RAPPORT MINI PROJET

Traitement d'images

Sujet : La Stéganographie

Encadré par :

Professeur Mounir EL OMARI

Réalisé par :

Ayoub EL HAMRAOUI

Hind ELKAMOUSI

Younes CHABRAOUI

Table de matières

Partie Théorique

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.

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 5eme 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,...

III. 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) 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.

IV. 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.

V. 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.

VI. 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 certains 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.

Partie Pratique

I. Partie fonctionnelle :

Le mini projet s'agit d'une application desktop qui cache un message de l'utilisateur A dans une image, l'image peut être envoyée à un utilisateur B, ce dernier doit avoir l'application installée afin qu'il puisse voir le message caché dans l'image.

Ce projet a été crée en utilisant le langage de python qui est connu des librairies open source, utilisées dans tous les domaines de recherche scientifique, il peut être utilisé dans tous les Systèmes d'exploitation.

II. Partie technique :

1. Prérequis :

Il faut tout d'abord s'assurer d'avoir tkinter, PILLOW, stegano et Image.

Ces derniers peuvent être installé via la ligne de commande Windows (CMD) ou le terminal UNIX / MacOs.

2. Installation :

→ Il suffit d'exécuter les commandes suivantes :

Tkinter

```
C:\Users\aelha>python -m pip install tk
```

PILLOW

```
C:\Users\aelha>
C:\Users\aelha>python -m pip install PILLOW
Requirement already satisfied: PILLOW in c:\users\aelha\appdata\local\programs\python\python310\lib\site-packages (9.1.1
)
```

Stegano

```
C:\Users\aelha>python -m pip install stegano
```

Image

```
C:\Users\aelha>python -m pip install Image
```

3. Explication du code :

```python
from tkinter import * # appel à tous les composants du module Tkinter
from tkinter import filedialog #appel à filedialog
import tkinter as tk #au lieu d'utiliser le mot tkinter on le remplace par tk
from PIL import Image, ImageTk #appel à Image et ImageTk
import os #appel à  os (fctions systeme)
from stegano import lsb #lsb du module stegano

```

L'instruction import effectue deux opérations ; elle cherche le module dont le nom a été donné puis elle lie le résultat de cette recherche à un nom dans la portée locale.

4. Configuration de la fenêtre :

```python

root = Tk() ## création de la fenêtre principale
root.title("Steganographie Mini Projet Traitement d'image") # donner un titre à la fenetre
root.geometry("700x500+150+180")
#Modifie la taille de la fenêtre. s est une chaîne de caractères de type "wxh±x±y".
# w et h sont la largeur et la hauteur. x et y sont la position du coin supérieur haut à l'écran.
root.resizable(False, False) #Spécifie si la fenêtre peut changer de taille. w et h sont des booléens.
root.configure(bg="#2f4155") #couleur de l'arrière plan

```

→ Cette dernière capture explique la configuration de la fenêtre qui s'affiche à l'utilisateur.

Les fonctions créées :

```
15
16  def Aff_Image():
17      global filename #variable globale sera utilisée dans d'autres fonctions
18      filename = filedialog.askopenfilename(initialdir=os.getcwd(),
19                                            title= 'selectionner une image',
20                                            filetype=(("PNG","*.png"),
21                                                      ("JPG","*.jpg"),
22                                                      ("Tous","*.txt"))) # l'ecran qui s'affiche pour choisir une image
23      IMG =Image.open(filename) # ouverture de l'image
24      IMG =ImageTk.PhotoImage(IMG) #creation et affichage de l'image
25      lbl.configure(image=IMG,width=250,height=250) # redimensionnement de l'image
26      lbl.image =IMG #LSB = Least significant bit : remplace le dernier bit avec un bit du message secret
27
28  def cacher_Image():
29      global secret # variable globale sera utilisée après dans d'autres fonction
30      message = text1.get(1.0, END) # prendre ce que l'utilisateur a saisie
31      secret = lsb.hide(str(filename), message) # cacher le message via la méthode LSB
32
33
34
35  def sauv():
36      secret.save("hidden.png") # sauvegarder l'image sous le nom hidden.png
37
38
```

→ Les fonctions dans cette capture s'occupent d'enregistrer et cacher le message saisie par l'utilisateur dans l'image.

```
38
39      def Aff_Img():
40          message_clair = lsb.reveal(filename) # trouver le message caché
41          text1.delete(1.0,END)
42          text1.insert(END,message_clair) #afficher le texte
43
```

→ Cette dernière fonction déchiffre le message caché dans l'image et l'affiche en clair.

III. Partie GUI :

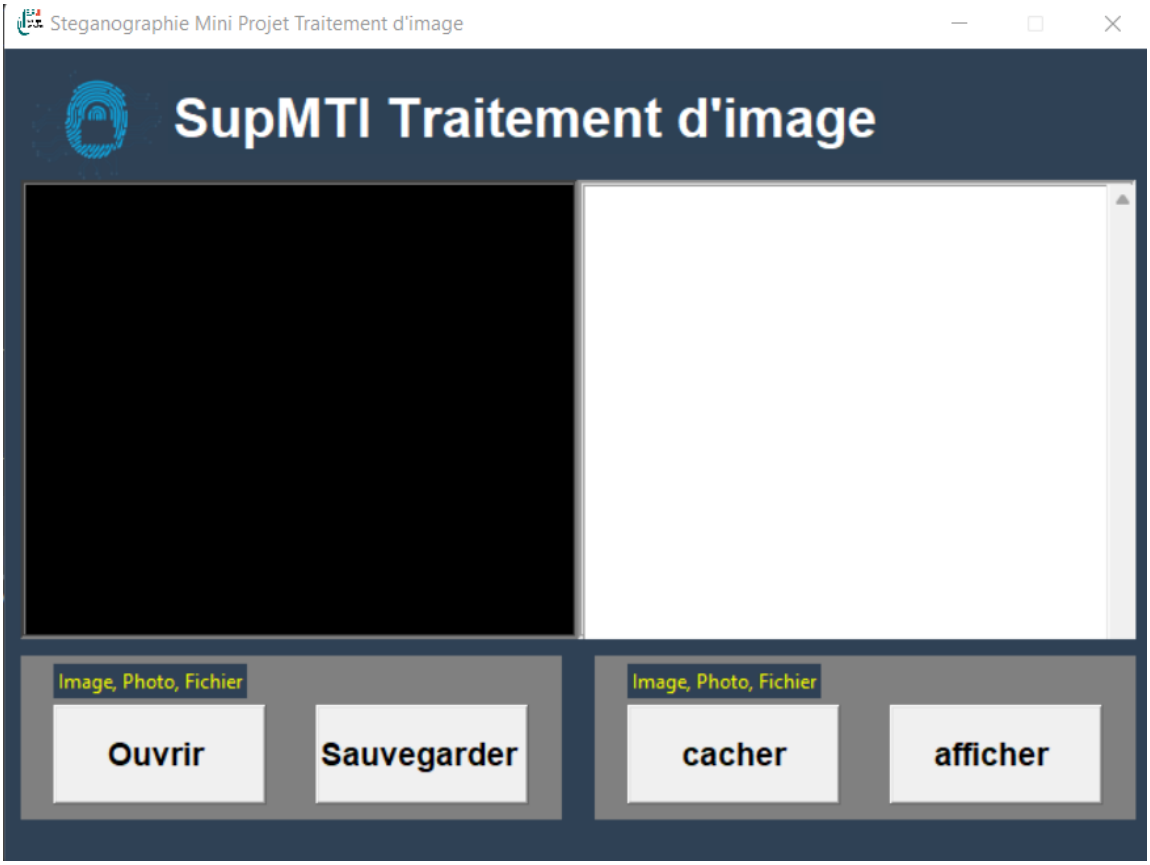
```
#icone en haut
image_icon =PhotoImage(file="logo.png")
root.iconphoto(False,image_icon) ##icone de l'interface ( dans ce cas j'ai choisi celle de supmti

#logo :
logo = PhotoImage(file='logo2.png')
Label(root,image=logo,bg='#2f4155').place(x=10,y=0)
Label(root,text="SupMTI Traitement d'image", bg='#2d4155',fg='white',font='arial 25 bold').place(x=100, y=20)

# Frame N° 1
f = Frame(root,bd=3,bg='black', width=340,height=280,relief=GROOVE)
f.place(x=10,y=80)
lbl = Label(f,bg='black')
lbl.place(x=40,y=10)

#Frame N°2
f2 = Frame(root,bd=3,bg='white', width=340,height=280, relief=GROOVE)
f2.place(x = 350, y=80)
text1 =Text(f2, font="Rebote 20", bg='white',fg='black', relief=GROOVE, wrap=WORD)
text1.place(x=0,y=0, width=320,height=295)
scrollbar1 = Scrollbar(f2)
scrollbar1.place(x=320,y=0,height=300)
scrollbar1.configure(command=text1.yview)
text1.configure(yscrollcommand=scrollbar1.set)
```

→ Partie front (affichage de l'interface)

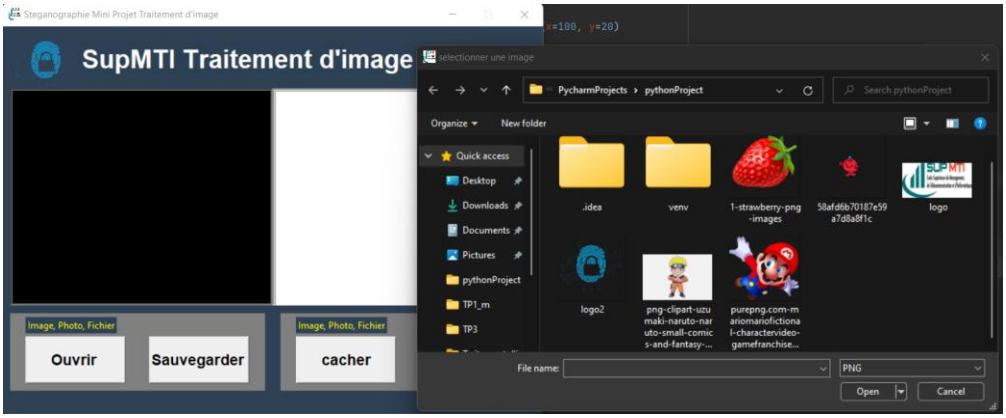
```

#Frame N°3
f3=Frame(root,bg='gray',width=330,height=100,relief=GROOVE)
f3.place(x=10,y=370)
Button(f3, text='Ouvrir ', width=10,height=2,font='arial 14 bold', command=Aff_Image).place(x=20,y=30)
Button(f3, text='Sauvegarder ', width=10,height=2,font='arial 14 bold', command=sauv).place(x=180,y=30)

Label(f3, text='Image, Photo, Fichier',bg='#2f4155',fg='yellow').place(x=20,y=5)

#Frame N° 4
f4=Frame(root,bg='gray',width=330,height=100,relief=GROOVE)
f4.place(x=360,y=370)
Button(f4, text='cacher', width=10,height=2,font='arial 14 bold',command=cacher_Image).place(x=20,y=30)
Button(f4, text='afficher ', width=10,height=2,font='arial 14 bold', command=Aff_Img).place(x=180,y=30)

Label(f4, text='Image, Photo, Fichier',bg='#2f4155',fg='yellow').place(x=20,y=5)
```

→ Suite de la partie Front.

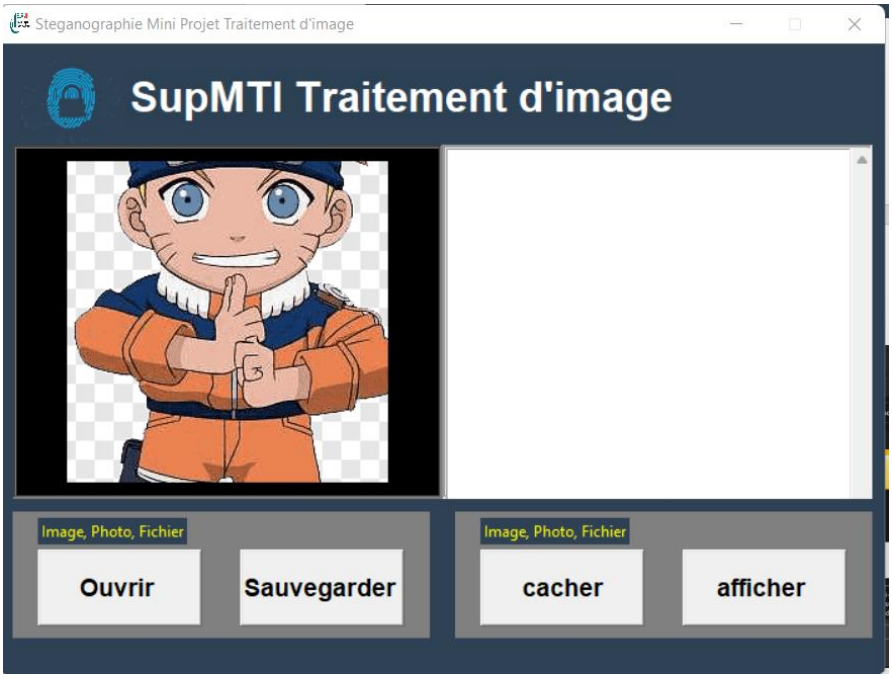
IV. Démonstration de l'application :



→ On clique sur le bouton **Ouvrir** :



→ On doit choisir une Image :

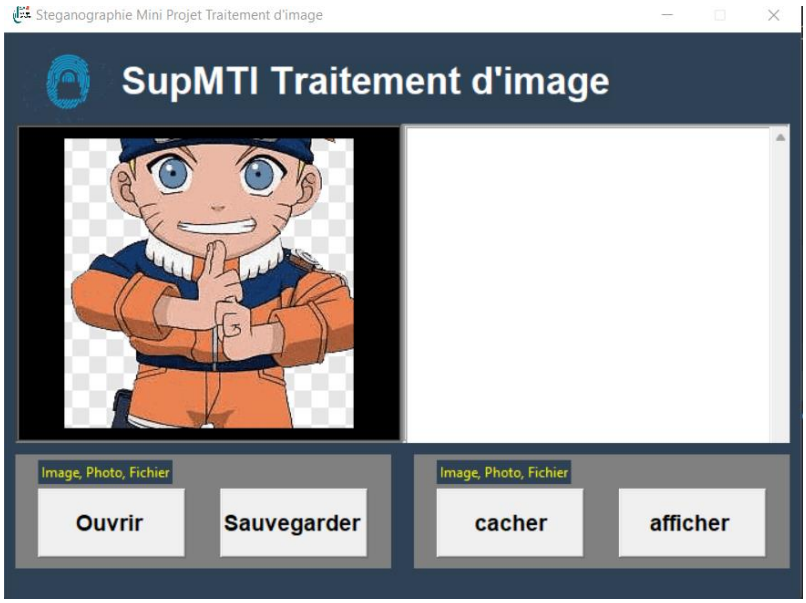


→ Puis j'écris un message quelconque dans le bloc du texte à droite :

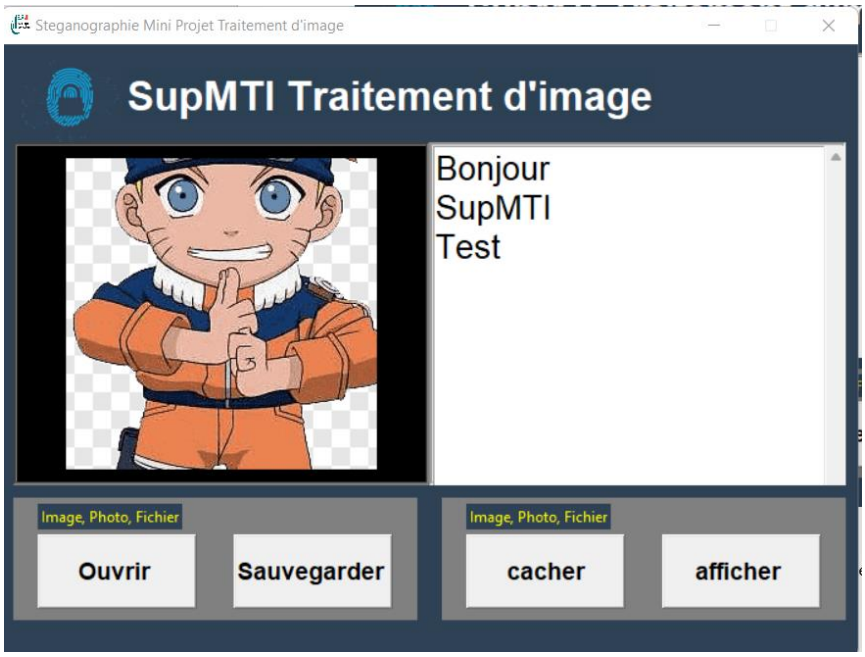
→ Je clique sur Cacher -> Sauvegarder

Dans notre répertoire on remarque la création d'une nouvelle image intitulée « Hidden » , C'est l'image qui contient le message caché.

→ Affichage du message caché :



→ Répéter les étapes mais cette fois choisir l'image « Hidden » au lieu de l'autre puis on clique sur le bouton afficher.



Cela affiche le message qu'on a caché.

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.