Stéganographie

1. Introduction

La stéganographie est l'art de la dissimulation. Elle consiste à cacher un message au sein d'un autre message anodin, de sorte que l'on ignore l'existence même du secret.

Alors que la cryptographie repose sur le fait que le message ne soit pas compris, la stéganographie repose sur le fait que le message ne soit pas trouvé.

C'est un mot issu du grec ancien steganós (« étanche ») et graphế (« écriture »).

La stéganographie est exploitable dans de nombreux domaines. Elle trouve ainsi comme application commerciale le watermarking[1] (apposition de filigranes électroniques), technique permettant de « tatouer » un fichier électronique (pour y introduire notamment des informations utile s à la gestion des droits d'auteur).

Il ne faut pas confondre le watermarking, par essence invisible, avec le fait que certains formats de fichiers offrent la possibilité d'inclure des méta-informations...

Nous allons étudier une technique de stéganographie appliquée à l'image, c'est-à-dire voir comment il est possible de cacher un message dans une image.

2. Message transporté dans une image

2.1. La Méthode LSB

L'idée est de prendre un message et de le modifier de manière aussi discrète que possible afin d'y dissimuler l'information à transmettre. Le message original est le plus souvent une image. La technique de base --- dite LSB pour Least Significant Bit --- consiste à modifier le bit de poids faible des pixels codant l'image : une image numérique est une suite de points, que l'on appelle pixels, et dont on code la couleur à l'aide d'un triplet d'octets, par exemple pour une couleur RGB sur 24 bits.

Chaque octet indique l'intensité de la couleur correspondante --- rouge, vert ou bleu (Red Green Blue) --- par un niveau parmi 256. Passer d'un niveau n au niveau immédiatement supérieur (n+1) ou inférieur (n-1) ne modifie que peu la teinte du pixel, or c'est ce que l'on fait en modifiant le bit de poids faible de l'octet.

Exemple :

00000000 10101000 11100101	00000000 00100000 11001101
00100100 00011111 00101000	11111111 11100000 10110010

Chaque entrée de ce tableau représente un pixel couleur, nous avons donc une toute petite image 2×2. Chaque triplet de bits (0 ou 1) code la quantité de l'une des trois couleurs primaires du pixel. Le bit le plus à droite de chaque triplet est le fameux bit de poids faible --- LSB. Si on souhaite cacher le message **111 111 001 111**, l'image est modifiée de la façon suivante : le bit de poids faible du $i^{ème}$ octet est mis à la valeur du $i^{ème}$ bit du message ; ici on obtient :

1 tatouage numérique

0000000**1** 1010100**1** 1110010**1**	0000000**1** 0010000**1** 1100110**1**
0010010**0** 0001111**0** 0010100**1**	1111111**1** 1110000**1** 1011001**1**

D'autres techniques similaires sont possibles. Par exemple, l'encodage du message peut être basé sur l e mode de colorisation TSL (Teinte Saturation Luminance) plutôt que RGB (Red Green Blue / Rouge Vert Bleu). Mais toutes ces techniques ont l'inconvénient d'entraîner une déformation - voire une perte - des informations de l'image et sont facilement détectables soit par comparaison avec l'image originelle, soit par analyse linéaire simple (de la parité par exemple !).

Ces techniques de stéganographie très basiques s'appliquent tout particulièrement au format d'image BMP[2], format sans compression destructive, avec codage des pixel s entrelacé sur 3 octets comme énoncé ci-dessus .

Réciproquement, tout procédé de compression-décompression d'images avec pertes ou de redimensionnement de l'image est susceptible de détruire un message stéganographique codé de ces façons. On parle alors de stérilisation.

2.2. Manipulation de la palette de couleurs

Certains formats graphiques tel que GIF ou PNG permettent le stockage des couleurs de l'image par référence à une palette de couleurs insérée dans le même fichier.

Ainsi, au lieu de stocker bleu, blanc, rouge dans une image du drapeau français, on trouve dans un format de fichier la description de l'objet la suite couleur1, couleur2, couleur3 ainsi qu'une palette qui définit que couleur1 est le bleu, couleur2 le blanc et couleur3 le rouge.

La même image peut-être stockée de la façon suivante : couleur2, couleur3, couleur1 avec une palette qui définit que couleur2 est le bleu, couleur3 est le blanc et couleur1 est le rouge.

Ces deux images sont visuellement identiques, mais le stockage de celles-ci est différent. Pour une image contenant 256 couleurs uniques dans sa palette, on a factorielle 256 (256!) façons de stocker cette image. En utilisant un code connu entre l'émetteur et le récepteur de l'image, on peut donc communiquer un message de petite taille ($\log_2(256!)$, un peu moins de 1 684 bits) caché dans la permutation des couleurs de la palette de l'image.

2.3. Message caché dans les choix de compression

Tout semble indiquer que l'on ne peut cacher un mes sage dans un format d'image utilisant une compression avec perte. En réalité la plupart des programmes de stéganographie sérieux s'attaquent justement au format JPEG[3] qui utilise ce type de compression.

L'idée n'est pas de cacher une information dans les couleurs ou dans la palette (puisqu'il n'y en a pas) mais dans les choix de compression. En effet, tout algorithme de compression nécessite une succession de choix.

Avec des algorithmes de compression tel que Zip ou Gzip[4], on peut choisir la puissance de compression. En consommant plus de temps calcul et/ou plus de mémoire pour les opérations intermédiaires, on peut obtenir de meilleurs résultats de compression. Ainsi deux fichiers compressés de tailles différentes peuvent être décompressés en deux fichiers identiques.

La compression dans le format JPEG est double. La première compression consiste à découper l'image en blocs de 8 fois 8 pixels et de transformer ces carrés sous une forme mathématique simplifiée. Cette compression introduit des pertes et la version mathématique peut être légèrement

2 Bitmap
3 Joint Photographic Experts Group
4 GNU zip

différente du carré original tout en étant visuellement très semblable. Une fois tous les blocs compressés, il faut coder les formes mathématiques en consommant le moins possible d'espace. Cette deuxième compression n'introduit pas de perte et elle est similaire dans les principes à ce que l'on peut retrouver dans Zip ou Gzip. C'est en introduisant dans cette phase des bits d'informations que l'on arrive à transporter un message caché.

2.4. Message transporté dans un son

Dans les formats sonores, il existe à peu près les mêmes possibilités de cacher des messages que dans les images :

- Dans un fichier sonore au format MIDI[5], il n'existe pas de palette de couleurs mais bien différentes pistes qui peuvent être permutées.
- Dans un fichier sonore avec compression sans perte, on peut cacher de l'information dans des variations imperceptibles du son, les bits faiblement significatifs.
- Dans un fichier sonore avec compression avec perte, on peut cacher de l'information dans les choix de compression.

5 Musical Instrument Digital Interface