Introduction

Qu'est-ce que la stéganographie ?
La stéganographie est le fait de cacher une information dans une autre anodine. Par exemple, une phrase peut être camouflée dans un texte. Plusieurs solutions existent pour qu'au final, la phrase soit cachée, en prenant le premier mot de chaque phrase d'un texte par exemple. Parfois, une image est le moyen de camouflage idéal. Grâce à d'infimes variations, un message est facilement caché dans un fichier image quelconque. Quelqu'un qui voudrait connaître ce message mais qui ignore que cette image contient un message, ne pourrait pas le trouver. On peut imaginer d'autres moyens, même les plus tordus pour camoufler ainsi des messages dans un contenant anodin. Donc dans cette introduction, nous allons voir comment ce rapport sur la stéganographie sera organisé.

La stéganographie a été définie par certains comme « l'art et la science de communiquer de manière à masquer l'existence même de la communication » [1]. C'est probablement la description la plus juste de ce qu'elle est. En effet, le but de la stéganographie n'est pas de rendre illisible un message, comme la cryptographie, mais de rendre cette transmission discrète. Bien sûr, il est tout à fait possible de combiner les deux. Le mot désigne d'ailleurs relativement bien ce que c'est, il dérive du grec et veut dire *écriture cachée*.
Historiquement, on en trouve des traces dès l'antiquité :

- Démarate transmit aux Grecs les projets d'invasion de la Grèce par le roi des Perses Xerxès en utilisant un messager avec une tablette de cire soit-disant vide (en fait la cire, supposée contenir le message était vierge mais le bois supportant la cire était gravé).
- Quand Histiée fait parvenir un message au gouverneur de Milet, message tatoué sur le crâne d'un esclave et envoyé quand les cheveux de cet esclave eurent repoussé.

Généralités
Depuis les techniques se sont un peu améliorées pour les applications qui nécessitent le secret, comme les applications militaires ou pour les applications liées au commerce, comme le watermarking. La prépondérance du numérique dans le monde moderne permet de multiplier les techniques. Et les méthodes pour les casser. Actuellement n'importe qui peut cacher des informations dans des fichiers numériques grâce à des logiciels (comme Steghide [2], un petit logiciel libre en lignes de commande).
Mais malgré ces évolutions, on peut toujours s'amuser à cacher des messages dans des textes. Différentes techniques existent là aussi. Prendre la première lettre ou le premier mot de chaque ligne, lire une ligne sur deux...
Il est possible, avec quelques astuces alors d'écrire un message qui lu comme ça dit quelque chose, mais dont le message caché dit l'inverse.

Dans ce rapport nous verrons, les principes de base de la stéganographie. Ensuite nous continuerons sur les applications possibles pour les différentes techniques, comme les messages cachés ou les signatures de fichiers. Enfin, nous examinerons les limites de la stéganographie. Nous pourrons alors conclure cette étude.

Les principes de base

Les techniques *classiques*.
Nous allons voir ici, quelles sont les techniques utilisées depuis des siècles et donc indépendantes de l'informatique. Certaines peuvent toujours être utilisées, par jeu par exemple.
Nous avons vu tout à l'heure deux exemples historiques. Le second est facile à comprendre, pour retrouver le message, il suffit de raser le crâne de l'esclave. Pour le premier, il est peut être nécessaire de faire une mise au point. A l'époque, un des support d'écriture était des tablettes de bois recouvertes de cire. On gravait le message sur la cire. Quand on devait l'effacer, on faisait fondre la cire et on avait une tablette à nouveau vierge. L'idée de Démarate était de graver sur la tablette de bois. Ainsi quelqu'un qui interceptait le messager ne voyait qu'une tablette vierge sans penser à regarder sur le bois.
Ces deux techniques n'ont pas vraiment traversé les siècles (quoi que les espions allemands utilisaient la technique des cheveux rasés au début du XXème siècle [3]). En revanche les jeux de lettres ont bien survécu. Un vieil exemple est l'Ictus [4] des premiers Chrétiens.

Ce poisson stylisé est un exemple de stéganographie à deux niveaux. D'abord, il s'agit d'un signe distinctif, dont seuls les Chrétiens connaissaient le sens (il en allait de leur vie au début). Mais ce n'est pas tout. Le nom d'Ictus cache aussi un message.

- I (*I*) : **I**ΗΣΟΥΣ (*IÊSOUS*) « Jésus » ;
- X (*KH*, *CH*) : **X**ΡΙΣΤΟΣ (*KHRISTOS*) « Christ » ;
- Θ (*TH*) : **Θ**ΕΟΥ (*THEOU*) « de Dieu » ;
- Y (*U*) : **Y**ΙΟΣ (*HUIOS*) « fils » ;
- Σ (*S*) : **Σ**ΩΤΗΡ Sôter *(*SÔTÊR*) « Sauveur ».*

Avec le temps, les jeux de lettres sont devenus plus courants pour ceux qui voulaient faire passer un message sans se faire voir ou alors simplement jouer. Plusieurs techniques existent. Il s'agit à chaque fois de lire tout ou partie du texte selon des règles précises. Quelques exemples :

lire une ligne sur deux, comme le texte prétendument écrit par George Sand.

```
Je suis très émue de vous dire que j'ai
bien compris, l'autre jour, que vous avez
toujours une envie folle de me faire
danser. Je garde un souvenir de votre
baiser et je voudrais que ce soit
là une preuve que je puisse être aimée
par vous. Je suis prête à vous montrer mon
affection toute désintéressée et sans cal-
cul. Si vous voulez me voir ainsi
dévoiler, sans aucun artifice mon âme
toute nue, daignez donc me faire une visite.
Et nous causerons en amis et en chemin.
Je vous prouverai que je suis la femme
sincère capable de vous offrir l'affection
```

la plus profonde et la plus étroite
amitié, en un mot, la meilleure amie
que vous puissiez rêver. Puisque votre
âme est libre, alors que l'abandon où je
vis est bien long, bien dur et bien souvent
pénible, ami très cher, j'ai le cœur
gros, accourez vite et venez me le
faire oublier. À l'amour, je veux me sou-
mettre entièrement.

Votre poupée.

La prétendu réponse d'Alfred de Musset et la nouvelle réponse de George Sand suivent la règle simple de prendre le premier mot de chaque ligne (au passage, un message comme ça est caché quelque part dans ce rapport).

Quand je vous jure, hélas, un éternel hommage
Voulez-vous qu'un instant je change de langage
Que ne puis-je, avec vous, goûter le vrai bonheur
Je vous aime, ô ma belle, et ma plume en délire
Couche sur le papier ce que je n'ose dire
Avec soin, de mes vers, lisez le premier mot
Vous saurez quel remède apporter à mes maux.

Cette grande faveur que votre ardeur réclame
Nuit peut-être à l'honneur mais répond à ma flamme.

Mais on peut aussi prendre la première lettre de chaque premier mot de chaque ligne.. ou toute autre règle de ce genre, du moment que le destinataire la connaît. Plus astucieux, il s'agit des vers brisés[5] où selon la lecture d'un poème, on trouve une signification ou une autre. Si on divise le poème en deux colonnes, on voit un autre sens.

Vive à jamais	l'empereur des Français
La famille royale	est indigne de vivre :
Oublions désormais	la race des Capets,
La race impériale	doit seule lui survivre !
Soyons donc le soutien	de ce Napoléon.
Du comte de Chambord	chassons l'âme hypocrite :
C'est à lui qu'appartient	cette punition.
La raison du plus fort	a son juste mérite.

Mais cela reste encore facilement cassable. D'autres messages cachés dans des textes sont plus difficilement repérables. Ainsi, en Grande-Bretagne, sous Cromwell, un des partisans emprisonnés du roi, reçu une lettre qui semble-t-il n'avait rien de bien secret. Sauf que certaines virgules étaient mal placées. Et en prenant la troisième lettre après chaque virgule, on pouvait y lire une information pour une évasion (réussie). Certains ont aussi caché des messages dans des partitions de musiques, un code permettait la traduction entre les notes et les lettres. Ainsi Bach utilisait parfois cette technique pour « signer » ses oeuvres d'une série de 4 accords qu'on peut traduire (dans la notation anglophone des accords) par B A C H.
Une autre manière de cacher un message dans un texte, est d'écrire un texte banal et entre les lignes écrire avec de l'encre sympathique, comme du jus de citron. L'encre est alors invisible et n'est révélée que dans certaines conditions (un révélateur chimique, ou, dans le cas du jus de citron, la chaleur d'une flamme). On peut également transmettre des oeufs durs à la personne à qui on veut transmettre un message sans que quelqu'un ne le comprenne. Le message étant écrit avec du vinaigre sur la coquille d'oeuf. A l'extérieur il est invisible, mais quand on enlève la coquille, le message apparaît sur le blanc d'oeuf.
Un des pères de la stéganographie est l'abbé Trithème, qui écrivit vers 1499 un ouvrage parlant de

stéganographie et de cryptage, ainsi que de magie noire : *Steganographia*.
De nombreuses autres techniques existent, comme faire avaler par le messager le message et attendre qu'il ressorte... Il serait vain d'essayer de faire une liste complète de toutes les techniques existantes ou ayant existé. Mais aux Etats-Unis, après Pearl Harbor, un système de 10000 censeurs fut mis en place et mêmes les messages les plus anodins (comme des affiches de chiens disparus ou des dessins d'enfants) furent interdits ou surveillés de peur qu'ils ne cachent des messages.
Mais ce n'est qu'en 1984 qu'un mathématicien Gustavus Simmons formalisera vraiment la stéganographie.

Les méthodes numériques.
A l'ère numérique, les méthodes ont considérablement évolué. Du simple jeu de mise en page au camouflage d'informations dans des images, beaucoup de techniques existent.
Une technique simple est de cacher le message en jouant sur une très légère variation de mise en forme. On écrit un texte que l'on copie en dessous en changeant presque imperceptiblement la taille des espaces entre certains mots. Quand on superpose les deux passage, le message apparaît clairement, exemple :

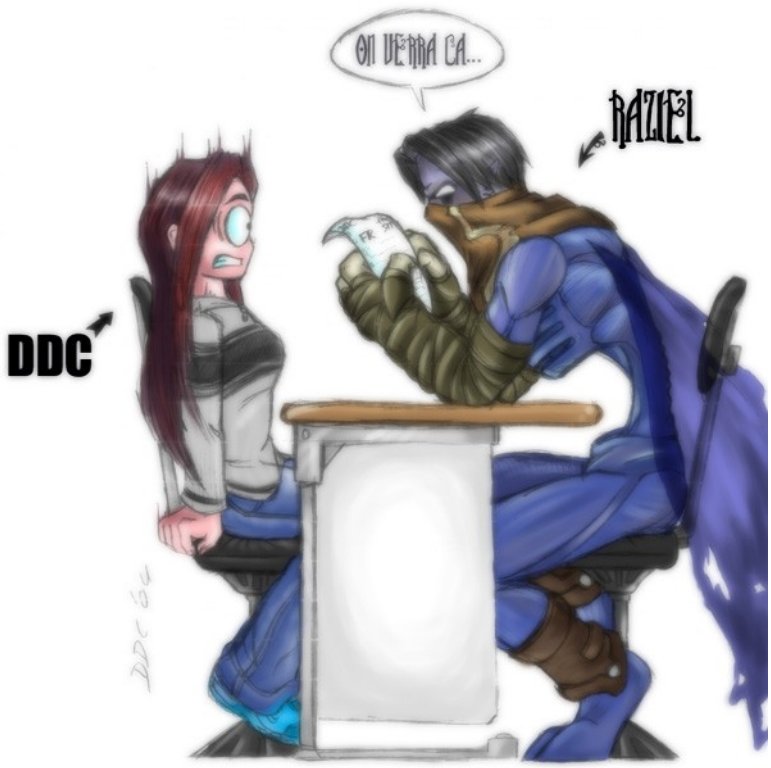
Donc ceci est un exemple basique.
Donc ceci est un exemple basique.

Donc **ceci est** un exemple **basique**.

Mais cette technique reste sensible, par exemple aux conversions de formats de fichiers.
Une méthode préférée par certains est celle qui utilise un arbre de traduction. Imaginons un une suite d'octets à transférer 100110. On prend un premier arbre, selon les valeurs des bits, on descend l'arbre de telle ou telle manière. Quand on arrive à la fin, on a un sujet pour une phrase. On va donc sur l'arbre des verbes et à partir du bit suivant, on descend l'arbre etc... A la fin on obtient une phrase anodine pouvant être transmise sans risque (mais là on s'approche de la cryptographie).

Ceci dit, ce que permet vraiment le numérique est de cacher les informations dans des « résidus ». L'exemple le plus connu est de cacher des informations dans une image sans que la différence soit visible à l'oeil nu. Comment faire ? Pour les images codées en RGB (red green blue), il s'agit de jouer sur les bits de poids faibles des images.
Une image de type bitmap est un ensemble de pixels dont la couleur est codée par 3 octets, un pour le rouge, un pour le bleu, un pour le vert, donc 256 valeurs possibles pour chaque couleur. De fait en décalant un peu une valeur (en passant de 60 à 61 pour la couleur verte par exemple), l'oeil humain ne voit pas la différence. Par conséquent, si l'on remplace les bits de poids faible de chaque couleur pour chaque pixel par des bits de poids fort d'autres images ou par des textes, on peut camoufler ces données dans une image. En pratique une image peut facilement contenir un texte brut de plusieurs pages. Pour les images, soit on accepte une légère dégradation de l'image cachée, soit l'image à cacher doit être bien moins importante en octets que celle dans laquelle elle va être cachée.
Il est aussi possible de jouer sur les formats de fichiers. En travaillant sur l'algorithme de compression des fichiers jpg, on peut camoufler aussi des messages. En effet, le jpg divise l'image en bloc de 8*8 pixels puis chaque bloc subit une opération permettant de réduire le poids (et la qualité) de l'image. Ceci donne des coefficients stockés de manière compressée. En jouant sur ces coefficients, on peut stocker des données : si le coefficient est pair positif ou impair négatif, il compte pour un bit à 1 du message à dissimuler. Sinon, c'est un 0 mais si le coefficient est égal à 0, on l'ignore pour éviter des confusions. En effet pour cacher un bit dans un coefficient, soit le coefficient correspond au bit à cacher (il est bien pair et positif quand on veut cacher un bit 1) et dans ce cas on ne fait rien, soit on décrémente le coefficient. Mais si après décrémentation on obtient un 0, on ne peut pas savoir quand on lira le message si le 0 vient du message caché ou était là à la base.
Pour les images gif, il est possible de jouer sur les bits de poids faible sur les couleurs de la table. Ou alors jouer sur la différence entre l'ordre des couleurs dans la table et un ordre de référence.

Exemple avec un jpg caché dans un autre (réalisé avec Steghide) :
Cette page contient d'abord les eux images avant traitement. Bien sûr la grande va camoufler la petite.
Ensuite le fichier traité avec Steghide et le fichier image extrait, toujours avec Steghide.
La chose qui saute au yeux, c'est que les images ne sont pas altérées de manière visible à l'oeil nu.



Toujours dans les fichiers multimédia, on peut utiliser des techniques pour camoufler des messages dans les sons. Dans les formats MP3, on utilise des techniques similaires à celles pour le jpg. Tandis que pour les fichiers codés en PCM, ce sont des techniques ressemblant à celles pour les images RGB. Mais on peut jouer aussi sur les échos. En effet l'oreille humaine n'est pas capable de distinguer des échos très rapprochés. Par exemple, on choisit deux valeurs d'écho. La première correspond à un 1, la seconde à un 0. Il est alors possible de coder un message binaire dans un son.

Mais ce n'est pas la seule façon de camoufler des données avec un ordinateur.
Une mémoire d'ordinateur est divisée en petits blocs de taille fixe, les clusters. Quand on enregistre un fichier dessus, il est divisé en autant de petits blocs que nécessaire. Bien entendu la plupart du temps, la taille du fichier n'est pas un multiple de la taille de ces clusters. Donc le dernier bloc n'est pas de la taille d'un cluster. Pour le système, le cluster est utilisé. Mais il reste de place de libre dans ce genre de clusters. Certains programmes permettent donc d'utiliser cet espace libre que le système considère comme occupé afin d'y cacher des données à camoufler.
On doit alors utiliser un fichier qui se rappelle ce qui est caché et à quel endroit afin de pouvoir y réaccéder. En fait, cela forme un second système de fichiers.
Un autre moyen moins utilisé, est de cacher des données dans un fichier exécutable. En effet ce genre de fichier a pas mal d'opérations NOP (*no operation*) qui, non seulement sont rarement exécutées, mais en plus quand elles le sont, elles ne font rien. Par conséquent, il est possible de les remplacer par des messages à faire passer. Du moment que personne ne regarde comment le fichier est codé (parce qu'un NOP est différent d'une suite d'octets qui transportent un message). On peut aussi écrire un programme qui contient des tests inutiles grâce auxquels on peut alors coder un message.

De plus, une affaire a choqué pas mal de personnes : on a découvert que les fabriquants d'imprimantes avaient, sur demande du gouvernement des USA, fait en sorte qu'en plus de ce qu'il fallait imprimer, les imprimantes imprimaient des minuscules points jaunes invisibles à l'oeil nu qui permettaient d'identifier ainsi l'imprimante de n'importe quel document ainsi imprimé. On peut alors se poser une question : combien d'autres accords de ce genre existent-ils entre les gouvernements et des groupes d'entreprises ?

On peut également utiliser les bits en trop ou inutilisés des protocoles réseaux comme IP ou TCP.
Une autre technique est d'utiliser un *canal caché* entre deux processus. Ce nom regroupe beaucoup de réalités. Bien entendu, ils ne sont cachés que lorsqu'en théorie un canal entre les deux processus ne devrait pas exister.

- **Covert storage channel** : un processus a les droits en écriture sur un espace mémoire et un autre a les droits en lecture.
- **Covert timing chanel** : un processus signale un événement à un autre en modifiant sa propre utilisation d'une ressource système de manière à changer le délai de réponse observé par l'autre processus.
- **Termination channel** : l'information est transmise lorsque le processus devant la récupérer vérifie si un premier processus a fini une tâche.
- Probabilistic channel : Un processus modifie la distribution de probabilités d'un événement. L'autre processus estime alors cette distribution.
- **Ressource exhaustion channel** : une ressource donnée est disponible ou non ?
- **Power channel** : mesure de la consomation électrique qui varie selon le bit à transmettre.

Les limites de la stéganographie

Détection des messages cachés.
Aucune technique n'est parfaite et il est toujours possible de trouver un moyen de voir un message caché... Au au moins limiter leur diffusion.
Quand il s'agit de jeux de lettres, quelqu'un d'habitué pourra trouver certains passages étranges dans les textes cachant d'autres messages. Par exemple imaginons un code basé sur le morse avec les points des **i** et des **j** ainsi que sur les barres des **t** et **f**, ça nous donnera un texte qui pourra paraître étrange, surtout si le message est long par rapport à la taille du texte, car alors les i,j,f et t seront plus nombreux dans le texte que dans un texte normal. Les techniques de base, comme lire une ligne sur eux, le premier caractère de chaque ligne etc... sont largement connu et si un message est suspect, un examen attentif du texte passant en revue toutes ces techniques permettra de dévoiler ces messages.
Pour des techniques comme l'encre sympathique, c'est plus limité mais quelqu'un qui écrit avec entre les lignes devra soit écrire petit, soit laisser un espacement assez important entre deux lignes écrite normalement ce qui sera suspect. Comme dit plus haut, durant la seconde Guerre Mondiale, les USA ont établis un gigantesque réseau de censeurs. De plus ils interdisaient tout message qui avait une chance de contenir un message caché par stéganographie (sauf les messages uniquement écrits avec du texte pur). Parfois, ils changeaient légèrement les textes transmis en mettant des synonymes. Un exemple connu est un message qui disait « Père est mort ». Les services secrets remplacèrent ce message par « Père est décédé ». La réponse fut un aveux flagrant : « Père est-il mort ou décédé ? ». Ainsi même sans être certain de l'existence de messages cachés, il est possible de les détecter en examinant le plus de messages possibles.

Pour les méthodes numériques, c'est plus complexe mais ça reste réalisable. Par exemple, pour les fichiers jpg, il est toujours possible de comparer les différents coefficients du fichier camouflant un message avec ceux qui auraient dû statistiquement tomber. Ce qui permet alors de détecter un message cachés. De même pour le son, il est possible d'avoir les mêmes résultats. Il est aussi possible de « disséquer » un fichier et de voir par exemple des échos impossibles à détecter sans appareil. Enfin si on a remplacé trop de bits de poids faible d'une image par trop de bits de poids fort d'une autre image, la seconde image commence à apparaître sur l'image censée la cacher.
Un examen attentif des trames envoyées dans un réseau peut aussi révéler certains messages, surtout ceux cachés superficiellement.
Et en ce qui concerne le stockage dans des espaces libres sur des clusters, il est possible dans certains cas de différencier un message caché d'un remplissage par défaut.
Enfin si quelqu'un regarde avec précision le contenu d'un fichier exécutable dont on a remplacé les NOP par des messages, il n'a aucun problème pour retrouver les messages cachés.

Destruction des messages.
Il est possible facilement de détruire les messages... outre la destruction du support pour les messages écrits physiquement. Modifier un message suspect sans en changer le sens peut ainsi détruire le message caché à l'intérieur.
Et niveau numérique, c'est aussi simple.
Si un message est caché dans une image, il suffit de légèrement retravailler l'image (agrandir puis rétrécir par exemple) pour que le message caché à l'intérieur devienne illisible. De même remettre un message à l'intérieur de l'image pose problème. Ce genre de modifications est très simple. Autre

faiblesse, le changement de format. Nous avons fait le test en cachant un texte dans un jpg. Nous avons enregistré ensuite ce jpg en bmp, et réenregistré en jpg, impossible de ressortir le message caché.
Enfin une image composée de larges zones de couleurs unies pourra avoir des différences visibles entre sa version avant le camouflage et celle après.
Tout ceci est également valable pour les messages cachés dans les sons. Surtout au niveau du changement de format. Prenons un son où le message est caché grâce à d'infimes différences non perceptibles par un homme, le transformer en mp3 serait une erreur vu que ce format supprime les informations non perceptibles à l'être humain. Par conséquent, le message caché est détruit.
Et la technique qui consiste à cacher les informations dans des clusters à moitié vides est risquée. Pour rappel, dans ce cas, on cache les données dans des clusters qui ne sont pas entièrement remplis (que le système considère comme occupés). Le problème dans ce cas est que le fichier qui occupe le cluster n'est pas figé. Il est tout à fait possible qu'il soit supprimé, qu'on y rajoute ou retranche des informations. De fait la taille de l'espace où on range les informations peut varier. Si elle augmente ça va mais si elle rétrécie, alors des informations sont perdues. De plus, si on supprime le fichier censé occuper le cluster, alors le système considérera le cluster comme inutilisé et donc réécrira dessus, effaçant alors les données cachées. Un moyen alors de réduire les risques est de cacher les informations de manière redondante afin que la perte de données d'un cluster ne fasse pas perdre d'information. Bien entendu, si on cache des données dans des clusters d'un support non réinscriptible (comme un CD), il n'est pas possible alors de détruire ces données sans détruire le support.

Déchiffrer les données.
La stéganographie n'est pas une fin en soit. En effet, même si un message est bien caché, s'il est découvert, il peut tout à fait être lu. Par conséquent, il est bien plus prudent de compresser, si possible, et de crypter les données. Bien entendu cela impose que le destinataire connaisse le code. Par exemple, Steghide demande un mot de passe lorsqu'il cache des données (il est possible de dire que le mot de passe ce n'est rien en appuyant sur Entrée). Sans ce mot de passe, il n'est pas possible de ressortir les informations.
Ceci dit, c'est sans compter sur l'ingéniosité de certains qui ont cassé les algorithmes de camouflage de données utilisés par la plupart des logiciels et qui grâce à ça peuvent retrouver les données cachées dans d'autres.
Autrement dit, actuellement, si un fichier est suspecté de transmettre des données cachées, il y de fortes probabilités qu'au mieux quelqu'un les rendra illisibles, au pire pourra déchiffrer ces données.
Mais tout n'est pas aussi facile. Trouver les données cachées dans un cluster ne permet pas facilement de les comprendre. En effet il s'agit, sauf pour les données de très petite taille, de données éparpillées un peu partout sur le disque, là où il y a de la place. Donc sans un index permettant de s'y retrouver, il faut fouiller tous les clusters, et, une fois qu'on a trouvé les données cachées, jouer au puzzle pour les reconstituer.

Les applications de la stéganographie

Les messages cachés.
Depuis que les humains peuvent communiquer, il ont des secrets à transmettre mais que toutes les oreilles ne peuvent ni ne doivent entendre. Pour cela, il a donc fallu transmettre des messages sans que les porteurs ou des oreilles indiscrètes les lisent. Les hommes ont appris à crypter les données de manière à ce que seuls ceux qui avaient le code puissent les comprendre. Mais les codes sont souvent cassables et un individu peut très bien donner le code à des personnes qui ne sont pas censées le connaître. Pour cela, donc, certains ont eu l'idée de cacher, non pas la manière de comprendre les informations mais le moyen de transmission de ces informations.
Cette mentalité est bien entendu utilisée pour les questions militaires et/ou politiques. On a prétendu, semble-t-il à tord, que les terroristes d'Al Quaida ont utilisé cette technique pour préparer les attentas de 2001. Mais il est probable que certains groupes, terroristes ou non, communiquent ainsi, surtout aujourd'hui avec internet. Comment surveiller tout ce qui se passe sur la toile ? Il est facile de communiquer via des images camouflant des textes, sans que cela ne se voit pour peut qu'on ne soit pas surveillé par des services spéciaux.
En revanche on sait que les Américains, durant la seconde Guerre Mondiale, réduisaient la tailles de certaines photos à celle d'un point sur un **i** ou un **j** et les mettaient sur des textes à la place des points sur les **i** et les **j**.
Mais tout n'est pas aussi secret, certains cachent des messages pour faire passer un message sans le dire officiellement. Comme par exemple le livre d'un auteur inconnu, (Hypnerotomachia Poliphili,1499, peut être écrit par Franscesco Columna), dont les premières lettres des chapitres forment la phrase *Poliam Frater Franciscus Columna Peramavit* (Frère Francesco Colomna aimait Polia intensément). Certains profitent de l'acrostiche pour faire passer un message, le nom de l'être aimé par exemple.
Ou tout simplement par jeu. Comme la prétendue correspondance entre George Sand et Albert de Musset, au XIXème siècle, il est courant, et à la mode de cacher des messages dans des textes. Ce n'est pas nouveau mais c'est à cette période que c'est le plus couramment utilisé.

Watermarking/Fingerprinting.
Le watermarking est le fait d'insérer une marque invisible dans un fichier afin d'en indiquer l'auteur du documents, le propriétaire, ainsi que les permissions autorisées au propriétaire du document.
Le fingerprinting est une marque sur un fichier indiquant le genre de choses qu'un fichier est censé faire, sous quelle système d'exploitation et autres informations de ce genre.
Ceci dit, il s'agit d'une variante améliorée de la stéganographie, le tatouage numérique. En effet, la nature même de leurs fonctions indique qu'il est nécessaire que les informations soient conservées même en cas de modification du fichier (comme un redimensionnement d'une image), on dit qu'ils doivent être robustes.
Les premiers algorithmes jouaient sur les bits de poids faible de la luminance d'une image, mais n'étaient pas robustes, étant donné que des techniques comme une compression jpg détruisent ce marquage. On peut aussi, c'est la méthode du patchwork, multiplier les bits modifiés afin qu'une étude statistique nous donne les bits marqués. On peut également renforcer un bit à 0 pour chaque bit renforcé à 1 pour que les statistiques globales de l'image ne changent pas. Et on peut imaginer utiliser une clé pour coder l'emplacement des bits à 0 et à 1. C'est plus résistant que les premiers

algorithmes mais ça reste très limité. Pour améliorer la robustesse, on peut utiliser l'algorithme de Koch et Zhao. Il propose de diviser l'image en bloc de 8*8 bits et de calculer dessus les bits à marquer. Ce genre de marquage résiste à la compression jpg. D'autres techniques existent aussi comme l'étalement de spectre, qui ressemble à celle du patchwork, ou encore le watermarking fractal qui fait appel à un traitement mathématique poussé de l'image pour calculer la marque.
On peut attaquer ces techniques de différentes manières :

- essayer d'enlever la marque ou la remplacer par une marque que l'attaquant a définie lui-même. Le logiciel StirMark est gratuit et est un programme d'attaque de marques.
- On peut attaquer l'image comme si on attaquait un code secret donc avec des techniques proches de celles utilisées en cryptologie.
- Plus simple, on peut tricher avec les marques. Par exemple si des robots parcourent internet à la recherche d'utilisations illégales d'une image marquée. Si on a cette image sans les droits de la mettre sur internet, il suffit de la découper en petit morceaux et de demander au navigateur de les afficher côte à côte.

Il ne faut pas confondre le marquage numérique (watermarking, fingerprinting) avec les méta-informations que certains formats de fichiers supportent.
Bien faites des ce genre de marque peut aller jusqu'à détecter les parties de l'image qui ont été modifiées. En effet ce sont celles dont la marque est totalement différente de l'image d'origine.
A noter que ces méthodes peuvent aussi servir sur des fichiers qui ne sont pas des images, et toujours pour des raisons commerciales. L'exemple type est l'industrie de la musique qui essaye de marquer ainsi les CD et les fichiers musicaux, ce qui permettrait de mieux gérer les différentes copies, ainsi que les autorisations ou refus de graver tel ou tel morceau.
Ceci dit, ce n'est pas encore parfait, il n'y a pas de solution de tatouage global. Cette technique doit donc être complémentaire aux système de gestions de droits.
Certains se demandent quand même si ce genre de protections contre les copies ne veut pas dire limiter l'accès à l'information et donc une sorte de censure.

Conclusion

La stéganographie est un ancien moyen de cacher des messages sans qu'un lecteur non autorisé en prenne connaissance. Elle a souvent été dans l'ombre, on parle souvent simplement de crypter les informations. C'est en partie exact, mais en partie faux. Quand on camoufle des données, il est fortement conseillé de les crypter également. En effet, pour reprendre un exemple de Wikipedia : *Si on utilise le coffre-fort pour symboliser la cryptographie, la stéganographie revient à enterrer son argent dans son jardin. Bien sûr, l'un n'empêche pas l'autre, on peut enterrer son coffre dans son jardin.*[6]
Si à la base la stéganographie était liée au domaine militaire, on l'utilise désormais dans beaucoup de domaines. Au point que maintenant ça soit utilisé dans des applications pour le commerce. Qui aurait pu dire que l'idée de raser un esclave pour tatouer sur le crâne un message aurait tellement évolué que certaines applications sont totalement sans rapport avec l'idée originale ?
Ce qui est aussi marquant avec la stéganographie, c'est que virtuellement, c'est indétectable dans une époque comme la notre où le nombre d'informations transmises chaque seconde croît en permanence. Il devient bien plus difficile de tout surveiller, même avec un réseau comme le réseau Echellon. De fait, si un particulier n'est pas surveillé, il peut alors transmettre des informations cachées sans risques réels de les voir lues par quelqu'un d'autre. Compte tenu du nombre d'image qui sont chaque semaines postées sur des forums et compte tenu du nombre de forums sur internet, compte tenu des hébergeurs d'images comme Imageshack, comment ne pas passer à côté de message cachés si l'expéditeur veut vraiment passer inaperçu ?

La stéganographie est en fait un monde vaste et varié mais totalement inconnu du grand public. Mais pouvait-il en être autrement de quelque chose dont la finalité est de dissimuler la communication ?

[1] LEYMONERIE, R., "Cryptage et droit d'auteur", (1998) 10 Les Cahiers de propriété intellectuelle, pp. 417-460.

[2] http://fr.wikipedia.org/Steghide

[3] http://www.lifl.fr/~fontaine/PUBLIS/fontaine_pls_02bis.pdf.gz

[4] http://fr.wikipedia.org/wiki/Ictus

[5] http://fr.wikipedia.org/wiki/Vers_brisés

[6] http://fr.wikipedia.org/wiki/Stéganographie

Bibliographie

Trouvé sur un texte de l'université de Montréal
LEYMONERIE, R., "Cryptage et droit d'auteur", (1998) 10 Les Cahiers de propriété intellectuelle, pp. 417-460.

Wikipedia
Wikipedia. *Steghide.* [en ligne]. [consulté le 4 avril 2006]. Disponible sur <http://fr.wikipedia.org/Steghide>

Wikipedia. *Ictus.* [en ligne]. [consulté le 4 avril 2006]. Disponible sur <http://fr.wikipedia.org/Ictus>

Wikipedia. *Vers brisés.* [en ligne]. [consulté le 4 avril 2006]. Disponible sur <http://fr.wikipedia.org/Vers brisés>

Wikipedia. *Stéganographie.* [en ligne]. [consulté le 4 avril 2006]. Disponible sur <http://fr.wikipedia.org/Stéganographie>

Technique
Fabien GALAND. *Stéganographie.* [en ligne]. [consulté le 4 avril 2006]. Disponible sur <http://www.techniques-ingenieur.fr/affichage/DispIntro.asp?nGcmID=H5870>

Frédéric RAYNAL. *Canaux cachés.* [en ligne]. [consulté le 4 avril 2006]. Disponible sur <http://www.techniques-ingenieur.fr/affichage/DispIntro.asp?nGcmID=H5860>

Mihai MITREA, Françoise PRÊTEUX, Adriana VLAD. *Le tatouage robuste, ou comment protéger les contenus visuels.* [en ligne]. [consulté le 4 avril 2006]. Disponible sur <http://www.techniques-ingenieur.fr/affichage/DispIntro.asp?nGcmID=H5360>

Frédéric RAYNAL. *Systèmes de protection électronique de droits d'auteur.* [en ligne]. [consulté le 4 avril 2006]. Disponible sur <http://www.techniques-ingenieur.fr/affichage/DispIntro.asp?nGcmID=H7318>

Frédéric RAYNAL. *Canaux cachés.* [en ligne]. [consulté le 4 avril 2006]. Disponible sur <http://www.techniques-ingenieur.fr/affichage/DispIntro.asp?nGcmID=H5860>

Sur Scirus
Frédéric SAINT-MARCEL. *Rapport pour l' Etudes D'Approfondissement* . [en ligne]. [consulté le 4 avril 2006]. Disponible sur <http://www-adele.imag.fr/~donsez/ujf/easrr0203/tatouagestegano/tatouagestegano.pdf>

Jean-Guillaume DUMAS. *Cryptographie & Algorithmique de la Sécurité.* [en ligne]. [consulté le 4 avril 2006]. Disponible sur <http://www-lmc.imag.fr/lmc-mosaic/Jean-Guillaume.Dumas/Enseignements/2005_M64_IUPMAI3/Securite.pdf>

V.BZNONY, A SEDOGLAVIC. *Stéganographie élémentaire avec un fichier BMP.* [en ligne]. [consulté le 4 avril 2006]. Disponible sur <http://www.lifl.fr/~sedoglav/C/steganographie.ps>

Google Scholar
Stéphanie VAN LOO, Marc MALENGREAUX. *Les mystères de la cryptographie, ou l'art du secret.* [en ligne]. [consulté le 4 avril 2006]. Disponible sur <http://www.jsb.be/expo2005/resumes/Les%20myst%C3%A8res%20de%20la%20cryptographie.pdf>

Stéphanie VAN LOO, Marc MALENGREAUX. *Les mystères de la cryptographie, ou l'art du secret.* [en ligne]. [consulté le 4 avril 2006]. Disponible sur <http://www.jsb.be/expo2005/resumes/Les%20myst%C3%A8res%20de%20la%20cryptographie.pdf>

Stéphanie VAN LOO, Marc MALENGREAUX. *Etude et comparaison de schémas d'analyse Stéganographique d'images numériques.* [en ligne]. [consulté le 4 avril 2006]. Disponible sur <http://www-rech.enic.fr/coresa2004/articles/p009-roue.pdf>

Google
Fabien PETITCOLAS. *Tatouage numérique: une nouvelle utopie ?.* [en ligne]. [consulté le 4 avril 2006]. Disponible sur <http://isdn.enssib.fr/archives/transversal/JDN/propintell/petitcolas.pdf>

Frédéric RAYNAL, Fabien PETITCOLAS et Caroline FONTAINE. *L'art de dissimuler les informations.* [en ligne]. [consulté le 4 avril 2006]. Disponible sur <http://www.lifl.fr/~fontaine/PUBLIS/fontaine_pls_02bis.pdf.gz>

Auteur inconnu (alors étudiant en DUT). *Stéganographie.* [en ligne]. [consulté le 4 avril 2006]. Disponible sur <http://steganographie.free.fr/>

INRIA. *La problématique du watermarking.* [en ligne]. [consulté le 4 avril 2006]. Disponible sur <http://www-rocq.inria.fr/codes/Watermarking/>