

Evaluation en Systèmes Distribués

Implémentation d'une blockchain basée sur les micro-services avec Spring cloud et et un Broker KAFKA

Durée : 3H | Documents autorisés

Prof. Mohamed YOUSSEFI : m.youssefi@enset-media.ac.ma

On souhaite créer une application distribuée basée sur les micro-services qui permet d'implémenter une plateforme de Blockchain en utilisant Spring Cloud et le Broker KAFKA. Ce projet sera réalisé en plusieurs parties.

Une blockchain est formée par un ensemble de blocks dont chacun est lié au block précédent dans la chaîne. Chaque block est formé par un ensemble de transactions.

Une transaction est définie par son id (String), sa date, adresse source, adresse destination et un montant.

Un block est défini par son id (String), sa date de création, le hash du block, le hash du block précédent, une liste de transactions et nonce (un entier utilisé au moment du minage des blocks).

Une blockchain est définie par son id (String), son nom, la difficulté (entier utilisé au moment du minage), miningReward (récompense attribuée au mineur du block).

Le service de gestion des blocks définit les opérations suivantes :

- Créer un nouveau block en définissant sa date de création à la date système et en calculant son hash en utilisant l'algorithme HSA256.
- Une opération qui permet de calculer le hash du block
- Une opération qui permet de miner un block avec une difficulté. Cette opération consiste à calculer dans une boucle le hash du block qui comment par un nombre de zéro équivalent au nombre représentant la difficulté. Pour chaque itération de la boucle on incrémente la propriété nonce. Cette opération est utilisée par la blockchain pour définir en combien de temps on souhaite qu'un block sera miné.

Le service de gestion de la blockchain définit une liste de transactions en cours (pendingTransactions), pour stocker en mémoire un lot transaction avant de les ajouter à un nouveau block. Ce service doit définir les opérations suivantes :

- Ajouter une blockchain en lui ajoutant un premier block appelé GenesisBlock
- Miner un block contenant les transactions courantes en lui associant le hash du block précédent (Last block). Cette opération reçoit en paramètre l'adresse du mineur. Une fois les transactions courantes sont minées, cette opération ajoute une transaction courante bonus à l'adresse du mineur.
- Une opération qui permet de retourner le dernier block ajouté à la blockchain
- Une opération qui permet vérifier si la blockchain est valide ou non. Une blockchain est valide si, pour chaque block

-
- Le hash recalculé du bloc correspond bien au hash calculé au moment de sa création
 - Le hash du block précédent n'a pas changé.
 - Une opération qui permet de calculer le solde d'une adresse.

Première partie : Micro-Service

Créer le micro-service blockchain-service permettant d'implémenter les bases de la blockchain

1. Créer la couche DAO
 - a. Créer les entités JPA
 - b. Créer les interfaces basées sur Spring Data
 - c. Tester la couche DAO
2. Créer la couche service
 - a. Créer l'interface BlockService et son implémentation
 - b. Créer l'interface Blockchain service et son implémentation
 - c. Tester la couche service
3. Créer la couche WEB
 - a. Créer les Rest Controllers
 - b. Tester les API REST
4. Sécuriser le Micro-service en utilisant Keycloak

Deuxième partie : Distribuer la blockchain

1. Mettre en place un cluster de broker KAFKA
2. Mettre en place une solution qui permet de synchroniser plusieurs instances du micro-service Blockchain de façon que, pour chaque block miné, toutes les instances doivent miner le même block de sorte que la blockchain est dupliquée sur toutes les instances avec une synchronisation parfaite qui garantit l'impossibilité de modifier les blocks.
3. Développer une solution de Real time Stream procession en utilisant KAFKA Streams permettant de fournir en temps réel de Data Analytics.