nomadic labs

Faire des transactions confidentielles sur Tezos

Sommaire

Introduction

Introduction

La plupart des blockchaines publiques permettent à n'importe qui ayant accès à Internet d'avoir accès à l'intégralité des opérations présentes sur la chaîne.
En outre, n'importe qui peut, en connaissant l'adresse d'un utilisateur (suite à un paiement par exemple) :

- lire la balance de son compte
- voir les transactions qu'il a émises et reçues depuis la création de son compte, leur montant, leurs destinataires et leurs émetteurs.

Ces aspects peuvent être incompatibles [1] avec la protection de la vie privée [2] des utilisateurs, ainsi que le droit à l'oubli [3].

Pour y remédier, Tezos propose des contrats basés sur le protocole **Sapling**. Ce protocole permet de contribuer à la fois **la protection de la vie privée des utilisateurs et à la transparence vis-à-vis des régulateurs**.

1. https://www.eublockchainforum.eu/sites/default/files/reports/20181016_report_gdpr.pdf
2. Concernant ce sujet, en Union Européenne, le R.G.P.D. encadre la protection des personnes physiques à l'égard du traitement des données à caractère personnel et de la libre circulation de ces données.
3. Le droit à l'oubli permet à un individu de demander le retrait d'informations le concernant et pouvant lui nuire.

Définition des principaux concepts

Définition des principaux concepts

- **Transaction protégée (ou shielded) :** Une transaction est dite protégée quand un observateur extérieur ne peut en connaître aucune donnée (notamment le montant, le destinataire et l'expéditeur [1]).
- **Transaction transparente :** Une transaction est dite transparente quand elle n'est pas protégée.
- **Shielded pool :** Une *Shielded pool* est un ensemble de jetons dont on peut connaître le nombre, mais dont les transactions internes sont protégées.
- **Smart-contrat :** Dans une cryptomonnaie, un smart-contrat est un compte autogéré dont les interactions sont paramétrées par un code et des données. À chaque transaction vers l'adresse d'un smart-contrat, son code détermine des modifications de ses données et les éventuelles transactions émises par celui-ci.

1. En pratique un observateur extérieur peut voir l'expéditeur s'il appelle lui même le contrat.

Gestion des clefs

Gestion des clefs

Sapling est un protocole cryptographique qui permet de faire des transactions protégées, tout en permettant de garantir à des tiers de confiance la visualisation des transactions. Il faut pour cela distinguer trois éléments :

- La clef de **dépense**, qui doit être tenue secrète et qui sert à signer les transferts.
- La clef de **visualisation** qui est dérivée de la clef de **dépense** et permet de visualiser les transactions émises ou reçues (il est possible de créer des clefs de visualisation qui permettent de voir uniquement les transactions émises ou les transactions reçues).
- Les adresses qui servent à recevoir des transactions sont toutes dérivées de la clef de **dépense**. On peut dériver autant d'adresses qu'on le souhaite, ainsi deux utilisateurs différents ne pourront pas savoir s'ils font un transfert à la même personne ou non.

Sapling est un protocole cryptographique utilisant des courbes elliptiques (Bls 12-381 et jubjub).

Démonstration

Exemple sans suivi de la transaction par un tiers

Dans cet exemple, Alice veut envoyer 8ꜩ à Bob, sans que quelqu'un d'autre puisse être au courant.

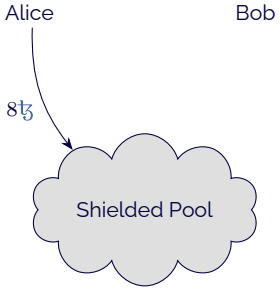
Alice

Bob

Exemple sans suivi de la transaction par un tiers

Dans cet exemple, Alice veut envoyer 8ꜩ à Bob, sans que quelqu'un d'autre puisse être au courant.

1. Alice commence par créer des jetons dans la *shielded pool* en envoyant 8ꜩ au contrat.



Exemple sans suivi de la transaction par un tiers

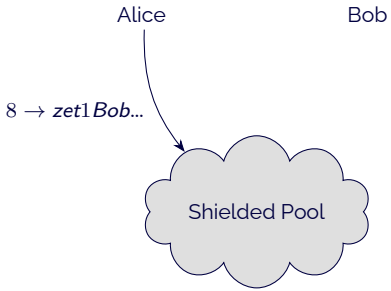
Dans cet exemple, Alice veut envoyer 8ꜩ à Bob, sans que quelqu'un d'autre puisse être au courant.

1. Alice commence par créer des jetons dans la *shielded pool* en envoyant 8ꜩ au contrat.
2. Bob dérive une adresse de sa clef de visualisation et l'envoie à Alice.

Exemple sans suivi de la transaction par un tiers

Dans cet exemple, Alice veut envoyer 8ꜩ à Bob, sans que quelqu'un d'autre puisse être au courant.

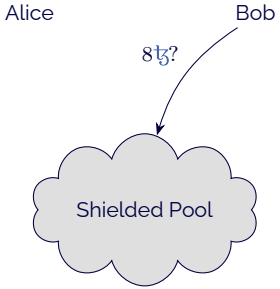
1. Alice commence par créer des jetons dans la *shielded pool* en envoyant 8ꜩ au contrat.
2. Bob dérive une adresse de sa clef de visualisation et l'envoie à Alice.
3. Alice envoie à Bob les jetons au travers de la *shielded pool*.



Exemple sans suivi de la transaction par un tiers

Dans cet exemple, Alice veut envoyer 8ꜩ à Bob, sans que quelqu'un d'autre puisse être au courant.

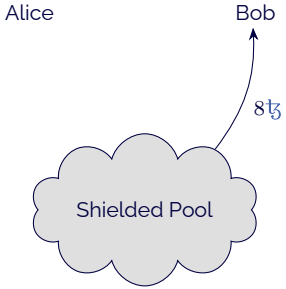
1. Alice commence par créer des jetons dans la *shielded pool* en envoyant 8ꜩ au contrat.
2. Bob dérive une adresse de sa clef de visualisation et l'envoie à Alice.
3. Alice envoie à Bob les jetons au travers de la *shielded pool*.
4. Bob peut alors réclamer les jetons qu'il a reçus.



Exemple sans suivi de la transaction par un tiers

Dans cet exemple, Alice veut envoyer 8ꜩ à Bob, sans que quelqu'un d'autre puisse être au courant.

1. Alice commence par créer des jetons dans la *shielded pool* en envoyant 8ꜩ au contrat.
2. Bob dérive une adresse de sa clef de visualisation et l'envoie à Alice.
3. Alice envoie à Bob les jetons au travers de la *shielded pool*.
4. Bob peut alors réclamer les jetons qu'il a reçus.
5. Le contrat lui envoie alors 8ꜩ et retire de la *shielded pool* les jetons correspondants.



Exemple sans suivi de la transaction par un tiers

Dans cet exemple, Alice veut envoyer 8ꜩ à Bob, sans que quelqu'un d'autre puisse être au courant.

1. Alice commence par créer des jetons dans la *shielded pool* en envoyant 8ꜩ au contrat.
2. Bob dérive une adresse de sa clef de visualisation et l'envoie à Alice.
3. Alice envoie à Bob les jetons au travers de la *shielded pool*.
4. Bob peut alors réclamer les jetons qu'il a reçus.
5. Le contrat lui envoie alors 8ꜩ et retire de la *shielded pool* les jetons correspondants.

Dans ce scénario, quelqu'un pourrait s'apercevoir qu'Alice envoie 8ꜩ et que Bob retire également 8ꜩ. Ceci pourrait permettre de deviner qu'il y a eu une transaction entre les deux surtout si le dépôt et le retrait sont proches temporellement.

L'idéal serait que Bob ne retire pas ses jetons, et qu'il s'en serve pour payer des transactions.

Pour accroître son anonymat, Bob peut faire en sorte de ne jamais interagir avec le contrat pour déposer ou retirer des tez.

Exemple sans suivi de la transaction par un tiers

Dans cet exemple, Alice veut envoyer 8ꜩ à Bob, sans que quelqu'un d'autre puisse être au courant.

1. Alice commence par créer des jetons dans la *shielded pool* en envoyant 8ꜩ au contrat.
2. Bob dérive une adresse de sa clef de visualisation et l'envoie à Alice.
3. Alice envoie à Bob les jetons au travers de la *shielded pool*.
4. Bob peut alors réclamer les jetons qu'il a reçus.
5. Le contrat lui envoie alors 8ꜩ et retire de la *shielded pool* les jetons correspondants.

Dans ce scénario, quelqu'un pourrait s'apercevoir qu'Alice envoie 8ꜩ et que Bob retire également 8ꜩ. Ceci pourrait permettre de deviner qu'il y a eu une transaction entre les deux surtout si le dépôt et le retrait sont proches temporellement.

L'idéal serait que Bob ne retire pas ses jetons, et qu'il s'en serve pour payer des transactions.

Pour accroître son anonymat, Bob peut faire en sorte de ne jamais interagir avec le contrat pour déposer ou retirer des tez.

Cet exemple est un cas simple de l'utilisation de Sapling.

Il est également possible de permettre à un tiers, par exemple un régulateur, de contrôler toutes les transactions du contrat Sapling.

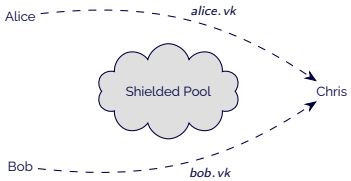
Exemple avec un suivi de la transaction par un tiers (1/2)

Dans ce scénario, seule l'adresse de Chris peut envoyer des transactions (hors dépôts et retraits) à la *shielded pool* :

Exemple avec un suivi de la transaction par un tiers (1/2)

Dans ce scénario, seule l'adresse de Chris peut envoyer des transactions (hors dépôts et retraits) à la *shielded pool* :

1. Alice et Bob s'identifient auprès de Chris et lui transmettent leurs clefs de visualisation respectives [1].



1. Ceci peut avoir été fait bien avant la transaction actuelle.

Exemple avec un suivi de la transaction par un tiers (1/2)

Dans ce scénario, seule l'adresse de Chris peut envoyer des transactions (hors dépôts et retraits) à la *shielded pool* :

1. Alice et Bob s'identifient auprès de Chris et lui transmettent leurs clefs de visualisation respectives [1].
2. Alice crée des jetons dans la *shielded pool* en envoyant 8ℓ au contrat.

1. Ceci peut avoir été fait bien avant la transaction actuelle.

Exemple avec un suivi de la transaction par un tiers (1/2)

Dans ce scénario, seule l'adresse de Chris peut envoyer des transactions (hors dépôts et retraits) à la *shielded pool* :

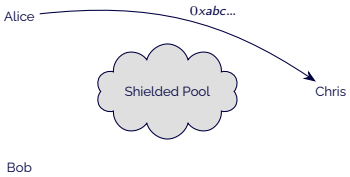
1. Alice et Bob s'identifient auprès de Chris et lui transmettent leurs clefs de visualisation respectives [1].
2. Alice crée des jetons dans la *shielded pool* en envoyant 8½ au contrat.
3. Bob dérive une adresse de sa clef de visualisation et l'envoie à Alice [2].

1. Ceci peut avoir été fait bien avant la transaction actuelle.
2. Alice pourrait également s'adresser à Chris car il a accès à la clef de Bob et peut donc également en dériver une adresse.

Exemple avec un suivi de la transaction par un tiers (1/2)

Dans ce scénario, seule l'adresse de Chris peut envoyer des transactions (hors dépôts et retraits) à la *shielded pool* :

1. Alice et Bob s'identifient auprès de Chris et lui transmettent leurs clefs de visualisation respectives [1].
2. Alice crée des jetons dans la *shielded pool* en envoyant 8ℓ au contrat.
3. Bob dérive une adresse de sa clef de visualisation et l'envoie à Alice [2].
4. Alice forge une transaction de 8ℓ vers Bob et l'envoie à Chris.

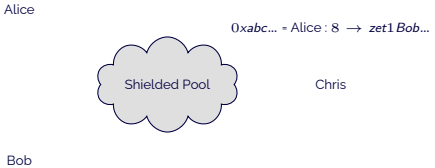


1. Ceci peut avoir été fait bien avant la transaction actuelle.
2. Alice pourrait également s'adresser à Chris car il a accès à la clef de Bob et peut donc également en dériver une adresse.

Exemple avec un suivi de la transaction par un tiers (1/2)

Dans ce scénario, seule l'adresse de Chris peut envoyer des transactions (hors dépôts et retraits) à la *shielded pool* :

1. Alice et Bob s'identifient auprès de Chris et lui transmettent leurs clefs de visualisation respectives [1].
2. Alice crée des jetons dans la *shielded pool* en envoyant $8\frac{1}{2}$ au contrat.
3. Bob dérive une adresse de sa clef de visualisation et l'envoie à Alice [2].
4. Alice forge une transaction de $8\frac{1}{2}$ vers Bob et l'envoie à Chris.
5. Chris constate, au moyen des clefs d'Alice, qu'il s'agit d'une transaction provenant d'Alice vers une adresse.

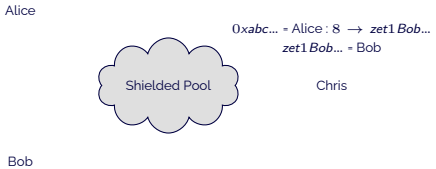


1. Ceci peut avoir été fait bien avant la transaction actuelle.
2. Alice pourrait également s'adresser à Chris car il a accès à la clef de Bob et peut donc également en dériver une adresse.

Exemple avec un suivi de la transaction par un tiers (1/2)

Dans ce scénario, seule l'adresse de Chris peut envoyer des transactions (hors dépôts et retraits) à la *shielded pool* :

1. Alice et Bob s'identifient auprès de Chris et lui transmettent leurs clefs de visualisation respectives [1].
2. Alice crée des jetons dans la *shielded pool* en envoyant 8ℓ au contrat.
3. Bob dérive une adresse de sa clef de visualisation et l'envoie à Alice [2].
4. Alice forge une transaction de 8ℓ vers Bob et l'envoie à Chris.
5. Chris constate, au moyen des clefs d'Alice, qu'il s'agit d'une transaction provenant d'Alice vers une adresse.
6. Il peut ensuite vérifier grâce à la clef de Bob que cette adresse est dérivée de la clef de Bob.

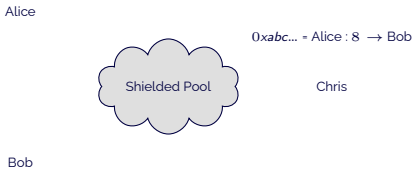


1. Ceci peut avoir été fait bien avant la transaction actuelle.
2. Alice pourrait également s'adresser à Chris car il a accès à la clef de Bob et peut donc également en dériver une adresse.

Exemple avec un suivi de la transaction par un tiers (1/2)

Dans ce scénario, seule l'adresse de Chris peut envoyer des transactions (hors dépôts et retraits) à la *shielded pool* :

1. Alice et Bob s'identifient auprès de Chris et lui transmettent leurs clefs de visualisation respectives [1].
2. Alice crée des jetons dans la *shielded pool* en envoyant 8ℓ au contrat.
3. Bob dérive une adresse de sa clef de visualisation et l'envoie à Alice [2].
4. Alice forge une transaction de 8ℓ vers Bob et l'envoie à Chris.
5. Chris constate, au moyen des clefs d'Alice, qu'il s'agit d'une transaction provenant d'Alice vers une adresse.
6. Il peut ensuite vérifier grâce à la clef de Bob que cette adresse est dérivée de la clef de Bob.
7. Les deux clefs lui indiquent le montant. Il sait alors qu'il s'agit d'une transaction de 8 jetons d'Alice vers Bob.

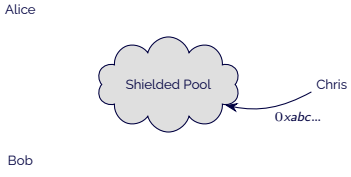


1. Ceci peut avoir été fait bien avant la transaction actuelle.
2. Alice pourrait également s'adresser à Chris car il a accès à la clef de Bob et peut donc également en dériver une adresse.

Exemple avec un suivi de la transaction par un tiers (1/2)

Dans ce scénario, seule l'adresse de Chris peut envoyer des transactions (hors dépôts et retraits) à la *shielded pool* :

1. Alice et Bob s'identifient auprès de Chris et lui transmettent leurs clefs de visualisation respectives [1].
2. Alice crée des jetons dans la *shielded pool* en envoyant 8ℓ au contrat.
3. Bob dérive une adresse de sa clef de visualisation et l'envoie à Alice [2].
4. Alice forge une transaction de 8ℓ vers Bob et l'envoie à Chris.
5. Chris constate, au moyen des clefs d'Alice, qu'il s'agit d'une transaction provenant d'Alice vers une adresse.
6. Il peut ensuite vérifier grâce à la clef de Bob que cette adresse est dérivée de la clef de Bob.
7. Les deux clefs lui indiquent le montant. Il sait alors qu'il s'agit d'une transaction de 8 jetons d'Alice vers Bob.
8. Il décide que cette transaction est régulière et envoie la transaction au contrat.

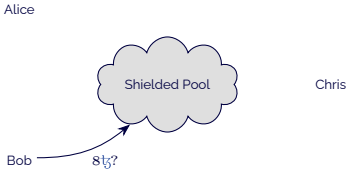


1. Ceci peut avoir été fait bien avant la transaction actuelle.
2. Alice pourrait également s'adresser à Chris car il a accès à la clef de Bob et peut donc également en dériver une adresse.

Exemple avec un suivi de la transaction par un tiers (1/2)

Dans ce scénario, seule l'adresse de Chris peut envoyer des transactions (hors dépôts et retraits) à la *shielded pool* :

1. Alice et Bob s'identifient auprès de Chris et lui transmettent leurs clefs de visualisation respectives [1].
2. Alice crée des jetons dans la *shielded pool* en envoyant 8ꜩ au contrat.
3. Bob dérive une adresse de sa clef de visualisation et l'envoie à Alice [2].
4. Alice forge une transaction de 8ꜩ vers Bob et l'envoie à Chris.
5. Chris constate, au moyen des clefs d'Alice, qu'il s'agit d'une transaction provenant d'Alice vers une adresse.
6. Il peut ensuite vérifier grâce à la clef de Bob que cette adresse est dérivée de la clef de Bob.
7. Les deux clefs lui indiquent le montant. Il sait alors qu'il s'agit d'une transaction de 8 jetons d'Alice vers Bob.
8. Il décide que cette transaction est régulière et envoie la transaction au contrat.
9. Bob peut alors réclamer les jetons qu'il a reçus et recevoir ses 8ꜩ.



1. Ceci peut avoir été fait bien avant la transaction actuelle.
2. Alice pourrait également s'adresser à Chris car il a accès à la clef de Bob et peut donc également en dériver une adresse.

Exemple avec un suivi de la transaction par un tiers (1/2)

Dans ce scénario, seule l'adresse de Chris peut envoyer des transactions (hors dépôts et retraits) à la *shielded pool* :

1. Alice et Bob s'identifient auprès de Chris et lui transmettent leurs clefs de visualisation respectives [1].
2. Alice crée des jetons dans la *shielded pool* en envoyant 8ꜩ au contrat.
3. Bob dérive une adresse de sa clef de visualisation et l'envoie à Alice [2].
4. Alice forge une transaction de 8ꜩ vers Bob et l'envoie à Chris.
5. Chris constate, au moyen des clefs d'Alice, qu'il s'agit d'une transaction provenant d'Alice vers une adresse.
6. Il peut ensuite vérifier grâce à la clef de Bob que cette adresse est dérivée de la clef de Bob.
7. Les deux clefs lui indiquent le montant. Il sait alors qu'il s'agit d'une transaction de 8 jetons d'Alice vers Bob.
8. Il décide que cette transaction est régulière et envoie la transaction au contrat.
9. Bob peut alors réclamer les jetons qu'il a reçus et recevoir ses 8ꜩ

1. Ceci peut avoir été fait bien avant la transaction actuelle.
2. Alice pourrait également s'adresser à Chris car il a accès à la clef de Bob et peut donc également en dériver une adresse.

Exemple avec un suivi de la transaction par un tiers (2/2)

Cet exemple montre le cas dans lequel Chris s'occupe de valider la conformité des opérations, potentiellement parce qu'il a à répondre de ce qui se passe dans le contrat, ou encore parce qu'il est chargé de protéger Alice (et les autres membres du contrat) contre d'éventuelles transactions frauduleuses. Par ailleurs, s'il décide qu'une opération n'est pas valide, il peut décider de contacter Alice pour lui en faire part et lui demander plus d'informations concernant cette transaction, ou tout simplement pour vérifier qu'elle ne s'est pas fait subtiliser son wallet et qu'elle est bien l'auteur de cette transaction.

Chris n'est pas forcément un humain, il peut aussi être un simple oracle[1] sécurisé qui possède des clefs de visualisation et effectue automatiquement les transactions dont les parties sont connues.

Dans les deux exemples, on a supposé que les jetons représentaient des tez. Ceci n'est pas obligatoire et l'on pourrait imaginer que les jetons représentent n'importe quelle autre bien fongible (actifs financiers, matières premières,...). Dans ce cas, le régulateur pourrait également gérer les dépôts et les retraits. Il n'y aurait plus d'interaction directe entre le bien réel et sa représentation sur le contrat (comme on a pu le faire avec le tez). Le contrat **pourrait alors avoir une valeur de preuve** en cas de litige.

1. Un oracle est un programme extérieur à la blockchain, généralement autonome, dont le rôle est de faire des appels à un smart-contrat pour lui donner des ordres en fonction de paramètres que le smart-contrat ne peut pas ou n'a pas besoin de contrôler.

nomadic labs

Continuons cet échange

https://tezos.com
https://developers.tezos.com
https://tezos.gitlab.io/alpha/sapling.html