Ascunderea Mesajelor in Fotografii - Criptare ElGamal

Andreea Musat, Victor Armegioiu

Scopul acestei prezentari este de a ilustra metoda de ascundere a mesajelor criptate in pixelii unei fotografii. De asemenea, urmarim sa descriem si procesele de criptare/decriptare implementate dupa schema ElGamal.

Introducerea Metodelor

Criptografia sau criptologia reprezinta practica si studiul tehnicilor de comunicare sigura in prezenta unui tert care ar putea intercepta mesajele, mai precis dezvoltarea unor metode care ar impiedica citirea mesajului de catre o persoana careia acesta nu ii este destinat. Steganografia reprezinta stiinta ascunderii anumitor date in cadrul altor date, cum ar fi ascunderea mesajelor in imagini sau chiar intr-un semnal audio. Inca din cele mai vechi timpuri, oamenii au incercat sa transmita informatii secrete unor anumiti destinatari, insa incercarea de a le trimite in text clar nu ar fi fost deloc sigura, asa ca au dezvoltat alte metode. In anii $500 - 600BC$, cel mai probabil, s-a inceput folosirea cifrurilor monoalfabetice cu substitutie de catre invatatii evrei, de catre indieni si de catre egipteni. De asemenea, Herodot documenta povestea unui sclav care a avut un mesaj scris pe capul ras si care a fost trimis sa il livreze dupa ce parul i-a crescut. Daca in acele timpuri se poate observa necesitatea unei comunicari sigure, este clar ca in zilele noastre, in era internetului, nevoia aceasta este din ce in ce mai mare. De aceea, proiectul implementat este reprezentat de ascunderea (si mai apoi extragerea de catre

destinatar) a unor mesaje secrete in imagini. Intrucat simpla ascundere ar fi fost un singur strat de securitate, pentru a imbunatati strategia am ales ca mesajele sa fie si criptate folosind sistemul de criptare ElGamal. Imaginile digitale sunt reprezentate de matrice de pixeli. Unul dintre cele mai folosite modele de culoare in cadrul fotografiei digitale, al televizoarelor si calculatoarelor este RGB, in care fiecare pixel este reprezentat folosind $3 \cdot 8$ biti, cate 8 biti pentru fiecare culoare - Red, Green si Blue, asta insemnand ca fiecare culoare este obtinuta cu un triplet (r, g, b), unde r, g, b sunt intregi intre 0 si 255. Putem, deci, obtine 255^3 culori in total, adica 16777216, un numar mult mai mare fata de numarul total de culori pe care ochiul uman le poate distinge, care este estimat intre 100.000 si $10.000.000$ [1]. Datorita acestui fapt, s-a observat ca schimbarea celor mai putin semnificativi biti dintr-un anumit canal de culoare din cele 3 nu modifica felul in care percepem imaginea, permitand ascunderea altor informatii. Astfel, un mesaj se va ascunde intr-o imagine folosind o multime de pixeli (metodele folosite pentru selectarea acestui set de pixeli vor fi discutate ulterior) pentru care se vor modifica ultimii lsb biti din canalul albastru.

Pentru ascunderea mesajului, acesta este mai intai criptat, iar apoi codul obtinut este transformat in binar, urmand ca in final bitii acestia sa fie fixati, cate lsb pe rand, in pixelii selectati. Pixeli in care se ascunde textul sunt selectati random, alegandu-se mai intai linia, apoi coloana fiecaruia in mod arbitrar; un generator de numere random care are ca seed o cheie introdusa de utilizator face totusi ca acest set de pixeli sa fie determinist la fiecare rulare, facand posibila citirea mesajului doar in prezenta cheii respective. Cheia se presupune a fi apriori cunoscuta de transmitator si destinatar. Pentru a face mai greu de detectat prezenta unui mesaj in imagine, lungimea mesajului ascuns trebuie sa fie mai mica decat un anumit prag setat sau un anumit procent din dimensiunea totala a imaginii.

Rezultate

Ascunderea functioneaza la fel de bine in cazul cazul imaginilor cu zgomot, dar si imaginilor fara zgomot, dupa cum se poate observa:

Figure 1: Stegano pentru imagine fara zgomot - 5 biti

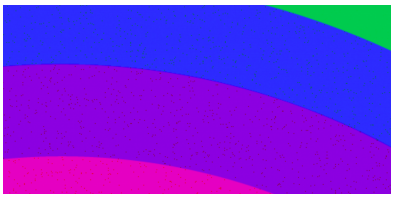

Figure 2: Stegano pe canalul albastru - 8 biti

Figure 3: Stegano - 2 biti

Algoritmul ElGamal

Schema de encriptie definita de Taher Elgamal[2] se bazeaza pe dificultatea rezolvarii problemei logaritmului discret. In acest sens, se definesc 3 etape relevante care descriu algoritmul folosit:

1. Generarea cheii

• se genereaza un grup ciclic G de ordin q cu generator g

• se alege un x (cheie privata) aleatoriu din multimea $\{1, ..., q-1\}$

• se calculeaza $h := g^x$

• h, impreuna cu o descriere a tuplului (G, q, g) definesc cheia publica

In general, grupul G este ales ca fiind $(Z/pZ)^\times$, unde p e un numar prim ales in prealabil. Astfel, se poate identifica usor un generator (radacina prima modulo p), iar ordinul grupului este trivial de gasit, deoarece $q := \varphi(p) = p - 1$.

2. Criptare

• se alege in mod aleatoriu y din $\{1, ..., q-1\}$ si se calculeaza $c_1 := g^y$

• se defineste secretul comun $s := h^y := g^{xy}$

• se mapeaza mesajul m pe un element m' din G

• se calculeaza $c_2 := m' \cdot s$

• se trimite perechea criptata $(c_1, c_2) = (g^y, m' \cdot h^y) = (g^y, m' \cdot g^{xy})$

3. Decriptare

• folosind perechea criptata primita (c_1, c_2) se calculeaza secretul comun $s := c_1^x$

• se obtine mesajul m' usor, intrucat $c_2 \cdot s^{-1} = m' \cdot h^y \cdot (g^{xy})^{-1} = m'$.

Detalii de Implementare

Am ales sa nu folosim biblioteci preexistente in scopul familiarizarii cu conceptele matematice din spatele algoritmului. Ca urmare, cateva din aspectele cu care ne-am confruntat sunt urmatoarele: generarea unui numar prim de n biti ($p \in [2^{n-1}, 2^n - 1]$), teste eficiente de primalitate, exponentiere modulara rapida ($\Theta(\log p)$)), gasirea unui generator (radacini prime in contextul dat) pentru grupul ales.

In ceea ce priveste gasirea unui numar prim de dimensiune data, am folosit testul nedeterminist Baillie-PSW[3]. Acest test este un hibrid intre testul Miller-Rabin si testul lui Lucas pentru pseudoprime. Pentru rapiditate, testul Miller-Rabin se efectueaza intr-o runda unica in etapa incipienta pentru a determina daca numarul este un pseudoprim fata de baza 2.
Daca testul initial este trecut cu succes, se continua prin a determina daca numarul este un pseudoprim Lucas. La nivel de rezultate, folosind prime de 1024 de biti, tot procesul (incluzand generare, criptare si decriptare pe un text de 500 de caractere) dureaza in jur de 5.2 minute.

Exponentierea rapida se face tinand cont de observatia ca orice numar intreg are o reprezentare binara unica. Ca urmare, date fiind numerele intregi (x, n) unde n se poate scrie ca $n = \sum_{k=0}^{\lfloor \log_2 n \rfloor} 2^k \cdot b_k$ iar b_k reprezinta valorile bitilor din reperezentarea lui n, obtinem:
$x^n = x^{\sum_{k=0}^{\lfloor \log_2 n \rfloor} 2^k \cdot b_k} = x^{b_0} \cdot x^{2 \cdot b_1} \cdot x^{2^2 \cdot b_2} \cdot \ldots (1)$.

Din (1) se observa usor ca x^n se poate calcula in complexitate logaritmica, prin ridicare succesiva la patrat a bazei initiale x.

Identificarea unui generator g pentru grupul $(Z/pZ)^\times$ nu este descrisa in punctul actual de un algoritm eficient. Aceasta implica gasirea tuturor factorilor primi ai ordinului grupului dati de $s := \varphi(p) = p - 1$, si testarea pe rand a valorilor m unde $m \in G$ pentru a gasi o valoare astfel incat $m^{\frac{\varphi(p)}{s_i}} \not\equiv 1 (\text{mod } p), (\forall) s_i | \varphi(p)$ [4].
In consecinta, pentru eficientizarea acestui proces, am ales sa cautam un numar prim de forma $p := 2q + 1$ unde q este de asemenea prim. Aceasta alegere determina existenta a doar 2 divizori pentru ordinul grupului dat $\varphi(p) = p - 1 = 2q$, acestia fiind 2, respectiv $\frac{p-1}{2}$.

Referinte

1. https://hypertextbook.com/facts/2006/JenniferLeong.shtml
2. https://en.wikipedia.org/wiki/ElGamal_encryption
3. http://mathworld.wolfram.com/Baillie-PSWPrimalityTest.html
4. https://math.stackexchange.com/q/133720