Steganokryptografia typu „Grayscale Image”

Grayscale Image Steganocryptography

Czesław Kościelny

Wrocławska Wyższa Szkoła Informatyki Stosowanej
ul. Wejherowska 28, 54-239 Wrocław

Treść. Przedstawiono nową metodę steganokryptografii, polegającą na przekształceniu pliku dowolnego typu na plik graficzny „Grayscale Image” o formacie TIF lub BMP. Wiadomość, zawarta w przekształcanym pliku jest najpierw szyfrowana w taki sposób, że plik po zaszyfrowaniu zawiera wyłącznie znaki kodu ASCII o numerach od 0 do 31. Następnie tak utworzony kryptogram jest zamieniany na plik graficzny, zwany steganokryptogramem, którego treścią jest obraz szachownicy utrzymany w skali szarości. Efektywność metody polega na tym, że steganokryptogram ma tego samego rzędu rozmiar, co plik, którego treść jest ukrywana.

S**łowa kluczowe**: Steganografia, kryptografia, steganokryptografia, pliki graficzne w skali szarości

Abstract. In the paper, a steganocryptographic method converting an arbitrary format disk file into grayscale image of the TIF or BMP format has been presented. A message contained in the file to be converted is encrypted into cryptogram first, in which characters with numbers from 0 to 31are contained. Then, the obtained kryptogram is transformed into a grayscale image file, presenting a chessboard, is transformed. This is so-called steganocryptogram. The method is effective, which means that the size of a message, contained in the steganocryptogram is comparable with the size of the latter.

Keywords: Grayscale image, steganography, cryptography, steganocryptography

1. Wstęp

Znana od starożytności metoda ukrywania tajnych wiadomości, zwana steganografią, poczynając od roku 1995 zaczęła być szeroko stosowana w informatyce jako alternatywa dla kryptografii oraz jako narzędzie do zabezpieczania plików przed kopiowaniem. W zastosowaniu do plików komputerowych steganografia polegała na ukrywaniu pliku z tajną wiadomością w nieużywanych bitach pliku, zawierającym wiadomość nieistotną. Dlatego też tajna wiadomość miała rozmiar od kilku do kilkunastu procent rozmiaru pliku, w którym sekretną informację ukrywano. Stosunkowo niedawno pokazano [1, 2] w jaki sposób efektywnie ukrywać wiadomości, tzn. tak, aby rozmiar pliku z ukrytą wiadomością był porównywalny z rozmiarem pliku zawierającego wiadomość tajną. Wprowadzono też pojęcie steganokryptografii, polegającej na tworzeniu steganogramów szyfrowanych, czyli steganokryptogramów. W pracy opisano oryginalną metodę generowania steganokryptogramów i odzyskiwania zawartych w nich wiadomości.

2. System kryptograficzny

Aby wygenerować steganokryptogram należy najpierw zaszyfrować plik z tajną wiadomością. Zastosowano tu jeden wariantów metody [3, 4], stosującej nieliniowe przekształcenie w postaci zapisu liczb w systemach liczbowych o różnych podstawach. System kryptograficzny składa się z trzech algorytmów: generowanie tajnego klucza, szyfrowanie pliku i deszyfrowanie pliku. Oryginalność metody polega nie tylko na zastosowanym przekształceniu kryptograficznym, ale też na wykonywaniu operacji szyfrowania i deszyfrowania na całym pliku, bez potrzeby dzielenia pliku na bloki.

Algorytm generowania klucza:
Wejście: liczba *kl*, oznaczająca liczbę bitów, czyli długość klucza.
Krok 1. Wygenerować *kl* -bitową liczbę systemu dziesiątkowego z niezerowym bitem o wadze $2^{(kl-1)}$

Algorytm szyfrowania:
Wejście: nazwa pliku *fn,* klucz szyfrujący *key.*
Krok 1. Przeczytać bajty pliku *fn* do listy *fp*, otrzymując $fp=[b_1,b_2,...,b_{fs}]$
Krok 2. Wyznaczyć liczbę
$N = b_1+b_2256+b_3256^2+,...,+b_{(fs-1)}256^{(fs-2)}+bfs256^{(fs-1)}+256^{fs}key$
Krok 3. Dokonać konwersji liczby N na liczbę systemu liczbowego o podstawie 32, czyli obliczyć *fc* oraz współczynniki $c_1,i=1,2,...,fc$ według wzoru
$N=c1+c232+c3322+,...,+cfc32(fc-1).$
Krok 4. Wpisać listę bajtów $[c_1,c_2,...cf_c]$ do pliku *fn.*

Algorytm deszyfrowania:
Wejście: nazwa pliku *fn*, klucz szyfrujący *key.*
Krok 1. Przeczytać bajty pliku *fn* do listy *fc*, otrzymując $fc=[c_1,c_2,...cf_c]$
Krok 2. Wyznaczyć liczbę
$N=c_1+c_232+c_332^2+,...,+c_{fc}32^{(fc-1)}+key$

Krok 3. Dokonać konwersji liczby *N* na liczbę systemu liczbowego o podstawie 256, czyli obliczyć *fs* oraz współczynniki *bi,i*=1,2,...,*fs* według wzoru
$N=b_1+b_2256+b_3256^2+,...,+b_3^{(fs-1)}256^{(fs-2)}+b_{fs}256^{(fs-1)}+256^{fs}$
Krok 4. Wpisać listę bajtów [*b1,b2,...,bfs*] do pliku *fn*.

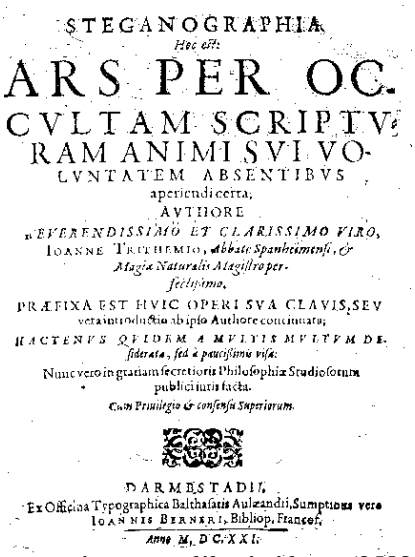
Prosty przykład:
Zakładając, że treścią pliku tekstowego jest napis TRITHEMIUS, a klucz jest w postaci 32-bitowej liczby = 2882724882, to algorytm szyfrowania będzie miał przebieg:
Krok 1. fp = [84, 82, 73, 84, 72, 69, 77, 73, 85, 83],
Krok 2. N = 1602455492893187086118466,
Krok 3. lista współczynników [2, 18, 27, 12, 7, 20, 30, 8, 5, 10, 19, 18, 20, 10, 13, 10, 1],
Krok 4. lista z poprzedniego kroku zostanie wpisana do pliku kryptogramu. Oznacza to, że steganokryptogramem 10-znakowego napisu TRITHEMIUS będzie 17 znaków
niedrukowalnych o numerach, pokazanych w liście.

Podczas deszyfrowania otrzyma się:
Krok 1. fc = [2, 18, 27, 12, 7, 20, 30, 8, 5, 10, 19, 18, 20, 10, 13, 10, 1],
Krok2. N = 1602455492893189968843348,
Krok 3. Otrzymana w tym kroku lista bajtów, [84, 82, 73, 84, 72, 69, 77, 73, 85, 83, 1] ,
bez ostatniego bajtu jest taka sama jak w 1. kroku procedury szyfrowania.
Krok 4. Lista z kroku 3., bez ostatniego bajtu, zostaje wpisana do pliku wyjściowego.

3. Przekształcanie kryptogramu w steganokryptogram

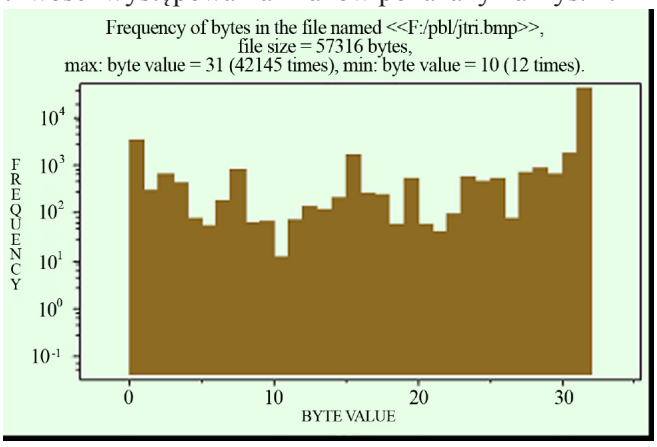
Ten etap przetwarzania plików polega na „narysowaniu" szachownicy przy pomocy bitów, zawartych w kryptogramie pliku, stanowiącym wiadomość tajną. Wymaga to dobrej znajomości pakietów bibliotecznych używanego środowiska programistycznego, dotyczących przetwarzania obrazów. Dzięki metodzie szyfrowania, generującej kryptogramy składające się wyłącznie ze znaków o numerach od 0 do 31, steganokryptogramy plików o różnych formatach są do siebie bardzo podobne i wyraziste.

Opis przykładu generowania steganokryptogramu i odzyskiwania ukrytej wiadomości
W tym przykładzie wiadomością do ukrycia jest strona tytułowa XV -wiecznego dzieła o steganografii i ta wiadomość jest zapisana w pliku jtri.bmp o objętości 35822 bajty (Rys. 1).


STEGANOGRAPHIA
Hoc eſt:
ARS PER OC.
CVLTAM SCRIPTV
RAM ANIMI SVI VO-
LVNTATEM ABSENTIBVS
aperiendi certa;
AVTHORE
R. EVERENDISSIMO ET CLARISSIMO VIRO,
IOANNE TRITHEMIO, Abbate Spanheimenſi, &
Magiæ Naturalis Magiſtro per-
ſpicuiſſimo.
PRÆFIXA EST HVIC OPERI SVA CLAVIS, SEV
vera introductio ab ipſo Authore concinnata;
HACTENVS QVIDEM A MVLTIS MVLTVM DE-
ſiderata, ſed a pauciſſimis viſa:
Nunc vero in gratiam ſecretioris Philoſophiæ Studioſorum
publici iuris facta.
Cum Priuilegio & conſenſu Superiorum.
DARMESTADII,
Ex Officina Typographica Balthaſaris Aulæandri, Sumptibus vero
IOANNIS BERNERI, Bibliop. Francof.
Anno M. DC. XXI.

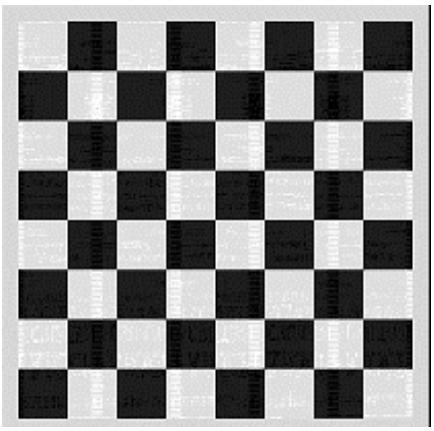
Rys. 1. Obraz zapisany w pliku jtri.bmp (35822 bajty)
Fig. 1. The image contained in the file jtri.bmp (35822 bytes)

Jak w każdym pliku graficznym, także i w tym pliku występują prawie wszystkie znaki kodu ASCII. Podstawową charakterystykę statyczną pliku z Rys.1 pokazano na Rys.2. W procedurze szyfrowania zastosowano 1025 bitowy klucz
key = 258336886841438853633549038007191828532868056745471672735362998052075538656810006153745439090605990420768146034351950576375613004215612669834810959945854795015629329485604310236232183115836846059598901589193318530700923353688489418223071772791792353624045605496946700965633275730897048879057257788360000545045,
i w efekcie otrzymano kryptogram pliku z Rys. 1. o objętości 57 316 bajtów, posiadający histogram częstotliwości występowania znaków pokazany na Rys. 2.



Rys. 2. Częstotliwość występowania znaków kodu ASCII w zaszyfrowanym pliku z Rys. 1.
Fig. 2. The frequency of bytes of the encrypted file from Fig. 1.

Ostatecznie wygenerowany steganokryptogram jest plikiem graficznym TIF, ma rozmiar 16 384 bajty i zawiera obraz, pokazany na Rys. 3.



Rys. 3. Obraz steganokryptogramu pliku z Rys. 1.
Fig. 3. The image of the steganocryptogram of the file shown in Fig. 1.

Podobne eksperymenty przeprowadzono z plikami o formatach TXT, JPG, BPP, TIFF, GIF, MID, WAV, RTF, AVI, uzyskując steganokryptogramy bardzo podobne do obrazu, pokazanego na Rys. 3.
Przykład zrealizowano za pomocą dość skomplikowanego programu, którego opis będzie przedmiotem oddzielnej publikacji.

4. Podsumowanie i wnioski

Przedstawiono prosty matematycznie sposób wyjątkowo skutecznego zabezpieczania plików dyskowych przed nieupoważnionym dostępem. Chociaż algorytmy kryptograficzne są bardzo proste, to wymagają wykonywania operacji na ogromnych liczbach całkowitych. Jeśli np. generuje się steganokryptogram pliku 100 kilobajtowego, to w procesie szyfrowania zawartość tego pliku jest zamieniana na liczbę liczącą ponad 240 824 cyfry. Dlatego też, chociaż czasy realizacji procedur kryptograficznych nie zależą od klucza, który może mieć praktycznie dowolną długość, to dla dużych plików trwają długo. Zaproponowana metoda nadaje się więc raczej do plików o rozmiarach nie przekraczających stu kilobajtów. Jeśli zaś chodzi o obraz steganokryptogramu, to „narysowanie" takiego obrazu zależy wyłącznie od umiejętności programisty i tutaj autor nie wykazał się nadmiarem pomysłowości. W każdym razie steganokryptogramy nie są podobne do typowych kryptogramów, mogą przypominać obiekty deterministyczne, a ponieważ są regularnymi plikami graficznymi, to bez znajomości niniejszej pracy nie ma możliwości wykrycia zawartych w nich tajnych wiadomości.

Według wiedzy autora, w dostępnej literaturze nie można znaleźć publikacji innych autorów, dotyczących steganokryptografii i algorytmów kryptograficznych z przekształceniem zapisu wiadomości do zaszyfrowania za pomocą konwersji i systemów liczbowych.

Literatura (References)

[1] Cz. Kościelny, *Steganocryptography with Maple* 8. 2003. http://www.maplesoft.com/applications/view.aspx?SID=4348
[2] Cz. Kościelny, *The MLA Steganography. 2006.* http://www.maplesoft.com/applications/view.aspx?SID=1707
[3] Cz. Kościelny, *A Symmetric-Key Block Cipher Generating Cryptograms Containing Characters Belonging to the Definite Set*. 2008. http://www.maplesoft.com/applications/view.aspx?SID=5646
[4] Cz. Kościelny, *Grayscale Image Steganography.* 2008. http://www.maplesoft.com/applications/view.aspx?SID=6878