Volume 4, Issue 7, July 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Image Steganography

Gurpreet Singh
HOD-Computer Science and Department,
Kurukshetra University/YIET,
Yamuna Nagar, India

Nivedita Sharma
Computer Science and Department
Kurukshetra University/YIET
Yamuna Nagar (9996088923), India

Abstract- Steganography is one of the methods of secret communication that hides the existence of hidden Message. It can be defined as the study of invisible communication that usually deals with the ways of hiding the existence of the communicated message. The hidden message may be text, image, audio, video, etc. The files can be a cover image after inserting the message into the cover image using stego-key. It is referred to as stego-image. Steganography is now more important due to the exponential growth and secret communication of potential computer users on the internet. In this paper I have analyzed various steganography techniques. It also given an overview of steganography, different methods of Steganography, its applications, how it is different from cryptography.

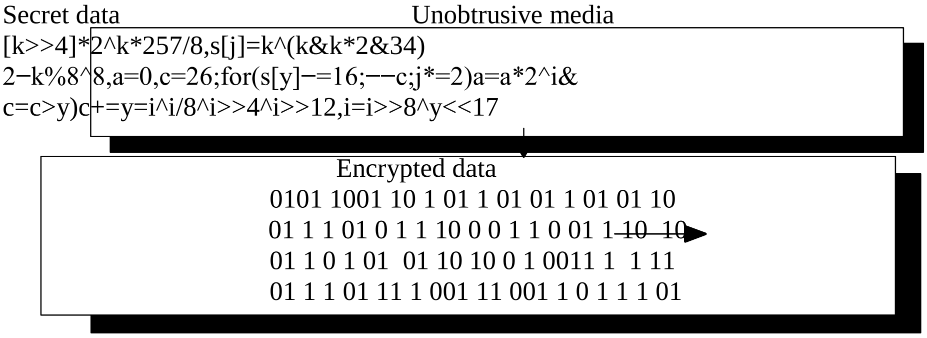
The image cryptography and steganography performed infrequency domain using random phase mask encoding are presented. These of random phase maskallowsto Dec correlate initial image and makes it unrecognized. This property is used for proposed image encryption and for steganography to increase the security level of the encoded image and to make it less visible. Finally, two keys are needed decrypt the image. The efficiency of the proposed approach is Demonstrated by the computer modeling.

Keywords – Steganography, LSB, spatial, frequency, masking, filtering, distortion.

I. INTRODUCTION

In this paper, we will introduce what steganography is and what kind of applications can be expected. Steganography is an art and science of hiding information within other information. The word itself comes from Greek and means “hidden writing”. First complex book covering steganography was written by Johannes Trithemius in 1499. The book *Steganographia* itself was published later in 1606 and immediately placed on the *Index Librorum Prohibit rum*.

In recent years cryptography become very popular science. As steganography has very close to cryptography and its applications, we can with advantage highlight the main differences. Cryptography is about concealing the content of the message. At the same time encrypted data package is itself evidence of the existence of valuable information. Steganography goes a step further and makes the cipher text invisible to unauthorized users. Hereby we can define steganography as cryptography with the additional property that its output looks unobtrusively.


Secret data Unobtrusive media

[k>>4]*2^k*257/8,s[j]=k^(k&k*2&34)
2−k%8^8,a=0,c=26;for(s[y]−=16;−−c;j*=2)a=a*2^i&
c=c>y)c+=y=i^i/8^i>>4^i>>12,i=i>>8^y<<17

Encrypted data
0101 1001 10 1 01 1 01 01 1 01 01 10
01 1 1 01 0 1 1 10 0 0 1 1 0 01 1 10 10
01 1 0 1 01 01 10 10 0 1 0011 1 1 11
01 1 1 01 11 1 001 11 001 1 0 1 1 1 01

One can ask what is it good for. Well, image the common situation when you encrypt your important business data. Suddenly robbers capture and torture you into revealing crypto graphics keys. As well police power may be abused. They ask you to give them the private keys or you are highly suspicious of committing crime. Next, what if the police is bribed. Would not it be better, if you can plausibly deny the existence of important data?

A famous example of steganography is Simmons’ ”Prisoners’ problem”, see [1]. Bob and Alice are in a jail and wish to escape. Their cells are far apart from each other and the only allowed communication is sending messages via prison officer. If warden detects any sign of conspiracy, he will secure their cells even more. Bob and Alice are well aware of these facts.

Happily, before they got arrested, they have agreed a stegosystem. Stegosystem describes the way the secret message is embedded into a cover text (seemingly innocent message). According to the standard terminology of information hiding a cover text with hidden information is called stegotext.

Examples of historical stegosystem can be secret inks, wax tablets or microdots used during WWII. In modern era these methods can invoke smile on face, but image their power, when they were not widely known. Nowadays methods still hold the same simplicity; just exploit advantages of digital media and communications.

We can distinguish between stegosystem with passive warder and active warder. Passive warder just monitors the communication channel. He can pass the cover texts through several statistical tests, but do not modify them. It is the same situation as when the network packets go through Intrusion Detect System. Applications from this field are often referred to as traffic security.

On the other side, active warden manipulates cover texts in order to preclude the possibility of hidden communication. Bob and Alice have to use very sophisticated embedding algorithm. Hidden information must withstand various recoding of covering medium, the use of error correction codes is recommended. Typical real-life application is watermarking and fingerprinting.

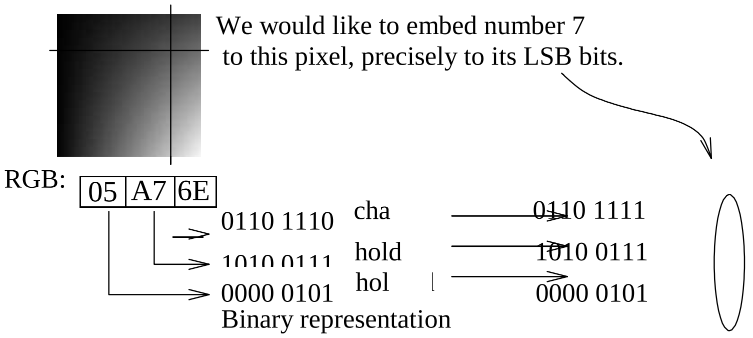
Watermark is a small piece of embedded information which can proof copyrighted material. Fingerprint is very similar, but is intended to track the concrete copy of copyrighted data.

II. EMBEDDING PROCESS

Data which hold effective information often has some redundancy. End users usually tend to think that redundancy is evil which cost extra money, as more disk space or network bandwidth is needed. Well, they are partially right, but optimal compression hardly ever exists. Moreover common compress ratio is mostly question of efficiency.

Anderson [2] pointed out one interesting conclusion in case we have optimal compression. Image an algorithm A, which can each wav file optimally compress. Algorithm B then does the decompression. Here, steganography is either impossible or trivial. We cannot add anything to the compressed file, otherwise the decompression will fail. On the other hand, we can take our secret information and algorithms B will trivial convert it to some fine familiar sound.

Now we know there are almost always few bytes; one can play with, without destroying carried information. Least Significant Byte (LSB) substitution is well known and widely used method. Take for example a True-Color BMP image file format. A color of pixel is coded in 3 byte array of indices to RGB palate. If you change only LSB bit in each color element, then the picture will seem still the same, but is not. It carries hidden information. A picture with size 120x100 pixels can hold approximately up to 4500B of hidden data, if this method is used.



To secure out our algorithm at least a bit, we can with advantage use some conventional pseudo-random number generator. Supplied pass- word will serve as initial seed. Generated numbers will specify which pixel to use for encoding next 3 bits of embedded data. The adversary, even with the complete knowledge of stegosystem, cannot extract the hidden message without the password. This system is secure in a sense of Keckhoffs' assumption, that everything is known except the password.

Similar procedures can be successfully applied to wide variety of multimedia formats. Only instead of the color indices we slightly modify the Discrete Cosine Transformation (DCT) or Fourier Transformation (FT) coefficients.

These changes have usually projection to specific statistical proper- ties of common multimedia data. For few formats these properties have been already discovered, for many not yet. The main problem is that these tests are still quite fragile. They tend to fail, if only a small amount of hidden information is embedded or algorithm is modified. Moreover, recent research of Neil's Provo's [3] has shown how to balance statistical properties after the embedding process.

Next to multimedia format we can even modify binary executable files without breaking their functionality. Hydan programme by Rakan El-Khalil [4] exploits redundancy in Intel x86 instruction set. Instruction *add* $*20,%eax* can be altered to *sub* $-*20,%eax* and vice versa. Then let the *add* means logical 1, *sub* means logical 0 and a solid base for stegosystem has born.

Possibilities how to add hidden information to existing data are al- most endless. Even in a very secure operating system, programs can still communicate by measuring CPU load or memory usage, and so on.

Information theoretic scientists try to seek for the limits of steganography, see [2, 5]. Currently we know two mathematical frameworks. The first one is informatictheoretic model by Cachin [6] and the latter is complex-theoretic view from von Ahn and Hopper [7].

III. NEW SCOPES

There are many real-life applications of steganography. Secure transition of sensitive data and watermarking are quite obvious. Let us take a look at something really new.

Steganography can be used to design a steganographic file system, where an adversary cannot deduce existence of any file. Surely, raw disk looks suspicious, but one can only guess if there are any files inside and what their names are.

There have been few proposals in recent years. Anderson in [8] introduced two ideas. User has to supply a file name and associated password to access the desired file in both schemes.

The first scheme initializes the file system with several randomly generated cover files. Newly created object is embedded as the exclusive-or of a subset of cover files. The subset is selected by the password and file name. The second construction fills the whole disk with random bits. Then the blocks of new objects are written to absolute disk addresses given by some pseudorandom process.

Other construction used HweeHwa Pang [9]. His scheme supports plain and hidden files at the same time. By hash value obtained from a file name and password a position of header of hidden file is located. The header contains a link to an inode table that indexes all the data blocks in the hidden file. Additionaly the header is encrypted. Then there are various blocks whose type and location confuse an adversary.

Next interesting application of steganography is developing a scheme, where the content is encrypted with one key and can be decrypted with several other keys. The relative entropy between encrypt and one specific decrypt key corresponds to the amount of information, which can be used to fingerprint the obtained data. Such fingerprinted data can be later easily tracked. This idea [10] together with partial extraction was presented at FEE CTU Poster 2004.

I V. CONCLUSION

This paper was a short introduction to the world of steganography. We have shown how the simplest methods work and how they can be explored. We have omitted public-key steganography as this subject requires advanced knowledge in mathematics, but research in this field has already begun. Relevant information can be found at [5, 7, 11].

Next to public-key steganography, one of the most active fields of research are mass detection tools for hidden contents. The problems are really big. At first, known statistical tests are fragile and for many embedding schemes we still do not know which properties to test. At second, the today traffic in public networks is so overwhelming, that is too hard to rigorously check each file.

REFERENCES

[1] G. J. Simmons. The prisoner's problem and the subliminal channel. In *Advances in Cryptology – CRYPTO '83"*, 1983.

[2] R. Anderson and F. Petitcolas. On the limits of stehanography. *IEEE Journal of Selected Areas in Communications*, 16(4):474–481, May 1998. Special issue on copyright & privacy protection.

[3] Niels Provos. Defending against statistical steganalysis. 2001.

[4] Rakan El-Khalil and Angelos D. Keromytis. Hydan: Hiding information in program binaries. Technical report, Department of Computer Science, Columbia University, 2004.

[5] Imre Csiszár. The method of types. *IEEE TIT: IEEE Transactions on Information Theory*, 44, 1998.

[6] Christian Cachin. An information-theoretic model for steganography. *Lecture Notes in Computer Science*, 1525:306–318, 1998.

[7] Luis von Ahn and Nicholas J. Hopper. Public-key steganography. In *Lecture Notes in Computer Science*, volume 3027 / 2004 of *Advances in Cryptology - EUROCRYPT 2004*, pages 323–341. Springer-Verlag Heidelberg, 2004.

[8] R. Anderson, R. Needham, and A. Shamir. The steganographic file system. In *IWIH: Inter- national Workshop on Information Hiding*, 1998.

[9] HweeHwa Pang, Kian-Lee Tan, and Xuan Zhou. Steganographic schemes for file system and b-tree. *IEEE Transactions on Knowledge and Data Engineering*, 16(6):701–713, June 2004.