. For each recording in the test set, we attempted to recognize the corresponding English alphabetic characters. Obviously, successful recognition happens when our classifier recognizes the typed letter correctly. It is worthwhile to note that the chance of getting the correct letter by random guess is $\frac{1}{26} = 3.84\%$.

We have implemented various classifiers, including Artificial Neural Network Classifier, Linear Discriminative Analysis, Naive Bayes with Gaussian distribution, Naive Bayes with Kernel Distribution, K-Nearest Neighbour, Decision Tree, Ensemble Discriminant, Ensemble KNN, Ensemble Decision Tree. Table 1 summarizes the best results from the top three classifiers.

As shown in Table 1, Discriminant Analysis gave the best recognition rate in our experiments, i.e., 84.31%. This is significantly higher than the rate of 1/26 (3.8%) that would be achieved by random guessing.

The analysis of the classifiers' confusion matrix showed in most of the wrong recognitions, the key is miss-recognized as one of the neighbouring keys. Therefore, the results could still improve by using linguistic methods such as Hidden Markov Models and considering the adjacent keys as the potential correct recognized key.

6.2 Rotor Movement Recognition

Because of the time that would have been involved, and the error-prone nature of the rotor re-positioning that would have been required, we investigated the effect of double rotor movements on just a single key, rather than for all twenty-six letters of the alphabet. Specifically, we recorded two samples of the "L" key in each data collection, the second of which involved movement of the second as well as the first rotor. In the classifiers for this section, the input was an array of MFCC coefficients and the output indi-

Table 2: Rotor Recognition results for three classifiers. Success Rate means the percentage of correct recognition

Classifier	Success Rate
Artificial Neural Networks	92.18%
SVM	89.06%
Discriminant Analysis	84.31%

cated whether there had been one or two rotor movements. We used three classifiers to recognize the output. Obviously, successful recognition means identifying the correct number of rotor movements. The results for this recognition are summarized in table 2.

In another experiment, we investigated how the recognition rate changes when we insert the records of “L” with two rotor movements among the records (all involving just single rotor movements) from the rest of the keyboard letters. We have set different labels for the one and two rotor movements of “L”. In our experiments, the recognition rate reduced by 1% in comparison with the results in section 6.1. Our observations of the confusion matrix showed in nearly half of the wrong recognitions, “L” with two rotor movements is mistaken for “L” with a single movement. Therefore, the correct letter has been recognized anyway. In addition, we investigated the recognition of the key “L” regardless of the number of rotor movements. In order to do so, we used the same label for all “L” records, without considering the number of the rotors involved. Our recognition results were the same as the previous set of experiments in section 6.1. Thus our experiments showed that rotor movement does not significantly impact on letter recognition, and that we can distinguish the number of rotor movements involved.

7 Conclusions

We have performed an acoustic side channel attack on an original World War II three-rotor Enigma machine. In our experiments we have identified four peaks in each sound sample and used an LPC based approach in this preprocessing. We extracted MFCC (Mel-frequency cepstrum coefficients), a common frequency based feature in speech recognition, from the samples in the feature extraction phase. We used 60% of the samples to train a classifier and implemented various classifiers in order to compare their performance. We found that statistical classifiers were the most efficient. Among them, the Discriminant analysis classifier gives the best recognition rate of 84.31%. Our experiments showed that the variability across the typists such as age, body type, gender and experience did not significantly affect the achievable recognition rate. Additionally, we performed an experiment to detect double rotor movements. We had two classes of recordings that used the same key. The first class is of samples in which only a single rotor moved and in the other class, two rotors moved. We applied the same preprocessing and feature extraction algorithms on the samples. In this experiment, 60% of the samples were used for training. An Artificial Neural Network classifier proved to have the highest recognition rate of 92.18%. To sum up, by utilizing state-of-the-art computing techniques, we were able to reliably decode the secret message typed by an Enigma machine by just analysing the sounds made. The improvement of the recognition rate from 1/26 (3.8%) to 84.31%, can be regarded as further evidence of the great advances in computing technology after World War II, and of how acoustic side channel attacks are now providing a powerful cryptanalysis tool.

Availability

All the Enigma recordings used are publicly available at `http://homepages.cs.ncl.ac.uk/ehsan.toreini/enigma/`. This is to facilitate others to improve the classifier algorithms against the benchmark performance reported in this paper.

References

[1] D. Asonov and R. Agrawal. Keyboard acoustic emanations. In *Security and Privacy, 2004. Proceedings of IEEE Symposium on*, pages 3–11, May 2004.

[2] M. Backes, M. Dürmuth, S. Gerling, M. Pinkal, and C. Sporleder. Acoustic side-channel attacks on printers. In *USENIX Security'10 Proceedings of the 19th USENIX conference on Security*, pages 20–20, August 2010.

[3] L. E. Baum and T. Petrie. Statistical inference for probabilistic functions of finite state markov chains. *The Annals of Mathematical Statistics*, 37(6):1554–1563, December 1966.

[4] 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*, pages 245–254, October 2006.

[5] T. Bullock. *Recognition of Complex Acoustic Signals.* Abakon-Verlagsgesellschaft [i. komm.], 1977.

[6] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, September 1995.

[7] D. Dannenberg and R. Mazzoni. Audacity project. Open Source Audio Processing Software available at `http://audacity.sourceforge.net/` – downloaded and installed in 2013.

[8] S. Davis and P. Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 28(4):357–366, August 1980.

[9] T. Dutoit and F. Marques. *Applied Signal Processing: A MATLAB^{TM}-Based Proof of Concept*. Springer, 2010.

[10] G. Fant. *Acoustic Theory of Speech Production*. Mouton De Gruyter, 1970.

[11] T. Ganchev, N. Fakotakis, and G. Kokkinakis. Comparative evaluation of various mfcc implementations on the speaker verification task. In *Proceedings of International Conferences Speech and Computer (SPECOM)'05*, pages 191–194, October 2005.

[12] D. Genkin, A. Shamir, and E. Tromer. Rsa key extraction via low-bandwidth acoustic cryptanalysis. *IACR Cryptology ePrint Archive*, 2013:857, 2013.

[13] H. Hinsley and A. Stripp. *Codebreakers: the inside story of Bletchley Park*. Oxford University Press, May 2001.

[14] A. Hodges. *Alan Turing: the Wnigma*. Vintage Books, March 1992.

[15] R. Jang. Audio signal processing and recognition. Web page available at `http://mirlab.org/jang/books/audiosignalprocessing/speechFeatureMfcc.asp` – accessed in 2013.

[16] P. Mermelstein. Distance measures for speech recognition, psychological and instrumental. *Pattern recognition and artificial intelligence*, 116:374–388, 1976.

[17] NumberPhile. 158,962,555,217,826,360,000 - numberphile. YouTube Video available at `https://www.youtube.com/watch?v=G2_Q9FoD-oQ` – accessed in 2013.

[18] J. Picone. Signal modeling techniques in speech recognition. *Proceedings of the IEEE*, 81(9):1215–1247, September 1993.

[19] L. Rabiner, M. J. Cheng, A. E. Rosenberg, and C. A. McGonegal. A comparative performance study of several pitch detection algorithms. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 24(5):399–418, October 1976.

[20] K. Teknomo. Linear discriminant analysis (lda). Web page available at `http://people.revoledu.com/kardi/tutorial/LDA/LDA.html` – accessed in 2013.

[21] E. Tromer. Acoustic cryptanalysis: on nosy people and noisy machines. *Eurocrypt2004 Rump Session*, May 2004.

[22] F. Zheng, G. Zhang, and Z. Song. Comparison of different implementations of mfcc. *Journal of Computer Science and Technology*, 16(6):582–589, 2001.

[23] L. Zhuang, F. Zhou, and D. Tygar. Keyboard acoustic emanations revisited. In *CCS'05 Proceedings of the 12th ACM conference on Computer and communications security*, pages 373–382, November 2005.