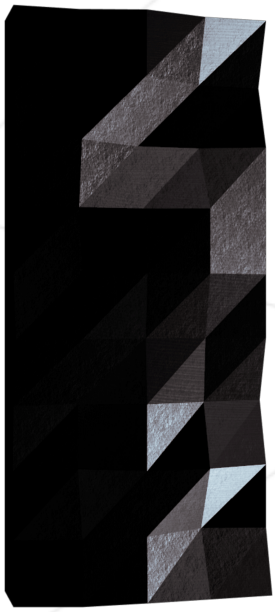

Bootcamp Cybersecurité | 42

ft_otp

Résumé: Rien ne dure jamais éternellement...

Version: 1

Table des matières

Chapitre I

Introduction

Les mots de passe sont `un des plus gros casse-tête de la sécurité informatique`
Les utilisateurs les oublient, les partagent, les réutilisent et les choisissent horriblement mal. En outre, les mots de passe font tôt ou tard l'objet de fuites lors de violations de la sécurité. Une façon d'éviter cela est d'utiliser des mots de passe à usage unique, basés sur des horodateurs, qui expirent après quelques minutes et deviennent ensuite invalides. Que vous utilisiez déjà ce système ou que vous n'en ayez jamais entendu parler, il est fort probable que l'un de vos mots de passe ait été compromis à un moment ou à un autre de votre vie.

Dans ce projet, l'objectif est de mettre en œuvre un système TOTP (Time-based One-Time Password), qui sera capable de générer des mots de passe éphémères à partir d'une clé maîtresse.
Il sera basé sur le `RFC` : https://datatracker.ietf.org/doc/html/rfc6238, vous pourrez donc l'utiliser dans votre vie quotidienne.

Chapitre II

Prologue

La route de la soie s'étendait sur tout le continent asiatique, reliant la Chine à la Mongolie, la Perse, l'Inde, le Moyen-Orient, la Turquie, l'Europe et l'Afrique. Malgré son nom, ce n'est pas le précieux tissu qui était principalement échangé. Le verre, le cuir, les armes ou les machines de guerre voyageaient à travers le monde, développant les découvertes industrielles, les techniques d'impression, la poudre à canon ou la boussole.

Password, please ?

Chapitre III

Partie obligatoire

Dans le langage de votre choix, vous devez implémenter un programme qui permet d'enregistrer un mot de passe initial, et qui est capable de générer un nouveau mot de passe chaque fois qu'il est demandé. Tu peux utiliser n'importe quelle bibliothèque qui facilite l'implémentation de l'algorithme, à condition qu'elle ne fasse pas le sale boulot, c'est-à-dire qu'il est strictement interdit d'utiliser une bibliothèque TOTP. Bien entendu, vous pouvez et devez utiliser une bibliothèque ou une fonction qui vous permet d'accéder au temps système.

Voici un exemple d'utilisation du programme :

- Le programme doit s'appeler `ft_otp`
- Avec l'option `-g`, le programme recevra en argument une clé hexadécimale d'au moins 64 caractères. Le programme stockera cette clé en toute sécurité dans un fichier appelé `ft_otp.key`, qui sera crypté.
- Avec l'option `-k`, le programme génère un nouveau mot de passe temporaire et l'imprime sur la sortie standard.

```
$ echo -n "NEVER GONNA GIVE YOU UP" > key.txt
$ ./ft_otp -g key.txt
./ft_otp: error: key must be 64 hexadecimal characters.
$ xxd -p key.txt > key.hex
$ cat key.hex | wc -c
64
$ ./ft_otp -g key.hex
Key was successfully saved in ft_otp.key.
$ ./ft_otp -k ft_otp.key
836492
$ sleep 60
$ ./ft_otp -k ft_otp.key
123518
```

Vous pouvez vérifier si votre programme fonctionne correctement en comparant les mots de passe générés avec `Oathtool` ou tout autre outil de votre choix.

```
oathtool -totp $(cat key.hex)
```

Chapitre IV

Partie bonus

L'évaluation des bonus se fera `SI ET SEULEMENT SI` la partie obligatoire est `PARFAITE`. Dans le cas contraire, les bonus seront totalement `IGNORÉES`.

Vous pouvez améliorer votre projet grâce aux fonctionnalités suivantes :

- Permet de choisir le mot de passe de cryptage de la clé principale ft_otp.key et de le demander à chaque fois qu'un nouveau mot de passe est généré.
- Développez un client qui génère le mot de passe principal et valide les résultats avec une interface graphique.
- Toute autre caractéristique que vous jugez réellement utile. Vos pairs jugeront si elles le sont.

Chapitre V

Évaluations peer 2 peer

Ce projet sera corrigé par d'autres étudiants. Remettez les fichiers dans le dépôt Git et assurez-vous que tout fonctionne comme prévu.