Codage et compression multimédia

HMIN 322

Rendu TP 2
Chiffrement multimédia

Élèves :
Yasmine Khodja

Enseignant :
Pauline Puteaux

1 Algorithme RSA

Implémentation de la méthode de chiffrement : Après avoir implémenter une fonction *bool premier (long n)* qui permet de savoir si un nombre est premier, une autre *int PGCD(int a, int b)* qui permet savoir si deux nombres sont premiers entre eux en calculant leur PGCD, une autre *vector<int> exposants(int p, int q)* qui permet d'obtenir tous les exposants de chiffrement e possibles et enfin la fonction *void chiffrementRSA(ImageBase imIn, int e, int p, int q, char nom[])* qui permet le chiffrement du système RSA. On obtient le résultat suivant :

FIGURE 1 – Image originale

FIGURE 2 – Image chiffrée

2 Implémentation de la méthode de déchiffrement

i. Le nom de l'algorithme généralement utilisé est l'algorithme d'Euclide étendu. Le résultat obtenu en calculant l'enverse modulaire de 17 et 220 (phi) est égal à 13.

ii. La fonction *void dechiffrement(ImageBase imIn, int e, int p, int q, char nom[])* nous permet de déchiffrer l'image précédemment chiffrée. Le résultat obtenue figure ci-dessous. La clé privée est égal à $d = 13$ qui représente l'inverse modulaire du couple de clés public *(17, 23)*.

FIGURE 3 – Image chiffrée

FIGURE 4 – Image déchiffrée

3 Analyse de sécurité

i. Après avoir calculé l'entropie des deux images de la question **(a)** on obtient :

— H(Image originale) = 6.67765bpp.

— H(Image chiffrée avec le RSA) = 6.67765bpp.

Les histogrammes des deux images figurant ci-dessous nous montrent clairement que les nuances de gris des pixels ont été uniformiser après RSA.

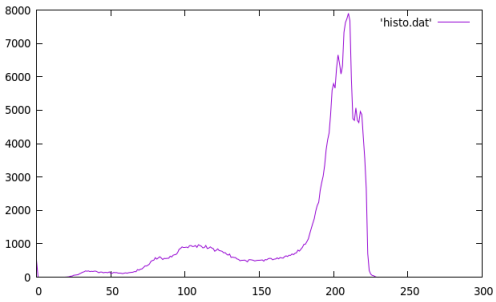

FIGURE 5 – Histogramme de l'image originale

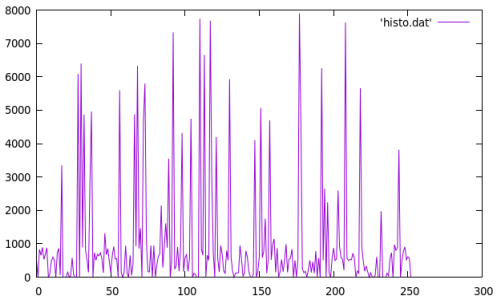

FIGURE 6 – Histogramme de l'image chiffrée

ii. Afin de binariser l'image, j'ai utilisé l'algorithme de seuil en seuillant l'image à 200. L'image obtenue figure ci-dessous.

FIGURE 7 – Image binarisée

Après chiffrement de l'image en utilisant l'algorithme RSA on obtient le résultat figurant dans la Figure 8. Ceci dit le résultat obtenu a permit d'afficher les objets de l'image dans un fond noir sans bruit.

FIGURE 8 – Image binarisée chiffrée

iii. L'algorithme ainsi implémentée offre un bon niveau de sécurité mais pas assez bon car il suffit de trouver la combinaison des deux nombres premiers et de tester avec tous les exposants en utilisant l'inverse modulaire et l'image sera forcément déchiffrée.

Une des solutions serait d'utiliser de choisir les clés publiques avec une autre relation qui ne serait pas symétrique.