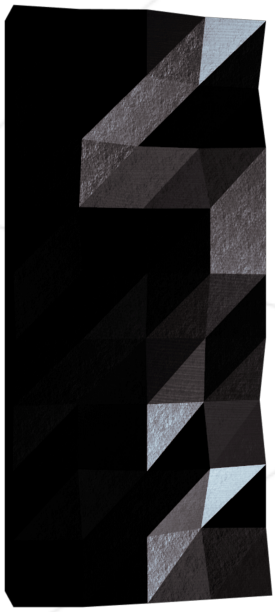

Bootcamp Cybersecurité | 42

ft_blockchain

Résumé: Chiffrement et décentralisation

Version: 1

Table des matières

Chapitre I

Introduction

L'objectif de ce projet est de créer une blockchain basée sur un algorithme (`Proof of work`). Pour ce faire, vous devrez implémenter la logique de la chaîne de blocs, ainsi qu'un serveur par lequel vous pourrez interagir avec elle.

Chapitre II

Prologue

Une version purement peer-to-peer de la monnaie électronique permettrait d'envoyer des paiements en ligne directement d'une partie à une autre sans passer par une institution financière. Les signatures numériques apportent une partie de la solution, mais les principaux avantages sont perdus si un tiers de confiance est toujours nécessaire pour empêcher le double paiement. Nous proposons une solution au problème de double dépense en utilisant un réseau peer-to-peer. Le réseau horodate les transactions en les hachant dans une chaîne continue de preuves de travail basées sur le hachage, formant un enregistrement qui ne peut être modifié sans refaire la preuve de travail. La chaîne la plus longue sert non seulement de preuve de la séquence d'événements observés, mais aussi de preuve qu'elle provient du plus grand pool de puissance CPU. Tant que la majorité de la puissance CPU est contrôlée par des nœuds qui ne coopèrent pas pour attaquer le réseau, ils génèrent la plus longue chaîne et devancent les attaquants. Le réseau lui-même nécessite une structure minimale. Les messages sont diffusés au mieux, et les nœuds peuvent quitter et rejoindre le réseau à volonté, en acceptant la plus longue chaîne de preuve de travail comme preuve de ce qui s'est passé pendant leur absence.

Satoshi Nakamoto, 2012

`0x000000000019d6689c085ae165831e934ff763ae46a2a6c172b3f1b60a8ce26f`

Chapitre III

Instructions générales

Pour ce projet, vous pouvez utiliser n'importe quel langage de programmation. Vous pouvez utiliser des bibliothèques cryptographiques comme `openssl` ou `hashlib` pour générer des `hashes`, mais la structure de la blockchain doit être implémentée par vos soins. De la même manière, un framework web comme `NestJS` ou `Flask` peut être utilisé pour l'implémentation du serveur.

```
block =
{
    'index': 4,
    'timestamp': 1644045050.00042,
    'transactions': [
        {
            'sender': '4c6e7e2a9f2f7f7ff8e7d3d6c8b7c6e8e23a7',
            'recipient': 'b3c6e7e2a9f2f7f7ff8e7d3d6c8b7c6e8e23a7',
            'amount': 42,
        }
    ],
    'proof': 324984774000,
    'previous_hash':
        '084c799cd551dd1d8d5c5f9a5d593b2e931f5e36122ee5c793c1d08a19839cc0',
}
```

Exemple de bloc. Le hachage du bloc ci-dessus a été généré en utilisant l'algorithme SHA-256.

Chapitre IV

Partie obligatoire

Le flux de travail consiste à ajouter différentes transactions au bloc actuel et à extraire le bloc afin d'ajouter la chaîne.

L'algorithme de la `preuve-de-travail` doit être simple, par exemple, trouver le nombre qui, concaténé avec la preuve de travail précédente, correspond au résultat du hachage SHA-256 se terminant par `4242`. La chaîne de blocs ne sera pas persistante, elle sera stockée dans la mémoire du serveur mais le serveur ne sera pas connecté à un logiciel de base de données spécifique. Lors du développement du minage, trois choses doivent être faites :

- Calculer la preuve de travail
- Récompenser les mineurs (une transaction)
- Création du nouveau bloc et ajout à la chaîne

Une fois la blockchain créée, vous pouvez interagir avec elle par le biais de différentes requêtes HTTP sur une API textuelle :

- `[POST] /transactions/new` : Enregistre une nouvelle transaction à ajouter au bloc suivant.
- `[GET] /mine` : Exécuter la preuve de travail et créer un nouveau bloc.
- `[GET] /chain` : Returns information about the full blockchain (blocks, transactions, etc).

Chapitre V

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 :

- Difficulté de l'algorithme PoW dynamique, croissant en fonction du nombre de blocs minés ou du temps écoulé.
- Mise en œuvre de la communication avec les autres nœuds du réseau par le biais d'un réseau décentralisé et d'un algorithme de consensus pour vérifier la chaîne correcte.
- Proof of Stake en plus de Proof of Work, comme alternative écologique, algorithme de consensus.
- Tout ce qui vous vient à l'esprit... vous pourrez tout justifier lors de la défense.

Chapitre VI

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.