Faculté des Sciences & Techniques — Université de Limoges

Licence 3ème année

Initiation Sécurité

TP n°4

Blockchain

Construction de « *blockchain* »

Qu'est ce qu'une « *blockchain* » ?

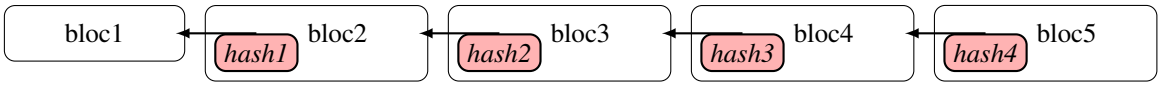
C'est une forme de cahier de compte contenant un ensemble de transactions :
- ▷ **distribué** : il peut être partagé, dupliqué et synchronisé ;
- ▷ **modifié par consensus** : sa modification doit être faite en accord avec tous ceux l'utilisant ;
- ▷ constitué de blocs de transactions liés entre eux : le nouveau bloc de transactions ajouté est **lié** au bloc précédent par la **cryptographie** empêchant toute modification du précédent bloc auquel il est lié.

L'ensemble de ces blocs liés entre eux est appelé **blockchain**.

La cryptograhie utilisée :
- garantie **l'intégrité** : protection contre les modifications ;
- est une fonction de hashage : c'est une fonction qui construit une **empreinte** de taille fixe à partir de données de tailles quelconque avec la garantie que si les données sont modifiées alors la valeur de l'empreinte sera différente.

Deux données peuvent produire une même empreinte, ce qui s'appelle une collision.
Il est statistiquement exceptionnellement dur de trouver deux documents proches, par exemple avec une signification humaine similaire, produisant une collision.

Chaque bloc de la blockchain est lié au précédent en contenant le hash de ce bloc précédent.

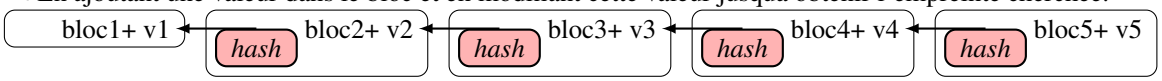


Comment modifier un bloc de la blockchain ?
⟹Il faut modifier l'ensemble des blocs qui le précédent et qui le suivent !

Mais ne pourrait-on pas le faire en créant une blockchain « alternative *» qui remplacerait la première ?*
⟹Il faut rendre difficile l'ajout d'un bloc dans la blockchain pour ralentir sa modification !

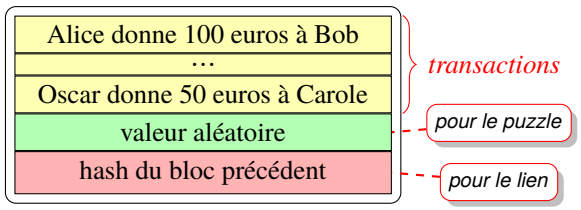
Comment « ralentir *» l'opération d'ajout ?*
⟹En créant un « *puzzle cryptographique* » avec l'empreinte : il faut que l'empreinte calculée présente une particularité comme un ensemble de zéro devant.

Comment modifier l'empreinte pour obtenir ces zéros devant ?
⟹En ajoutant une valeur dans le bloc et en modifiant cette valeur jusqu'à obtenir l'empreinte cherchée.



Combien faut-il de temps pour trouver cet ensemble de zéros ? Combien en faut-il ?
⟹Cela dépend de la puissance des ordinateurs.
⟹Pour garder le « *puzzle cryptographique* », on peut régulièrement au cours des avancées technologiques, augmenter le nombre de zéros voulus.

Quel est le contenu d'un block ?



Chaque bloc contient un nombre quelconques de transactions.
Pour commencer, vous pourrez ne mettre qu'une seule transaction par bloc.

Exemple:

```
#!/bin/bash

DATA='Alice donne 100 euros à Bob'
VALEUR=1

calculmd5()
{
  RESULTAT=$(echo -n $1 | md5sum | cut -d' ' -f1)
}

while
calculmd5 "${DATA}${VALEUR}"
SUCCES=$(echo -n $RESULTAT | grep -o '^0')

if [[ -n "$SUCCES" ]]
then
  echo "SUCCES !"
  echo "$RESULTAT"
  exit 0
fi
VALEUR=$(( VALEUR+1 ))
echo $VALEUR
do true; done
```

On l'exécute :

```
xterm
~/BLOCKCHAIN  ./chercher_lien
2
3
4
...
15
SUCCES !
06461419a03e8ee8ce4154c83cf7007a
```

Si on modifie le script pour chercher deux zéros :

```
...
SUCCES=$(echo -n $RESULTAT | grep -o '^00')
...
```

```
xterm
~/BLOCKCHAIN  ./chercher_lien
2
3
4
...
227
228
229
230
231
SUCCES !
0041177263d2d8db0d899288624a2583
```

Pour 3 zéros :

```
xterm
~/BLOCKCHAIN  ./chercher_lien
1
...
7726
7727
SUCCES !
000a99ecdea15adc9c35e196b087bf28
```

1 – a. Définissez un **format** pour exprimer une transaction entre deux individus.

b. Faites un programme pour enregistrer **plusieurs transactions** en un même bloc de texte.

c. Écrire un programme calculant le **hash d'un bloc de texte** contenant des transactions sous le format défini.

Vous choisirez une fonction de hachage rapide et dont la sécurité est encore prouvée.

d. Écrire un programme qui **résoud le puzzle cryptographique** avec un nombre de zéro choisi et une **valeur** de type entier.
Vous commencerez avec un zéro devant.

e. Évaluez le **temps** pris en fonction :
- ◇ du nombre de zéros demandés ;
- ◇ du contenu du bloc (vous essaierez avec des contenus de bloc différents) ;
- ◇ de la nature de la valeur ajoutée : un nombre entier sur 32bits, sur 64bits ;

Vous calculerez la moyenne sur des blocs différents et la variation (linéaire, exponentielle, ...) pour des nombres de zéros de plus en plus grand.

2 – Écrire le programme :
- ▷ réalisant la **blockchain** en utilisant 3 ou 4 bloc de transactions ;
- ▷ **enregistrant** cette blockchain dans un **fichier** ;
- ▷ **vérifiant** si la blockchain lue depuis le fichier est **correcte**.