

1 Problématique

La stéganographie est l'art de la dissimulation. Le principe est de cacher un message dans un autre message. Il est ainsi possible de cacher une image dans une autre image.

Comment les informations d'une image sont-elles codées dans la mémoire de l'ordinateur ?

2 Notation binaire

À retenir

Un **bit** est la plus petite unité de stockage. Il ne peut contenir qu'une valeur parmi 0 ou 1.

Pour stocker une couleur on a besoin de huit bits (**1 octet**).

couleurs	bits
0	00000000
1	00000001
2	00000010
...	...
239	11101111
240	11110000
...	...
254	11111110
255	11111111

On peut différencier les huit bits en deux catégories :

$\underbrace{1\ 1\ 1\ 1}_{\text{bit de poids fort}}\ \underbrace{1\ 1\ 1\ 1}_{\text{bit de poids faible}}$

Activité 1 :

1. Se rendre sur le site <https://htmlcolorcodes.com/fr/> .
2. Choisir la couleur rouge (R = 255, G = 0, B = 0).
3. La comparer avec le rouge (R = 240, G = 0, B = 0).
4. Que peut-on dire des rouges compris entre 240 et 255 ?
5. Comparer l'écriture binaire de ces couleurs.

3 Principe de la stéganographie

En ne gardant que les bits de poids fort pour chaque couleur, l'image n'est que très peu dégradée. Le principe de la dissimulation est de placer les bits de poids fort de l'image secrète à la place des bits de poids faible de l'image visible.

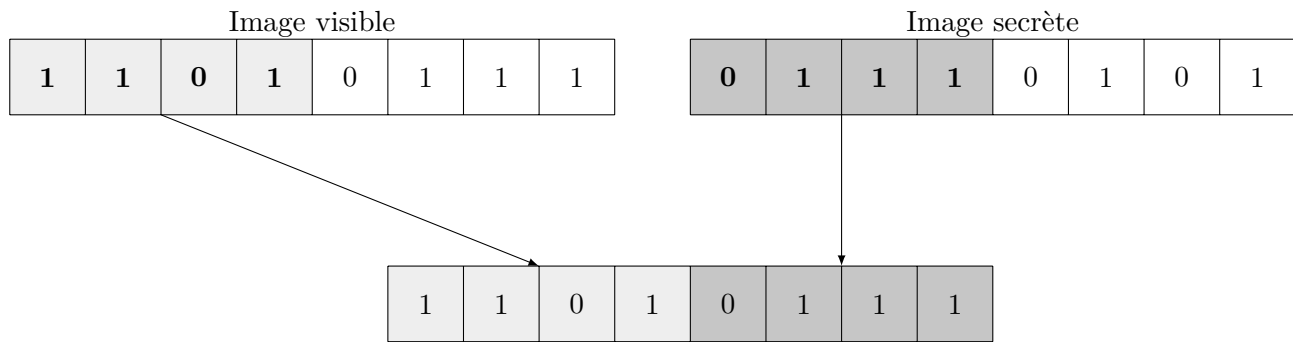


FIGURE 1 – Cacher une image dans une autre.

Activité 2 :

1. Détailler les étapes du protocole pour cacher une image secrète dans une image visible.
2. Télécharger le fichier *steganographie.zip* sur le site <https://cviroulaud.github.io> et extraire les dossiers.
3. Ouvrir le logiciel *Spyder*.
4. Dans le dossier *cache* ouvrir le programme *cache-image.py*.
5. Observer le code. Quelles lignes permettent de sélectionner les bits de poids fort de l'image secrète ?
6. Compléter les lignes 11 et 12 pour cacher l'image de l'éléphant dans l'image du magasin de porcelaine.
7. Observer l'image obtenue. L'éléphant est-il correctement caché ?

4 Retrouver une image cachée

Un fabricant de smartphone vous a envoyé la photographie du prototype de son nouvel appareil. Afin de rester discret l'image est cachée dans une autre.

Activité 3 :

1. Dans le dossier *retrouver* ouvrir l'image mystère. Peut-on distinguer l'image cachée ?
2. Avec *Spyder* ouvrir le programme *retrouver-image.py*.
3. Compléter la ligne 11.
4. Quel est le rôle des lignes 20 à 22 ?
5. Exécuter le programme pour retrouver l'image cachée.