1INDF activité 5 : preuve d'enjeu (PoS, proof of stake)

Nom:	Groupe:	Date:

Le but de la troisième activité est de comprendre le consensus basé sur la preuve d'enjeu (proof of stake, PoS) sur une chaîne de blocs.

Vocabulaire

La principale critique concernant les chaîne de blocs basées sur la preuve de travail (PoW) et leur utilisation d'énergie électrique démesurée. Certaines chaînes de blocs optent pour le consensus basé sur la preuve d'enjeu (PoS).

Dans ce type de chaînes, les nœuds n'ont plus besoin d'effectuer un calcul mathématique pour qu'un nouveau bloc soit inséré dans la chaîne : les mineurs s'appellent maintenant **validateurs**. Le nœud/validateur qui validera le prochain bloc est tiré au hasard parmi tous les blocs qui mettent de côté une partie de leurs avoirs en cryptomonnaie. La chance de remporter la loterie est proportionnelle au montant gelé par le nœud. Notons que le montant gelé n'est plus disponible pour effectuer des transactions.

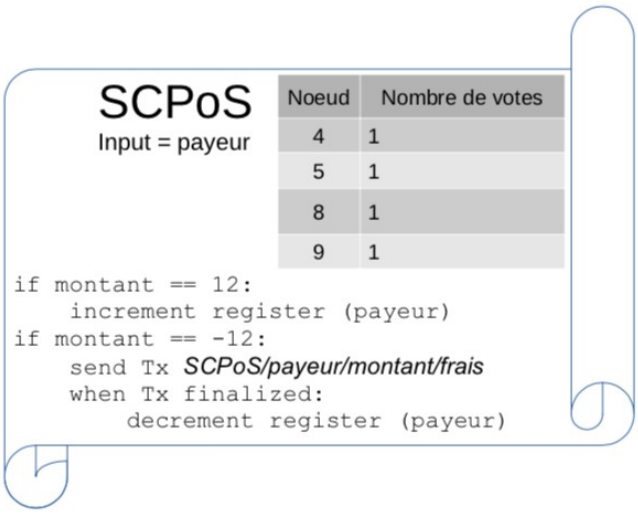
Les PoS sont insérées dans la chaîne de blocs grâce à des contrats intelligents spécialisés.

Si un validateur ne fait pas son travail, n'est pas en ligne, ne détecte pas un bloc invalide, ou laisse délibérément un bloc invalide dans la chaîne de blocs, il perd une partie ou la totalité de la cryptomonnaie gelée dans sa PoS.

Création de PoS

Dans cette activité, les validateurs peuvent augmenter leur nombre de droits de vote en mettant de côté une garantie, fixée à 12 R$ par vote supplémentaire. Les garanties des nœuds sont gérées par le contrat intelligent « SCPoS », qui contient les champs suivants :

- l'adresse du payeur / validateur
- le registre du nombre de votes de chaque validateur
- le code qui effectue automatiquement les tâches suivantes :
    1. incrémenter le registre du nombre de votes ;
    2. décrémenter le nombre de votes si le nœud souhaite récupérer sa garantie.



Par exemple, si le validateur 4 souhaite augmenter son nombre de votes, il soumet la transaction « 4_SCPoS_12_3 » :

- le payeur est le validateur 4
- le destinataire est le SCPoS
- le montant est de 12 R$
- le validateur décide des frais de transaction, ici de 3 R$.

Cette transaction déclenche automatiquement le code du SCPoS, qui augmente le nombre de vote pour le validateur 4 dans le registre.

Transaction:
4_SCPoS_12_3

SCPoS
Input = payeur

Noeud	Nombre de votes
4	~~1~~ 2
5	1
8	1
9	1

```
if montant == 12:
    increment register (4)
if montant == -12:
    send Tx SCPoS/payeur/montant/frais
    when Tx finalized:
        decrement register (payeur)
```

Pour récupérer la garantie associée à un droit de vote, le validateur doit soumettre une transaction avec comme destinataire le SCPoS, qui lui-même soumettra une deuxième transaction avec comme destinataire le validateur afin de lui rembourser la garantie. Le validateur doit donc prévoir deux frais de transaction :

- des frais à son choix pour soumettre la transaction avec le SCPoS comme destinataire,
- des frais de 3 R$ sont fixés pour la transaction que le SCPoS soumettra pour lui rembourser la garantie.

Par exemple, si le validateur 8 souhaite récupérer la garantie associée à un de ses droits de vote, il soumet la transaction « 8_SCPoS_-12_3 » :

- le payeur est le validateur 8
- le destinataire est le SCPoS
- le montant est de -12 R$
- le validateur décide des frais de transaction.

Cette transaction déclenche automatiquement le code du SCPoS, qui soumet la transaction « SCPoS_8_12_3 » :

- Le payeur est le SCPoS
- le destinataire est le validateur 8
- le montant est de 12 R$
- les frais de transaction sont fixés à 3 R$.

Transaction:
8_SCPoS_-12_3

SCPoS
Input = payeur

Noeud	Nombre de votes
4	~~1~~ 2
5	~~1~~ 2 3
8	~~1~~ 2 3 4
9	~~1~~ 2

```
if montant == 12:
    increment register (payeur)
if montant == -12:
    send Tx SCPoS_8_12_3
    when Tx finalized:
        decrement register (payeur)
```

Lorsque la transaction « SCPoS_8_-12_3 » est finalisée, le validateur 8 a récupéré les 12 R$ de sa garantie (mais a dû payer 6 R$ en frais de transaction au passage), et le registre est mis à jour : le nombre de votes du validateur 8 passe de 3 à 2.

Transaction:
8_SCPoS_-12_3

SCPoS
Input = payeur

Noeud	Nombre de votes
4	~~1~~ 2
5	~~1~~ ~~2~~ 3
8	~~1~~ ~~2~~ **3**
9	~~1~~ 2

```
if montant == 12:
    increment register (payeur)
if montant == -12:
    send Tx SCPoS_8_12_3
    when Tx finalized:
        decrement register (payeur)
```

Matériel

- le matériel de l'activité 3 (tableau d'affichage de la chaîne de blocs, mempool)
- un tableau d'affichage « machine virtuelle » où est affiché le SCPoS et son registre
- une « urne de tous les validateurs » contenant un jeton par droit de vote, chacun identifié au validateur.

Déroulement

But : valider le plus de blocs possible.

1. Au début de l'activité, chaque nœud/validateur possède un droit de vote.
2. Les nœuds et les validateurs gèrent les transactions comme dans l'activité 3, mais cette fois, les validateurs ne calculent pas de PoW : le validateur du prochain bloc est tiré au hasard dans l'urne.
3. Le validateur insère dans le bloc la transaction de récompense, et 0, 1 ou 2 transaction(s) choisies dans la mempool, puis enchaîne le nouveau bloc au bloc-parent de son choix.
4. Les autres validateurs sont tenus de vérifier la validité des transactions insérées dans le bloc. Si un validateur a inséré une transaction non-valide, il doit payer une amende de 5 R$ et son bloc en entier est éliminé de la chaîne, même si certaines transactions sont valides.
5. En plus des transactions d'échange de R$ (comme dans l'activité 3), les nœuds/validateurs peuvent soumettre des transactions avec le SCPoS pour gagner des droits de vote, ou récupérer les garanties associées à des droits de vote.

Attention : les R$ mis en gage ne sont plus disponibles pour les transactions.

Pour gagner des droits de vote, une seule transaction est nécessaire, mais pour récupérer une garantie, deux transactions sont nécessaires :

- une première avec le SCPoS comme destinataire,
- puis lorsque la première transaction est finalisée, une deuxième transaction de remboursement de garantie doit être soumise.

6. À la fin de l'activité, le nombre de blocs validés par chaque validateur est compté. Le validateur ayant validé le plus de blocs l'emporte !

Dans la vraie vie...

Les jetons mis en garantie font chacun l'objet d'un contrat intelligent, avec sa propre paire de (clé, solde).

La récompense pour la validation d'un bloc est calculée selon la participation du validateur à la chaîne de blocs ; elle n'est pas fixe, comme c'était le cas avec la PoW.