Speaker Recognition in Encrypted Voice Streams

Bachelor Thesis

Department of Computer Science
Saarland University

Submitted by Goran Doychev

Supervisors:
Dr. Boris Köpf,
Markus Dürmuth

Reviewers:
Prof. Dr. Michael Backes,
Dr. Boris Köpf

Saarbrücken, 2009

Abstract

Digital telecommunications play a vital role in our society. However, some aspects of their security have not been fully investigated yet. In the current work we show that the use of the voice activity detection technique makes encrypted telephone conversations vulnerable to attacks against the anonymity of speakers. This technique is utilized in methods for reducing the amount of transmitted voice data and is widely applied in mobile communications and in VoIP. We propose several methods for speaker recognition which are based on analysis of encrypted telephone traffic. We conducted experiments using voice recordings of 13 speakers. Our best-performing classifier achieved a recognition rate of 73%, in contrast to an expected recognition rate of 7.7% of random guessing.

Statement

I hereby confirm that this thesis is my own work and that I have documented all sources used. I agree to make my thesis accessible to the public by having it added to the library of the Computer Science Department.

Goran Doychev
December 18, 2009

Acknowledgements

First of all, I would like to thank Markus Dürmuth and Dr. Boris Köpf for their insightful guidance and supervision during my work on this thesis. I am also grateful to Prof. Dr. Michael Backes for letting me pursue my interests in the productive environment of his group. I want to thank my parents Nelly and Valentin for their financial and moral support throughout my studies and my sister Cveta for being such a good example to me. I thank my friends Violeta, Nadezhda, Kalina, Yassen, Alexi, Diana, Vitaly, Daniel, Elena, Gancho, Anna, Daniela for never giving up on me and supporting me in the past several months. Last (and least), I would like to thank my gym for the long opening hours, Trey Parker and Matt Stone for the philosophical inspiration, and all my favorite musicians for contributing with a fruitful atmosphere for my work.

Contents

1 Introduction

In the past 20 years, digital telecommunications have seen a rapid development. After the introduction of the first GSM cellular network in 1991, mobile communications have spread so much that in 2008 the estimated number of mobile cellular subscriptions worldwide was over 4 billion [44]. Around the same time, in 1989 the first computer network for commercial use was launched, resulting in what is now known as the Internet. The rapid growth of the Internet boosted the development of new technologies for communication and in the late 1990s a technology known as Voice over Internet Protocol (VoIP) emerged, which allows transmission of voice over computer networks. The wide deployment of those technologies for telecommunication carries serious concerns about two main security aspects – the secrecy of the transmitted data and the privacy of the users of those technologies. To address those concerns, developers of telecommunication technologies apply encryption to the transmitted data. If the applied encryption scheme is considered secure, it is supposed that the security of the encrypted data is granted.

However, there exist the so-called *side-channel attacks*, which exploit certain implementation properties of the systems and make it possible to bypass the used encryption. In the last several years a new class of side-channel attacks based on traffic analysis of encrypted telephone conversations was discovered. Those attacks are targeted against VoIP conversations encrypted using a length-preserving encryption scheme. They exploit a widely used bandwidth-saving sound encoding technique called *variable bit-rate* (VBR), which allows varying of the amount of bits per second (*bit-rate*) in an encoded sound file. Wright et al. [47] show that analysis of the Internet traffic of encrypted VBR-encoded VoIP conversations can reveal the language spoken in the conversations. The severity of this security threat is underlined by later work of Wright et al. [46]. They show that involved analysis of encrypted VBR-encoded VoIP traffic makes it possible to uncover the content of whole spoken phrases in a conversation.

Inspired by those findings, we developed a further attack based on traffic analysis of encrypted telephone conversations. Our attack is targeted against the anonymity in encrypted telephone conversations. We exploit a widely deployed technique called *voice activity detection* (VAD), which distinguishes between segments in a conversation containing voice and the rest of the conversation, containing silence or some background noise. VAD is used both in mobile communications and in VoIP. Note that the use of VAD instead of VBR is more general because pauses in speech are encoded by VBR using the lowest several bit-rates [47] and therefore attacks that

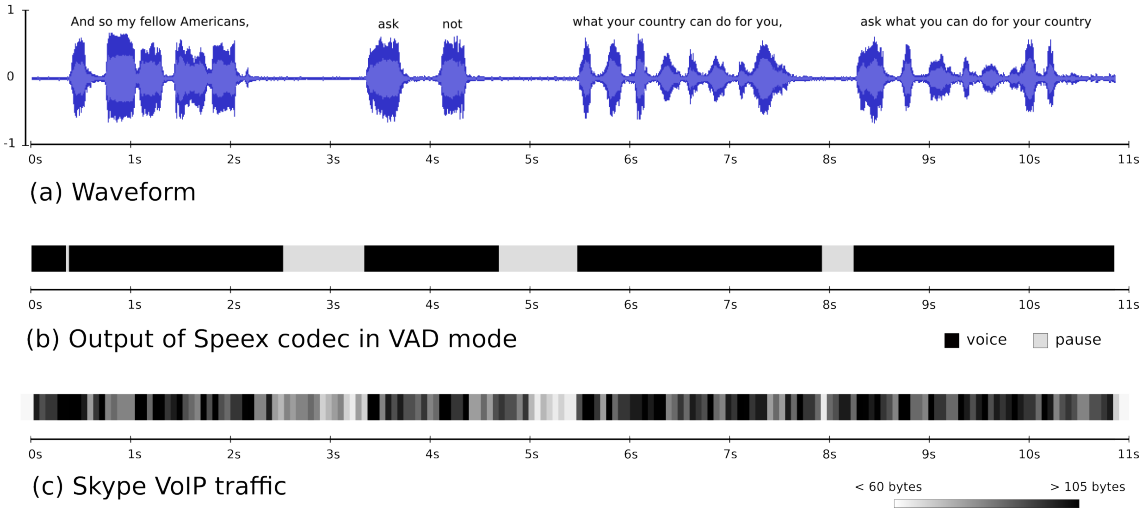


Figure 1.1: The output of the Speex codec and the Internet traffic of the Skype VoIP client, corresponding to the audio signal of a sentence taken from the inaugural address of John F. Kennedy from January 20th, 1961

exploit the use of VAD can be easily transferred to VBR-technologies. Furthermore, VAD is used not only in VoIP, but also in mobile communications, and therefore the attack discussed in the current work has a broader application area than previous attacks exploiting VBR. The operation of VAD and VBR is illustrated in Figure 1.1, where the output of the Speex sound codec [48] in VAD mode is compared to the Internet traffic generated by the Skype VoIP client [24], which uses VBR codecs.

The objective of our attack is revealing the identity of an unknown speaker. This task is called *speaker recognition*. For the proposed attack, we use durations of pause and voice segments, which, albeit not commonly, have been considered in previous research of speaker recognition (see Chapter 2).

1.1 Attack overview

In our attack against the anonymity in encrypted telephone conversations, we assume a scenario including two actors. One is conducting encrypted telephone conversations using either the GSM network, or VoIP. Another one is eavesdropping on the encrypted traffic. We call the first one *the victim*, because he is being spied on, and second one *the attacker*.

We divide the process into two phases – a training and an attack phase. In the *training phase*, the victim performs conversations with N different people and reveals their identity to the attacker. In the *attack phase*, the victim picks one of the N speakers uniformly at random and makes a telephone conversation with him. The goal of the attacker is to find out which one of the N speakers the victim spoke to.

We have the precondition that voice activity detection (VAD) is used when data is transmitted and that the attacker is able to distinguish whether the VAD-algorithm has classified segments of the conversation as *voice* or as a *pause*. Conversations are assumed to be encrypted. We don't require the encryption scheme to be length-preserving, as it is the case in the attacks proposed by Wright et al. [47, 46]. However, it has to allow distinguishing between voice and pause segments in a telephone conversation. All meta-data (IP headers, application layer headers, etc.) is assumed to be of an approximately known size.

In this setting, the attacker inspects the collected traffic in order to extract the necessary data and conduct the attack. The attack has the following steps:

1. Distinguish segments of traffic corresponding to voice from segments corresponding to pauses. For this, some knowledge about the way voice data is transported (see Chapter 3).

2. Compute the durations of voice and pause segments, obtaining a stream of alternating voice and pause segment durations. Those durations can be measured either in milliseconds or in an application-specific measure, such as number of audio packets.

3. The traffic is analyzed accordingly (see Chapter 4):
   (a) In the training phase, the attacker uses the segment durations to train the underlying models for each speaker.
   (b) In the attack phase, the attacker uses the trained models and the observed segment durations corresponding to the speech produced by the unknown speaker to compute the most likely identity of the speaker.

1.2 Outline

This work is organized in 6 chapters. Chapter 2 presents some background from linguistics that justifies the features we have used for speaker recognition. Chapter 3 gives technical details about how sound is encoded and transmitted in telecommunication technologies that use VAD. Chapter 4 describes the classifiers used in the proposed attack. The experiments we executed and their results are shown in Chapter 5. We conclude in Chapter 6.

2 Linguistic Background for Speaker Recognition

Speaker recognition is the task of discovering the identity of a speaker from a set of known speakers using passages of his or her speech. There has been significant research in obtaining good algorithms for speaker recognition [9, 35]. An extensive overview on speaker recognition is to be published in [6]. Current speaker recognition technologies are deemed accurate enough so that human speech is being used as a *biometric* for confirming or rejecting a claimed identity, or for determining who an unknown speaker is. Such technologies have been applied in systems that require authentication of users, as well as in forensic and intelligence applications [26]. Organizations which have deployed speaker recognition technologies include AT&T, British Telecom, VeriSign, Visa and the US Department of Homeland Security [27].

In our study we investigate speaker recognition in encrypted voice streams. We construct an attack against the anonymity in encrypted telephone conversations, using durations of voice and pause segments in speech. We refer to features related to durations in speech or pauses as *temporal features*. In the following, we summarize previous research on speaker recognition and other types of speaker classification, focusing on the use of temporal features for those tasks.

2.1 Features used for speaker recognition

Several different types of features are used for speaker recognition. One set of features that is traditionally related to this task reflects acoustic voice parameters. We refer to such features as *lower-level features*. Those features are gained from the frequency spectrum of speech and are estimated from a short (usually 10-50 ms) segment of the waveform [36]. Common lower-level features include mel-frequency cepstral coefficients and linear prediction-based cepstral coefficients [43].

Recently, there has been considerable research in another set of features, the so-called *higher-level features*, which are based on information gained at longer time spans [42]. Higher-level features include several types of features which are not connected to the human voice, such as features based on phonetics, lexical usage and prosody. Such features were shown to have significant discriminatory ability and although they lack the accuracy of lower-level features, they are believed to reveal aspects of a speaker's identity that cannot be captured by lower-level features [33]. Systems based on combinations of higher and lower-level features have shown better

results in distinguishing speakers than systems based only on traditional lower-level features [30, 14].

2.2 Temporal features for speaker recognition

Many of the proposed higher-level features for speaker recognition are based on temporal properties of speech, which are of interest to our work. Ferrer et al. [14] consider durations of *phones* (i.e., the smallest distinguishable sound units in speech) occurring in speech, as well as in-word phone durations. They conclude that those features are highly effective knowledge sources for automatic speaker identification and their use in combination with traditional speaker recognition features achieves a reduction in identification error of 50%. Peskin et al. [30] consider so-called *prosodic features*, which describe patterns of stress and intonation in languages. Among the 19 prosodic features they examined, six were related to word, phone and segmental durations and five were related to pause durations and pause frequency. Those prosodic features, especially in a combination, were found to add new and useful information to the speaker recognition task.

Patterns in the interaction between speakers in a conversation (also called *dialogic features*) were also considered for speaker recognition [30, 34]. Such dialogic features include duration of the turns in a conversation, as well as number of words in a turn. Although conversational patterns haven't given as good results for speaker recognition as prosodic patterns, those features were considered to be a promising area for further investigation.

2.3 Temporal features for other types of classification

Speaker recognition, the task we consider in our study, is a particular type of speaker classification. Aside from this task, there are other speaker classification tasks which make use of temporal features of the human speech. In the following, we discuss three such tasks.

2.3.1 Determining speaker's age

With the advance of human age, temporal aspects of speech are strongly affected by the age of speakers [39, 40, 25]. Features considered for determining speaker's age include pause duration and frequency, as well as syllable, consonant, vowel and subphonemic durations.

Those effects can be explained by the fact that through the process of aging, various changes occur in the human body, accounting for changes in the way speech is produced. Those changes include inadequate laryngal valving, decreased lung

capacity, stiffening of thorax, weakening of respiratory muscles, and changes in neuromuscular control. They affect human coordination of articulators and breath support and result in the need of elderly people to pause more frequently while speaking [25, 39].

2.3.2 Determining speaker's stress levels

Temporal features have been considered for determining a speaker's stress levels [18]. In stressful situations, an increase in people's respiration rate is observed. In loud, fast, slow, clear, and angry speech, there have been found changes in the mean duration of words. Additionally, the listener's ability to perceive a speaker's information context is found to be highly dependent on word and subword durations. Subword durations include duration of vowels versus consonants and consonant presence.

2.3.3 Determining deceptive speech

Temporal features have also been considered for determining deception. The use of silent pauses in speech, as well as the use of filled pauses (*ums* and *uhs*) have been found to correlate more with truthful than with deceptive speech [7]. This observation can be explained by the hypothesis that deceptive speech is more carefully planned and thus people produce less pauses when lying. This hypothesis is utilized in several interviewing and interrogating methods (cf. [5]).

2.4 Summary

The literature presented in this chapter shows that temporal features have been used in different types of speaker classification. For speaker recognition, those features have shown a worse performance than classical lower-level features. For that reason, in usual speaker recognition applications such as for authentication of users, to our knowledge, only the combined use of temporal features with lower-level features has been considered. Nevertheless, the presented studies indicate that the discriminatory ability of those features may be good enough for our application domain – to enable us conduct an attack against the anonymity in encrypted telephone conversations.

3 Digital Telephony Using Voice Activity Detection

In modern digital telephony, sound is first encoded using a speech codec, e.g. the GSM codec [17], G.728 [21], or Speex [48]. The encoded voice data is then transmitted over a network, such as a fixed-line telephone network, a mobile network, or as it is the case with Voice over IP (VoIP) – over a computer network.

Our attack against the anonymity in encrypted telephone conversations assumes the use of a technique called voice activity detection (VAD), which is widely applied in methods for reducing the amount of transmitted voice data in mobile communications and in VoIP. In mobile communications, VAD technologies are used in the GSM standard [17], as well as in the UMTS third-generation mobile telecommunication technologies [2]. In both standards for mobile communication, VAD operates similarly and in our work we discuss the GSM standard only.

In the following, we describe the setting in which the proposed attack can be conducted. We give details on speech encoding and the operation of VAD in the GSM standard and in VoIP, as well as on voice transportation. This information is important for understanding how the attack can be performed in practice.

3.1 Speech coding

Speech produced by a speaker is recorded using a microphone. The analog sound signal is digitalized, usually at a sampling rate of 8000 samples per second (8 kHz) or 16000 samples per second (16 kHz). To achieve reasonable sound quality, each sample should contain more than 8 bits; at 16 bits/sample the sound quality is considered high [11]. The number of bits per second (*bit-rate*) of a digital signal encoded at 8 kHz with 16 bits per sample would thus be 8 kHz $\cdot$ 16 bits/sample = 128 kbps. To reduce the bit-rate of digital signal, audio data is compressed using compression algorithms called *audio codecs*, which comprise of sound coders and sound decoders. A sound coder takes as input several digital sound samples and compresses them to *audio packets*, typically at a rate fixed at between 10 ms and 50 ms. In the Speex codec an audio packet is 14.5 ms long.

3.2 Voice activity detection (VAD)

In a two-way conversation, roughly 63% of the time a speaker is silent [11]. Some technologies make use of this observation and discriminate actual speech from silence to achieve certain benefits. The technique for distinguishing voice from non-voice in an audio recording is called *voice activity detection* (VAD) [32]. In the following sections we describe the uses of VAD in mobile communications and in VoIP.

3.2.1 VAD in mobile communications

In the GSM system for mobile communications, the *discontinuous transmission* (DTX) option allows halting the transmission of the signal when no voice activity is detected [45]. It is implemented in order to reduce the battery power needed for transmission and to minimize co-channel interference. On the receiving side, so-called *comfort noise* is generated during the breaks of speech, which gives the impression to users that the call is still active. The comfort noise is generated from parameters of the background noise, which are measured during the preceding voice phases and are transmitted in regular intervals to the receiving side.

3.2.2 VAD in VoIP

VoIP is transmitted over packet-switched computer networks. In such networks, efficient bandwidth utilization is a primary concern. In order to reduce network bandwidth, VAD may be applied in two ways: so that the transmitted segments of pause are encoded in a lower quality than segments of voice, and thus sound at a lower bit-rate is sent over the network during pauses; or, similarly to the DTX mode in the GSM standard, packets containing silence may not be transmitted at all.

Network overhead may be further reduced by using variable bit-rate (VBR), where speech is encoded using one of multiple bit-rates according to the nature of the encoded sound. VAD-coders can be seen as a type of VBR-coders using only two distinct bit-rates. Usually, VBR codecs use the lowest bit-rates for encoding silence [47], hence silence is similarly distinguishable in general VBR-encoded recordings as it is in a VAD-encoded recordings. For that reason the conclusions we make about VAD codecs can be easily transferred to VBR. Many popular VoIP applications support VAD or VBR codecs, including Google Talk [20], Ekiga [1], Skype [24], Twinkle [13].

3.3 Voice transportation

After voice is encoded, it is transported from the sender to the receiver using the underlying network infrastructure. In the current section we present the technologies

for voice transportation used in mobile communications and in VoIP, thus describing the setting in which the attack against the anonymity in encrypted telephone conversations takes place.

3.3.1 Voice transportation in mobile communications

In mobile communications, voice data is transmitted between a *mobile station*, usually a handheld device, and a transceiver known as a *base station*, using a frequency from a dedicated radio frequency range [45]. Between base stations, data is usually transmitted using a partner network, such as the public fixed-line telephone network. Figure 3.1 depicts the architecture of a mobile network. The GSM system usually operates on 900 MHz or 1800 MHz frequency bands (known as GSM-900 and GSM-1800 respectively). In USA, Canada and other American countries, 850 MHz or 1900 MHz frequency bands are used.

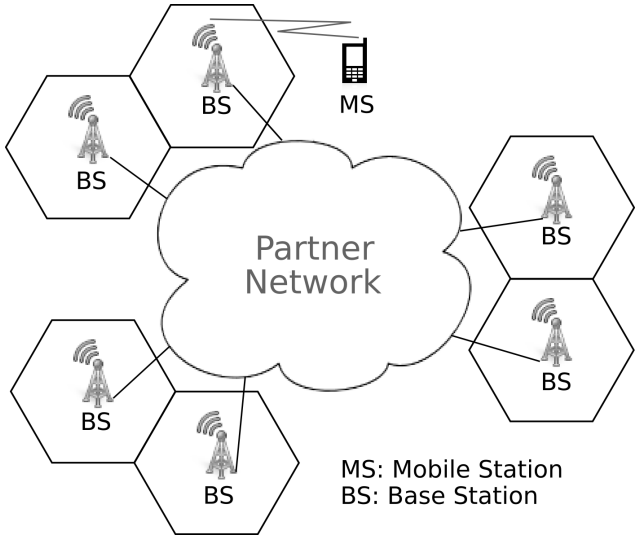


Figure 3.1: Mobile network architecture

In GSM-900, the range from 890 MHz to 915 MHz is used for *uplink traffic* (from the mobile station to the base station), and the 935 to 960 MHz range is used for *downlink traffic* (from the base station to the mobile station). In order to provide several simultaneous connections, GSM provides a combination of *frequency division multiplexing* (FDM) and *time division multiplexing* (TDM). In GSM's FDM, the range is divided into channels that have a width of 200 kHz, thus 124 FDM channels for uplink and 124 FDM channels for downlink traffic are provided. GSM's TDM uses 8 channels (time-slots) per FDM channel. Figure 3.2 shows how a physical channel is realized in GSM using a combination of TDM and FDM.

Optionally, *frequency hopping* can be applied to reduce co-channel interference. In this mode, transmitters and receivers synchronously change the frequency after

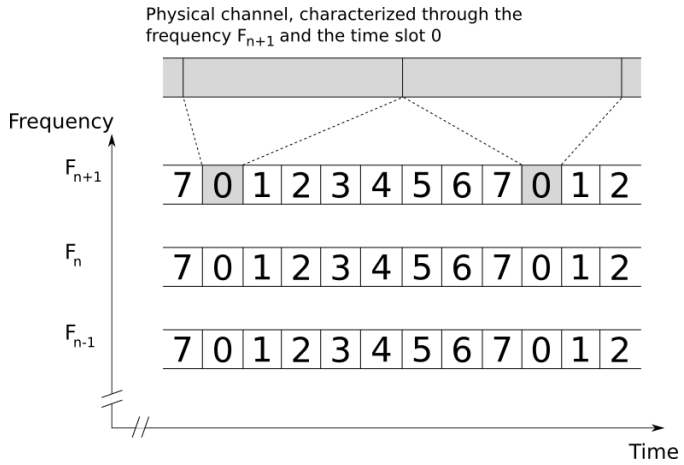


Figure 3.2: Realization of physical channels in GSM using FDM and TDM (adapted from [45]). The frequency spectrum is divided into frequency channels and each frequency channel is divided into 8 time-slots.

each transmitted frame.

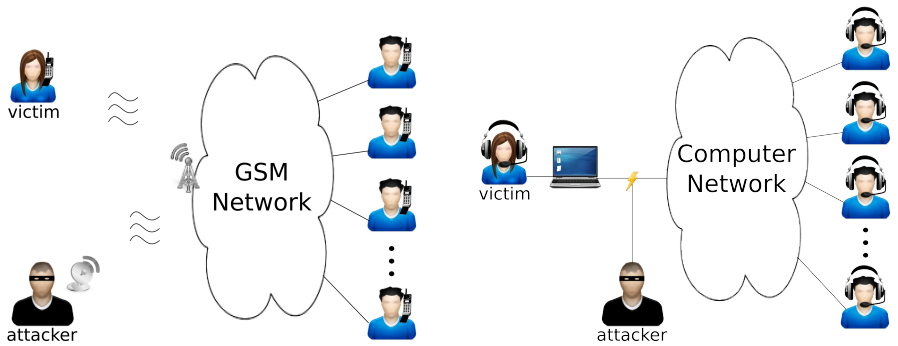
3.3.2 Voice transportation in VoIP

In VoIP, the encoded sound signal is transported between two or more participants through the Internet Protocol. Session control (connection setup and teardown) are accomplished using a control channel, usually over TCP (Transmission Control Protocol). Protocols frequently used for session control include SIP (Session Initiation Protocol) [37] and XMPP (Extensible Messaging and Presence Protocol) [38]. A second channel is dedicated to the transportation of speech data. This is accomplished through an application layer protocol such as RTP (Real-time Transport Protocol) [41], preferably through UDP (User Datagram Protocol). Some applications, most notably Skype [24], do not use standardized protocols such as SIP, XMPP and RTP but use their own proprietary protocols for both tasks.

3.4 Performing the attack

To perform an attack against the anonymity in encrypted telephone conversations, we have an attacker-victim scenario, as described in Section 1.1. In the training phase, the victim performs telephone conversations with N speakers. In the attack phase, the victim is having a conversation with one of the N speakers and the attacker wants to guess the speaker's identity.

First of all, the attacker needs to find a way to eavesdrop on the telephone traffic. Figure 3.3 depicts the attack setting in the mobile communications and in the VoIP scenario. In the case of mobile communications (Figure 3.3(a)), the attacker has to



(a) The attack setting against GSM (b) The attack setting against VoIP

Figure 3.3: The attack setting

capture the signal transported between the mobile station and the base station. To accomplish this, he has to be in the proximity of the base station and to possess the appropriate radio receivers. In the case of VoIP (Figure 3.3(b)), the attacker has to capture the Internet traffic corresponding to the particular conversation. Likely attackers in this scenario are participants in the same local area network or the victim's Internet provider.

The collected traffic can be divided into *uplink traffic*, which flows from the victim to the N speakers, and *downlink traffic*, which flows from the N speakers to the victim. As the task of the attacker is to recognize one of the N speakers, only downlink traffic is relevant to the attack.

Being able to eavesdrop on telephone traffic, the attacker has to apply some traffic analysis in order to find a way to distinguish between voice and pauses in a conversation. After that, the attacker measures the durations of voice and pause segments and collects a stream of alternating voice and pause segment durations.

4 Data Analysis Techniques

In order to perform an attack against the anonymity in encrypted telephone conversations, the attacker first needs to collect data in the form of a stream of alternating voice and pause segment durations. Using this data as input, the attacker makes a guess about the identity of an unknown speaker. In the training phase, the attacker builds probabilistic models for all speakers, based on distributions of the observed features. In the attack phase, the attacker builds a model for the unknown speaker, compares it to the trained models and picks the actor whose model is most similar. In the following we describe the statistical methods used for those tasks.

4.1 Building probabilistic speaker models

The first question we are addressing is which features of the extracted data can be used to conduct the attack and how raw data is transformed into probabilistic models for each speaker.

4.1.1 Features

The first step of any classification task is feature selection. In our scenario, the attacker has at his disposal a stream of durations of voice and pause segments. Therefore, for the analysis the attacker can use the features *voice segment durations*, *pause segment durations*, or a combination of the two features.

Only voice or pause durations

Voice segments include everything the voice activity detection (VAD) algorithm has classified as voice (see Section 3.2). Semantically this corresponds to segments in a conversation where someone is speaking. Pause segments, which include segments that were not considered by the VAD algorithm to be voice, correspond semantically to pauses, which may be silent or filled with background noise.

Combining voice and pause durations

Analyzing only voice or only pause durations would be in accord to theories claiming that there is a high inter-speaker variability in one of those features (see Section 2.2). However, such analysis would ignore half of the data input, thus the attacker would

possibly lose one source of valuable information. For that reason, we also consider the combined use of both features.

We assume that the collected stream of alternating voice and pause durations is measured in number of audio packets, starting and ending with a voice segment duration. Formally, for an odd $k \in \mathbb{N}$, the attacker collects a stream $d_1, \ldots, d_k$, where $d_i \in \mathbb{N}$ corresponds to a voice duration if i is odd and to a pause duration if i is even. To make use of both voice and pauses, we observe n-grams of segment durations. In our experiments, we used overlapping 3-grams which we refer to as *(voice,pause,voice)-triplets*. For example, if we are given a stream of segment durations $123, 32, 83, 71, 43$, we observe the 3-grams $(123, 32, 83)$ and $(83, 71, 43)$.

4.1.2 Optimization techniques

Having determined which features are to be used, we consider two optimization techniques which are aimed at improving the performance of the classifiers – *clustering audio packets* and *excluding short segments*.

Clustering audio packets

Audio data is usually encoded at a constant frequency and therefore the resulting audio packets have a constant duration. For example, Speex audio packets are 14.5 ms long. For that reason, the segment durations we analyze will inevitably be a multiple of the audio packet duration. However, this granularity may not correspond to practically relevant semantical units. For example, in speech encoded by the Speex codec in VAD mode, it is not clear whether it really makes any difference from a semantical point of view if a speaker has spoken 5075 ms or 5089.5 ms, which corresponds to 350 and 351 audio packets respectively, or whether both segments have the same semantics.

To address this issue, we explored how clustering of packet durations changes the classification results. This method has additionally the advantage that it reduces the complexity of the used models. We regard sequences of n audio packets to fall into the same cluster, e.g. for cluster size of $n = 5$, the first cluster consists of segments of length 1 to 5 packets, the second one of segments of length 6 to 10 packets, etc. In our experiments, we adjusted the cluster size in order to find a granularity size of the segment durations which gives the best classification performance.

Excluding short segments

In the English language, syllables were observed to have a mean duration of 190 ms [16], and pauses of less than 200 ms account for only 3.9% of the observed pauses [10]. Thus, *short segments* (less than 200 ms long) might carry little valuable information for our purposes. However, in the course of our experiments, we observed that in speech encoded by the Speex codec in VAD mode, there is a high number of short

voice and pause segments. The percentage of one-audio-packet-long segments (14.5 ms long) reached 65% for pauses and 40% for voice. As short segments may just be an artifact of the particular codec used, their use in our analysis may have a negative effect on the classifiers' performance. To examine the impact of short segments to our classification task, we considered excluding them from our experiments. We varied the length of the excluded segments to determine when the classifiers perform best.

4.2 Classifiers

This section introduces the classifiers we consider in our work. Those classifiers are *goodness of fit* tests, comparing how well the probability distribution over segment durations of the unknown speaker fits into distributions over durations collected in the training phase. Before describing the particular tests, we introduce some notation.

Assume we are given a set of speakers S with $|S| = N$ and a set of possible segment durations L with $|L| = n$. In the training phase, from the training data of each speaker $s \in S$, the attacker computes the *trained probability distribution* P_t^s over segment durations in L. In the attack phase, from the collected data of the unknown speaker, the attacker computes the *observed probability distribution* P_o segment durations in L. Using the goodness of fit function f, the attacker computes the goodness of fit $f(P_o, P_t^s)$ of the observed distribution P_o in all trained distributions P_t^s for speakers $s \in S$. The speaker $s^* = \arg\min_{s \in S} f(P_o, P_t^s)$ is considered by the classifier to be the most likely speaker.

4.2.1 L_1 distance

The L_1 distance, also known as L_1 norm, taxicab geometry and Manhattan distance, is a classical measure for distance between two vectors. We use it to determine the distance between the observed and the trained probability, treating probability distributions as vectors. We used the L_1 distance because of its simplicity – it compares two distributions by summing the differences between the distributions at each point.

For two distributions P_o and P_t, the L_1 distance is computed as:

$$L_1(P_o, P_t) = \sum_{i=1}^{n} |P_o(l_i) - P_t(l_i)|$$

4.2.2 χ^2 test

The χ^2 test [29] is a widely used goodness of fit test. In contrast to the L_1 distance, the χ^2 test gives more weight to points which have a lower trained probability. For

two distributions P_o and P_t, the χ^2 test is computed as:

$$\chi^2(P_o, P_t) = \sum_{i=1}^{n} \frac{(P_o(l_i) - P_t(l_i))^2}{P_t(l_i)}$$

4.2.3 Kolmogorov-Smirnov test

A further test we consider is the Kolmogorov-Smirnov test (or *K-S test*) [23]. It is known to perform good in some situations when the performance of the χ^2 test is limited, such as when the sample size is small or is scattered throughout a relatively large number of discrete categories [28]. For two distributions P_o and P_t, the K-S test is computed as:

$$\text{K-S}(P_o, P_t) = \max_{k \leq n} \{ | \sum_{i=1}^{k} (P_o(l_i) - P_t(l_i)) | \}$$

The K-S test searches for the maximum difference between two cumulative distributions. However when regarding the cumulative distribution function, segments of smaller length have a higher weight in the resulting evaluation than longer segments. As we don't have a reason to believe that short segments are more important than long ones (see the discussion in Section 4.1.2), we slightly modified the K-S test so that it searches for the longest sequence of consecutive segment durations. We call this test *K-S-modified* and it is defined as follows:

$$\text{K-S-mod}(P_o, P_t) = \max_{j \leq k \leq n} \{ | \sum_{i=j}^{k} (P_o(l_i) - P_t(l_i)) | \}$$

4.2.4 Support distance

In order to obtain a better understanding about what distinguishes the analyzed distributions, we developed a further goodness of fit test[1], which we call *support distance*. Let the support of a probability distribution P be defined as:

$$\text{supp}(P) = \{x | P(x) > 0\}$$

For two probability distributions P_o and P_t, we compute the support distance as the percentage of points where only one of the probabilities is positive, or formally:

$$\text{supp\_dist}(P_o, P_t) = \frac{1}{n}(|\text{supp}(P_o) \setminus \text{supp}(P_t)| + |\text{supp}(P_t) \setminus \text{supp}(P_o)|)$$

[1] This test was inspired by a bug we had in one of our classifiers.

4.3 Classifier evaluation

In the attack phase, the classifiers described above take as input a particular observation and calculate how well the observed model fits into different trained models. If we have N trained speakers and we input observed data from an unknown speaker, we obtain a vector of scores $\langle s_1, s_2, \ldots, s_N \rangle$, s_i corresponding to the score of the unknown speaker's model when compared to the model of speaker i. From this vector we have to infer which model describes best the observation and make a guess about the identity of the speaker. After performing t experiments, we obtain t such vectors of scores. In the following we present several techniques to evaluate the performance of the classifiers after several performed experiments.

4.3.1 Recognition rate

The first evaluation technique we consider is *recognition rate* (RR). After each experiment the classifier makes a *guess* about the identity of the unknown speaker. We take as a guess the speaker with the optimal score from the vector of scores, which in the above described classifiers is the minimal score. A guess is considered *correct* if it is unique and is equal to the true identity of the unknown speaker. After performing t experiments, we count how many correct guesses the classifier has performed, and the recognition rate is obtained as follows:

$$\mathrm{RR} := \frac{\#\text{ correct guesses}}{t}$$

The recognition rate is an intuitive measure for the accuracy of classifiers. However, it is a quite conservative measure, as it ignores all results where the speakers are classified close to correctly. For example, if the correct result is always classified with the second score from 15 speakers, we would obtain a recognition rate of 0. Nevertheless, we would still be content with this result because it would mean that the classifier performs far better than random guessing, which would give on average the 8th score. Therefore, we are not only interested in the best-scored speaker, but in a subset of the highest-ranked speakers. In the following sections we describe two classifier evaluation techniques that address this issue.

4.3.2 Average rank

A possible scoring method is computing the *average rank* (AR) over the all obtained ranks. This is a well-known method for classifier evaluation (e.g., see [8]). Let N be the number of speakers and t the number of trials. After performing a trial, from the set of scores $\langle s_1, s_2, \ldots, s_N \rangle$ we compute the *rank* as the position at which the correct speaker was ranked. In case of a score tie, we take the lowest ranking position among all speakers with the same score. Thus, after t trials, we obtain the

ranks $r_1, r_2, \ldots, r_t$, where r_i rank in the i-th trial. We compute the average rank as follows:

$$\mathrm{AR} := \sum_{i=1}^{t} \frac{r_i}{t}$$

Reading the results of this measure is very intuitive as it shows which position is output by the classifier on average, results closer to position 1 being preferred. However, as we are only interested in the highest several ranks, the use of average ranks may result in misleading conclusions. For example, assume a classifier that for $N = 10$ speakers and $t = 2$ trials outputs once rank 3 and once rank 9, thus the average rank here is 6. Here we would prefer another classifier, which outputs once rank 2 and once rank 10, but nevertheless in both cases we obtain the same average rank of 6.

To address those shortcomings of average ranks, we consider giving more weight to ranks closer to position one. A technique which does this is called *discounted cumulative gain* and is discussed in the following section.

4.3.3 Discounted cumulative gain

Discounted cumulative gain (DCG) is a scoring technique used mainly in information retrieval for rating web search engine algorithms [22]. We adapt this measure to our purposes and define it in the following.

Let for $i \in \{1, \ldots, N\}$, the relevance rel_i be defined as number of trials where the correct speaker was ranked i-th. The DCG-measure is defined as:

$$\mathrm{DCG} := \sum_{i=1}^{N} \frac{rel_i}{d(i)},$$

where $d(i)$ is called *discounting function*. In information retrieval literature, $f(i) = \log_2(i+1)$ is usually applied. Using this measure, top-ranked speakers will have a higher weight than lower-ranked ones, but lower ranks will still have a relevance to the final score of a classifier. The smoothness of the discounts can be adjusted by selecting a different base of the logarithm or by selecting a different discounting function, such as a linear (e.g. $d(i) = i$), a polynomial (e.g. $d(i) = i^2$), or an exponential one (e.g. $d(i) = 2^i$).

5 Empirical Evaluation

We conducted a series of experiments to evaluate how well the classifiers described in Chapter 4 perform if we execute an attack against the anonymity in VoIP. In the assumed attack setting, the traffic observed by the attacker contains communication between the victim and N speakers. We simulated VoIP conversations by encoding speeches of 13 speakers using a popular speech codec and we analyzed the durations of what was classified by the codec as voice and pause segments. In the following we describe the conducted experiments and their results.

5.1 Experimental setup

As target speakers, we chose 13 politicians (see Table 5.1): 11 American presidents, one Russian president, and one German chancellor. They were aged between 43 and 77 at the time of recording, 12 of them are male, 11 are English native speakers. This set of voice recordings is homogeneous with respect to the setting in which the speeches were given, as they are official addresses to the nation that were broadcast on radio or television. The collected speeches are available online on [19], [3], [4] and [31]. The length of the collected audio data per speaker varied between 47 and 114 minutes.

We encoded the data using the Speex codec [48], version 1.2rc1. This is an open source speech codec, supported by popular VoIP applications, such as Google Talk [20], TeamSpeak [15], Ekiga [1] and Microsoft Netmeeting [12]. We used the Voice Activity Detection (VAD) mode offered by Speex.

5.2 Results

We tested the performance of the analysis techniques presented in Chapter 4 on the above-described data set. As features, we used voice, pause and (voice,pause,voice)-triplets (cf. Section 4.1.1). For each tested speaker, the speech data was divided into two parts, a *training set* and an *attack set*, which were used respectively for training the classifiers and for performing the actual attack. We additionally tested the robustness of the analysis techniques by applying them on noisy data and on data sets recorded several years apart.

Speaker	Language	Number speeches	Duration (mm:ss)	Speaker's age
Angela Merkel	German	15	53:53	54
Barrack Obama	English	15	68:33	47
Dimitry Medvedev	Russian	12	66:40	43
Dwight D. Eisenhower	English	7	67:28	62-70
Franklin D. Roosevelt	English	4	80:38	54-57
George W. Bush	English	15	50:24	61
Harry S. Truman	English	5	60:48	51-55
Jimmy Carter	English	6	47:56	52-55
John F. Kennedy	English	8	47:10	44-46
Lyndon B. Johnson	English	8	50:25	55-59
Richard Nixon	English	6	113:43	56-58
Ronald Reagan	English	12	51:06	70-77
William J. Clinton	English	20	82:05	53-54

Table 5.1: Data used in the experiments

5.2.1 Performance of the classifiers

We performed the experiments applying the techniques presented in Chapter 4.2, using half of a speaker's data as the training set and the other half as the attack set, and then we repeated the experiments using the second half for training and the first for the attack. Thus, for the 13 tested speakers, a total of 26 experiments were conducted. To evaluate the performance of the classifiers after conducting those experiments, we analyzed the classifiers' recognition rate (RR), i.e., percentage of correctly guessed speakers, average rank (AR), and discontinuous gain (DCG) (cf. Section 4.3). In the following we will only discuss the first two, as the computed values are more intuitively understandable.

Table 5.2, 5.3 and 5.4 show the best results of the performed experiments, using voice, pause and (voice,pause,voice)-triplets as features. They show a comparison between *raw results*, i.e. when the experiments were conducted directly on the collected data, and *optimized results*, i.e. after applying some of the optimization techniques discussed in Section 4.1.2. A comprehensive list of results is presented in Appendix A.

Raw results When no optimization techniques were applied, the best obtained results for each feature were a recognition rate of 34.6% using χ^2 on pauses (Table 5.2), 46.2% using support distance on voice (Table 5.3), and 38.5% using χ^2 on (voice,pause,voice)-triplets (Table 5.4). It is notable that almost always the classifiers performed better than random guessing, which gives an expected recognition rate of 7.7%.

	Raw results			Optimized results			
Classifier	RR	AR	DCG	Variant	RR	AR	DCG
L_1	0.269	4.000	0.581	2	0.346	4.269	0.601
χ^2	0.346	3.577	0.631	0	0.346	3.577	0.631
K-S	0.192	4.115	0.542	2	0.231	4.500	0.545
K-S-mod	0.269	4.192	0.579	2	0.346	3.962	0.611
supp_dist	0.269	3.885	0.514	5	0.385	4.346	0.566
Random	0.077	7	0.412		0.077	7	0.412
Best case	1	1	1		1	1	1

Table 5.2: Best results of the experiments when using pause segments. The *variant* is the longest omitted segment duration when optimization is applied, in number of packets. (RR = recognition rate, AR = average rank, DCG = discontinuous gain)

	Raw results			Optimized results			
Classifier	RR	AR	DCG	Variant	RR	AR	DCG
L_1	0.308	3.231	0.580	2	0.346	3.385	0.587
χ^2	0.269	3.962	0.539	2	0.308	3.692	0.561
K-S	0.154	3.962	0.474	11	0.462	2.692	0.661
K-S-mod	0.115	4.500	0.443	11	0.385	2.923	0.612
supp_dist	0.462	4.538	0.667	5, 7-11	0.462	4.538	0.668
Random	0.077	7	0.412		0.077	7	0.412
Best case	1	1	1		1	1	1

Table 5.3: Best results of the experiments when using voice segments. The *variant* is the longest omitted segment duration when optimization is applied, in number of packets. (RR = recognition rate, AR = average rank, DCG = discontinuous gain)

	Raw results			Optimized results			
Classifier	RR	AR	DCG	Variant	RR	AR	DCG
L_1	0.269	4.577	0.539	100	0.423	3.192	0.694
χ^2	0.385	2.808	0.638	10	0.692	2.577	0.818
K-S	0.269	2.962	0.587	200	0.308	3.346	0.631
K-S-mod	0.077	3.923	0.488	100	0.462	3.654	0.683
supp_dist	0.308	5.346	0.564	100	0.423	3.077	0.694
Random	0.077	7	0.412		0.077	7	0.412
Best case	1	1	1		1	1	1

Table 5.4: Best results of the experiments when using (voice,pause,voice)-triplets. The *variant* is the number of packets per cluster when optimization is applied. (RR = recognition rate, AR = average rank, DCG = discontinuous gain)

Classifier	RR	AR	DCG
supp_dist	0.731	2.577	0.841
Random	0.077	7.000	0.412
Best case	1	1	1

Table 5.5: Results when using support distance on voice data, with clusters of 8 packets (RR = recognition rate, AR = average rank, DCG = discontinuous gain)

Optimized results Using only pause and only voice segments as features, we repeated the experiments omitting short segments of voice and pauses. After tuning the amount of omitted segment durations, we obtained an improved recognition capability of the used classifiers. For pauses, the best recognition rate of 38.5% was obtained using support distance when omitting segments of less than or equal to 5 packets (Table 5.2); the remaining classifiers also improved their performance. For voice, a recognition rate of 46.2% was achieved using the K-S test when omitting segments of less than or equal to 11 packets (Table 5.3), and the remaining classifiers also showed an improved performance.

Using (voice,pause,voice)-triplets, we applied clustering of the data. After tuning the number of the audio packets contained in a cluster, an improvement was observed in all used classifiers, and the best result of 69.2% was obtained using the χ^2 classifier, with 10 packets per cluster (Table 5.4).

In the course of the experiments, we tried out several different combinations of the above-mentioned techniques. The best result of 73.1% recognition rate was obtained using support distance on voice data, with 8 packets per cluster (see Table 5.5). It is interesting to note that this classifier was robust against missing short segments, having similar performance when the first 1 to 20 packet-long segments were omitted. This can be explained by the fact that short segments were very common in our Speex-encoded data, and common durations do not have much weight in the computation of the support distance.

5.2.2 Robustness to noise

In a real-life scenario, the use of various devices and differences in connection quality may cause differences in the audio quality of the conversations. We tested how audio quality would affect the recognition of a speaker by adding *pink noise* to Angela Merkel's speeches. Unlike other types of noise, such as white noise or some other background sound, the energy of pink noise is logarithmically distributed across the human hearing and it is known to be hard to be removed by noise-removal algorithms [46]. The resulting recordings had perceivably worse audio quality than the originals, although the speech was still recognizable by humans. We divided Angela Merkel's data into two equal parts and added noise to each of the parts. One part containing noisy data was used for training and one part containing the original

data (or *normal data*) was used for the attack and vice versa, resulting in a total of four experiments. We used χ^2 on voice, pauses and (voice,pause,voice)-triplets, as this classifier showed the most stable performance in the previous experiments, as well as support distance on voice, as this classifier achieved the best recognition rate. Table 5.6 shows a comparison of the performance of those classifiers when only normal data was used and when noise was added to Angela Merkel's speech (*mixed data*). On voice and (voice,pause,voice)-triplets, χ^2 showed a worsening in its performance, but nevertheless a recognition rate of 25% for voice and 50% for (voice,pause,voice)-triplets was obtained. On pauses, χ^2 performed very poorly on noisy speech. The support distance classifier showed a relatively good performance with a recognition rate of 50%. However, it did not show a stable performance because half of the samples were rated extremely low, and therefore it had an average rank of 7, which is equal to the expected average rank in random guessing.

	Normal data			Mixed data		
Analysis techniques used	RR	AR	DCG	RR	AR	DCG
χ^2 on pauses, no omitted segments	0.5	1.5	0.815	0	8.5	0.298
χ^2 on voice, longest omitted segment duration 2	1	1	1	0.25	3	0.612
χ^2 on (voice,pause,voice), cluster size 10	1	1	1	0.500	3.250	0.659
support on voice, cluster size 8	1	1	1	0.5	7	0.631
Random	0.077	7	0.412	0.077	7	0.412
Best case	1	1	1	1	1	1

Table 5.6: Results when testing noisy vs. normal data of Angela Merkel (RR = recognition rate, AR = average rank, DCG = discontinuous gain)

5.2.3 Age-related changes

As discussed in Section 2.3.1, changes in the way of speaking occur as people get older. It is interesting to investigate whether the classifiers' performance will deteriorate if the data used for training and for the attack is recorded several years apart. We tested this on Bill Clinton's speeches from his first and his second term, which were recorded 7 years apart. For this, we additionally collected 168 minutes of recordings from the time when Clinton was 47 years old. With the best-performing classifiers from the previous experiments, two experiments were performed: using recordings from 1993-1994 for training and recordings from 2000-2001 for the attack and vice versa. Table 5.7 shows a comparison of the classifiers' performance when both the older and the newer datasets (*mixed data*) were used and when only the newer recordings were used (*normal data*). The χ^2 classifier performed similarly on mixed data as on normal data and we observed a worsening only in the results for (voice,pause,voice)-triplets, where the recognition rate sank from 0.5 to 0 and the

average rank sank from 1.5 to 3. Support distance showed a drastic worsening in its performance, showing no recognition ability whatsoever.

	Normal data			Mixed data		
Analysis techniques used	RR	AR	DCG	RR	AR	DCG
χ^2 on pauses, no omitted segments	0	2.5	0.565	0	2	0.630
χ^2 on voice, longest omitted segment duration 2	1	1	1	1	1	1
χ^2 on (voice,pause,voice), cluster size 10	0.5	1.5	0.815	0	3	0.5
support on voice, cluster size 8	1	1	1	0	11.5	0.275
Random	0.077	7	0.412	0.077	7	0.412
Best case	1	1	1	1	1	1

Table 5.7: Results when testing older vs. newer recordings of Bill Clinton (RR = recognition rate, AR = average rank, DCG = discontinuous gain)

It is important to note that the audio quality of the older and the newer recordings was different, probably due to the use of different recording equipment. Thus, the decrease in discrimination ability may be due to differences in audio quality only, and not due to Clinton's age. Furthermore, 7 years may be not a long enough time-span, so that age-related changes in speech occur.

5.2.4 Discussion of results

The experiments we conducted show that using L_1, χ^2 and the two variants of the K-S test, we obtained a recognition rate of 23.1% to 38.5% using pauses, 30.8% to 46.2% using voice and 30.8% to 69.2% using (voice,pause,voice)-triplets. All these results are a clear improvement over random guessing where the expected recognition rate is 7.7%. χ^2 shows the most robust performance, delivering reasonable results even when noise is applied. The best result using this classifier was a recognition rate of 69.2%, which was obtained using (voice,pause,voice)-triplets.

The best results we obtained was a recognition rate of 73.1% using the support distance classifier on voice. It is interesting to analyze the performance of this classifier, which in some cases delivered good results and in other showed a much worse performance than that of the other classifiers, as for example in the tests of older versus newer speeches of Bill Clinton (Section 5.2.3). A deeper look at its construction shows that it distinguishes well between two speakers if there are many segment durations which one of the speakers does not use at all and the other one uses at least once. It is naive to infer that there are some durations that are specific only to a particular speaker and that even given an unlimited number of other people's speeches, those particular durations will never be found in their speech. We conjecture that given enough speech data from all speakers, at some point all possible segment durations will be used at least once by each speaker, and

thus the support distance will be constantly 0. Thus, we do not recommend the broad usage of this classifier. However, the good performance of support distance in some cases indicates that some segment durations are more important than others. This finding may be used when constructing classifiers which give more weight to particular durations and ignore other durations.

6 Conclusion

In the current work we show that in areas where voice activity detection (VAD) techniques are applied, such as mobile telephony and VoIP, even though encryption is commonly applied, the anonymity of users is compromised. We present methods for analysis of VAD-encoded telephone traffic which take the durations of what the VAD algorithm has classified as voice and pauses and output a guess about the identity of the speaker. At first sight, being able to distinguish whether one is speaking or not during a conversation looks like a minor information leakage. However, in the experiments we executed, using speeches of 13 speakers, two of the presented classifiers achieved a recognition rate of 69.2% and 73.1% respectively. Those results imply that this information leakage is sufficient to make a good guess about the identity of the participating speakers. This finding makes us believe that the use of VAD technologies in digital telephony presents serious threat for the anonymity of its users.

Future work

The aim of this work is to show the presence and severity of a particular security threat by presenting tools that can be used for attacks on the anonymity in encrypted telephone conversations. We believe that the performance of the proposed classifiers can be improved by combining some of the proposed techniques or by incorporating new techniques in the data analysis.

Our experiments were not conducted on real telephone conversations, but we used an artificial experimental environment instead. More experiments are needed to evaluate the performance of the proposed methods in a practical scenario by placing an attack against an actual mobile network or a VoIP application. Furthermore, the test data we used contains monologues which have different dynamics than conversations between two speakers. Thus, tests on actual conversations will shed more light on the severity of those security threats.

Bibliography

[1] Ekiga. http://ekiga.org/.

[2] 3GPP. 3GPP - The 3rd Generation Partnership Project. http://www.3gpp.org/.

[3] Administration of the President of the Russian Federation. Videoblog of the president of the russian federation. http://blog.kremlin.ru/.

[4] The American Presidency Project. Audio/video archive. http://www.presidency.ucsb.edu/media.php.

[5] John E. Reid and Associates Inc. The Reid technique of interviewing and interrogation, 2005.

[6] Homayoon Beigi. *Fundamentals of Speaker Recognition*. Springer, January 2010 (to appear).

[7] Stefan Benus, Frank Enos, Julia Hirschberg, and Elizabeth Shriberg. Pauses in deceptive speech. In *Proc. ISCA 3rd International Conference on Speech Prosody*, 2006.

[8] Pavel B. Brazdil and Carlos Soares. A comparison of ranking methods for classification algorithm selection. In *Proceedings of the European Conference on Machine Learning ECML2000*, pages 63–74. Springer-Verlag, 2000.

[9] Joseph P. Campbell. Speaker recognition: a tutorial. *Proceedings of the IEEE*, 85(9):1437–1462, 1997.

[10] Estelle Campione and Jean Véronis. A large-scale multilingual study of silent pause duration. In B. Bel and I. Marlien, editors, *Proceedings of the Speech Prosody 2002 Conference (pp. 199-202). Aixen Provence: Laboratoire Parole et. Langage*, 2002.

[11] Wai C. Chu. *Speech Coding Algorithms: Foundation and Evolution of Standardized Coders*. John Wiley & Sons, Inc., New York, NY, USA, 2003.

[12] Microsoft Corporation. Microsoft Netmeeting. http://www.microsoft.com/downloads/details.aspx?FamilyID=26c9da7c-f778-4422-a6f4-efb8abba021e&displaylang=en.

[13] Michel de Boer. Twinkle. `http://www.twinklephone.com/`.

[14] Luciana Ferrer, Harry Bratt, Venkata R. R. Gadde, Sachin Kajarekar, Elizabeth Shriberg, Kemal S Andreas, and Stolcke Anand Venkataraman. Modeling duration patterns for speaker recognition. In *Proceedings of the EUROSPEECH*, pages 2017–2020, 2003.

[15] TeamSpeak Systems GmbH. Teamspeak. `http://www.teamspeak.com/`.

[16] Steven Greenberg, Dan Ellis, and Joy Hollenback. Insights into spoken language gleaned from phonetic transcription of the Switchboard corpus. In *ICSLP-96*, pages 24–27, Philadelphia, PA, 1996.

[17] GSM-Association. GSM - global system for mobile communications. `http://www.gsmworld.com/`.

[18] John H. Hansen and Sanjay Patil. Speech under stress: Analysis, modeling and recognition. pages 108–137, 2007.

[19] The White House. Your weekly address. `http://www.whitehouse.gov/briefing-room/weekly-address`.

[20] Google Inc. Google talk. `http://www.google.com/talk/`.

[21] International Telecommunication Union ITU. Recommendation g.728: Coding of speech at 16 kbit/s low-delay code excited linear prediction, 1992.

[22] Kalervo Järvelin and Jaana Kekäläinen. Ir evaluation methods for retrieving highly relevant documents. In *SIGIR '00: Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 41–48, New York, NY, USA, 2000. ACM Press.

[23] Frank J. Massey Jr. The Kolmogorov-Smirnov test for goodness of fit. *Journal of the American Statistical Association*, 46(253):68–78, 1951.

[24] Skype Limited. Skype. `http://www.skype.com/`.

[25] Sue Ellen Linville. *Vocal Aging*. Singular, 2001.

[26] Judith A. Markowitz. Voice biometrics. *Commun. ACM*, 43(9):66–73, 2000.

[27] Judith A. Markowitz. Speaker identification and verification (SIV) applications and markets. In *Workshop on Speaker biometrics and VoiceXML 3.0*, SRI International, Menlo Park, CA, US, March 2009.

[28] Bruce Mitchell. A comparison of chi-square and Kolmogorov-Smirnov tests. *Area*, 3:237–241, 1971.

[29] David S. Moore. *Goodness-of-Fit Techniques*, chapter 3. Tests of Chi-Squared Type. Marcel Dekker, New York, 1986.

[30] Barbara Peskin, Jiri Navratil, Joy Abramson, Douglas Jones, David Klusacek, Douglas A. Reynolds, and Bing Xiang. Using prosodic and conversational features for high-performance speaker recognition. In *Proc. Internatonal Conference on Acoustics, Speech, and Signal Processing (ICASSP 2003) Hong Kong*, pages 792–795, 2003.

[31] The Press and Information Office of the Federal Government. Podcasts. `http://www.bundeskanzlerin.de/Webs/BK/De/Aktuell/Podcasts/podcast.html`.

[32] Javier Ramirez, Juan Manuel Gorriz, and José C. Segura. *Robust Speech Recognition and Understanding*, chapter 1. Voice Activity Detection. Fundamentals and Speech Recognition System Robustness, pages 1–22. ISBN 978-3-902613-08-0, I-Tech Education and Publishing, Vienna, Austria, 2007.

[33] Douglas Reynolds, Walter Andrews, Joseph Campbell, Jiri Navratil, Barbara Peskin, Andre Adami, Qin Jin, David Klusacek, Joy Abramson, Radu Mihaescu, Jack Godfrey, Doug Jones, and Bing Xiang. The supersid project: Exploiting high-level information for high-accuracy. In *Proc. International Conference on Audio, Speech, and Signal Processing, Hong Kong*, pages 784–787, 2003.

[34] Douglas Reynolds, Joe Campbell, Bill Campbell, Bob Dunn, Terry Gleason, Doug Jones, Tom Quatieri, Carl Quillen, Doug Sturim, and Pedro Torres-Carrasquillo. Beyond cepstra: Exploiting high-level information in speaker recognition. In *Proceedings of the Workshop on Multimodal User Authentication*, pages 223–229, Santa Barbara, Calif, USA, December 2003.

[35] Douglas A. Reynolds. An overview of automatic speaker recognition technology. volume 4, pages IV–4072–IV–4075 vol.4, 2002.

[36] Douglas A. Reynolds, Thomas F. Quatieri, and Robert B. Dunn. Speaker verification using adapted gaussian mixture models. In *Digital Signal Processing*, volume 10, pages 181–202, 2000.

[37] Jonathan Rosenberg, Henning Schulzrinne, Gonzalo Camarillo, Alan Johnson, Jon Peterson, Robert Sparks, Mark Handley, and Eve Schooler. SIP: Session Initiation Protocol, 2001.

[38] Peter Saint-Andre. Extensible Messaging and Presence Protocol (XMPP): Core, RFC 3920, October 2004.

[39] Susanne Schötz. Acoustic analysis of adult speaker age. pages 88–107, 2007.

[40] Susanne Schötz and Christian Müller. A study of acoustic correlates of speaker age. In *Speaker Classification II*, pages 1–9. Springer-Verlag, 2007.

[41] Henning Schulzrinne, Stephen Casner, Ron Frederick, and Van Jacobson. RTP: A Transport Protocol for Real-Time Applications, 1996.

[42] Elizabeth Shriberg. Higher-level features in speaker recognition. pages 241–259, 2007.

[43] D. E. Sturim, W. M. Campbell, and D. A. Reynolds. Classification methods for speaker recognition. pages 278–297, 2007.

[44] International Telecommunication Union. ICT statistics database. `http://www.itu.int/ITU-D/icteye/Indicators/Indicators.aspx`.

[45] Bernhard H. Walke. *Mobile Radio Networks: Networking and Protocols*. John Wiley & Sons, Inc., 2nd edition, 2002.

[46] Charles V. Wright, Lucas Ballard, Scott E. Coull, Fabian Monrose, and Gerald M. Masson. Spot me if you can: Uncovering spoken phrases in encrypted VoIP conversations. In *SP '08: Proceedings of the 2008 IEEE Symposium on Security and Privacy*, pages 35–49, Washington, DC, USA, 2008. IEEE Computer Society.

[47] Charles V. Wright, Lucas Ballard, Fabian Monrose, and Gerald M. Masson. Language identification of encrypted VoIP traffic: Alejandra y Roberto or Alice and Bob? In *SS'07: Proceedings of 16th USENIX Security Symposium on USENIX Security Symposium*, pages 1–12, Berkeley, CA, USA, 2007. USENIX Association.

[48] Xiph.Org. Speex: A free codec for free speech. `http://speex.org/`.

Appendix A

Comprehensive List of Results

This appendix presents a full list of the results of the experiments described in Section 5.2.1.

Using pauses segments

RR = recognition rate, AR = average rank, DCG = discontinuous gain;
the *variant* is the longest omitted segment duration

L_1 classifier			
Variant	RR	AR	DCG
0	0.269	4.000	0.581
1	0.308	4.423	0.588
2	0.346	4.269	0.601
3	0.269	4.231	0.576
4	0.269	4.385	0.562
5	0.269	4.500	0.560
6	0.231	4.615	0.547
7	0.269	4.769	0.552
8	0.308	4.769	0.565
9	0.269	4.885	0.546
10	0.269	5.000	0.541
11	0.269	5.000	0.541
12	0.231	5.000	0.526
13	0.231	4.962	0.529
14	0.231	4.962	0.529
15	0.231	4.962	0.529
16	0.231	4.962	0.529
17	0.231	4.962	0.529
18	0.231	4.962	0.529
19	0.231	4.962	0.529
20	0.231	4.962	0.529
Random	0.077	7	0.412
Best case	1	1	1

χ^2 classifier			
Variant	RR	AR	DCG
0	0.346	3.577	0.631
1	0.308	3.808	0.613
2	0.308	3.692	0.619
3	0.308	3.615	0.621
4	0.269	3.769	0.601
5	0.269	3.769	0.598
6	0.231	4.308	0.577
7	0.308	4.500	0.593
8	0.269	4.808	0.563
9	0.346	4.808	0.588
10	0.269	5.346	0.540
11	0.192	5.500	0.509
12	0.192	5.462	0.505
13	0.192	5.615	0.493
14	0.231	5.500	0.512
15	0.231	5.538	0.510
16	0.269	5.462	0.529
17	0.269	5.346	0.533
18	0.269	5.269	0.537
19	0.269	5.269	0.537
20	0.269	5.231	0.540
Random	0.077	7	0.412
Best case	1	1	1

RR = recognition rate, AR = average rank, DCG = discontinuous gain;
the *variant* is the longest omitted segment duration

K-S classifier			
Variant	RR	AR	DCG
0	0.192	4.115	0.542
1	0.192	4.885	0.513
2	0.231	4.500	0.545
3	0.192	4.731	0.518
4	0.192	4.846	0.520
5	0.231	4.846	0.532
6	0.231	4.769	0.540
7	0.192	4.923	0.517
8	0.231	5.000	0.531
9	0.192	5.269	0.508
10	0.154	5.269	0.494
11	0.192	5.269	0.506
12	0.192	5.192	0.509
13	0.192	5.154	0.509
14	0.192	5.077	0.510
15	0.231	5.000	0.527
16	0.231	5.000	0.527
17	0.231	5.000	0.527
18	0.231	5.000	0.527
19	0.231	5.000	0.527
20	0.231	5.000	0.527
Random	0.077	7	0.412
Best case	1	1	1

K-S-mod classifier			
Variant	RR	AR	DCG
0	0.269	4.192	0.579
1	0.269	4.154	0.584
2	0.346	3.962	0.611
3	0.269	4.462	0.562
4	0.269	4.692	0.558
5	0.231	4.885	0.532
6	0.269	4.769	0.552
7	0.269	4.808	0.550
8	0.269	4.846	0.549
9	0.231	4.962	0.529
10	0.231	4.962	0.529
11	0.231	5.000	0.527
12	0.231	5.000	0.527
13	0.231	5.000	0.527
14	0.231	5.000	0.527
15	0.231	5.000	0.527
16	0.231	5.000	0.527
17	0.231	5.000	0.527
18	0.231	5.000	0.527
19	0.231	5.000	0.527
20	0.231	5.000	0.527
Random	0.077	7	0.412
Best case	1	1	1

RR = recognition rate, AR = average rank, DCG = discontinuous gain;
the *variant* is the longest omitted segment duration

supp_dist classifier			
Variant	RR	AR	DCG
0	0.269	3.885	0.582
1	0.308	4.346	0.590
2	0.308	4.423	0.581
3	0.346	4.308	0.603
4	0.385	4.308	0.616
5	0.385	4.346	0.615
6	0.308	4.308	0.592
7	0.308	4.231	0.593
8	0.308	4.385	0.586
9	0.192	4.615	0.543
10	0.231	4.615	0.552
11	0.269	4.577	0.568
12	0.192	4.577	0.544
13	0.231	4.615	0.555
14	0.231	4.577	0.553
15	0.192	4.731	0.535
16	0.269	4.769	0.561
17	0.269	4.731	0.568
18	0.231	4.846	0.542
19	0.154	4.885	0.518
20	0.154	4.769	0.521
Random	0.077	7	0.412
Best case	1	1	1

supp_dist classifier, cluster size 8			
Variant	RR	AR	DCG
0	0.692	2.538	0.811
1	0.692	2.500	0.815
2	0.692	2.615	0.812
3	0.692	2.654	0.809
4	0.692	2.692	0.807
5	0.654	2.885	0.791
6	0.654	3.077	0.787
7	0.654	3.269	0.785
8	0.654	3.423	0.778
9	0.615	3.269	0.766
10	0.500	3.346	0.717
11	0.423	3.538	0.681
12	0.346	3.615	0.642
13	0.462	2.808	0.735
14	0.346	3.423	0.657
15	0.346	3.385	0.675
16	0.385	3.577	0.682
17	0.385	3.462	0.690
18	0.385	3.115	0.703
19	0.385	3.423	0.690
20	0.346	3.731	0.663
Random	0.077	7	0.412
Best case	1	1	1

Using voice segments

RR = recognition rate, AR = average rank, DCG = discontinuous gain;
the *variant* is the longest omitted segment duration

L_1 classifier			
Variant	RR	AR	DCG
0	0.308	3.231	0.653
1	0.269	3.192	0.638
2	0.346	3.385	0.652
3	0.308	3.385	0.643
4	0.308	3.385	0.643
5	0.308	3.500	0.633
6	0.308	3.500	0.634
7	0.308	3.500	0.635
8	0.192	3.538	0.602
9	0.231	3.577	0.608
10	0.269	3.654	0.615
11	0.269	3.731	0.609
12	0.269	3.923	0.594
13	0.231	3.769	0.581
14	0.231	3.846	0.578
15	0.192	3.846	0.566
16	0.192	3.731	0.577
17	0.192	3.808	0.569
18	0.154	3.808	0.556
19	0.115	3.654	0.550
20	0.115	3.769	0.548
Random	0.077	7	0.412
Best case	1	1	1

χ^2 classifier			
Variant	RR	AR	DCG
0	0.269	3.962	0.610
1	0.269	3.500	0.624
2	0.308	3.692	0.630
3	0.308	3.731	0.628
4	0.308	3.769	0.623
5	0.231	3.923	0.586
6	0.269	3.846	0.602
7	0.269	3.846	0.604
8	0.231	3.846	0.591
9	0.231	3.962	0.587
10	0.231	4.000	0.585
11	0.231	4.154	0.580
12	0.192	4.269	0.563
13	0.192	4.192	0.568
14	0.192	4.231	0.567
15	0.192	4.192	0.564
16	0.192	4.231	0.562
17	0.192	4.077	0.565
18	0.192	4.000	0.566
19	0.192	3.885	0.574
20	0.192	3.962	0.565
Random	0.077	7	0.412
Best case	1	1	1

RR = recognition rate, AR = average rank, DCG = discontinuous gain;
the *variant* is the longest omitted segment duration

K-S classifier			
Variant	RR	AR	DCG
0	0.154	3.962	0.561
1	0.308	3.423	0.638
2	0.346	3.269	0.655
3	0.462	3.231	0.694
4	0.423	3.077	0.689
5	0.423	3.192	0.684
6	0.346	3.231	0.650
7	0.385	2.846	0.680
8	0.423	2.654	0.703
9	0.423	2.808	0.698
10	0.462	2.731	0.713
11	0.462	2.692	0.716
12	0.346	2.885	0.665
13	0.308	3.077	0.643
14	0.269	3.192	0.618
15	0.269	3.038	0.632
16	0.269	3.115	0.629
17	0.308	3.000	0.645
18	0.385	2.808	0.680
19	0.385	2.769	0.690
20	0.385	2.923	0.678
Random	0.077	7	0.412
Best case	1	1	1

K-S-mod classifier			
Variant	RR	AR	DCG
0	0.115	4.500	0.532
1	0.192	4.308	0.569
2	0.231	4.115	0.605
3	0.231	4.192	0.601
4	0.269	4.000	0.604
5	0.269	4.115	0.602
6	0.231	3.769	0.595
7	0.346	3.192	0.668
8	0.308	2.962	0.650
9	0.308	2.962	0.648
10	0.385	2.962	0.669
11	0.385	2.923	0.675
12	0.308	3.077	0.640
13	0.308	3.192	0.628
14	0.231	3.346	0.596
15	0.231	3.115	0.610
16	0.231	3.154	0.609
17	0.231	3.154	0.605
18	0.231	2.962	0.621
19	0.231	3.000	0.624
20	0.346	3.000	0.665
Random	0.077	7	0.412
Best case	1	1	1

RR = recognition rate, AR = average rank, DCG = discontinuous gain;
the *variant* is the longest omitted segment duration

supp_dist classifier			
Variant	RR	AR	DCG
0	0.462	4.538	0.667
1	0.462	4.538	0.667
2	0.462	4.577	0.667
3	0.462	4.615	0.666
4	0.462	4.577	0.667
5	0.462	4.538	0.668
6	0.462	4.577	0.668
7	0.462	4.538	0.668
8	0.462	4.538	0.668
9	0.462	4.538	0.668
10	0.462	4.538	0.668
11	0.462	4.538	0.668
12	0.423	4.538	0.655
13	0.423	4.538	0.655
14	0.423	4.538	0.655
15	0.423	4.500	0.655
16	0.423	4.538	0.655
17	0.423	4.500	0.655
18	0.462	4.500	0.668
19	0.423	4.577	0.654
20	0.423	4.615	0.653
Random	0.077	7	0.412
Best case	1	1	1

supp_dist classifier, cluster size 8			
Variant	RR	AR	DCG
0	0.731	2.577	0.841
1	0.731	2.577	0.841
2	0.692	2.654	0.826
3	0.692	2.654	0.826
4	0.692	2.615	0.826
5	0.692	2.654	0.826
6	0.692	2.615	0.826
7	0.692	2.615	0.826
8	0.731	2.654	0.839
9	0.731	2.654	0.839
10	0.731	2.654	0.839
11	0.731	2.654	0.839
12	0.731	2.654	0.839
13	0.692	2.731	0.825
14	0.692	2.731	0.825
15	0.692	2.731	0.825
16	0.692	2.731	0.825
17	0.692	2.731	0.825
18	0.692	2.731	0.821
19	0.692	2.769	0.821
20	0.692	2.769	0.821
Random	0.077	7	0.412
Best case	1	1	1

Using (voice,pause,voice)-triplets

RR = recognition rate, AR = average rank, DCG = discontinuous gain;
the *variant* is the number of packets per cluster

L_1 classifier			
Variant	RR	AR	DCG
1	0.269	5.577	0.539
5	0.346	4.846	0.594
10	0.269	4.462	0.570
20	0.154	4.192	0.535
40	0.308	3.654	0.628
80	0.423	3.385	0.680
100	0.423	3.192	0.694
200	0.308	3.346	0.644
400	0.385	3.692	0.646
800	0.231	4.962	0.557
1000	0.154	6.346	0.462
1200	0.192	5.846	0.510
Random	0.077	7	0.412
Best case	1	1	1

χ^2 classifier			
Variant	RR	AR	DCG
1	0.385	3.808	0.638
5	0.577	2.615	0.776
10	0.692	2.577	0.818
20	0.462	2.462	0.737
40	0.462	3.000	0.698
80	0.423	3.000	0.690
100	0.385	3.115	0.682
200	0.231	3.500	0.598
400	0.385	3.538	0.654
800	0.269	4.808	0.556
1000	0.192	5.846	0.502
1200	0.115	6.346	0.465
Random	0.077	7	0.412
Best case	1	1	1

K-S classifier			
Variant	RR	AR	DCG
1	0.269	3.962	0.587
5	0.154	4.846	0.521
10	0.269	4.346	0.579
20	0.154	5.538	0.486
40	0.308	4.192	0.586
80	0.462	3.808	0.688
100	0.308	3.615	0.620
200	0.308	3.346	0.631
400	0.231	3.577	0.590
800	0.231	5.308	0.528
1000	0.077	5.808	0.440
1200	0.115	5.192	0.497
Random	0.077	7	0.412
Best case	1	1	1

K-S-mod classifier			
Variant	RR	AR	DCG
1	0.077	4.923	0.488
5	0.154	5.692	0.483
10	0.154	5.077	0.491
20	0.154	5.077	0.500
40	0.269	4.231	0.583
80	0.346	3.654	0.640
100	0.462	3.654	0.683
200	0.308	3.423	0.635
400	0.231	3.885	0.587
800	0.154	5.423	0.512
1000	0.077	6.154	0.431
1200	0.192	4.923	0.536
Random	0.077	7	0.412
Best case	1	1	1

RR = recognition rate, AR = average rank, DCG = discontinuous gain;
the *variant* is the number of packets per cluster

supp_dist classifier			
Variant	RR	AR	DCG
1	0.308	5.346	0.564
5	0.346	4.692	0.596
10	0.269	4.423	0.570
20	0.154	4.154	0.535
40	0.308	3.538	0.630
80	0.423	3.385	0.680
100	0.423	3.077	0.694
200	0.308	3.346	0.644
400	0.385	3.615	0.647
800	0.231	4.923	0.558
1000	0.154	6.346	0.462
1200	0.192	5.769	0.511
Random	0.077	7	0.412
Best case	1	1	1