Compte Rendu - Projet Sécurité

    - MediaPlayer.exe
    - Utilitaires

Composition

- Lafosse Alexy
- Lahrouri Yasin

Introduction & Contexte

Explication du projet

Le projet visait à concevoir un virus de type compagnon. Ces virus ont pour objectif de se dissimuler en prenant le nom d'un programme exécutable existant. Lorsque l'utilisateur lance le programme, le virus compagnon s'exécute en premier, avant de céder la main au programme d'origine. Cette approche vise à éviter tout soupçon de la part de l'utilisateur, car l'exécution initiale semble être celle du programme attendu.

Contexte

Nous fournissons un cd Rom rempli d'images et de vidéos accompagné d'un programme s'intitulant MediaPlayer.exe sensé afficher uniquement des images, ou de procéder à la lecture de vidéos. En réalité, ce programme est un virus et va infecter d'autres programmes.

Démarrage

Tout d'abord, nous avons commencé le projet en sélectionnant six images libres de droit appartenant à trois formats distincts : jpg, png et bmp.

Par la suite, nous avons développé plusieurs utilitaires, chacun doté de fonctionnalités distinctes. Ils ont été conçus selon trois critères principaux, à savoir :

- La présence d'une interface graphique,
- La réalisation de calculs concrets,
- L'interaction avec l'utilisateur.

Présentation des programmes

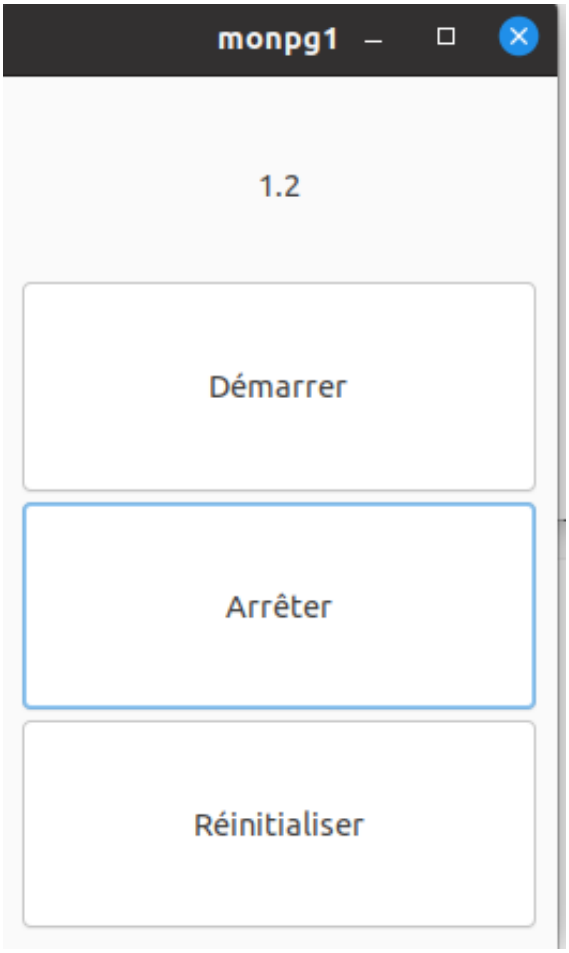
GTK

Pour ces programmes nous avons utilisé la bibliothèque GTK pour créer une interface graphique.

Nous l'avons installé à l'aide des commandes :

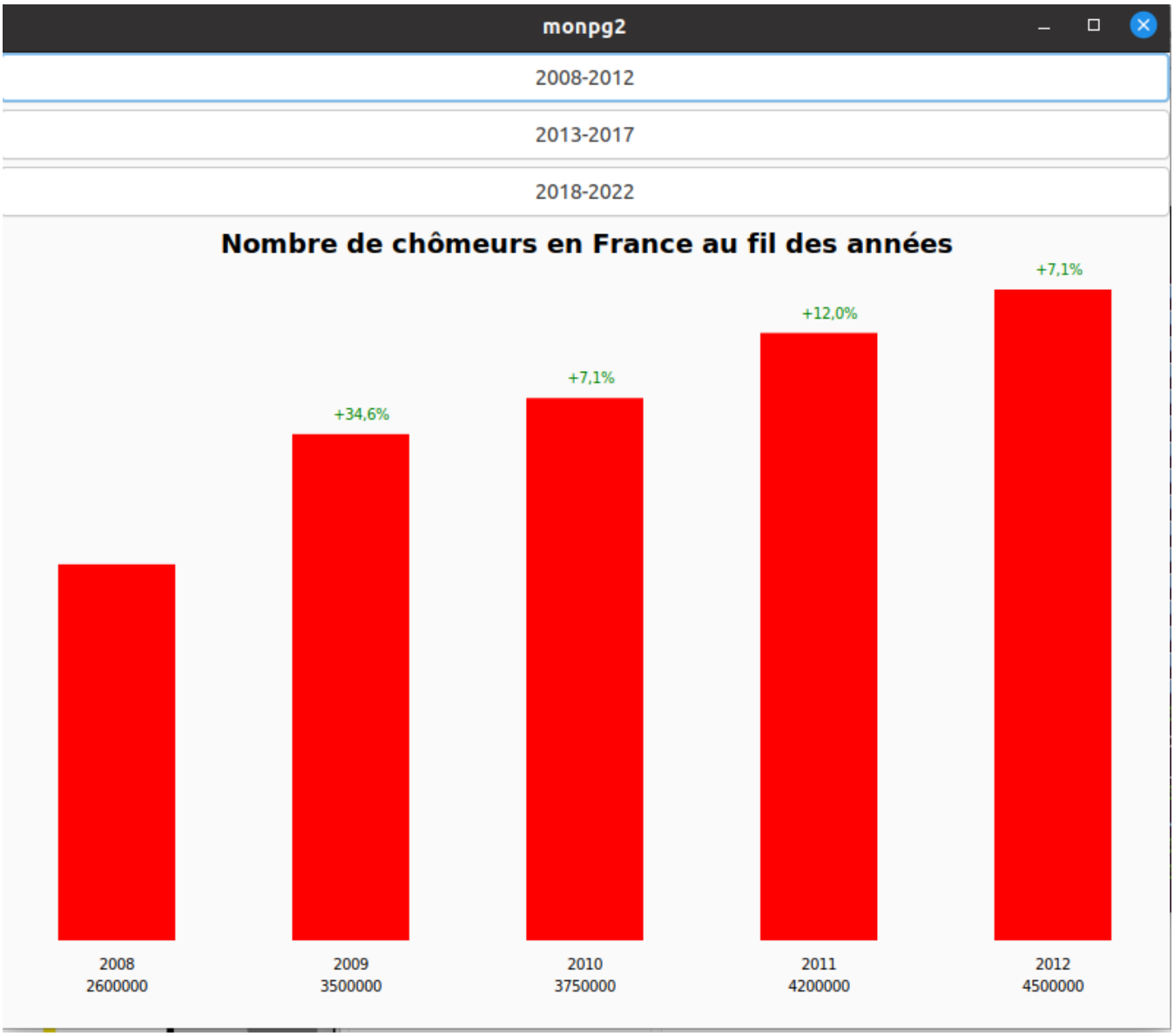
```
sudo apt-get update
sudo apt-get install libgtk-3-dev
```

MonPG1



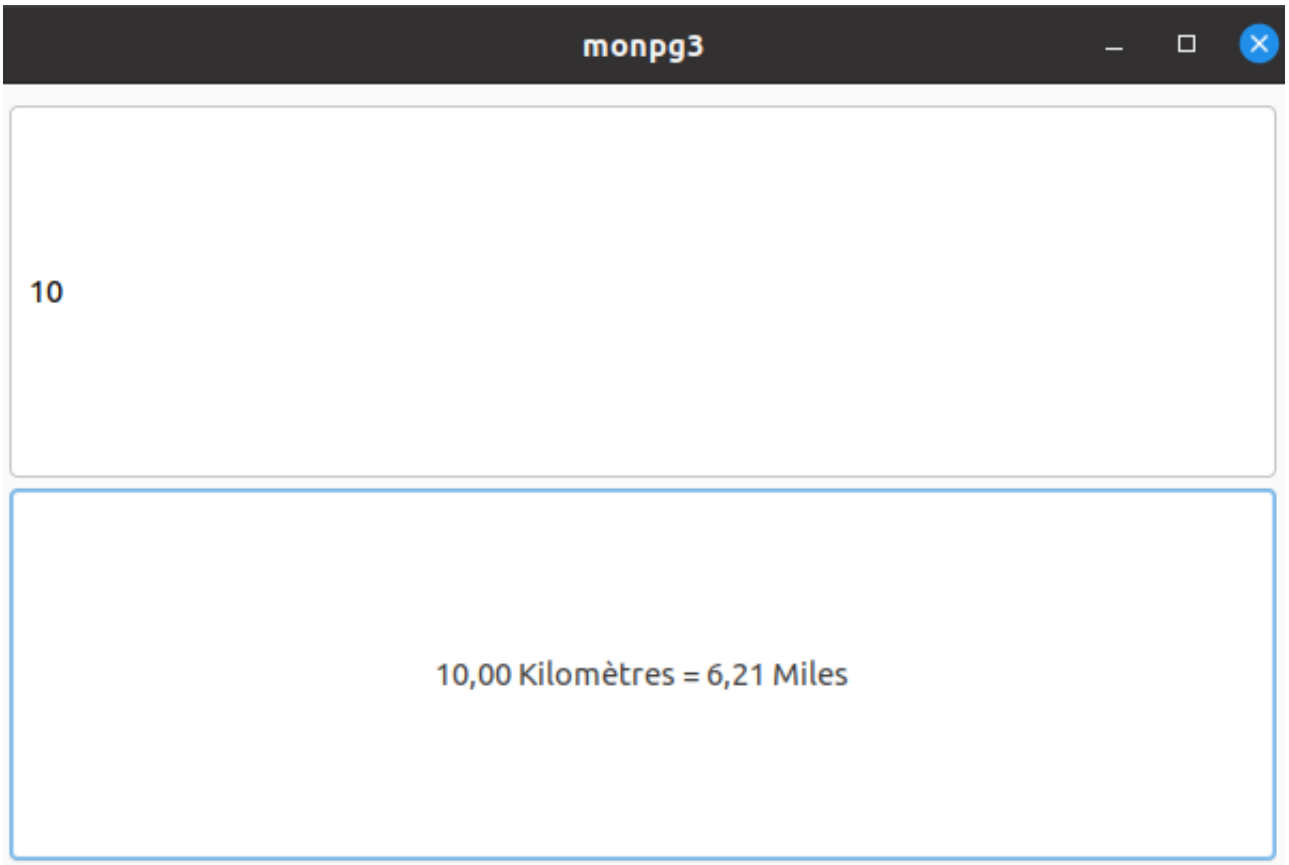
Ce programme agit comme un chronomètre, permettant aux utilisateurs de démarrer, arrêter et réinitialiser le temps à leur convenance.

MonPG2



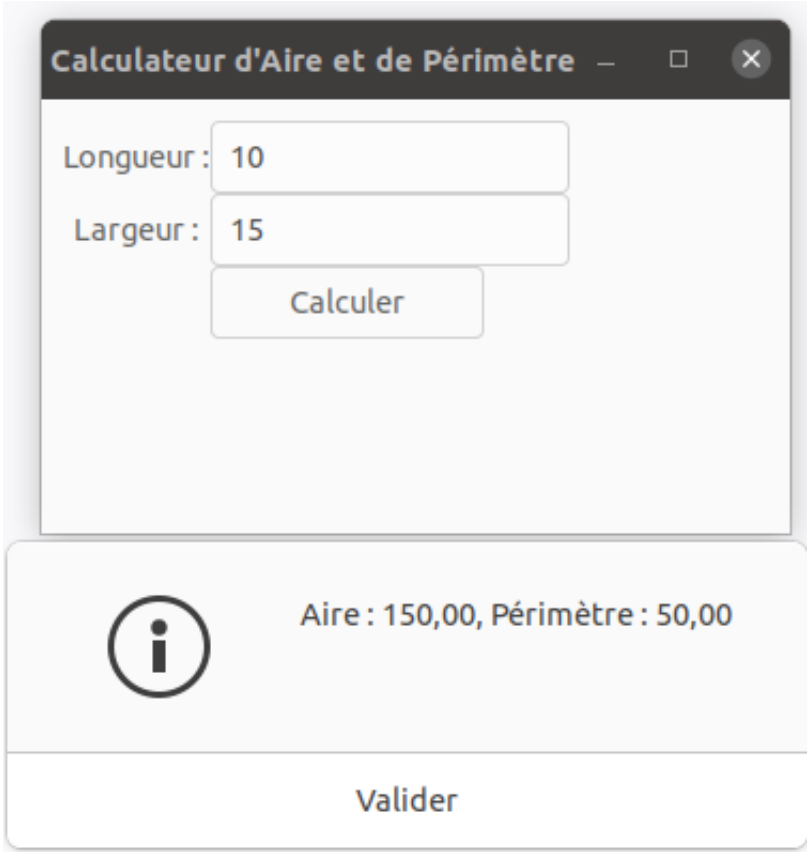
Ce programme affiche un graphique interactif des taux de chômage en France de 2008 à 2012, enrichi d'annotations indiquant les pourcentages d'augmentation ou de diminution d'une année à l'autre. Grâce à des boutons dédiés, les utilisateurs peuvent facilement basculer entre différents intervalles de temps, y compris les périodes de 2013 à 2017 et de 2018 à 2022, permettant de constater l'évolution du chômage en France au fil des années.

MonPG3



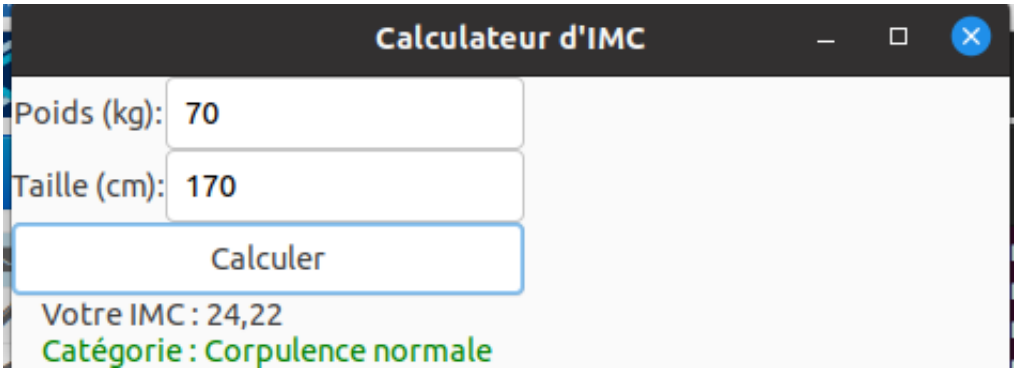
Ce programme est un convertisseur pratique qui transforme les kilomètres en miles. Pour effectuer une conversion, il suffit d'entrer le nombre de kilomètres et de cliquer sur le bouton "Convertir". Le résultat en miles s'affiche instantanément.

MonPG4



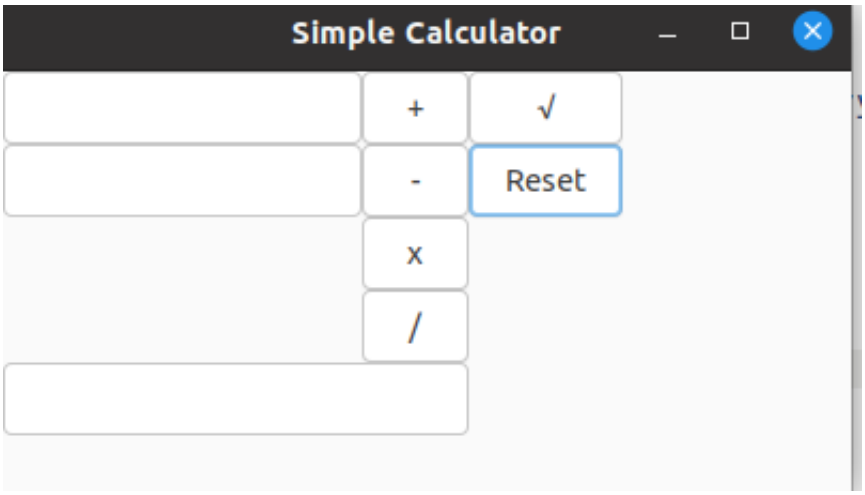
Ce programme permet de calculer l'aire et le périmetre d'un rectangle. Dans celui-ci, on peut entrer une largeur et une longueur. Après avoir cliqué sur 'calculer', un pop-up apparait et donne l'aire et le périmetre.

MonPG5



Il s'agit d'un programme qui permet de calculer l'Indice de Masse Corporelle (IMC) d'un individu. En saisissant leur poids et leur taille, les utilisateurs obtiennent instantanément leur IMC, accompagné d'une indication claire sur leur catégorie de poids, allant de l'anorexie à l'obésité morbide.

MonPG6



Cette calculatrice, conçue pour la simplicité, permet d'effectuer des opérations élémentaires et inclut une fonctionnalité de réinitialisation pour commencer de nouveaux calculs facilement.
Le resultat est inscrit dans l'output en bas.

Le virus

Un virus, c'est quoi ?

Un virus informatique est un programme dissimulé à l'intérieur d'un autre programme. Il est conçu pour infiltrer votre machine en se dupliquant dans des fichiers cibles ou des zones spécifiques. Il agit de manière discrète et son objectif final est de déclencher une nuisance, bien que cela ne se produise pas systématiquement.

Mécanismes d'attaque d'un virus

En règle générale, un virus suit un ensemble d'étapes bien définies.

Initialement, il va chercher à **pénétrer** dans une machine. Pour cela, il a besoin que son programme hôte soit exécuté. Ainsi, il adopte souvent une stratégie d'appât, cherchant à séduire l'utilisateur en se présentant de manière attrayante. Cette phase marque le début de l'infection.

Ensuite, il va chercher à **se propager**. Pour ce faire, il va passer par plusieurs étapes. Tout d'abord, il va **chercher les fichiers cibles**, généralement, des fichiers exécutables. Après ça, il va chercher à **se reproduire**, c'est à dire qu'il va essayer d'infecter d'autres programmes (il existe plusieurs méthodes, par exemple, par recouvrement ou bien par virus compagnon), tout en évitant d'infecter ceux qu'il a déjà infecté. Par la suite, il va chercher à **ne pas se faire détecter**. Il va faire en sorte

de préserver les fonctions du programme hôte et plus largement à utiliser des méthodes de polymorphisme, notamment l'obfuscation de code.

Pour finir, il va **déclencher une nuisance**, comme par exemple, des ralentissements ou encore des pertes de données.

Visualisateur d'images

Avant d'élaborer le virus, nous avons développé la fonctionnalité du programme permettant de rendre service à l'utilisateur, pour éviter des soupçons de sa part. La fonction "normalement" principale de MediaPlayer.exe est d'être un visualisateur d'images permettant à l'utilisateur de parcourir et de visionner les images disponibles. Pour implémenter cette fonctionnalité, nous avons opté pour l'utilisation de la bibliothèque SDL.

Actuellement, notre visualisateur d'images reste rudimentaire : la qualité des images n'est pas exceptionnelle et seules les six images du répertoire peuvent s'afficher. De plus, la navigation entre les images se fait uniquement à l'aide des touches de direction du clavier. Cependant, ces fonctionnalités de base sont amplement suffisantes pour que l'utilisateur ne se doute pas de la présence éventuelle d'un virus.

Nous l'avons installé à l'aide des commandes :

```
sudo apt-get update
sudo apt-get install libsdl2-dev
sudo apt-get install libsdl2-image-dev
```

Cas du virus compagnon

Dans le contexte du virus compagnon, explorons ses différentes étapes.

Pénétration

Rappellons que le programme MediaPlayer.exe est accompagné d'images.
La présence de ces images incite l'utilisateur à exécuter le programme pour les visualiser, amorçant ainsi la première phase d'infection.

Propagation

Fichiers cibles

Le virus commence par repérer les programmes exécutables et réguliers, à savoir nos utilitaires. Cette tâche est effectuée par la fonction `recherche` dans notre code. Elle parcourt le répertoire courant et récupère les fichiers exécutables réguliers autre que MediaPLayer.exe.

Reproduction

Le virus prend le nom de ces programmes. Pour se faire, il va d'abord s'assurer de ne pas attaquer un fichier déjà infecté, c'est la lutte contre la sur-infection, gérée par la fonction `lutteSurinfection` dans notre code.

Si le programme n'est pas déjà infecté, alors il le contamine. Ce processus implique le renommage du programme avec l'extension ".old" suivi de sa duplication en une copie portant le nom du programme (sans l'extension .old). Il est essentiel que cette copie soit exécutable, condition nécessaire pour éviter toute détection, gérée par la fonction `infection` dans notre code.

Double vérification de l'infection

La vérification de l'infection d'un programme doit être double. D'abord, le virus vérifie si le fichier n'a pas déjà l'extension .old, indiquant qu'il est déjà infecté. Ensuite, il examine si un autre fichier portant le même nom dans le répertoire courant possède l'extension .old. Cette double vérification est cruciale pour éviter d'écraser le programme d'origine, ce qui pourrait éveiller les soupçons de l'utilisateur.

Assurer l'exécution de la copie

Lors de la duplication du virus, il est impératif que la copie soit exécutable. Si cette permission manque, l'utilisateur ne pourrait pas exécuter son utilitaire, ce qui pourrait conduire à la détection du virus.

Cependant, à notre avis personnel, nous ne comprenons pas vraiment cette vérification. Lorsqu'un fichier exécutable est dupliqué, la copie hérite automatiquement des mêmes permissions que le fichier d'origine. Par conséquent, nous trouvons ça étrange de vérifier explicitement si la copie est exécutable, puisqu'elle devrait l'être par défaut.

Amplification de l'infection

Grâce à l'étape d'infection qui génère de nouveaux fichiers infectés, chaque fichier contaminé devient un nouveau point d'accès potentiel pour le virus. Plus le nombre de fichiers infectés augmente, plus les chances d'exécution du virus augmentent

également, ce qui accroît les possibilités de propagation et amplifie l'infection.

De plus, en infectant des fichiers cibles non encore infectés, le virus peut se propager à d'autres utilisateurs. Cela peut conduire à une propagation virale à grande échelle si ces utilisateurs partagent les fichiers infectés.

Déclenchement d'une nuisance

Après s'être proprement dupliqué en évitant la sur-infection, le virus procède à sa tâche principale : déclencher une nuisance. Bien que nous n'ayons pas implémenté cette fonctionnalité dans notre projet, une fonction de "nuisance" est présente dans notre code, illustrée par la fonction destruction qui affiche simplement un message de nuisance dans la console.

Transfert d'exécution

Pour finir, le virus n'a plus qu'à transférer l'exécution au programme hôte. Si le programme demandé par l'utilisateur est MediaPLayer.exe, alors il n'est pas nécessaire de transférer le contrôle à un autre programme. Cependant, s'il demande un autre programme, il est alors nécessaire de transférer l'exécution au programme demandé.

Si le virus ne transfert pas l'exécution au programme demandé, alors l'utilisateur n'aura pas accès à son programme (dans notre cas, son utilitaire) et va donc détecter le virus.

Spécificités

GTK et SDL

Nous avons opté pour l'utilisation simultanée des bibliothèques GTK et SDL afin de montrer que notre virus est capable de fonctionner avec des programmes exploitant des bibliothèques différentes.

Particularités de MediaPlayer.exe

```
int main(int argc, char* argv[]) {
    nomFichierActuel = argv[0];
    recherche();
    destruction();
    if (!strcmp(nomFichierActuel, "./MediaPlayer.exe")){
        afficherImages();
```

```
    }
    else {
        char commande[100];
        sprintf(commande, "%s.old", nomFichierActuel);
        system(commande);
    }
    return 0;
}
```

Dans le contexte du virus, une logique particulière est appliquée dans la fonction principale (main).

Si le programme en cours d'exécution est MediaPlayer.exe, alors le virus lance la fonction d'affichage des images.
Si le programme en cours d'exécution n'est pas MediaPlayer.exe, le contrôle est transféré au programme d'origine en renommant le fichier actuel avec l'extension ".old".

Compilation

MediaPlayer.exe

```
gcc -Wall MediaPlayer.c -o MediaPlayer.exe -lSDL2 -lSDL2_image -lm
```

Utilitaires

```
gcc -Wall MonPG1.c -o MonPG1 `pkg-config --cflags --libs gtk+-3.0`
gcc -Wall MonPG2.c -o MonPG2 `pkg-config --cflags --libs gtk+-3.0`
gcc -Wall MonPG3.c -o MonPG3 `pkg-config --cflags --libs gtk+-3.0`
gcc -Wall MonPG4.c -o MonPG4 `pkg-config --cflags --libs gtk+-3.0`
gcc -Wall MonPG5.c -o MonPG5 `pkg-config --cflags --libs gtk+-3.0`
gcc -Wall MonPG6.c -o MonPG6 `pkg-config --cflags --libs gtk+-3.0`
```