Computer Based Steganography: How it works and why therefore any restrictions on cryptography are nonsense, at best.*+

Elke Franz[1], Anja Jerichow[1], Steffen Möller[2], Andreas Pfitzmann[1], Ingo Stierand[2]

[1]University of Dresden, Institute of Theoretical Computer Science, D–01062 Dresden Germany, {ef1, jerichow, pfitza}@inf.tu-dresden.de

[2]University of Hildesheim, Institute of Computer Science, PF 101 363, D-31113 Hildesheim, Germany, {moeller, stierand}@informatik.uni-hildesheim.de

Abstract: In the future, messages, e.g. speech, text or pictures, will be transmitted digitally since this is cheaper, more perfect and more flexible. It is possible to hide messages, which are of necessity much shorter, nearly unrecognizable for outsiders in such digitized messages. In this article we describe how computer based steganography works and give a summary on the results of our implementation.

Keywords: encoding, encryption, digitization, hissing noise, background noises, listening tests, cryptography, ISDN, method with threshold values, sonogram, steganography, hidden transmission

1 Introduction

It is a fact that at all times people want to hide certain things of their private life or of business interests. Presumably it was one of the first inventions by people to do something against the fact that strangers get to know any secret.

There are two basic ideas how to keep something as a secret. An object can be hidden at a secret place. One can hope that nobody will find this object except people who are familiar with the secret. Another possibility is to put the object somewhere in such a way that only certain people are able to use or to see it. One example of the latter is a safe. Only people who know the correct combination to open the safe can access the object. In this case it does not matter how many people know the content of the safe.

But not only objects or things should be hidden. Often messages are supposed to be readable only by the intended recipient. The idea how to do so was already used in antiquity: a message will be encrypted. So only the person who can decrypt it is able to read the contents of the message.

Things are different now. When in former times a herald had to be on his way for many days to deliver a message, nowadays computers take on the communication for us. Digital communication networks are used all over the world. By introducing computers for communication, cryptography can also be realized. A computer can work with algorithms to encrypt a single page of text which would take a person many years. Thus encrypting and transmitting data is very fast and user-friendly in the computer age.

---

* Most of the text was originally published by Steffen Möller, Andreas Pfitzmann and Ingo Stierand in German in the Journal "DuD, Datenschutz und Datensicherung" 18/6 (1994) 318-326.

+ We thank the German Science Foundation (DFG) and the Gottlieb-Daimler - and Karl-Benz Foundation, Ladenburg (Germany) for their financial support.

1.1 Cryptography under Supervision

Since the technological progress has made the use of encryption potentially cheap and easy, there has been a discussion about restricting the use of cryptography. In some countries different restrictions on the use of cryptography are already laid down by law. The official argument is that without effective restrictions on cryptography, the fight against international criminality would be considerably obstructed.

It is proposed that the registration of cryptographic systems should be a duty, whether they are products of hardware or software. Moreover, some people would like that for supervision, everybody who wants to encrypt his data, should receive the secret key from a nationalized allocation office. This was motivated by governmental efforts in order to decode secret messages which were intercepted. Definitely, such restrictions would succeed in a serious degrading of cryptography, one of the most important and efficient protection methods available in computer science.

"I want to have the right to protect my data against unwanted access. But the interest of the investigation offices to catch a criminal person should go forward." Certainly, it is legitimate to argue like that. But probably it is not clear to everybody that the restriction of cryptography can as little protect against the transmission of secret data (e.g. data about illegal drug transports) as the ban on disguising can protect against bank robbery. Even if we assume that criminals would adhere to these restrictions there are still methods to overcome them. Steganography is one simply to use method. Investigators are not able to establish the mere existence of a hidden message, not to mention its content. Computer based steganography cannot be prevented as long as the estate of programmable computers is widespread. It supplies enough "hidden" bandwidth for planning and coordination of crimes. Hence, any effort to restrict encryption is ineffective with respect to criminal prosecution. Moreover, such a law would make it even more difficult to observe their activities. Cryptography by law would be a great restriction of the personal rights of everybody.

Trustworthy encryption is essential for the citizens if they want to protect their right of informational self-determination. Trust in a protective measure requires its free knowledge, organization and usage. Therefore, we strongly advise against any efforts to restrict the use of cryptography.

With this article we make a contribution for a broad disapproval of a law of regulating cryptography. In a few years investigation offices would also have to recognize that a restriction of cryptography would not be a good solution. We proved that it is possible to hide the transmission of data which we want to keep secret!

1.2 Hiding Messages to Overcome Restrictions on Cryptography

1.2.1 In General

Though the encoding of data can be monitored, one may send a letter or a painting which appears absolutely inconspicuous to an observer. E.g. one could send a letter to his brother containing a painting from his three years old daughter – the painting of an apple tree. A normal observer like a police man would see the picture – the painting of the apple tree. The covering letter will tell him that the proud father has sent the painting of his daughter. But the brother would count the number of apples on the picture and know: "Our meeting will be at 3 pm."

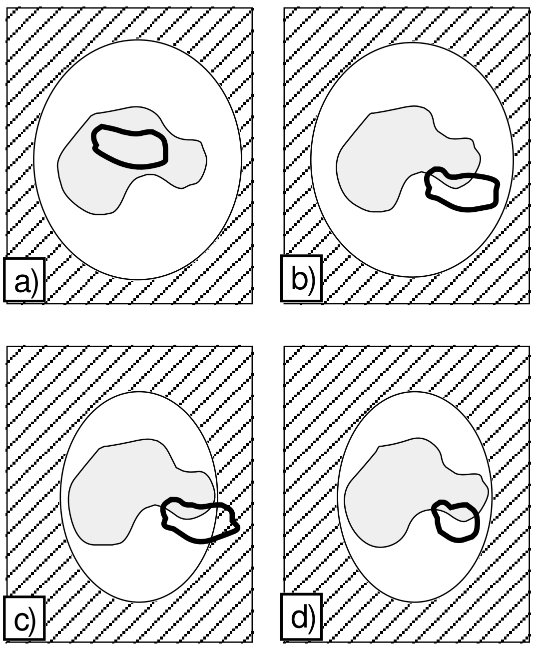
If a letter is sent which apparently can be read by everybody, then nobody would become suspicious and start looking for secret messages. Also if somebody suspects a criminal act, e.g. if it is known that the sender of the letter does not have a daughter, both parties can still find an explanation about sending this picture. Anyway, how could one definitely prove the transmission of a secret information?

1.2.2 Digitized Speech

In this article we do not want to count apples. Our "painting" is a usual telephone talk. It could be very short like between two foreign parties: "... Sorry, I must have dialed the wrong number ... doesn't matter ..." or anything unsuspicious. We tested if it was possible to transmit secret data in addition to the speech information when making a phone call – without giving the opportunity of observation.

To analyze the telephone has two reasons: First, the telephone is the most widespread telecommunication tool (e.g. nearly every household owns a telephone). Second, the new ISDN (Integrated Services Digital Network, see also Section 2) offers the basic requirement for our manipulations: The transmission of telephone talks is **digital**. With the fast growing internet it is no longer necessary - by now there are more internet accounts than ISDN phones. The internet adds the possibility to stay anonymous to the possibility of hidden data transmission.

1.3 Using Methods for Detecting Steganography to Improve Steganography



a) perfectly secure stegosystem: It modifies only inside the nondeterminism. It is impossible to find a stegoanalytical method for detecting, because each method must accept the existing nondeterminism (see also §4.3.2).
b) pragmatically secure stegosystem: It also modifies samples which would be determined. The modifications, however, are unrecognizable again if the stegoanalytical method is not good enough.
c) insecure stegosystem: The stegoanalytical method for detecting hidden data was improved. Now secret messages are found.
d) improved stegosystem: By using the stegoanalytical method of c), the stegosystem has been improved. The modifications are again in the sector in which they are not recognized.

nondeterminism, i.e. the area in which bits are not determined because of background noises and also quantization errors
stegosystem
area in which modifications cannot be recognized by the stegoanalytical method
area in which modifications are recognized by the stegoanalytical method

Figure 1: Methods for detecting and improving steganography

In the long run, it is not possible to prove steganography to court. Even if there would be a way to prove it once, in any case it is not possible in the long run:

For hiding data, we use a stegosystem. Let us suppose that a stegoanalytical method is found which allows to detect hidden data. But the detection is not enough. If somebody is blamed for using steganography, this use must be proved by explaining the stegoanalytical method. Now this stegoanalytical method can be used to improve the stegosystem: It serves as a method of testing. When our secret data could be detected by this method, the stegosystem is not allowed to modify anything. That means we now hide our data in such a way that the stegoanalytical method cannot find them. A new stegoanalytical method would be necessary to see our data, and if one has been developed we can use it again as a method of testing. That way, we successively improve the security of our stegosystem, cf. Figure 1. The price we have to pay is that the capacity of the stegochannel decreases.

2 ISDN – Integrated Services Digital Network

2.1 ISDN in General

Twin wires, already used in the plain old telephone network, form the physical channel of ISDN. By sophisticated signal processing, two 64 kbit/s-channels for data and one control channel of 16 kbit/s are available per direction. It is also possible to use as many as 30 data channels in this network, but for this four wires are necessary. A special connection device, the so-called NT-socket, is needed to be able to link up the various devices, e.g. the telephone, with the net.

For our application it is very important that in the ISDN data is transmitted digitally and without any changes.

2.2 Important for us: Digitization of Speech

Sound waves, like speech or music, can be transformed into analogous voltages by a microphone. An oscilloscope shows this voltages as overlaid sine curves. But it is not easy to see that there are really only sine curves, because there is a jumble of lines on the screen.

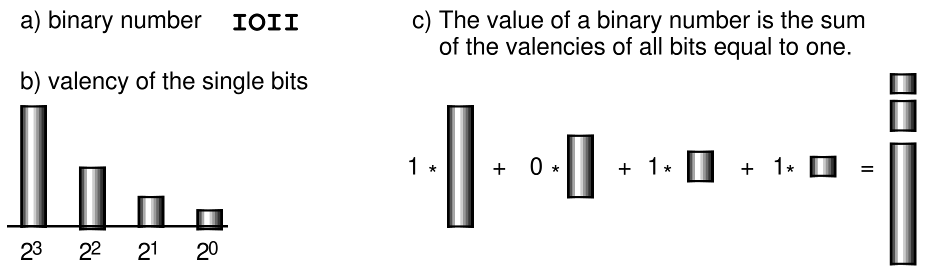


Figure 2: Transforming binary numbers into analogous values and conversely

These voltages can be digitized by a so-called analog/digital-converter. For that the A/D-converter measures the current amplitude (i.e. the value of the voltage) in short periods of time and transforms it into a binary number (Figure 2).

This "sampling" takes place several times per second, and thus nearly the original signal can be reconstructed from the established data (Figure 3).

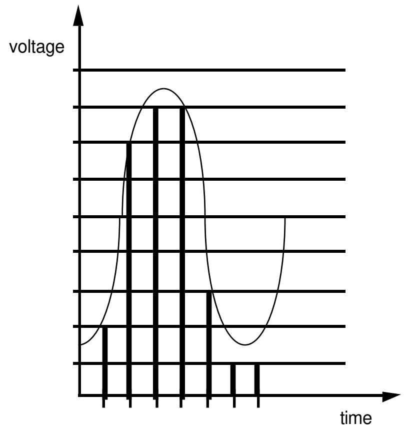


Figure 3: Encoding of noises as a succession of binary numbers

3 Hiding of Messages in Digitized Speech

Our goal is to transmit data simultaneously and unobservably in addition to the speech during a call. Therefore we want to "hide" data in the speech. We are allowed to do so since in every quantization there is some noise. Especially in phone calls additional noise is produced by cheap microphones and from the background. It is due to the noise that we can insert our data undetectably into the speech. Then they have themselves transmitted unnoticed like a fare dodger and being drowned out by the other noises.

If we speak about hiding we do not mean insertion of data into the speech but replacement of hissing noise. We replace the hissing noise "in the line" by our own one.

3.1 Our Idea of Hissing Noise

In this article we do not want to formalize the basic idea of noise but try to find out how background noises can be used in a pragmatic way.

The stegosystem must pick the hissing noise to pieces to decode the message. But for this it is necessary that the hissing noise is received exactly like it was sent. In analogous lines there is always a hissing noise caused by amplification of the signal. That is the reason why we cannot be sure that our requirement will be fulfilled. Our secret data would be falsified. Therefore we use the telephone service of the ISDN. There all data are transmitted digitally and without any modifications. Thus our noise can safely reach its destination.

Another very important source for noises is quantization. Each digitization is tainted with an error caused by the resolution of the A/D- and D/A-converters. These quantization errors are also important for the (im)possibility to detect our manipulations. We will later discuss this point.

3.2 Basic Ideas about Hiding Messages in Digitized Speech

It is very important to bear the fact in mind how sensitive ears and measuring instruments can be. If we really want to stay undetected, we must basically think about the points we have to respect by our "manipulations".

We want to change the speech as unobtrusive as possible. Surely, there are various approaches. Here we describe a first simple approach. If we set definite samples to an even or an odd value the recipient will get information of one bit with each of these values. But the recipient must know which values have been changed. There is no difference between the modified values and the others around them, because the hissing noise described above superimposes the speech.

Section 3.3 describes a method to apply this approach by modifying only the least significant bits.

It is wise to select the places for modification randomly and to insert a random bit stream. The latter would be automatically done by transmitting only compressed or encrypted data. These facts would complicate the proof of a hidden transmission. Besides this, the following is clear: The less bits we modify, the less modifications can be detected.

In addition to that, any number of conditions can be found which decide whether it is allowed to modify the value of the amplitude or not. We will also describe a modification which depends on the amplitude. This is an example for such conditions.

What happens if a 'censor' adds noise to the audio signal? If the artificial noise is small it should be sufficient to use error correcting codes. If the least significant bit becomes unusable due to "white noise" added by the government, it is still possible to use a slightly more significant bit. Therefore one has to adapt the bits prior and posterior to the modified position. But this is beyond our experiment. If that is also no longer possible one would start to use usual steganographic techniques like coding something in a melody that is played with a flute or on a keyboard - what can be automatically deciphered with a computer. As long as any information can pass from one end at a phone line to the other, it is possible to hide data during the connection. As a consequence noise would have to be added until the phone is no longer usable for normal speech, too. The moment the state adds "noise" to digital phone calls might be the moment one should consider to leave the country.

3.3 The Process

The principle of our simulation can be seen in Figure 4. The lowest bit of any byte of a file with digitized speech (original byte stream) can be set to the value of a bit stream of a second file (bit stream to be hidden). This second file represents the information which should be transmitted unnoticed. The distance between two encoded bits of the secret message is controlled by a random value, the maximum of which is a parameter for our process. The maximum is adjustable between 1 and 255. The distance is defined as the number of bytes between the two bytes which should be modified. The random values to control the distances are equally distributed between 0 and the maximum, they are a secret between the sender and the recipient. They correspond to a secret key in a symmetric cryptosystem and allow our stegosystem to be published in any detail and to be shared by anybody. The secrecy of the encoded message depends only on the secrecy of this key – this is Kerckhoff's principle applied to steganography.

There are two ways of generating the random values. They could be established from a file with random numbers which the recipient must also have. The more elegant way to produce these random numbers is the usage of a pseudo-random-generator. The latter has the advantage that only a "starting value" would have to be transmitted between sender and recipient. Such a starting value can be seen as a password which has to be fed in before using the real information.

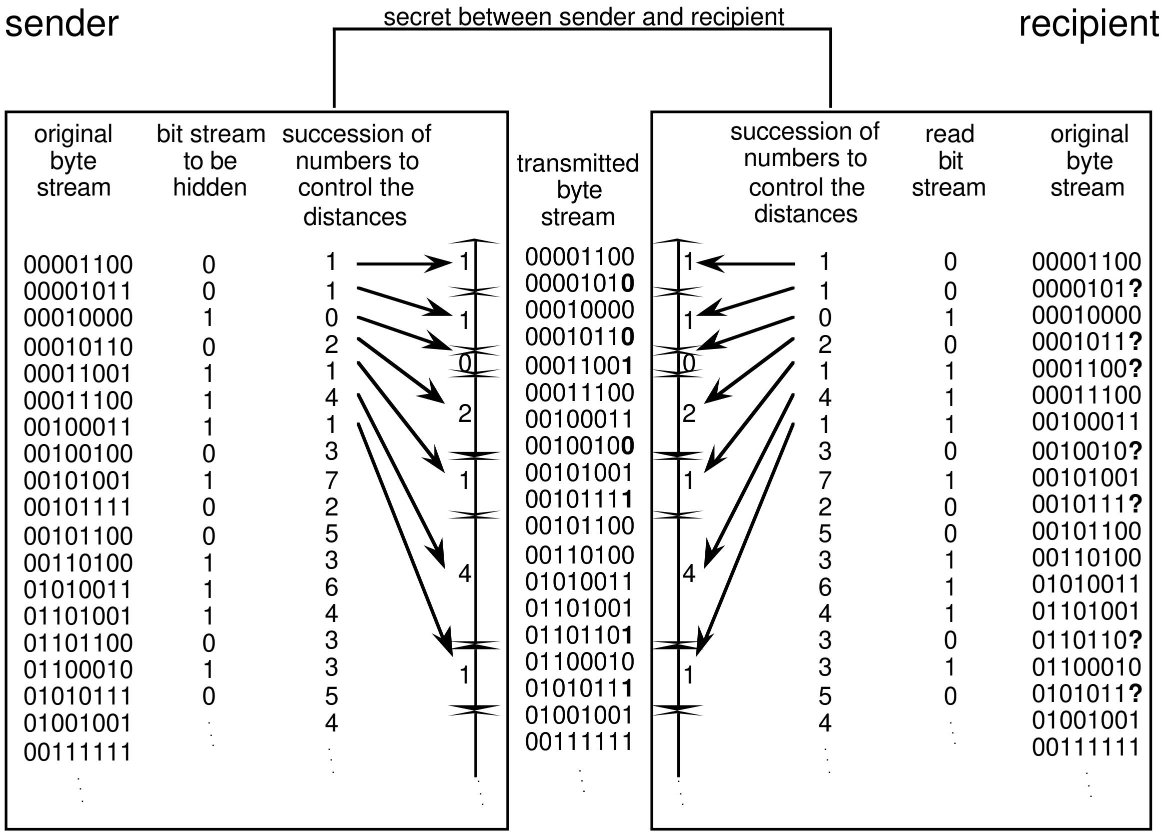


Figure 4: Random values, which are known to the sender and to the recipient, control the distances between the modified samples.

If you want to concentrate on a supposed sound, you will automatically lower your voice to better concentrate on it. You possibly ask bystanders to be quiet for a moment.

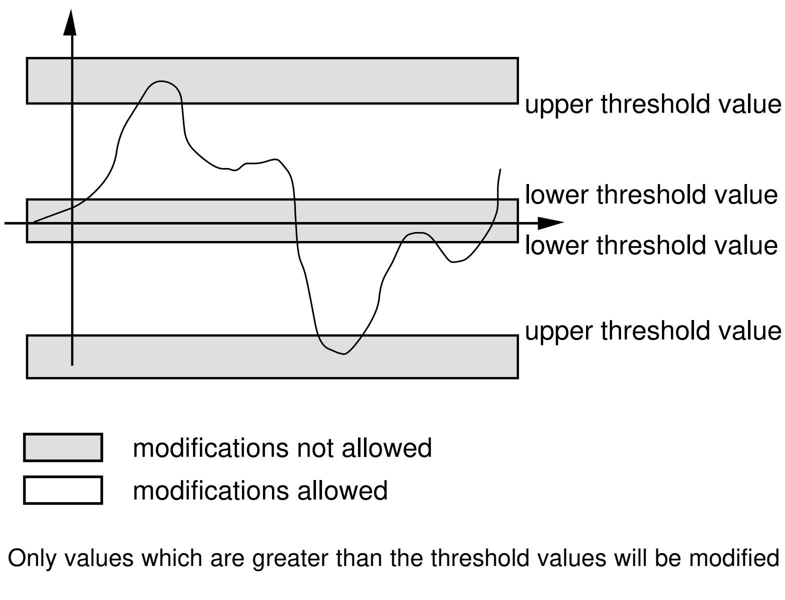


Figure 5: Only values which are greater than the threshold values will be modified a little bit.

So we came to the conclusion that it would be wise to transmit our data only when people were speaking. For this we implemented a method, in which a sample will only be modified when the current volume is greater or less than a threshold value, cf. Figure 5.

Even if the quantization is linear, this modification keeps the relative error small. This method does not exclude any of the others. They can be used in combination.

If the modification depends on the amplitude, it must be questioned if the recipient can safely decide whether the respective bit was set or not. Is it possible that the modified value changes the fulfillment of the conditions for modification and therefore the recipient must assume that this bit was not set? In order to prevent this situation we adhere to the following rule. If we want to set the lowest bit we must check whether the bits before fulfill our condition, i.e. our threshold values do not include the lowest bit. Generally, our threshold values should not include the bits we possibly want to change. Thus the recipient tests the bits which are not modified and finds the same conditions as the sender.

4 Fixing the Quality

We have implemented a small program for Apple Macintosh, called DigiStilz, to simulate such a covert transmission and to gain experience for the possibility to hear such a transmission.

4.1 Fundamental Questions about Measuring the Quality

To determine the quality of a machine, at first it is necessary to know its purpose and second how the fulfillment of this purpose can be measured. The purpose can be defined easily: We have a machine which should transmit additional data unnoticed by others. Measuring will be more difficult. How can it be measured whether transmitted speech was noticeably falsified?

4.2 Methods of Testing and Measuring

4.2.1 Listening Tests

The sense of hearing cannot be designated as objective, but our ears were the simplest and surely not the worst aid for testing our modifications. By listening tests we wanted to find out whether it was possible to hear our data. In case of hearing any change between original and modified version, we concluded that the modification selected was not suitable. An important criterion of quality was the "empty test". For this we used a sound file where "total silence" was recorded. Everything possible to hear must be caused by our modifications. We have also recorded talks in different natural environments. We wanted to test the influences of background noises on the sensitivity for our interferences.

For evaluation of the modified recordings you should remember that we always compared with the original to judge the hiding. An eavesdropper does not have this possibility. That is why we can assume that our results are on the safe side.

4.2.2 Sonogram

The shares of frequency in our modified sound files were analyzed by the program "SoundEdit": A sonogram filters out several shares of frequency and shows the intensity of the shares in relation to time. If a single frequency, e.g. 1000 Hz, is measured by the sonogram, a clear line will be seen at the 1000 Hz mark. Thus we could find out whether and how strong our manipulations changed the characteristic of our noises. This was a good measuring tool.

During transmitting ASCII-text with the constant distance 0 you could see a constant line in the sonogram for the also audible "data noise". It is interesting that this line was not yet recognizable by a variable distance of one at maximum. As expected, the sonogram always yields most minimal differences. But not any recognizable characteristics are provided which would be typical of hidden sounds. In the evaluation we will not further deal with the sonogram.

4.2.3 Distributions

Of course, only real existing things can be measured. Thus, the number of bits modified on average by the respective method should be used as a standard for quality. Therefore we assumed the following: If we set a bit we really change one. Let us suppose that we use the method with random distances. There is a file which controls the distances between the bits to be modified. If the bits of the file are equally distributed, then we can assume that the distance is also equally distributed. By a maximum distance of d_{max}, we receive on average a distance of $d_{max}/2$. E.g. we have selected a maximum distance of 4 bits. On average we then modify with a distance of 2, by a maximum distance of 32 bit with a distance of 16, etc. It can be seen that very small probabilities of modifications are achievable by expanding the maximum distance.

However, you cannot say for sure whether our modifications are detectable or not. If the bits of a text are not equally distributed and random, the text will be more observable. To draw conclusions from that would mean to ignore the features of the original data, i.e. the data, in which our secret data shall be encoded. Making specific statements which are based on distributions requires knowledge of the distribution of the digitized noises. Noise is usually equally distributed. Nevertheless, we are forced to allow correlations between the bits of the original data. But we still believe that the unobservability of inserting our data will be further maintained. The reasons will be explained in the following section.

Such assumptions are meaningful for the hiding algorithm. Surely, also other and possibly more skillful methods for hiding data can be used. E.g. several values of a sample could be changed. In order to prevent the noticeability of this change the environment will be adapted by modifying additional bits. That way, these vast modifications may be no more recognizable than single-bit modifications.

4.3 Results of the Methods

It is very difficult to make absolute statements about the quality. Especially it is difficult to speculate which measuring methods will be used or invented to discover steganographic modifications. However, in the next sections we will represent the results of applying our methods and critically analyze them.

4.3.1 Sonogram

Comparing the original and the modified sound file has shown which frequencies have changed their intensities and how strongly. But again, it is difficult to make absolute statements. When are these modifications audible? The answer of this question strongly depends on the respective listener.

4.3.2 Statistical Approach

A reasonable statistical approach would allow us to make precise statements about the quality of our modifications of noises. For this it must be questioned before how the original data are distributed.

Do chronological correlations between different samples exist, or even correlations between the separate bits of a single sample?

To answer these questions would mean to specify and to check all imaginable influences. You must consider that there are no simple, yet accurate models for all conceivable sounds. The possibilities for influences are nearly endless.

It is safe to say that the succession of numbers which comes into being during the digitization is not completely determined a priori. So there must be randomness in the distributions of the bits. That means again that it must be possible to modify randomly. Therefore we can assume that our modifications will not be detected if we will make them rarely enough and without destroying existing correlations.

A simple example for the existing nondeterminism are errors caused by quantization. In order to make this clear we want to give an example:

The signal, which should be digitized, amounts to 4,9 mV at a special moment. The linear 8 bit A/D-converter can digitize voltages in steps of 2 mV. (This is equivalent to a possible maximum voltage of about 0,5 V.) The result is 2, or 10 as a binary number.

If we change this value to 11 (binary), or 3 (decimal), a voltage of 6 mV will be produced by reconversion instead of 4 mV. However, the error will be only 0,2 mV greater than the error by reconverting the original established value of 2.

But what will happen, if new statistical methods will be developed which are able to track down our modifications? We can argue with the nondeterminism again (Figure 1). Each stegoanalytical method must work with this nondeterminism. That means it must include gaps. Thus each such stegoanalytical method must accept constellations of bits which cannot be identified as modified. On condition that this stegoanalytical method will be known, these gaps can be used in our algorithms to change these bits anyway.

Probably, we will never gain final certainty that our steganographic changes are undetectable. But using surely existing nondeterminism (which has to exist despite of possibly existing correlations) we may assume that after all our stegosystem can resist even great stegoanalytical expenditure.

4.3.3 About the Tests

At the beginning we recorded a number of "sounds" with our program for the listening tests. For that purpose we used a PowerBook to be able to record talks in various environments. In all, we so recorded 12 "sounds": in big and in small rooms, with much or less background noise, in a telephone box, at the street and in the countryside. We wanted to simulate various standard situations which can occur during a telephone call. For all tests apart from the last we used the same text as secret message protected by steganography. Thus we have not to expect falsifications of the results caused by different texts.

We started several series of tests:

1. In the first place we confined ourselves to the method with constant distances. We expanded the distances in each sound till it was not yet possible to hear any difference between the original and the sound with the inserted data. From this we obtained an overview about the influence of the environment. So we could see how the background noises themselves influence the quality of the methods. This phase serves as a basis for the following series of tests.
2. Here we used the method with the random distances. For this we started with the distance of which differences were not yet audible by constant distances. Then it was reduced till the steganographically modified sound began to noise. This series of tests should answer the question how much better the usage of random distances is in comparison to constant distances.
3. In the 3rd series of tests we inserted the text with constant distance 0 into a sound. For this we used various threshold values, see §4.3.3.3.
4. In this experiment we looked for an optimal method (on a very heuristic way). We modified the methods and parameters till especially good results were achieved. Good means that so much data as possible could have been hidden unnoticed.
5. Here we checked the influences of the distribution of the bits in the data to transmit on the quality of our modification. Like in the first series of tests, we expanded the constant distances till the modifications were not audible. However, we used only one "sound".

Of course, this is only a small selection of experiments, which are possible or meaningful. But we had to decide for some things, because growing of possibilities causes growing of the number of tests. All in all, we made over 200 listening tests.

In the following paragraphs we will describe the results and draw conclusions from them. The table gives an overview about the results.

4.3.3.1 Series of Tests 1

This test should give information on the influence of the environment noises. It was clear that the environment is surely important, but we were astonished how influential this factor is. Thus we could insert into the sound which was recorded directly at the street, 2750 bits per second (each 8th bit). In a quiet environment, it was only possible to insert about 170 bits per second (each 128th bit). This shows us that it will be useful to reflect on this topic in the future.

Description of the Sounds

(1) big room with only little background noises	(5) telephone box, very quiet
(2) ‚countryside‘ with much background noises	(6) telephone box, situated on a busy road
(3) ‚countryside‘ with less background noises	(7) very busy road (directly by the pavement)
(4) telephone box in a quiet area	(8) rather small, quiet room

Table for the 1st and the 2nd Series of Tests:

distance	sound 1		sound 2		sound 3		sound 4	
	constant	random	constant	random	constant	random	constant	random
0	++++	–	++++	–	++++	–	–	–
1	++++	–	++++	–	++++	–	++++	–
3	++++	++++	+++	–	+++	+++	++++	–
7	+++	+++	+	+++	++	++	++++	–
15	+++	+++	0	++	+	+	+	–
31	+++	++	–	0	0	0	+	+
63	+	0	–	–	–	–	0	0
127	0	–	–	–	–	–	–	–

distance	sound 5		sound 6		sound 7		sound 8	
	constant	random	constant	random	constant	random	constant	random
0	++++	–	++++	–	+++	–	++++	–
1	++++	–	++++	–	++	+++	+++	–
3	+++	–	+++	–	+	++	++	–
7	++	–	+++	–	0	+	+	++
15	+	+	++	–	–	0	0	0
31	+	+	+	+	–	–	–	–
63	0	0	0	0	–	–	–	–
127	–	–	–	–	–	–	–	–

++++ = well audible ++ = hardly audible 0 = not audible
+++ = audible + = at the border of audibility -- = not tested

Comparison of the 1st and the 2nd Series of Tests
1st series of tests: constant distances, no threshold values
2nd series of tests: random distances (the maximum distance is declared, thus nearly twice as much bits than by constant distances can be inserted on average), no threshold values

4.3.3.2 Series of Tests 2

We intentionally strived for a comparison of the two methods. It can clearly be seen that the method with random distances is substantially better than the method with constant distances. We could hide about twice as much of data by random distances in most of the sounds.

4.3.3.3 Series of Tests 3

We supposed that it would be wise to transmit hidden data only when the basic noise is loud. Our supposition was confirmed by this series of tests. We could affect the hiding by usage of an upper and a lower threshold value, which are adjustable between 0 and 128. A modification was only allowed when the amplitude was between these threshold values.

The relative errors of the modification are big by using an upper threshold value of 5 (and a lower one of 0), and this was really audible. On the other hand, modifications were not audible by a lower threshold value of 80 (and an upper one of 128).

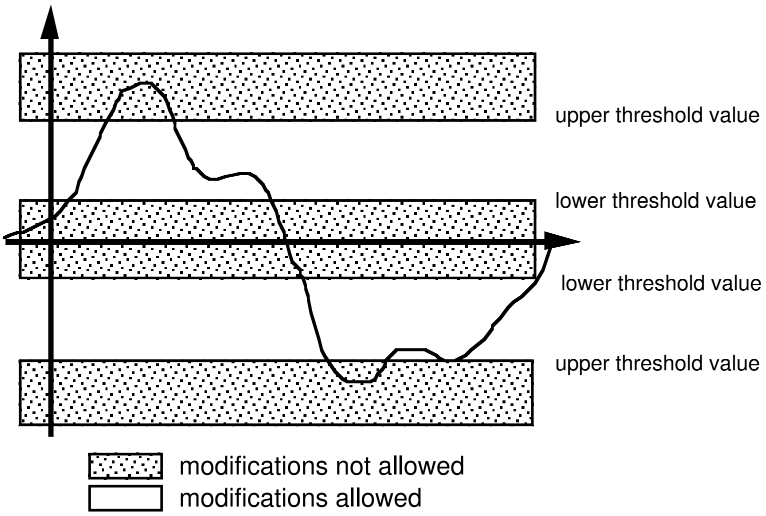


Figure 6: Ban on modifications depending on threshold values

4.3.3.4 Series of Tests 4

We selected the sound with the "worst" features as a reference, i.e. the sound which requires the greatest distances. We started with the random distance, on which the modifications began to be audible. Then we raised the distance till it was not yet possible to hear our modifications. The same was done with various smaller distances. In order to compare we established the quantity of data which could be inserted into the sound by different random distances. By random distance with a maximum of 1 and a lower threshold value of 60 more data could secretly be transmitted on average than by all other methods, on which there were no noise caused by modification.

Table for the 3rd Series of Tests:
sound (1), constant distance 0, variable threshold values (adjustable between 0 and 128)

distance	threshold value lower	threshold value upper	sound 1
0	0	1	+++
0	0	5	+++
0	0	10	+++
0	120	128	0
0	100	128	0
0	80	128	0
0	70	128	+
0	60	128	++
0	50	128	++
0	40	128	+++

++++ = well audible
+++ = audible
++ = hardly audible
+ = at the border of audibility
0 = not audible
-- = not tested

Table for the 4th Series of Tests:
sound (1), random distances, variable threshold values

distance (random)	threshold value lower	threshold value upper	sound 1
15	30	128	+++
15	40	128	0
7	40	128	+
3	40	128	+++
3	50	128	0
1	50	128	+
1	60	128	0
0	80	128	0

++++ = well audible
+++ = audible
++ = hardly audible
+ = at the border of audibility
0 = not audible
-- = not tested

This result has to be expected in so far as the quantity of data which can be transmitted is mostly determined by the distances. This method was developed on the sound with the worst features. As we expected, it could also be used for all other sounds.

4.3.3.5 Series of Tests 5

In the tests described above we used for secret transmission a text which consists of a random and equally distributed succession of letters, numbers and some standard special characters.

Table for 5th Series of Tests: sound (5), constant distance, several different texts

distance (constant)	sound 5			
	text with zeros	**text with zeros and ones**	**ASCII text**	**equally distributed text**
3	+++	–	–	–
7	+++	–	–	–
15	++	++	++	++
31	+	+	+	+
63	0	+ / 0	0	0

++++ = well audible
+++ = audible
++ = hardly audible
+ = at the border of audibility
0 = not audible
-- = not tested

In the last experiment we inserted a text with only zeros, an ASCII-text, a text with alternating zeros and ones and a text which consists of an equally distributed stream of zeros and ones. To our surprise the results did not show great differences. So it might be assumed that the distribution of the text is not so important. It may be that the values of the bits do not make any difference by greater distances.

4.3.3.6 Remarks

Our expectations were not in any case confirmed by our series of tests. Something was quite unexpected. Thus hissing noises which were caused by too low or too high threshold values are more similar to distortions than to noise. However, we tried to be as objective as possible. We mostly did "blind tests". For that, one had to guess which the original and which the stego-modified version is. Besides, we used headphones in order to achieve a good quality for playback. A lot of tests were repeated on the next day to reduce mistakes.

5 Outlooks

5.1 Can the Method be Used in Reality?

Our system would not be a relevant contribution to the existing discussion if there were no possibility for a real time application. Therefore we wanted to find out which possibilities are available to implement this system into the ISDN. The Macintosh Toolbox supports processing of bit streams very well. We could achieve a throughput of data of more than the required 8000 bytes per second by its help. To be less audible means less to modify. Thus the program can work faster by raised demands, even though that sounds paradoxical.

As mentioned in Section 3.3, a linear quantization was used for the simulation. The relative error was kept small by modifying only great amplitudes.

If logarithmic quantization is used the relative error is equal for all amplitude values. Therefore another approach is needed. One possible way is to modify when the amplitude changes. But this is beyond the scope of this paper.

Further work has also to be done e.g. in the area of statistical analyses.

5.2 Conclusion

Steganography is the science of hiding data. We gave a clear example how to hide data in telephone calls by using the ISDN. We do not show a solution working in real time, but we hope that it is clear that this would not be a serious problem. We think it would take a software house only a few months to realize such an application by using suitable hardware. Using such a secret-communication-by-stego-in-ISDN-application prevents any observation of communication in networks.

But there are also other possibilities for hiding data. There are the numerous programs for hiding data in pictures, cf. e.g. `http://www.site.gmu.edu/~njohnso5/stegdoc/stegdoc.html`. Peter Wayner introduced a method to construct a text from data. There is not any meaning in this text, but you will see this only at a closer look. However, all secret information is included.

In addition to hide the data by steganography, the messages which should be kept secret can be encoded by cryptography first. Even if the hidden data should be detected, they cannot be understood by persons who do not know the cryptographic key. Thus it is even more difficult to prove the existence of a secret hidden message. So we can see that it is not possible to succeed in getting a ban on steganography done in a state founded on the rule of law.

A law restricting the use of cryptography will cause criminals to use steganographical measures. Therefore it is dangerous and must not become effective.

Literature

Steffen Möller, Andreas Pfitzmann, Ingo Stierand: Rechnergestützte Steganographie: Wie sie funktioniert und warum folglich jede Reglementierung von Verschlüsselung unsinnig ist; Datenschutz und Datensicherung DuD 18/6 (1994) 318-326.

Peter Bocker: ISDN – The Integrated Services Digital Network; Concepts, Methods, Systems; In collaboration with G. Arndt, V. Frantzen, O. Fundneider, L. Hagenhaus, H. J. Rothamel, L. Schweizer; Springer-Verlag, Heidelberg 1988

Peter Wayner: Mimic Functions; Cryptologia XVI/3 1993, pp. 193-213

Newsgroup: sci.crypt.research

Romana Machado: Stego; http://www.fqa.com/romana/romanasoft/stego.html

Further information on steganography at www–serveers of the internet such as
http://www.iquest.net/~mrmil/stego.html and
http://www.cs.hut.fi/ssh/crypto/software.html#stego