

---

## 1. Introduction

Afin de tester notre programme, nous vous fournissons deux fichiers :

- `antivirus.sh` : Le script principal contenant l'intégralité du programme respectant le cahier des charges.
- `prepare_test.sh` : Un programme d'environnement qui permet de générer automatiquement une arborescence et des fichiers (sains et corrompus) pour effectuer vos tests sans création manuelle.

## 2. Détails des fonctions

Voici les fonctionnalités disponibles dans le script principal que vous pourrez tester :

- **Fonction 1 (Chercher)** : Recherche les fichiers correspondant au critère (extension du virus) dans un répertoire donné.
- **Fonction 2 (Déplacer)** : Déplace les fichiers identifiés comme virus vers un dossier de quarantaine.
- **Fonction 3 (Annoter)** : Renomme les fichiers infectés en ajoutant un suffixe (ex: `_virus`).
- **Fonction 4 (Compter)** : Affiche le nombre de virus détectés dans l'historique.
- **Fonction 5 (Afficher contenu)** : Affiche le contenu complet d'un fichier choisi.
- **Fonction 6 (Historique)** : Affiche le contenu du fichier `historique.txt`.
- **Fonction 7 & 8 (Head/Tail)** : Affiche respectivement les X premières ou X dernières lignes d'un fichier.

## 3. Programme d'environnement

Pour éviter de créer les dossiers à la main, il vous est fourni un programme `prepare_test.sh`.

### Installation

1. Créez le fichier sur votre machine : `nano prepare_test.sh`
2. Copiez le contenu fourni.
3. Ajoutez les droits d'exécution : `chmod 700 prepare_test.sh`

### Utilisation de l'environnement

Lancez le script :

Bash

```
./prepare_test.sh
```

Sélectionnez l'option "1. Créer les fichiers test".

Cela va générer un répertoire dossier\_test contenant :

- fichier2.sha (Virus)
- Dossier1/fichier4.sha (Virus dans un sous-dossier)
- Des fichiers textes inoffensifs ( .txt , .sh ).

Une fois l'environnement créé, quittez ce script pour lancer l'antivirus.

## 4. Lancement du test (Script Intégral)

Le programme principal se nomme antivirus.sh . Il nécessite **un argument** au lancement : le chemin d'un fichier contenant l'extension du virus à chercher.

### Prérequis

Créez un fichier contenant l'extension du virus (le critère) :

Bash

```
echo ".sha" > critere.txt
```

### Exécution

Lancez le script avec le fichier critère en paramètre :

Bash

```
./antivirus.sh critere.txt
```

## 5. Vérification test manuel

Afin de vérifier l'efficacité des fonctions, un retour sous forme d'echo vous sera fait dans le terminal2.

Vous pouvez également ouvrir un second terminal pour effectuer des commandes de vérification (ls, cat, tree) et voir si les modifications ont bien été prises en compte3.

### Test Fonction 1 : Chercher les fichiers corrompus

**Description :** Cette fonction cherche tous les fichiers correspondant au critère ( .sha ) dans un répertoire donné et enregistre le résultat dans `historique.txt` .

- **Protocole :**

1. Dans le menu, choisissez l'option **1**.
2. À la demande du chemin, entrez : `dossier_test` .

- **Vérification (Test manuel) :**

- Le script doit afficher "Voici les virus détectés..." .
- Ouvrez un autre terminal et vérifiez la création du fichier historique : Bash

```
cat historique.txt
```

- **Résultat attendu :** Le fichier doit contenir les chemins vers `fichier2.sha` et `Dossier1/fichier4.sha` .

---

## Test Fonction 2 : Déplacer les fichiers corrompus

**Description :** Cette fonction déplace les fichiers identifiés comme virus d'un dossier source vers un dossier de destination.

- **Préparation :** Créez un dossier de quarantaine : `mkdir quarantaine` .

- **Protocole :**

1. Dans le menu, choisissez l'option **2**.
2. Dossier source : `dossier_test` .
3. Dossier destination : `quarantaine` .

- **Vérification (Test manuel) :**

- Le script doit indiquer "OK : ... déplacé".
- Vérifiez le déplacement : Bash

```
ls quarantaine
```

- **Résultat attendu :** Le dossier `quarantaine` doit contenir `fichier2.sha` et `fichier4.sha` . Le dossier `dossier_test` ne doit plus les contenir.

---

## Test Fonction 3 : Annoter les fichiers corrompus

**Description :** Renomme les fichiers infectés en ajoutant le suffixe `_virus` .

- **Préparation (Reset)** : Si vous avez fait le test 2, les virus ne sont plus là. Quittez l'antivirus, relancez `./prepare_test.sh`, choisissez "**Réinitialiser**" (2), puis relancez l'antivirus.
  - **Protocole** :
    1. Dans le menu, choisissez l'option **3**.
    2. Dossier à traiter : `dossier_test` .
  - **Vérification (Test manuel)** :
    - Vérifiez le changement de nom : Bash

```
ls -R dossier_test
```
    - **Résultat attendu** : Vous devriez voir `fichier2.sha_virus` au lieu de `fichier2.sha` .
- 

## Test Fonction 4 : Compter les fichiers corrompus

**Description** : Compte et affiche le nombre de lignes présentes dans le fichier `historique.txt` .

- **Préparation** : Assurez-vous d'avoir lancé la **Fonction 1** au préalable pour générer l'historique.
  - **Protocole** :
    1. Dans le menu, choisissez l'option **4**.
  - **Vérification (Test manuel)** :
    - **Résultat attendu** : Le script doit afficher le nombre **2** (correspondant aux deux fichiers virus créés par le script de préparation).
- 

## Test Fonction 5 : Afficher contenu fichier

**Description** : Affiche le contenu complet d'un fichier spécifié par l'utilisateur.

- **Protocole** :
    1. Dans le menu, choisissez l'option **5**.
    2. Fichier à afficher : `dossier_test/fichier1.txt` .
  - **Vérification (Test manuel)** :
    - **Résultat attendu** : Le terminal doit afficher : "Ceci est le contenu du fichier1.txt".
- 

## Test Fonction 6 : Historique des corrompus

**Description :** Affiche le contenu du fichier `historique.txt` généré lors de la recherche.

- **Protocole :**
    1. Dans le menu, choisissez l'option **6**.
  - **Vérification (Test manuel) :**
    - **Résultat attendu :** L'affichage doit être identique au résultat de la commande `cat historique.txt` effectuée lors du Test 1.
- 

## Test Fonction 7 : Afficher X premières lignes (Head)

**Description :** Affiche les premières lignes d'un fichier.

- **Protocole :**
    1. Dans le menu, choisissez l'option **7**.
    2. Fichier : `antivirus.sh` (Testons sur le script lui-même pour avoir assez de lignes).
    3. Nombre de lignes : `3` .
  - **Vérification (Test manuel) :**
    - **Résultat attendu :** Le terminal doit afficher uniquement les 3 premières lignes du script (le shebang et l'en-tête de commentaire).
- 

## Test Fonction 8 : Afficher X dernières lignes (Tail)

**Description :** Affiche les dernières lignes d'un fichier.

- **Protocole :**
  1. Dans le menu, choisissez l'option **8**.
  2. Fichier : `antivirus.sh` .
  3. Nombre de lignes : `2` .
- **Vérification (Test manuel) :**
  - **Résultat attendu :** Le terminal doit afficher les deux dernières lignes du script (fermeture du `case` et du `done` ).