Stéganographie : Sécurité par Dissimulation

***A. ALI-PACHA*[1] ; *N. HADJ-SAID*[1] ; *A. BELGORAF*[1]; *A. M'HAMED*[2]**

[1]*Université des Sciences et de la Technologie d'Oran –USTO,*
[2]*Institut National des Télécommunications Evry- Paris*
E.Mail : alipacha@yahoo.com

1. Introduction

Pour éviter un jour de devoir fournir des versions décryptées de ces messages à la justice, la stéganographie s'avère bien entendu indispensable.

La stéganographie a pour fonction de permettre la transmission sécurisée de messages dans des circonstances où la cryptographie ne peut être mise en œuvre (par exemple parce qu'elle est interdite !). La stéganographie, un procédé de codage des données et qui met à profit l'immensité d'Internet et les caractéristiques des images numériques, est l'art et la science de dissimuler un message à cacher dans un message quelconque. On peut, par exemple, remplacer le dernier bit significatif de chaque point d'une image par celui du message. Ainsi, l'aspect graphique est quasiment inchangé.

L'insertion d'un message dans le fichier choisi implique la modification de parties de son code. Tout l'art de la stéganographie consiste à faire en sorte que ces changements soient invisibles ou inaudibles. Plus le message est réduit et le fichier volumineux, plus cette altération a des chances de passer inaperçue. D'où l'utilisation de fichiers image, son ou vidéo plutôt que de fichiers texte, de taille plus réduite. A cet aspect quantitatif s'ajoute un aspect qualitatif. Le message caché est inséré là où il sera le plus imperceptible : dans les fichiers son, par exemple, le message caché est intégré aux basses fréquences qui correspondent au bruit de fond.

Exemple, un email non crypté sur Internet peut être comparé à une vulgaire carte postale : tout comme il nous parait logique de mettre nos lettres sous enveloppe lorsque nous utilisons la poste physique, il devrait paraître naturel de crypter nos emails afin que seul leur correspondant légitime puisse les lire. Seulement à la différence de l'enveloppe facilement ouvrable et refermable discrètement, la cryptographie forte n'autorise le déchiffrement du message qu'en connaissance de la clé secrète (dans le cas de cryptographie symétrique) ou de la clé privée (dans le cas de cryptographie asymétrique).

La stéganographie repose sur l'idée de sécurité par l'obscurité : si personne ne sait qu'il y a un fichier caché, personne ne cherchera à le regarder ou le récupérer. Et avec tout ce qui passe sur l'Internet, et le nombre de fichiers joints que les gens s'échangent, personne ne dispose de suffisamment de ressources informatiques pour scanner tous ces transferts d'images, sons et autres fichiers.

La stéganographie permet de nos jours à dissimuler un fichier, une musique, un dessin, un texte dans un autre document numérique, musique, texte, code html, ... Avec l'informatique la stéganographie prend une nouvelle ampleur, mais la dissimulation de message ne date pas d'hier.

2. Historique de la Stéganographie

I- Stéganographie sur Support Physique

Bien avant la stéganographie informatique existait la stéganographie sur support physique [1, 3]. Ces techniques consistaient à camoufler l'information secrète dans le support physique même du message anodin. Bien entendu avec l'avènement du traitement numérique de l'information, cette stéganographie a disparu mais reste néanmoins assez intéressante voire même amusante.

La stéganographie sur support physique nécessite bien entendu l'usage de courrier de physique (ou de messagers).

L'histoire veut que les premières utilisations de la steganographie date du 5^{eme} siècle avant Jésus-Christ. Herodotus, auteur grec, relate les communications secrètes entre deux chefs de guerre qui utilisaient des esclaves pour passer des messages et plans de batailles. L'idée était simple, ils tatouaient sur le crâne des esclaves le message, laisser repousser les cheveux.

Les Grecs vont mettre en place plusieurs mécanismes dédiés à la stéganographie. Des trous sur un disque représentant des lettres. Des fils, de couleurs différentes, permettaient de lire un mot. Une autre technique était de percer un petit trou, sur les lettres d'un document pour en faire un message. Une prémisse aux messages enchâssés.

Avec le développement de la chimie, par la suite on utilisait des encres sympathiques pour communiquer des messages en toute discrétion, (l'encre "invisible", souvent du jus de citron, d'ognon ou de chlorure d'ammoniac ou avec une solution de vinaigre et d'alun). Il suffisait d'écrire un message sans importance et d'inscrire entre les lignes quelques mots du message secret à transmettre à l'aide de l'encre sympathique. Passées quelques minutes l'encre sympathique devenait invisible. Le message sans importance n'éveillait pas l'attention et le destinataire légitime était le seul à connaître le procédé : pour lire le message secret, il suffisait de chauffer le papier ou de le tremper dans un bain d'espèce chimique spécifique.

II- La stéganographie informatique

Grâce aux moyens informatiques dont nous disposons, nous pouvons exprimer toute notre créativité et stéganographier à loisir tout en éveillant au minimum l'attention. En effet l'information numérique à l'état brut peut généralement subir de nombreuses compressions destructives par élimination de données inutiles.
L'idée est alors de remplacer ces données inutiles, ces bruits de fond parasites par des données plus utiles qui seras en fait les données que l'on veut cacher.
Pour cacher des données, on peut utiliser toute sorte de types de fichiers numériques : images, sons, vidéos,...

3. Définitions et Terminologies

La stéganographie (en anglais: steganography ou data hiding) est encore une technique peu connue du grand public : pour preuve aucun dictionnaire ne lui consacre une entrée (à noter au passage qu'il ne faut pas confondre sténographie et stéganographie). En fait le mot stéganographie (en anglais : steganography) tire son origine d'une étymologie grecque : steganos signifiant caché, couvert et graphos signifiant écriture, dessin. Ainsi nous pouvons en déduire que la stéganographie est l'art de cacher des messages secrets au sein de messages plus anodins.
La stéganographie est la technique consistant à insérer un fichier dans un second fichier, sans que l'aspect extérieur de ce dernier ne soit modifié (hormis sa taille).

En d'autres termes, on peut insérer à l'intérieur d'une image, d'un fichier son, d'un fichier Adobe Acrobate, ou même d'une page html, un fichier de son choix, quelque soit sa nature. Par exemple, tapez votre courrier dans un simple éditeur de texte, encryptez-le, insérez le fichier ainsi créé dans la dernière photo de vos vacances, et envoyez cette photo au destinataire de votre courrier.

En apparence, il recevra une photo, qu'il pourra d'ailleurs visionner sans problème. Mais s'il *sait* que la photo contient un courrier et s'il a le logiciel adéquat, il pourra alors extraire votre courrier de la photo, puis le décrypter par la méthode habituelle.

Ceci a un intérêt énorme, à l'heure ou la sécurité des transferts de fichiers par Internet n'est plus assurée.

Le terme dissimulation d'information est très général ; il désigne le fait de cacher une information dans un support. Cependant, selon les objectifs, et les contraintes qui en découlent, on distingue différentes variantes.

Tout d'abord, le **médium vierge** dans lequel des informations sont cachées est appelé **médium de couverture**, ou plus simplement le **médium**. Une fois que les informations sont insérées, nous utilisons alors l'expression **stégo-médium**.

D'une manière générale, nous appelons données l'information dissimulée dans le **stégo-médium**.

Le fichier "destinataire" doit être de taille suffisante pour accueillir votre fichier de données. Dans certains cas, sa taille initiale va varier, dans d'autres non. Mais ceci n'a strictement aucune importance tant que personne ne peut comparer le fichier initial et le nouveau fichier créé.

Le fichier destinataire peut être de différents types: Graphique (jpg, gif, bmp, pcx, tif, etc), Son (wav), Text (txt), ou autres formats divers (html, pdf, etc). Par contre il n'existe pas de logiciel permettant d'utiliser tous ces types à la fois. Il vous faudra donc choisir un logiciel en fonction du type de fichier que vous désirez utiliser.

Le processus complet de dissimulation d'information repose sur deux opérations :

- la dissimulation, qui consiste à insérer l'information dans le médium ;
- l'extraction, qui récupère cette information. Le mot détection est également utilisé lorsqu'il s'agit de vérifier la présence d'une information (représentée grâce à un signal, une caractéristique particulière du médium...) dans le stégo-médium, sans pour autant vouloir l'extraire.

Selon les objectifs poursuivis, les schémas de dissimulation d'information portent des noms différents, on en distingue trois principaux :

1) La stéganographie (Data Hiding) [4] cherche à cacher un message secret, ou message plus sommairement, dans un médium de sorte que personne ne puisse distinguer un médium vierge d'un stégo-médium. La nature de l'information dissimulée ne revêt pas d'importance : il peut tout aussi bien s'agir d'un texte en clair que de sa version chiffrée. Ce message n'a a priori aucun lien avec le stégo-médium qui le transporte.

2) Le tatouage cherche à répondre au problème de la protection des droits d'auteur [5, 6, 7]. Un client essaye d'abord de détecter le possible présence d'une marque dans un médium, puis dans l'affirmative, de vérifier si l'utilisateur a bien acheté une licence. Il s'agit bien de dissimulation d'information puisque, pour y parvenir, on insère un tatouage (ou marque, ou filigrane) dans le médium spécifique au propriétaire. Comme celui-ci souhaite protéger son médium et non une version trop déformée, l'insertion doit minimiser les modifications subies par le médium afin d'être imperceptible. Ensuite, chaque copie du stégo-médium contient la même marque, celle du propriétaire légal. Ici, la dissimulation ne signifie pas la même chose qu'en stéganographie : un attaquant sait qu'un tatouage est présent dans le stégo-médium, mais cette connaissance ne doit cependant pas lui permettre de le retirer.

3) Enfin, le fingerprinting cherche à permettre la détection des copies illégales d'un stégo-médium. Chaque utilisateur authentifié reçoit sa propre copie du médium qui contient une empreinte l'identifiant. Ainsi, lorsqu'une copie illégale est découverte, la lecture de l'empreinte indique la source de la fuite. A la différence du tatouage où l'origine du médium importe, le fingerprinting se préoccupe plutôt de l'utilisateur final. Chaque copie du médium contient une information différente, relative à son utilisateur, rendant alors chaque stégo-médium différent.

Lorsqu'un attaquant tente uniquement de détecter si un message transite dans un médium sur le canal de communication, on dit de lui qu'il **est passif**. La plupart des solutions de stéganographie ne considèrent que ce type d'attaquant, au contraire des deux domaines suivants où il est **actif** : l'attaquant sait alors que le stégo-médium contient une information et il tente de la modifier ou de la retirer.

4. Objectifs de la stéganographie

Malgré leurs objectifs distincts, ces trois variantes n'en requièrent pas moins des paramètres communs :

- chaque approche nécessite des données, que ce soit un message, un tatouage ou une empreinte ;
- ces données sont dissimulées dans un support, le médium, qui possède plus ou moins d'importance selon le schéma : aucune pour la stéganographie, capitale pour les deux autres ;
- il est indispensable de pouvoir distinguer des personnes différentes, utilisant des données identiques dans un même médium : chacune doit donc posséder sa propre **stégo-clé** (ou plus simplement **clé**) afin que l'insertion de ces données identiques permettent quand même de différencier les protagonistes.

Toutefois si le but de la stéganographie est de dissimuler un message sans éveiller l'attention humaine, avec la stéganographie informatique il faut également veiller à ne pas éveiller l'attention des logiciels d'analyse. Car si une image est soupçonnée de contenir un message stéganographié, on pourra toujours la soumettre à un logiciel chargé de traquer tout bruit de fond trop organisé et statistiquement non aléatoire : on peut alors facilement repérer un message stéganographié et effectuer une stéganalyse (tentative de récupération du message en clair caché).

Il faut donc que le message à cacher soit en tout point comparable à une suite de bits aléatoires : pour cela une seule solution: *il faut préalablement crypter le message*.

5. Conditions requises

Les objectifs de la dissimulation d'information peuvent changer de manière subtile. Classiquement, les applications sont triées en fonction de trois critères :

- l'imperceptibilité : les données ne doivent pas être « perceptibles » dans le stégo-médium. Pour le tatouage ou le fingerprinting, l'objectif est de ne pas détériorer le stégo-médium protégé. Cependant, la contrainte est plus forte en stéganographie où il s'agit plutôt d'une indétectabilité statistique afin qu'une personne surveillant le canal ne remarque pas la présence du message ;
- la capacité est la quantité de bits significatifs dissimulés dans le stégo-médium par unité d'accès (par exemple, le nombre de bits par seconde en musique) ;
- la robustesse correspond à l'aptitude de préservation des données cachées face aux modifications du stégo-médium.

En stéganographie, une propriété essentielle est l'indétectabilité statistique puisqu'une personne surveillant le canal de communication ne doit pas pouvoir différencier un médium d'un stégo-médium. De plus, comme le message constitue l'information principale, la capacité doit aussi être assez élevée. Quant à la robustesse, elle constitue une défense contre les modifications subies par le stégo-médium. Néanmoins, la meilleure défense reste l'incapacité de l'adversaire à détecter le message. Ainsi, la plupart du temps, le canal ne modifie pas le stégo-médium et les besoins en robustesse sont minimes. En revanche, des mesures doivent être prises lorsque que l'adversaire est actif, soit en terme de robustesse, soit pour contrôler l'intégrité du message afin de détecter un éventuel changement dans celui-ci.

En tatouage, les contraintes diffèrent largement. Tous les utilisateurs savent, ou soupçonnent très fortement, qu'une marque est dissimulée dans le stégo-médium, et il n'est donc nul besoin de chercher à en détecter la présence. Le stégo-médium doit toutefois rester aussi proche, au sens d'une mesure de similitude sur l'espace des média, que possible de l'original afin de ne pas être dénaturé. La capacité dépend étroitement de l'application. Si le tatouage est suffisamment discriminatoire, un bit d'information suffit à répondre à la question : cette marque est-elle présente dans ce stégo-médium ? Même lorsque les données sont extraites et une mesure de confiance calculée, l'utilisation d'un seuil pour valider ou non la présence de la marque ne fournit toujours qu'un bit d'information. Au contraire, lorsque les données extraites servent ensuite à diriger une action, on considère alors que le tatouage transporte de l'information.

Enfin, les besoins du fingerprinting sont à peu près identiques à ceux du tatouage pour l'imperceptibilité et la robustesse (pour ce dernier critère, les raisons diffèrent : on ne

souhaite pas voir un utilisateur distribuer sa propre copie... avec l'empreinte de quelqu'un d'autre insérée). En revanche, la capacité est importante car un médium doit contenir une empreinte spécifique à un utilisateur. Dans ces conditions, il n'est pas réaliste de se contenter, comme en tatouage, d'une réponse binaire sur la présence d'une empreinte dans un stégo-médium car il faudrait alors tester toutes les empreintes pour un stégo-médium. Ainsi, tout comme en stéganographie, l'extraction de l'empreinte est indispensable.

6. Domaine d'utilisation

De nombreux usages peuvent exister dans des domaines très variés [2] mais souvent sensibles comme :

- **Communiquer en toute liberté même dans des conditions de censure et de surveillance :**

Les Américains ont été surpris par la parfaite synchronisation de ces attaques du 11 septembre, une seule attaque ne nécessite pas d'échanges d'informations entre des personnes, mais quatre… Il fallait forcément échanger des éléments, soit par courrier, téléphone, fax ou par e-mail. Et ceci pendant plusieurs semaines précédant l'attaque.

- **Contrebalancer toutes les législations ou barrières possibles empêchant l'usage de la cryptographie :**

Dans certain pays la cryptographie est interdite et quiconque est surpris en train de l'utiliser risque des peines importantes. En effet en utilisant une stéganographie robuste, il est impossible de suspecter la moindre trace d'un message crypté.

- **Publier des informations ouvertement mais à l'insu de tous des informations qui pourront ensuite être révélées et dont l'antériorité sera incontestable et vérifiable par tous :**

Les attestations officielles (de diplômes, par exemple), faites sur du papier spécial, comportant éventuellement un filigrane, des dessins, une signature manuscrite, des tampons, etc. Pensons aussi aux nouvelles cartes d'identité plastifiées, aux procédés nombreux employés pour sécuriser les billets de banque, de telle sorte que les destinataires soient sûrs qu'ils proviennent bien de l'établissement habilité à les émettre et non de quelque faux-monnayeur.

7. Supports & techniques de la stéganographie

Nous passer en revue les différents supports numériques utilisés pour dissimuler des données: le texte, l'image et le son, page HTLM, programme, disque dur ou CD.

7.1 Texte

La stéganographie sur un support texte est quelque chose qui ne supporte pas la fantaisie. En effet chaque retouche est directement visible par le lecteur. D'autre part un texte signé par les méthodes décrites est relativement facilement identifiable.

La dissimulation de données dans du texte est une chose bien particulière, et comme nous allons le voir elle n'a pas grande chose à voir avec l'image ou le son. Et ceci pour plusieurs raisons: on ne travaille pas avec le texte dans "l'a peu près". Je m'explique, dans une image on peut considérer qu'"endommager" celle ci avec un filtre passe-haut suffisamment "léger" ne change quasiment rien à la perception que nous allons avoir de celle-ci. Par contre avec un texte, soit le texte est comme l'original soit il ne l'est pas. Celui ci ne permet quasiment aucune modification. Une des exigences de la dissimulation de données est d'endommager le moins possible le texte original. Pour cela nous allons utiliser la méthode dite des "espaces".

7.1.1 Méthodes des "espaces" (en fin de phrases)

Il y a 2 méthodes différentes quoi que reposant sur le même principe. La première méthode consiste à mettre des espaces en fin de ligne. On se définit un code à suivre et l'on commence:

0 espaces en fin de ligne correspondent à 0, 1 espace en fin de ligne correspond à 1.

Ex: (ici les espaces sont remplacés par des "_" pour plus de lisibilité).

Bonjour ceci est un message caché. A vous de le lire. _Je pense que vous commencez _à comprendre le principe. Malheureusement tout n'est pas _rose. Mais bon, nous arrivons quand même à _dissimuler un octet_.

Comme vous pouvez le voir dans notre exemple nous avons codé: 0 espace; 1 espace; 1 espace; 0 espace; 1 espace 0 espace; 1 espace; 1 espace <=> 01101011 soit un octet de dissimulé.

Inconvénient :

- Il faut énormément de lignes pour coder peu de texte. En effet, il faut 8 lignes pour coder 1 octet. Donc imaginons que l'on veuille coder une phrase de 20 mots (chaque mot faisant environ 4 caractères) et que l'on code chaque caractères sur 7 bits (on optimise comme on peut), il va nous falloir environ 560 lignes.

- Super visible par une personne extérieure qui s'y attend un peu. Et donc facilement manipulable.

Avantages

- Très facile et donc très simple à implémenter. Et puis ça peut marcher avec de nombreuses personnes.

- On peut rajouter des espaces pour coder plus de caractères sur moins de texte:

En utilisant 3 espaces: 0 espace <=> 00, 1 espace <=> 01, 2 espace <=> 10, 3 espace <=> 11. Il faut alors 4 lignes pour coder 1 octet (au lieu de 8).

7.1.2 Méthodes des "espaces" (entre les mots)

Cette méthode est basée sur le même principe, mais cette fois-ci nous allons coder notre texte dans le nombre d'espaces entre chaque mot. C'est encore plus visible que la méthode précédente, mais le rapport texte codé sur texte hôte est beaucoup plus important. On se met d'abord d'accord sur une convention :

Un espace entre 2 mots suivit de deux espaces entre les 2 mots suivants <=> 0, deux espaces entre 2 mots suivit d'un espace entre les 2 mots suivants <=> 1

Pour mieux comprendre voici un exemple avec le texte suivant:

Ceci_est__essai__de_texte__caché_dans__un_texte_hôte. Vous__devez__avouer que_ce__ n'est_ pas_ très__subtil.

_ __ : 0, __ _ : 1, __ _ : 1, __ _ : 1_ __ : 0, __ _ : 1, __ _ : 1, _ __ : 0 => 01110110 soit un octet.

Le rapport texte à coder sur texte hôte. Il vous faut 2 mots pour un bit, donc pour une phrase de 20 mots (20*7=240 bits), il faut un texte hôte de 480 mots, en comptant 10 mots par ligne on arrive à environ 50 lignes. On gagne un rapport de 10 comparé à la méthode des espaces en fin de ligne.

7.2 Son

De faibles variations, imperceptible pour l'oreille, dans les basses fréquences ou ce que l'on appelle le bruit de fond peuvent contenir une grande quantité d'information. Un grésillement infime peut cacher des secrets.
Evidemment, ce bruit doit de préférence être transmis de façon numérique sans quoi les vraies pertes de transmission pourraient effacer entièrement le message caché.

7.3 Image

Une image est constituée de points (ou pixels) qui sont autant de données permettant à l'ordinateur de recréer l'image lors de la lecture du fichier. Il est possible d'insérer de nouvelles lettres et chiffres dans ces données, afin de constituer un message caché dans l'image initiale.

« Chaque pixel est constitué de trois couleurs : le rouge, le vert et le bleu. Certains de ces points peuvent être remplacés par une autre information sans que les changements apportés dans l'image soient perceptibles à l'œil humain. ».
Reste que, pour pouvoir lire ce message caché, la personne recevant l'image doit connaître la clef permettant de lire les informations contenues dans celle-ci. Sans cette clef privée, il est impossible de lire le contenu caché, et l'image garde tout son mystère.

Loin d'être complexe, les méthodes pour cacher une image numérique dans une autre image sont simples [6, 7]. Plusieurs techniques sont utilisées. La première est basée sur les couleurs présentes dans une photo. De façon similaire au son, le spectre des couleurs peut être traduit par des courbes. Les logiciels de codage insèrent dans ces spectres (rouge vert bleu) de micros variations qui ne seront pas perceptibles pour les personnes qui regardent les photos. Tous les types de données peuvent être insérés de cette façon, seule la capacité de l'image à "absorber" ces données cachées, limite les transferts. Une autre méthode consiste à modifier imperceptiblement la valeur de chaque pixel de l'image. Là aussi, la dégradation de l'image passe inaperçue et seul le logiciel adéquat permet de retrouver les données cachées.
De façon plus sophistiquée, il est aussi possible de cacher des données dans des constructions mathématiques (fractales en particulier). Ces constructions qui se répètent à l'infini selon une suite de chiffres peuvent inclure des bouts de fichiers sans en altérer la représentation graphique.
Une image peut en cacher une autre. En remplaçant les bits qui altèrent le moins (bits de poids faible) l'image de couverture par les bits les plus représentatifs (bits de poids fort) de l'image à cacher. Pour une image codée sur 32 bits, on peut en remplacer environ 4 bits sans problèmes. Voici quelque exemple :

Format JPEG : Les images les plus utilisées sur le net sont des images format JPEG. La compression JPEG (Joint Photographic Experts Group) utilise « Discrete Cosine Transform » le DCT pour transformer d'une façon successive chaque bloc de 8×8-pixel d'une image en un 64 DCT coefficients. Les bits les moins signifiants du DCT coefficient sont utilisés comme des bits redondants pour dissimuler le message secret. La modification d'un seul DCT coefficient affect les 64 pixels d'une image.

Format GIF : Dans certaines images de format GIF (Graphic Interchange Format), les couleurs des pixels ne sont pas indépendantes mais codées selon une palette de 256 couleurs. On donne une valeur comprise entre 0 et 255 à chaque pixel et lorsque l'on lit les pixels, on se réfère à la palette pour y mettre la couleur.

Exemple :

11111111 <=> 255 : et la couleur 255 est définie comme étant un rouge vif.

11111110 <=> 254 : et la couleur 254 est définie comme étant un bleu foncé.
En ne changeant que le dernier bit vous allez complètement détruire notre image. Se qui n'est pas vrai pour le format JPEG car la modification se fait dans le domaine des fréquences et non pas dans le domaine spatial.
Il s'agit ici du domaine le plus vaste et certainement du plus intéressant. En effet, les applications sont très nombreuses, et les méthodes à utiliser sont assez complexes (comparées à celles utilisées pour le texte). Le marquage d'images a de nombreux buts: copyright des images (très important à l'époque d'Internet où des milliers d'images circulent sur le web), mais aussi marquage de papiers ou de billets (pour éviter le photocopillage), vérification de l'intégrité de documents, etc. A chaque utilisation correspond une méthode.
En fait les techniques doivent répondre à certaines règles très importantes et souvent difficiles à concilier:

- Endommager le moins possible le support sur lequel le marquage va avoir lieu. (l'œil humain ne doit pas être choqué).
- Le marquage doit pouvoir supporter le plus grand nombre de transformations possible sans être dégradé (compression JPEG, filtres, passage analogique-numérique, changement de palettes etc.. voir les différentes attaques possibles sur le marquage).
- La complexité de l'algorithme doit être minimum afin de pouvoir effectuer le marquage et/ou la détection en temps réel.
- Le marquage peut avoir lieu au niveau de l'image dans le domaine des fréquences:

7.4 HTML

Enfin certains logiciels de stéganographie se proposent de cacher des messages dans des pages HTML : ils ne font que toucher au source pour camoufler le fichier secret en insérant des espaces entre balises, variant minuscules et majuscules dans les balises,... Astucieux mais cela peut toutefois se détecter par analyse statistique et même par un coup d'oeil au source dont l'indentation exotique pourra attirer l'attention.

Exemple :

Dans toute page HTML comme celle-ci, il y a (ou il peut y avoir) des indications qui n'apparaissent pas sur l'écran des navigateurs – ce sont les balises 'META'. On peut y dissimuler tout ce qu'on veut. Le non-initié n'y verra que du feu. Mais le procédé n'est pas très performant, puisqu'il suffit de cliquer, généralement dans le menu Affichage du navigateur, sur Source de la page pour voir tout ce que contiennent les balises META.

7.5 Programmes

- Dans les "zones mortes" du code (commentaires, branche morte d'un organigramme)
- Dans le programme lui-même à l'aide d'une commande jamais utilisée (quintuple clic)

7.6 Disques dur ou CD

Mais les formes de stéganographie informatique ne s'arrêtent pas à cela : il est également possible de cacher des fichiers à l'intérieur de l'espace disque libre d'un disque dur ou d'une disquette. Car il faut bien différencier 2 choses : ce qui est inscrit sur le disque et ce que y est inscrit dans la table d'allocation du disque géré par le système d'exploitation. En effet pour répertorier tous les fichiers d'un disque, le système d'exploitation accède et met à jour une table d'allocation des fichiers : seuls les entrées de fichiers figurant dans cette table sont accessibles par le système d'exploitation. L'idée est alors d'inscrire un fichier physique sur le disque dur sans que la table d'allocation en ait connaissance : ainsi le fichier est sur le disque mais le système d'exploitation ne le voit pas. Pour ce faire il suffit d'utiliser un logiciel spécialisé écrivant et lisant directement sur le disque sans passer par le système d'exploitation. Un tel système de stéganographie est efficace si l'intercepteur n'en a pas conscience, par contre si celui-ci est au courant alors il n'aura aucune difficulté pour retrouver le fichier. L'autre point faible de cette technique est que le disque ne doit subir aucune modification en écriture par le système d'exploitation une fois que la stéganographie a été effectuée car sinon le système d'exploitation pourrait écraser sans en avoir connaissance le fichier caché car les secteurs du disque abritant ce fichier sont considérés comme espace libre pour le système d'exploitation.

Inconvénients

Facile d'aller récupérer des fichiers sur le disque à l'aide de logiciels spécialisés.

- en plus, on crypte les données les fichiers non répertoriés sont considérés comme inexistant
- le système peut malencontreusement réécrire dessus
- données redondantes

8. Conclusion

La technique qui aurait été employée, la stéganographie, consiste à cacher un message dans un support "innocent". Elle peut, de surcroît, se combiner à la cryptographie, qui se charge de dissimuler le sens de la missive et non plus son existence. Le résultat est alors particulièrement efficace. Le message secret s'abrite d'abord derrière son invisibilité. En cas de découverte, il restera à le décoder. Un défi pour les services secrets qui, dans le cas du terrorisme, doivent réaliser ces deux opérations au plus vite pour que l'information recueillie ne soit pas obsolète.

Il faut savoir que si la steganographie est très pratique, son utilisation informatique est détectable. Il ne faut pas oublier que cela est avant tout un code informatique. La sécurité de la stéganographie repose sur le fait que le message ne sera sans doute pas détecté.

Enfin, on peut dire que le filigrane électronique (Watermarking) est sans doute la principale application industrielle des algorithmes de dissimulation.

Webographie et Bibliographie

[01] http://www.stega.net/ste/stegano.htm: L'histoire de la stéganographie

[02] http://www.intelligenceonline.fr/: Les efforts de la NSA vis-à-vis la steganographie

[03] Le coût du piratage http://rvp.tvt.fr/themes/INFORMATIQUE/art1 du 9 mars 1999.

[04] F.A.Petitcolas, R. J. Anderson, et M. G. Kuhn. Information hiding-a survey. Proceedings of the IEEE (USA), 87(7) : 1999.

[05] M.Barni, F.Bartolini. De Rosa et A.Piva. Capacity of the watermarking channel : How many bits can be hidden within a digital image? In Proc. SPIE, january 1999.

[06] F. Hartung et M.Kutter “Multimedia watermarking techniques”. Proceedings of the IEEE, 87(7) :1079-1107, juillet 1999.

[07] Marthin Kutter, Sviatoslav, Voloshynovskiy et Alexandre herrigel. Watermarking copy attack. Sympasium, electronic imaging 2000: Security and watermarking of multimedia. Content