Design and Evaluation of Steganography for Voice-over-IP

Christian Krätzer, Jana Dittmann, Thomas Vogel, Reyk Hillert
Advanced Multimedia and Security Lab (AMSL)
Otto-von-Guericke-Universität
Magdeburg, Germany
Email: {kraetzer, jdittman, tvogel, hillert}@iti.cs.uni-magdeburg.de

***Abstract*—According to former results from [1] in this paper we summarize the design principles from the general approach and introduce extended experimental test results of a Voice-over-IP (VoIP) framework including a steganographic channel based on [1], [13], [15] and [16]. We show that using this framework it is largely secure to transmit hidden messages during a VoIP session and demonstrate results with respect to perceptibility for music and speech data.**

I. INTRODUCTION

For digital images as well as digital audio there exist many steganographic techniques [3][4][5], and furthermore there exist a lot of approaches to detect steganography in digital images [6][7][8]. However there are only few methods published considering Voice-over-IP (VoIP) as a new field for applied steganography. The term "VoIP" describes the digitalization, compression and transmission of analogue audio signals (in the majority of cases speech) from a sender to a receiver using IP packets. The receiver applies the reverse process and gets the reconstructed audio signal. After that he can act as the sender. For transmission the size of the used network and the distance between communication partners are of little relevance which means VoIP can be and already is used for worldwide telephony. Many applications of VoIP technology have been developed and are currently under development. For that reason embedding hidden messages in VoIP communication is a very interesting task and may become subject of further studies. In this paper we use the JVOIPLIB 1.3.0 [2] as VoIP framework as basis for our analysis, since this framework is platform-independent and can be used free of charge. Details about the software and our specific implementation are given in [1]. Beyond the work described in former publications we present an extended design and extensive test results. The tests we performed aim at perceptibility, probability of detection, as well as possible malfunction while handling large amounts of data over a long time. In [16] we presented first test results for this properties and it could be shown by using a well established steganographic approach (time domain LSB embedding), that a reliable steganographic side channel communication could be established. The tests in [16] did show a transparency which was lower than the results for comparable steganographic algorithms like Steghide [3]. This fact was identified in [15] where steganographic algorithms were benchmarked for their embedding transparency. Based on these compromising results we started to improve the perceptual quality for speech data by introducing silence detection. The paper is organised as follows: In chapter II an abstract overview of the analyzed VoIP scenario is summarized and the silence detection approach is introduced. Chapter III gives a brief overview of the design concept including the participating components. In chapter IV test performance and test results are presented and compared with the first tests from [16] in accordance to perceptibility as one measure for detectability (as discussed in [18] for visual attacks). The paper concludes with a summary and future work.

II. OVERVIEW

The environment of an active VoIP communication using a steganographic channel is illustrated in Figure 1. Alice (A) on the left and Bob (B) on the right are talking over an unsuspicious VoIP connection, as introduced in [1], [13] and [16]. Let us assume, Alice wants to send a hidden message to Bob, that means Alice acts as sender and Bob as receiver. She embeds her message in the VoIP stream by using secret side information which is known by Bob, too, but no one else has it.

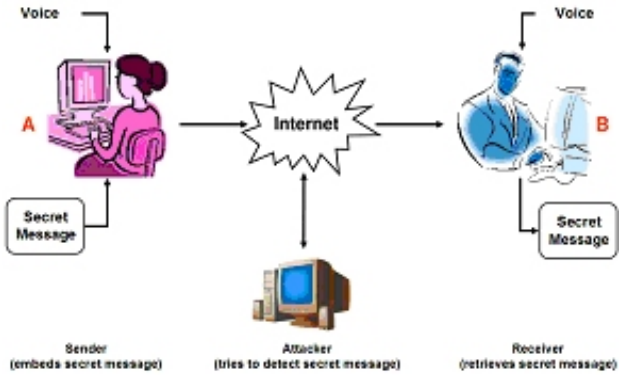

Figure 1. VoIP scenario with steganographic channel.

For describing that communication in a more formal manner we use the following statements. First sender and receiver choose from a set of codecs denoted by $SC=\{sc_1,sc_2,sc_3,..,sc_i \mid i\in N\}$ one audio codec for their communication. From the set of steganographic embedding techniques $SE=\{se_1,se_2,se_3,...,se_i \mid i\in N\}$ Alice chooses one algorithm while Bob selects an according retrieving technique from $SR=\{sr_1,sr_2,sr_3,...,sr_i \mid i\in N\}$. If both techniques match Bob is able to reconstruct the message from Alice. In order to increase security the hidden message is encrypted by using a symmetric cryptographic scheme from the set of all cryptographic methods $CM=\{cm_1,cm_2,cm_3,...,cm_i \mid i\in N\}$. For applying a cryptographic scheme a secret key K is necessary which is generated from a secret password. The mapping of a password to a secret key of a fixed length is applied by choosing a cryptographic hash function. After encrypting the hidden message the message bits are uniformly distributed and spread over the whole audio stream by using a mixing algorithm. As input the mixing algorithm gets a pseudo random number which is generated by a pseudo random number generator (PRNG).

Silence Detection

Based on the results from [16] and [15] we see that the overall approach has the lowest transparency for speech data. This fact is assumed to origin from the limited dynamic range of speech signals and the many occurrences of pauses (silence) in speech. From our investigations we deduced that a silence detection for speech becomes important as a transparency enhancing method, since it is assumed that an embedding into digital silence is the most perceptible embedding scenario. At this point it has to be differentiated between digital and analogue silence (the first one being an encoded "Zero"-signal, the later being the signal produced by a microphone in a silent environment). The silence detection method implemented for the LSB embedder used here is used to detect samples containing digital silence.

III. Design of VoIP Scenario

For our prototypical scenario we concentrate on one codec $sc_1\in SC$, i.e. we use RAW PCM (8,000 Hz, 8 Bit). As a main requirement we postulate embedding and retrieving must be possible without causing delays or interferences during audible communication. Hence, for embedding the hidden message we choose for $se_1\in SE$ a Least Significant Bit (LSB) scheme, providing a high capacity and low complexity. According to this the retrieving scheme is chosen. For encryption we use *Twofish* cipher [9] and for the cryptographic hash function *Tiger* [12], since for the well-known cryptographic hash functions as MD5 (128 Bit), SHA-1 (160 Bit) and RIPEMD (160 Bit) collisions have been found (see [10] and [11]). In addition to *Tiger* we use MD5 which produces a shorter hash value to calculate a checksum over the hidden message which is only used for detecting errors during transmission.

A. Sender

Using the above techniques Alice starts the embedding process which is illustrated in Figure 2. She uses a secret password from which a cryptographic hash value is calculated referred to as secret key *K*. The secret key is used for encrypting the hidden message with *Twofish* cipher. After encryption a hash value (*CHK*) is calculated, which helps the receiver to detect errors caused by packet loss or intense network traffic. Furthermore a binary pattern is generated referred to as *Begin Of Message* (*BOM*). *BOM* indicates the start of the hidden message in the audio stream and contains also the length of the message. Combining the *BOM* pattern, the encrypted message and the hash value results in the data to be embedded on sender side. After constructing the message the data is embedded by a spatial domain LSB scheme. The message bits are stored in the least significant bits of the audio samples taken from the VoIP stream. All packages are considered for embedding where no silence is detected. The packet interval I_S is set to 20ms which means 50Hz. Therefore the maximum capacity C_P of one VoIP packet is given by equation (1) depending on the chosen sample rate P_{samp}. In Table 1 typical sample rates and the resulting maximum capacity are listed. These theoretical values given are valid under the assumption that we have a variable package size and that we are sending a mono signal (for stereo signals the maximum capacity is doubled since we could embed in the LSB for each channel). Moreover, the capacity for a transmission of hidden messages in one using the according sample rate is given.

P_{samp} in Hz	C_P in Bits/package	C in Mbytes/h
8,000	160	3.6
11,025	222	4.96
22,050	441	9.92
44,100	882	19.85

Table 1. Maximum capacity depending on the used sample rate.

In general it is not advisable to use the complete capacity C_P for embedding a secret message. Using maximum capacity may introduce perceptible distortions, especially in silent parts of audio data. Furthermore statistical detection of modifications becomes easier since each least significant bit is changed and contains a part of the embedded message. For that reason only a subset of the possible embedding positions should be used for embedding. Introducing the payload factor *packet_usage* we define the payload C_P^* as illustrated in (1).

$$C_P^* = round\left(\frac{packet\_usage \cdot C_P}{100}\right), \quad C_P = \frac{P_{samp}}{I_S} \qquad (1)$$

For example, choosing *packet_usage*=1% we get for the sample rate P_{samp}=8,000 a payload C_P^*=2 bits/packet. Beside encryption we try to achieve better security by using a mixing algorithm in order to distribute the hidden message over the

VoIP stream. For each packet we determine individual positions for embedding the message bits by using pseudo random numbers. As PRNG we use sp_1=*MT19937* which has a period of 2^{19937}-1 and outputs more than 16 million uniformly distributed pseudo random numbers per second. It is initialized by using the secret key K and its output indicates the position for embedding the next message bit. For a better understanding the process of mixing is illustrated in Figure 3. The cover data is represented by exemplary sample values of a VoIP packet S_P. Let 100_2 be the hidden message M^*. Applying the mixing algorithm the positions idx_P={0,1,2,...} are shuffled and changed to idx_P^*={6,2,5,...}. By using the new sequence the encoder embeds the message M^* at the defined positions by adjusting the least significant bit and output the modified VoIP packet S_P^*.

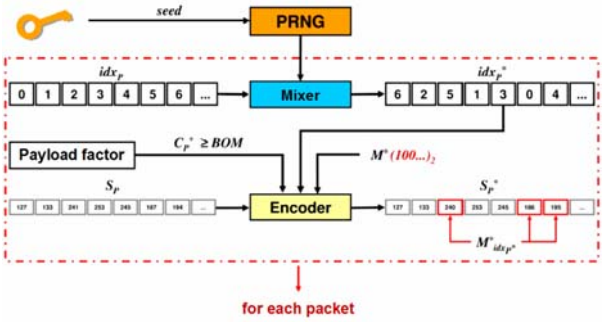

Figure 2. Encoder scheme.

B. Receiver

Sender and Receiver must share the same knowledge of the used parameters which means both of them must know the secret key *K* and the applied payload factor *packet_usage*. After receiving the VoIP packets the recipient uses its key *K* to reconstruct the pseudo random numbers that have been used by the sender. These numbers are used to find the positions within the audio stream *SP* and are stored in a shift register alike packet buffer *MP**. After each writing to *MP** it is checked if the packet buffer contains the synchronization pattern *BOM*. If this is true all following bits of *MP** will be appended to a message buffer *M**. If the given length is reached the following bits are interpreted as a hash value (*CHK*) which is compared to the hash value calculated over the message in *M**. If both hash values are equal the transmission of the hidden message has been finished successfully. Then the message is decrypted and saved to disk.

C. Attacker

In order to make statements about security concerns and non-perceptibility of the described scenario a third person Willy (W) acts as an attacker which is interested in detecting the hidden message of Alice. Let us assume, Willy is capable of accessing the network and detecting VoIP communication. Using their abilities she finds the communication between Alice and Bob and tries to detect the hidden message by analyzing the transmitted VoIP packets. For analysis she uses the Intrusion Detection/Prevention System (IDS/IPS) introduced in [13] and the included module for steganalysis. In [1] the 13 implemented attacks are listed and described in detail.

IV. Evaluation and Test Results

The following results have been reached for the test goals of the usefulness of perceptibility as a steganalytical approach for this algorithm and the impact of silence detection on the transparency. The tests from [16] implied for an embedding with *packet_usage* =100% an average |ODG| value of 0.24 for music and 0.40 for speech signals. Since these test results were computed on a small test-set and without the benefit of working silence detection they had to be confirmed with a larger test-set and improved by using silence detection.

For the tests in [15] an audio test set of 389 files was used (as set of covers), which is divided into four main categories (music, sounds, speech and SQAM) and 24 sub-categories. All audio files are PCM coded WAVE files with 44100 Hz sampling rate, 16 bit quantisation and stereo (audio CD format). The audio test set is described in more detail in [15]. In an offline test for the evaluation of the embedding transparency the ODG value between the original file and the file with the embedded steganographic message is computed using the Open Source software tool EAQUAL (Evaluating of Audio QUALity [17]). A global transparency value for the algorithm is given as the means (sum of all absolute values of the ODGs divided by the number of files) of the ODG values between all 389 files of the test set and the corresponding stego files. Since the ODG value is defined within the range of [0,-4] and EAQUAL in some cases returns values slightly larger than 0 those values are considered to be rounding errors. In the presentation of the results the absolute value of the ODG (|ODG|) is used for a better understanding. In all 389 cases the message *M*=”UniversityOfMagdeburg” was embedded using a constant *K*. The embedding strength (*packet_usage*) for tests performed was set to 100% (all LSBs were used for embedding the message) since this is assumed to be the most perceptible embedding strength.

Test Results

For the version of the embedder (version *Heutling051208*) used in [15] an average |ODG| of 0.017969152 (values ranging from 0.0 to 0.07) was computed. One file of the test set (which does only contain digital silence for the complete duration) was (rightfully) discarded by the algorithm due to its silence detection capabilities. The steganographic message was embedded into all other files and retrieved successfully in any case.

Since the preliminary test results discussed in [16] implied a strong context dependency this had to evaluated in more detail in [15] since it would have an implication for possible capacities for different context classes for a given transparency threshold. In the tests performed in [15] the average |ODG| values for the 24 sub-categories of the audio test set of 389 files have been computed. From the results for the different

categories it has to be noted that embedding into music signals still leads to better |ODG| values (average: 0.01298) than embedding into speech signals (average: 0.03213).

Comparison to the results of first tests

If the results from the preliminary tests described in [16] and the evaluation done in [15] are compared a large improvement of the |ODG| values for the same embedding strength can be noticed. These results are compared in figure 3. In this figure the results from [15] are coloured grey while the results from [16] are white.

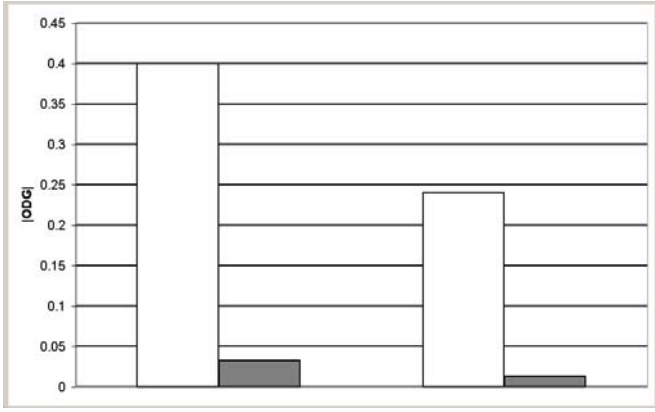

Figure 3.comparison of the results (left speech, right music)

The reason for this has to be sought mostly in the added silence detection feature. The context dependency first indicated by [16] is affirmed by the results from [15].

V Summary and Future Work

Our tests have demonstrated that VoIP communication can be practically used for steganographic applications. Tests with a large test set have shown that an embedding (and detection) can be done reliably and very transparent. The embedding transparency was improved by using a silence detection. For further work the silence detection should be expanded to cover also analogue silence, a step which might further improve the transparency but requires hardware dependent calibration. Furthermore the impact of transmission errors during real VoIP sessions on the introduced steganography approach and the steganalytical transparency have to be researched.

Acknowledgments

We wish to thank Sebastian Heutling for providing us the used offline LSB embedder (version *Heutling051208*) for the VoIP scenario.

Effort sponsored by the Air Force Office of Scientific Research, Air Force Materiel Command, USAF, under grant number FA8655-04-1-3010. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Office of Scientific Research or the U.S. Government. The work about transparency evaluation described in this paper has been supported in part by the European Commission through the IST Programme under Contract IST-2002-507932 ECRYPT. The information in this document is provided as is, and no guarantee or warranty is given or implied that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

References

[1] Jana Dittmann, Danny Hesse, Reyk Hillert: Steganography and steganalysis in voice-over IP scenarios: operational aspects and first experiences with a new steganalysis tool set, Proc. of SPIE, Vol. 5681, Security, Steganography, and Watermarking of Multimedia Contents VII, San Jose, 2005, pp. 607-618.

[2] JVOIPLIB, Jori's Voice over IP library, Website: http://research.edm.luc.ac.be/jori/jvoiplib/jvoiplib.html, 2004.

[3] StegHide - Homepage: http://steghide.sourceforge.net/, 2005.

[4] Steganos - Homepage: http://www.steganos.de/, 2005.

[5] Westfeld, F5 - Homepage: http://wwwrn.inf.tu-dresden.de/~westfeld/f5.html, 2005.

[6] J. Fridrich. M. Goljan, D. Hogea, Steganalysis of JPEG Images: Breaking the F5 Algorithm, 5th Information Hiding Workshop 2002.

[7] Nils Provos, Steganography Detection with Stegdetect, Website: http://www.outguess.org/detection.php, 2004.

[8] Andreas Westfeld: Detecting Low Embedding Rates, in Fabien A. P. Petitcolas (Ed.): Information Hiding, 5th International Workshop, IH 2002, Noordwijkerhout, The Netherlands, October 7-9, 2002, Lecture Notes in Computer Science 2578 Springer 2003, pp. 324-339.

[9] Schneier: Twofish Cipher. http://www.schneier.com/twofish.html, 2005.

[10] Wang, Feng, Lai: Collisions for Hash Functions MD4, MD5, HAVAL-128 and RIPEMD. Proc. of the Eurocrypt, 2004.

[11] Wang, Yin, Yu: Collision in the Full SHA1. Crypto, Santa Barbera (USA), 2005.

[12] Biham, Anderson: Tiger - A Fast New Cryptographic Hash Function, http://www.cs.technion.ac.il/biham/, 1995.

[13] Jana Dittmann; Danny Hesse: Network Based Intrusion Detection to Detect Steganographic Communications Channels -- on the Example of Audio Data: Multimedia Signal Processing; IEEE 2004 IEEE 6th Workshop on Multimedia Signal Processing, Sep. 29th - Oct. 1st 2004, Siena, Italy, ISBN 0-7803-8579-9, 2004.

[14] K. Gopalan: Audio Steganography by Amplitude or Phase Modification, Proc. Of 15th Annual Symposium on Electronic Imaging -- Security, Steganography, and Watermarking of Multimedia Contents V, San Jose, 2003.

[15] Christian Kraetzer, Jana Dittmann, Andreas Lang; Transparency benchmarking on audio watermarks and steganography; to appear in SPIE conference, at the Security, Steganography, and Watermarking of Multimedia Contents VIII, IS&T/SPIE Symposium on Electronic Imaging, 15-19th January, 2006, San Jose, USA, 2006

[16] Thomas Vogel, Jana Dittmann, Reyk Hillert and Christian Kraetzer, Design und Evaluierung von Steganographie für Voice-over-IP, To appear in Sicherheit 2006 GI FB Sicherheit, GI Proceedings, Feb 2006, Magdeburg, Germany

[17] EAQUAL, Alexander Lerch, zplane.development, EAQUAL - Evaluation of Audio Quality, Version: 0.1.3alpha, http://www.mp3-tech.org/programmer/misc.html, 2002

[18] Andreas Westfeld, Andreas Pfitzmann: Attacks on Steganographic Systems. S. 61–76 in Andreas Pfitzmann (Hrsg.): Information Hiding. Third International Workshop, IH'99, Dresden, Germany, September/October, 1999, Proceedings, LNCS 1768, Springer-Verlag Berlin Heidelberg 2000.