(19) RÉPUBLIQUE FRANÇAISE

INSTITUT NATIONAL
DE LA PROPRIÉTÉ INDUSTRIELLE

COURBEVOIE

(11) **N° de publication :** ***3 077 151***

(à n'utiliser que pour les commandes de reproduction)

(21) **N° d'enregistrement national :** **18 50570**

(51) Int Cl[8] : ***G 06 Q 20/00*** *(2018.01), H 04 L 9/14*

(12) **BREVET D'INVENTION** **B1**

(54) SYSTEME SECURISE DE TRANSACTIONS ENTRE TERMINAUX.

(22) **Date de dépôt :** 25.01.18.

(30) **Priorité :**

(43) **Date de mise à la disposition du public de la demande :** 26.07.19 Bulletin 19/30.

(45) **Date de la mise à disposition du public du brevet d'invention :** 24.06.22 Bulletin 22/25.

(56) **Liste des documents cités dans le rapport de recherche :**

Se reporter à la fin du présent fascicule

(60) **Références à d'autres documents nationaux apparentés :**

◯ **Demande(s) d'extension :**

(71) **Demandeur(s) :** *REPUTACTION Société par actions simplifiée* — FR.

(72) **Inventeur(s) :** SEIGNEUR JEAN-MARC.

(73) **Titulaire(s) :** REPUTACTION Société par actions simplifiée.

(74) **Mandataire(s) :** INNOVATION COMPETENCE GROUP.

FR 3 077 151 - B1

SYSTÈME SÉCURISÉ DE TRANSACTIONS ENTRE TERMINAUX

[001] L'invention concerne un système et un procédé sécurisés de transactions entre terminaux. L'invention concerne aussi un terminal de paiement pour la réalisation de ce système ainsi qu'un support d'enregistrement d'informations pour mettre en œuvre ce procédé.

[002] Des systèmes sécurisés connus de transactions entre terminaux comportent

- un ensemble de serveurs programmés pour vérifier et, après vérification, enregistrer des transactions dans une chaîne de blocs, chaque bloc comportant plusieurs transactions et une empreinte numérique d'un bloc précédent dans la chaîne de blocs, chaque transaction étant un message numérique qui contient au moins les informations suivantes :
  - au moins une adresse d'arrivée de cette transaction, cette adresse d'arrivée ayant été construite à partir d'une paire de clefs publique/privée contenant une clef privée et une clef publique correspondant à cette clef privée,
  - au moins une adresse de départ qui identifie sans ambiguïté une adresse d'arrivée d'une transaction précédemment enregistrée dans la chaîne de blocs,
  - un identifiant d'objet transféré depuis l'adresse de départ vers l'adresse d'arrivée,
  - une signature numérique de la transaction obtenue en signant au moins les adresses de départ et d'arrivée avec la clef privée, cette signature permettant à l'ensemble de serveurs de vérifier que cette transaction a bien été générée par un terminal ayant accès à cette clef privée,
- un terminal de paiement apte à communiquer avec l'ensemble de serveurs, ce terminal de paiement étant équipé :
  - d'un cryptoprocesseur, ce cryptoprocesseur comportant une mémoire sécurisée uniquement accessible par le cryptoprocesseur,
  - d'un émetteur/récepteur apte à établir une liaison d'échange d'informations avec un autre terminal,
- un terminal d'encaissement apte à communiquer avec l'ensemble de serveurs, ce terminal d'encaissement étant équipé :
  - d'un microprocesseur,
  - d'une mémoire, et
  - d'un émetteur/récepteur apte à établir une liaison d'échange d'informations avec un autre terminal,
- l'ensemble (6) de serveurs est apte à enregistrer une première transaction contenant une première adresse d'arrivée et un premier identifiant d'objet transféré vers cette première adresse d'arrivée,
- la mémoire sécurisée comporte une première paire de clefs publique/privée à partir de laquelle la première adresse d'arrivée a été générée, cette première paire de clefs publique/privée comportant :

- une première clef privée qui est la seule à pouvoir valablement signer une seconde transaction ayant une adresse de départ qui identifie sans ambiguïté la première adresse d'arrivée, et
- une première clef publique correspondant à la première clef privée,

- le terminal d'encaissement est apte à obtenir et à enregistrer dans sa mémoire, une seconde adresse d'arrivée et une seconde paire de clefs publique/privée à partir de laquelle la seconde adresse d'arrivée a été générée, cette seconde paire de clefs publique/privée comportant une seconde clef privée et une seconde clef publique correspondant à cette seconde clef privée,
- le cryptoprocesseur est apte, en réponse à la réception de la seconde adresse d'arrivée, à construire la seconde transaction, signée avec la première clef privée, entre une adresse de départ identifiant sans ambiguïté la première adresse d'arrivée et cette seconde adresse d'arrivée,
- les terminaux de paiement et d'encaissement sont aptes à transmettre toutes transactions construites ou reçues à l'ensemble de serveurs lorsqu'ils sont connectés à cet ensemble de serveurs,
- l'ensemble de serveurs est apte, en réponse à la réception de toute transaction :
  - à vérifier la validité de cette transaction reçue à l'aide de la signature qu'elle contient,
  - à vérifier l'absence de double-dépense en vérifiant qu'il n'existe pas une transaction du même objet depuis la même adresse de départ déjà enregistrée dans la chaîne de blocs, et
  - seulement si ces vérifications confirment la validité de la transaction reçue et l'absence de double-dépense, à enregistrer la transaction reçue dans la chaîne de blocs,
- même en absence de connexion avec l'ensemble de serveurs, les terminaux de paiement et d'encaissement sont aptes à établir, par l'intermédiaire de leurs émetteurs/récepteurs respectifs, une liaison d'échange d'informations entre eux.

[003] Par exemple, l'un de ces systèmes connus est celui développé pour transférer de la crypto-monnaie connue sous le terme de « Bitcoin » entre terminaux. Par la suite, ce système est simplement appelé « système Bitcoin ». Dans le système Bitcoin, l'ensemble de serveurs réalise les opérations connues sous le terme de « minage » ou « mining » en anglais.

[004] Dans ces systèmes connus pour qu'une transaction entre deux terminaux soit réalisée et validée, il faut que le terminal de paiement construise la transaction, puis la transmette à l'ensemble de serveurs. L'ensemble de serveurs vérifie la validité de la transaction et si la transaction est valide, l'enregistre dans la chaîne de blocs. Cette chaîne de blocs est plus connue sous le terme anglais de « blockchain ».

[005] L'une des fonctions essentielles de l'ensemble de serveurs est d'empêcher les fraudes. En particulier, les systèmes connus sont conçus pour rendre, si possible, impossible la fraude connue sous le terme de « double-dépense » en français ou sous le terme anglais de « double-spending ». Cette fraude est définie ci-

dessous dans le cas particulier du système Bitcoin. Toutefois, elle peut être définie de façon similaire dans tout système sécurisé de transaction. Dans le système Bitcoin, cette fraude consiste :

- à construire une première transaction qui transfert, par exemple, une somme X de crypto-monnaie d'une première adresse de départ, appelée UTXO (« Unspent Transaction Output ») dans le système Bitcoin, vers une première adresse d'arrivée, appelée « Bitcoin address » dans le système Bitcoin, puis
- à construire une seconde transaction qui transfert, la même somme X de crypto-monnaie de la même première adresse de départ vers une seconde adresse Bitcoin d'arrivée différente de la première adresse d'arrivée.

[006] En faisant cela, la même somme X de crypto-monnaie a été dépensée deux fois, ce qui doit absolument être évité.

[007] Dans le système Bitcoin développé initialement, ce sont les serveurs informatiques de l'ensemble qui vérifient notamment si l'adresse de départ de la transaction reçue ne correspond pas à une adresse de départ déjà enregistrée dans la chaîne de blocs. Si l'adresse de départ de la transaction reçue a déjà été utilisée, l'ensemble de serveurs informatiques considère que cette transaction est invalide et elle n'est donc pas enregistrée dans la chaîne de blocs. Grâce à cela, la double-dépense est rendue impossible.

[008] Dès lors, pour que le terminal d'encaissement soit sûr que la transaction vers l'une de ses adresses d'arrivée soit valide, il doit attendre que celle-ci soit validée par l'ensemble de serveurs, puis vérifier qu'elle a bien été enregistrée dans la chaîne de blocs. Pour que la transaction soit enregistrée dans la chaîne de blocs puis ensuite vérifiée par le terminal d'encaissement, il faut que le terminal de paiement et le terminal d'encaissement disposent chacun d'une connexion à cet ensemble de serveurs. Ainsi, dans le système Bitcoin initialement développé, la sécurité d'une transaction entre deux terminaux ne pouvait être garantie que si ces terminaux disposaient chacun d'une connexion à l'ensemble de serveurs. Ce type de transactions, qui nécessitent de disposer d'une connexion à l'ensemble de serveurs pour être sécurisé contre les fraudes telle que la double-dépense, est par la suite appelé « transaction en-ligne » ou « on-chain transaction» en anglais.

[009] Par contre, le système Bitcoin initialement développé ne permettait pas de garantir la sécurité d'une transaction entre deux terminaux sans connexion à l'ensemble de serveurs informatiques. Ce deuxième type de transactions est appelé par la suite « transaction hors-ligne » ou « off-chain transaction » en anglais. Attention, une transaction hors-ligne n'est pas nécessairement une transaction lors de laquelle aucune connexion à un réseau de transmission d'informations n'existe mais seulement une transaction réalisée sans qu'une connexion à l'ensemble de serveurs informatiques soit nécessaire.

[0010] Récemment, une solution connue sous le terme de « Lightning network » a été développée pour garantir en plus la sécurité des transactions hors-lignes tout en restant compatible avec le fonctionnement des transactions en-ligne. Cette solution

est décrite, par exemple, dans l'article suivant : Joseph Poon et Al : «The Bitcoin Lightning Network : Scalable off-chain Instant paiement », 14/01/2016, draft version 0.5.9.2, accessible en-ligne sur le site Internet : https://lightning.network/.

[0011] En résumé, cette solution consiste à créer un compte commun sur lequel est transféré un montant en Bitcoin. Le montant en Bitcoin ne peut être valablement transféré de ce compte commun vers une adresse d'arrivée que si cette transaction a été signée à la fois à l'aide d'une clef privée du terminal de paiement et d'une clef privée du terminal d'encaissement. De plus, pour assurer la sécurité de cette transaction hors-ligne, un serveur informatique particulier supplémentaire, appelé « Watcher Node », est ajouté à l'ensemble de serveurs existant.

[0012] Dans cette solution « Lightning network », un compte commun doit être créé en-ligne par les terminaux de paiement et d'encaissement. Ainsi, une transaction sécurisée hors-ligne n'est possible qu'entre un terminal de paiement et un terminal d'encaissement qui se connaissaient déjà avant d'être déconnectés de l'ensemble de serveurs. Cette solution ne marche donc pas si avant de se déconnecter de l'ensemble de serveurs, le terminal de paiement ne connaît pas le terminal d'encaissement. En effet, dans ce cas, le compte commun n'a pas pu être créé.

[0013] L'invention vise à résoudre les inconvénients de la solution « Lightning network » en assurant la sécurité d'une transaction hors-ligne entre un terminal de paiement et un terminal d'encaissement, même si le terminal d'encaissement était inconnu avant que le terminal de paiement se déconnecte de l'ensemble de serveurs. En d'autres termes, l'invention vise à sécuriser une transaction hors-ligne entre le terminal de paiement et le terminal d'encaissement sans qu'il soit nécessaire, avant d'être déconnecté de l'ensemble de serveurs, que ces terminaux de paiement et d'encaissement aient besoin de coopérer entre eux pour réaliser certaines opérations en-ligne préparatoires à la sécurisation de la transaction hors-ligne qui sera ensuite réalisée. En particulier, la transaction hors-ligne doit être sécurisée contre la double-dépense.

[0014] De plus, comme la solution « Lightning network », la transaction réalisée hors-ligne doit rester compatible avec le fonctionnement des transactions en-ligne. En particulier, l'adresse d'arrivée vers laquelle l'objet de la transaction hors-ligne a été transféré doit ensuite pouvoir être utilisée en tant qu'adresse de départ d'une autre transaction réalisée, cette fois-ci, en-ligne.

[0015] Elle a donc pour objet un tel système sécurisé de transactions entre terminaux dans lequel même en absence de connexion avec l'ensemble de serveurs, le cryptoprocesseur est apte :

- à transmettre au terminal d'encaissement, par l'intermédiaire de la liaison d'échange d'informations, la première clef privée ou la seconde transaction signée construite par le cryptoprocesseur, puis
- à enregistrer, dans la mémoire sécurisée, un second identifiant de l'objet transféré par la seconde transaction propre à empêcher tout nouveau transfert du même objet à partir de la première adresse d'arrivée, et ensuite

- lorsqu'une nouvelle transaction est construite par le cryptoprocesseur, à comparer l'identifiant de l'objet transféré par cette nouvelle transaction au second identifiant de l'objet transféré enregistré dans la mémoire sécurisée, et

- seulement dans le cas où l'identifiant de l'objet transféré par cette nouvelle transaction correspond au second identifiant de l'objet transféré enregistré dans la mémoire sécurisée, à interdire la transmission de cette nouvelle transaction vers le terminal d'encaissement.

[0016] Les modes de réalisations de ce système peuvent comporter une ou plusieurs des caractéristiques suivantes :

- le microprocesseur du terminal d'encaissement est un cryptoprocesseur, ce cryptoprocesseur comportant une mémoire sécurisée uniquement accessible par le cryptoprocesseur, la mémoire du terminal d'encaissement est la mémoire sécurisée du cryptoprocesseur du terminal d'encaissement ;
- l'ensemble de serveurs est apte à vérifier l'absence de double-dépense en vérifiant seulement que l'adresse de départ de la transaction reçue n'est pas égale à une adresse de départ déjà enregistrée dans la chaîne de blocs.

[0017] L'invention a également pour objet un terminal de paiement spécialement conçu pour la réalisation du système revendiqué dans lequel le terminal de paiement est apte à communiquer avec l'ensemble de serveurs, ce terminal de paiement étant équipé :

- d'un cryptoprocesseur, ce cryptoprocesseur comportant une mémoire sécurisée uniquement accessible par le cryptoprocesseur,
- d'un émetteur/récepteur apte à établir une liaison d'échange d'informations avec un autre terminal,

- la mémoire sécurisée est apte à comporter une première paire de clefs publique/privée à partir de laquelle la première adresse d'arrivée a été générée, cette première paire de clefs publique/privée comportant :
  - une première clef privée qui est la seule à pouvoir valablement signer une seconde transaction ayant une adresse de départ qui identifie sans ambiguïté la première adresse d'arrivée, et
  - une première clef publique correspondant à la première clef privée,
- le cryptoprocesseur est apte, en réponse à la réception de la seconde adresse d'arrivée, à construire la seconde transaction, signée avec la première clef privée, entre une adresse de départ identifiant sans ambiguïté la première adresse d'arrivée et cette seconde adresse d'arrivée,
- le terminal de paiement est apte à transmettre toutes transactions construites à l'ensemble de serveurs lorsqu'il est connecté à cet ensemble de serveurs,
- même en absence de connexion avec l'ensemble de serveurs, le terminal de paiement est apte à établir, par l'intermédiaire de son émetteur/récepteur, une liaison d'échange d'informations avec le terminal d'encaissement,

et dans lequel, même en absence de connexion avec l'ensemble de serveurs, le cryptoprocesseur est apte :

- à transmettre au terminal d'encaissement, par l'intermédiaire de la liaison d'échange d'informations, la première clef privée ou la seconde transaction signée construite par le cryptoprocesseur, puis
- à enregistrer, dans le mémoire sécurisée, un second identifiant de l'objet transféré par la seconde transaction propre à empêcher tout nouveau transfert du même objet à partir de la première adresse d'arrivée, et ensuite
- lorsqu'une nouvelle transaction est construite par le cryptoprocesseur, à comparer l'identifiant de l'objet transféré par cette nouvelle transaction au second identifiant de l'objet transféré enregistré dans la mémoire sécurisée, et
- seulement dans le cas où l'identifiant de l'objet transféré par cette nouvelle transaction correspond au second identifiant de l'objet transféré enregistré dans la mémoire sécurisée, à interdire la transmission de cette nouvelle transaction vers le terminal d'encaissement.

[0018] Les modes de réalisation de ce terminal de paiement peuvent comporter une ou plusieurs des caractéristiques suivantes :

- le cryptoprocesseur est apte à transmettre au terminal d'encaissement, par l'intermédiaire de la liaison d'échange d'informations, la première clef privée ;
- le terminal de paiement est apte à chiffrer la transmission, par l'intermédiaire de la liaison d'échange d'informations, de la première clef privée ou de la seconde transaction signée ;
- le terminal de paiement est apte à signer la transmission, par l'intermédiaire de la liaison d'échange d'informations, de la première clef privée ou de la seconde transaction signée ;
- le cryptoprocesseur est apte :

- en absence de connexion avec l'ensemble de serveurs, à construire et à enregistrer dans sa mémoire sécurisée la première transaction, puis
- après la transmission de la première clef privée ou de la seconde transaction signée, en réponse à une connexion avec l'ensemble de serveurs, à transmettre automatiquement la première transaction enregistrée à l'ensemble de serveurs ;

- la mémoire sécurisée est une mémoire non-volatile.

[0019] L'invention a également pour objet un procédé sécurisé de transactions entre terminaux destiné à être mis en œuvre dans le système revendiqué, dans lequel :

- l'ensemble de serveurs enregistre une première transaction contenant une première adresse d'arrivée et un premier identifiant d'objet transféré vers cette première adresse d'arrivée,
- le cryptoprocesseur du terminal de paiement enregistre dans sa mémoire sécurisée une première paire de clefs publique/privée à partir de laquelle la première adresse d'arrivée a été générée, cette première paire de clefs publique/privée comportant :

- une première clef privée qui est la seule à pouvoir valablement signer une seconde transaction ayant une adresse de départ qui identifie sans ambiguïté la première adresse d'arrivée, et

- une première clef publique correspondant à la première clef privée,

- le terminal d'encaissement obtient et enregistre dans sa mémoire, une seconde adresse d'arrivée et une seconde paire de clefs publique/privée à partir de laquelle la seconde adresse d'arrivée a été générée, cette seconde paire de clefs publique/privée comportant une seconde clef privée et une seconde clef publique correspondant à cette seconde clef privée,

- en absence de connexion avec l'ensemble de serveurs, les terminaux de paiement et d'encaissement établissent, par l'intermédiaire de leurs émetteurs/récepteurs respectifs, une liaison d'échange d'informations entre eux,

dans lequel, même en absence de connexion avec l'ensemble de serveurs, :

- le cryptoprocesseur transmet au terminal d'encaissement, par l'intermédiaire de la liaison d'échange d'informations, la première clef privée ou la seconde transaction signée construite par le cryptoprocesseur, puis

- le cryptoprocesseur enregistre, dans la mémoire sécurisée, un second identifiant de l'objet transféré par la seconde transaction propre à empêcher tout nouveau transfert du même objet à partir de la première adresse d'arrivée, et ensuite

- lorsqu'une nouvelle transaction est construite par le cryptoprocesseur, le cryptoprocesseur compare l'identifiant de l'objet transféré par cette nouvelle transaction au second identifiant de l'objet transféré enregistré dans la mémoire sécurisée, et

- seulement dans le cas où l'identifiant de l'objet transféré par cette nouvelle transaction correspond au second identifiant de l'objet transféré enregistré dans la mémoire sécurisée, le cryptoprocesseur interdit la transmission de cette nouvelle transaction vers le terminal d'encaissement.

[0020] Enfin, l'invention a également pour objet un support d'enregistrement d'informations, lisible par un cryptoprocesseur, dans lequel ce support d'enregistrement d'informations comporte des instructions pour la mise en œuvre du procédé revendiqué, lorsque ces instructions sont exécutées par le cryptoprocesseur.

[0021] L'invention sera mieux comprise à la lecture de la description qui va suivre, donnée uniquement à titre d'exemple non limitatif et faite en se référant aux dessins sur lesquels :

- la figure 1 est une illustration schématique de l'architecture d'un système sécurisé de transactions entre terminaux ;

- la figure 2 est une illustration schématique d'une mémoire sécurisée d'un terminal de paiement utilisé dans le système de la figure 1 ;

- la figure 3 est une illustration schématique d'une mémoire utilisée dans un terminal d'encaissement du système de la figure 1 ;

- la figure 4 est un organigramme d'un procédé sécurisé de transactions entre terminaux mis en œuvre dans le système de la figure 1.

[0022] <u>Chapitre I : Définitions et notations :</u>

[0023] Dans les figures, les mêmes références sont utilisées pour désigner les mêmes éléments.

[0024] Dans la suite de cette description, les caractéristiques et fonctions bien connues de l'homme du métier ne sont pas décrites en détail.

[0025] Par « paire de clefs publique/privée », on désigne une paire de clefs contenant une clef privée et une clef publique. La clef publique est différente de la clef privée. La clef publique permet de chiffrer un message numérique qui est alors uniquement déchiffrable à l'aide de la clef privée. La clef publique permet aussi de déchiffrer un cryptogramme obtenu en chiffrant un message numérique avec la clef privée. Les notions de clef publique et clef privée sont bien connues dans le domaine de la cryptographie asymétrique. Par la suite, on dit aussi que la clef privée correspond à la clef publique et vice versa lorsqu'il s'agit des clefs publique et privée d'une même paire de clefs publique/privée.

[0026] Une liaison point-à-point d'échanges d'informations est une liaison d'échanges d'informations établie par l'intermédiaire d'un réseau de transmission d'informations entre seulement deux terminaux.

[0027] On dit qu'une transaction est « hors-ligne », ou « off-chain » en anglais, lorsqu'une transaction est réalisée entre deux terminaux sans qu'il soit nécessaire pour cela qu'au moins l'un des terminaux soit connecté à l'ensemble de serveurs informatiques chargés de valider et d'enregistrer chaque transaction dans la chaîne de blocs.

[0028] Par opposition, on qualifie de transactions « en-ligne », ou « on-chain » en anglais, toute transaction qui nécessite, pour être sécurisée, une connexion à l'ensemble de serveurs informatiques. Les transactions en-ligne correspondent aux transactions les plus classiques des systèmes de transactions tel que le système Bitcoin.

[0029] La terminologie utilisée pour décrire le système Bitcoin est celle définie dans le glossaire du développeur Bitcoin accessible à l'adresse suivante :

- https://bitcoin.org/en/developer-glossary pour les termes anglais, et
- https://bitcoin.org/fr/vocabulaire pour les termes en français.

[0030] La description technique du fonctionnement du système Bitcoin actuel est par exemple donnée dans le guide « Bitcoin developer guide » accessible à l'adresse suivante : https://bitcoin.org/en/developer-guide. Par conséquent, en ce qui concerne les caractéristiques et fonctionnalités connues du système Bitcoin, le lecteur est renvoyé aux documentations citées ci-dessus.

[0031] Par « adresse de départ » d'une transaction, on désigne une donnée numérique qui identifie sans ambiguïté l'adresse d'arrivée d'une précédente transaction. Par exemple, dans le cas d'un système Bitcoin, l'adresse de départ est connue sous le terme de « Entrée » (« Input » en anglais) d'une transaction. Elle correspond alors à la concaténation d'un identifiant de la précédente transaction et d'un indice d'une sortie (« Output index» en anglais) de cette précédente transaction. Dans le système Bitcoin, une adresse de départ correspond donc à l'identifiant d'une

UTXO (« Unspent Transaction Output »). Attention, avec la terminologie adoptée ici, une adresse de départ n'est pas une adresse Bitcoin car une même adresse Bitcoin peut avoir plusieurs sorties. Dans le cas du système Ethereum, l'adresse de départ correspond à l'adresse d'un compte de départ.

[0032] Par « adresse d'arrivée » d'une transaction, on désigne une donnée numérique qui identifie sans ambiguïté l'adresse d'arrivée vers laquelle est transféré l'objet de la transaction. Dans le système Bitcoin, cela correspond à une sortie (« Output » en anglais) d'une transaction ou à une adresse Bitcoin si celle-ci ne comporte qu'une seule sortie. Dans le système Ethereum, cela correspond à l'adresse d'un compte d'arrivée.

[0033] Chapitre II : Exemples de modes de réalisation :

[0034] La figure 1 représente un système sécurisé 2 de transactions entre terminaux. Le système 2 est décrit ici dans le cas particulier où l'objet de chaque transaction est une monnaie couramment appelée crypto-monnaie. Dans ce mode de réalisation, la crypto-monnaie est le Bitcoin. Par la suite, seules quelques caractéristiques conventionnelles du système Bitcoin utiles pour la compréhension sont rappelées. La description se focalise surtout sur les modifications et fonctionnalités nouvelles implémentées dans le système Bitcoin conventionnel pour permettre la réalisation de transactions sécurisée hors-ligne entre terminaux.

[0035] On rappelle qu'une transaction est un message numérique qui contient typiquement au moins les informations suivantes :

- au moins une adresse de départ d'où provient la crypto-monnaie transférée vers la ou les adresses d'arrivée,
- un identifiant d'objet transféré, c'est-à-dire ici typiquement un montant de crypto-monnaie transféré exprimé en Bitcoin ou en Satochis,
- un premier script qui, lorsqu'il est exécuté par un calculateur électronique, détermine si la transaction est valide ou non, ce premier script contenant l'adresse d'arrivée vers laquelle doit être transférée la crypto-monnaie, et
- un second script qui, lorsqu'il est exécuté par un calculateur électronique, permet de charger les informations traitées par le premier script pour déterminer si cette transaction est valide ou non.

[0036] Dans le système 2, l'adresse de départ et l'adresse d'arrivée sont appelées, respectivement, entrée (« input ») et sortie (« output »). L'adresse de départ identifie sans ambiguïté l'adresse d'arrivée d'une précédente transaction enregistrée dans la chaîne de blocs, c'est-à-dire un UTXO. À cet effet, l'adresse de départ comporte un identifiant Txid de la transaction précédente et un index de sortie connu sous le terme anglais de « output index » qui identifie sans ambiguïté la transaction précédente.

[0037] L'adresse d'arrivée comporte une adresse unique construite à partir d'une paire de clefs publique/privée. Par exemple, dans le système Bitcoin, la clef privée est utilisée pour générer la clef publique correspondante et la clef publique est ensuite

utilisée pour générer l'adresse d'arrivée en mettant pour cela en œuvre une fonction de hachage (« hash function » en anglais).

[0038] Le premier script est connu sous le terme de « pubkey script » ou « scriptpubkey ». Le second script est connu sous le terme « signaturescript » ou « scriptsig ».

[0039] Le second script contient la signature de la transaction. Cette signature est obtenue en chiffrant une empreinte numérique de la transaction avec une clef privée connue seulement du terminal de paiement qui a généré cette transaction. L'empreinte numérique est construite à partir, notamment, de l'adresse de départ et de l'adresse d'arrivée de la transaction et à l'aide d'une fonction de hachage (« hash function » en anglais). Le second script contient également la clef publique qui permet de vérifier cette signature. Dans le cas du système Bitcoin, cette clef publique est la même que celle qui a été utilisée pour générer l'adresse de départ de cette transaction.

[0040] Lorsque le premier script est exécuté, il permet notamment de vérifier les deux conditions suivantes :

- Condition 1) : l'adresse de départ correspond bien à la clef publique chargée en mémoire par l'exécution du second script, et
- Condition 2) : la signature chargée par l'exécution du second script est une signature valide, c'est-à-dire qu'elle a pu être vérifiée à l'aide de la clef publique également chargée par l'exécution du second script et du contenu de la transaction.

[0041] La condition 1) permet de confirmer que l'adresse de départ est une adresse de départ générée par un terminal de paiement qui contient la clef publique. La condition 2) permet de confirmer que la transaction a bien été autorisée par le terminal de paiement qui contient la clef privée qui correspond à cette clef publique.

[0042] Le système 2 comporte :

- un réseau 4 de transmission d'informations,
- un ensemble 6 de serveurs informatiques connecté au réseau 4, et
- des terminaux électroniques entre lesquels sont échangés des Bitcoins.

[0043] Le réseau 4 est ici le réseau Internet sur lequel se déploie la toile d'araignée mondiale ou « World Wide Web ». Par l'intermédiaire de ce réseau 4, les différents serveurs de l'ensemble 6 ainsi que les différents terminaux sont connectés les uns aux autres sous la forme d'un réseau connu sous le terme de réseau pair-à-pair ou « Peer to Peer » en anglais.

[0044] L'ensemble 6 comporte typiquement des milliers de serveurs informatiques programmés pour vérifier les transactions entre terminaux et enregistrer les transactions valides dans une base de données connue sous le terme de « chaîne de blocs ». Pour simplifier la figure 1, seuls trois serveurs informatiques 10 à 12 de l'ensemble 6 ont été représentés. Le symbole « … » entre les serveurs 11 et 12 indique que seule une partie des serveurs informatiques de l'ensemble 6 a été représentée.

[0045] La chaîne de blocs est répartie entre les nombreux serveurs informatiques. À cet effet, chaque serveur informatique comporte une mémoire dans laquelle est enregistrée une partie ou la totalité de la chaîne de blocs. Sur la figure 1, ces mémoires portent les références M_{ref}, où l'indice « ref » est égal à la référence numérique attribuée au serveur informatique.

[0046] Chaque bloc de la chaîne de blocs comporte plusieurs transactions et une empreinte numérique d'un bloc précédent de la chaîne de blocs. Avant d'enregistrer une transaction dans un bloc, le serveur informatique vérifie la validité de cette transaction en exécutant les premier et second scripts qu'elle contient. Le serveur informatique vérifie aussi que l'adresse de départ que contient cette transaction n'a pas déjà été utilisée pour une transaction précédente déjà enregistrée dans la chaîne de blocs. Cette dernière vérification rend impossible ou presque impossible la double-dépense.

[0047] Pour exécuter les différentes opérations nécessaires au fonctionnement du système 2, chaque serveur informatique comporte un calculateur électronique programmé ou configuré pour exécuter le procédé décrit en référence à la figure 4. Sur la figure 1, le calculateur électronique d'un serveur informatique porte la référence CE_{ref}, où l'indice « ref » est égal à la référence numérique affecté à ce serveur informatique.

[0048] Les serveurs informatiques de l'ensemble 6 sont connus sous le terme de « mineur » ou « miner » en anglais.

[0049] Le système 2 comporte également de nombreux terminaux susceptibles de réaliser des transactions entre eux. Typiquement, le nombre de terminaux est dix fois ou cent fois ou mille fois supérieur au nombre de serveurs informatiques de l'ensemble 6. Pour simplifier la figure 1, seuls deux terminaux 20 et 22 sont représentés.

[0050] Le terminal 20 est ici un terminal de paiement. Ce terminal 20 comporte :

- un microprocesseur 24 programmable,
- une mémoire électronique non volatile 26,
- un cryptoprocesseur 28,
- un émetteur/récepteur 30,
- une interface homme-machine 32, et
- un bus 34 de transmission d'informations qui permet à ces différents composants du terminal 20 de communiquer entre eux.

[0051] Le cryptoprocesseur 28 peut se présenter sous la forme d'une carte à puce (« smart card » en anglais) ou d'un TPM (« Trusted Platform Module ») ou d'un module de sécurité matérielle. Le cryptoprocesseur 28 est un microprocesseur spécialement conçu pour être résistant vis-à-vis des tentatives de cryptanalyses telles que les attaques par canaux cachés. En particulier, il est plus résistant aux tentatives de cryptanalyse que le microprocesseur 24. De plus, le cryptoprocesseur 28 est configuré pour exécuter des algorithmes de chiffrement et de déchiffrement ainsi que des algorithmes de génération de paire de clefs publique/privée.

[0052] Le cryptoprocesseur 28 est par exemple réalisé à partir d'un FPGA (« Field-Programmable Gate Array ») ou d'un ASIC (« Application-Specific Integrated Circuit ») ou d'un microcontrôleur programmable.

[0053] Ici, le cryptoprocesseur 28 forme ce qui est connu sous le terme anglais de « Trusted Execution Environment ». Par exemple, de tels cryptoprocesseurs sont commercialisés par la société INTEL® sous le nom de « INTEL TXT » (« Intel Trusted eXecution Technology ») ou par la société GEMALTO® sous l'acronyme HSM (« Hardware Security Module »).

[0054] Le cryptoprocesseur 28 comporte une mémoire électronique non volatile sécurisée interne 36. La mémoire 36 est uniquement accessible par le cryptoprocesseur 28. Ainsi, le propriétaire du terminal 20 ne peut donc pas lire et obtenir les clés privées générées et stockées dans cette mémoire. En particulier, le microprocesseur 24 ne peut pas accéder au contenu de la mémoire 36. Cette mémoire 36 est destinée à contenir des informations secrètes telles que des clefs privées et des algorithmes cryptographiques. Elle comporte en particulier les instructions, exécutables par le cryptoprocesseur 28, pour la mise en œuvre du procédé de la figure 4.

[0055] Ici, le cryptoprocesseur 28 comporte en plus un mécanisme de détection de tentative de cryptanalyse qui permet de savoir si le cryptoprocesseur 28 a fait l'objet de telles tentatives de cryptanalyse.

[0056] L'émetteur/récepteur 30 est capable d'établir une liaison point-à-point 40 d'échange d'informations avec un autre terminal du système 2. Par exemple ici, cette liaison 40 peut être :

- une liaison filaire réalisée à l'aide d'un câble tel qu'un câble USB (« Universal Serial Bus »), ou
- une liaison sans fil telle qu'une liaison conforme à l'une des normes suivantes : NFC (« Near Field Communication ») ou Bluetooth.

[0057] La liaison 40 peut être établie entre les terminaux 20 et 22 indépendamment du fait que ces terminaux soient ou non connectés en même temps au réseau 4. Par exemple, ici, il s'agit d'une liaison directe entre les terminaux 20 et 22 conforme à la norme Bluetooth.

[0058] L'interface homme-machine 32 comporte typiquement au moins un écran pour afficher des informations et des moyens d'acquisition d'informations de la part de l'utilisateur de cette interface homme-machine. Par exemple, l'interface homme-machine 30 comporte un écran tactile.

[0059] Le terminal 20 est aussi capable de se connecter à l'ensemble 6 de serveurs informatiques par l'intermédiaire du réseau 4. Par exemple, le terminal 20 est un smartphone ou un ordinateur fixe ou portable.

[0060] Le terminal 22 est un terminal d'encaissement. Ici, à titre d'exemple, sa structure est identique à celle du terminal 20 sauf qu'il ne comporte pas le cryptoprocesseur 28. Par la suite, les composants du terminal 22 identiques à ceux du terminal 20 portent les mêmes références numériques que celles des composants

du terminal 20 mais incrémentées du chiffre 20. Ainsi, dans cet exemple de mode de réalisation, le terminal 22 est moins sécurisé que le terminal 20 puisque les opérations de chiffrement/déchiffrement et similaires sont réalisées par le microprocesseur 44 qui est moins sécurisé que le cryptoprocesseur 28.

[0061] Dans ce mode de réalisation, le système 2 comporte en plus un serveur informatique 60 d'informations de réputation et un serveur informatique 62 de révocation de certificats cryptographiques. Ces deux serveurs 60 et 62 sont raccordés au réseau 4. Chacun des serveurs 60 et 62 comporte :

- un microprocesseur programmable, respectivement 64 et 66, et
- une mémoire électronique non volatile respectivement 68 et 70.

[0062] Le microprocesseur 64 est programmé pour stocker, mettre à jour et transmettre, en réponse à une requête, des informations sur la confiance (« trustworthiness » en anglais) que l'on peut avoir dans les terminaux du système 2.

[0063] Le microprocesseur 66 est programmé pour stocker, mettre à jour et transmettre, en réponse à une requête, des informations sur la validité des certificats cryptographiques utilisés dans le système 2. En particulier, le microprocesseur 66 est capable de transmettre, en réponse à une requête, une liste contenant l'ensemble des certificats cryptographiques révoqués.

[0064] La figure 2 représente plus en détail le contenu de la mémoire 36. La mémoire 36 comporte :

- une liste L_{Rep},
- une liste L_{Rev},
- un certificat cryptographique C_{y20} contenant une clef publique K_{pub20},
- une clef privée K_{pr20} correspondant à la clef publique K_{pub20},
- une liste L_u de données de transactions déjà utilisées.

[0065] La liste L_{Rep} comporte, associée à des identifiants de terminaux du système 2, un indice de confiance. Plus la valeur numérique de cet indice de confiance est élevée, plus le terminal associé à cet indice est considéré comme fiable. Par exemple, tous les terminaux du système 2 équipés d'un cryptoprocesseur qui n'a pas été corrompu ont un indice de confiance plus élevé que les terminaux qui, comme le terminal 22, sont dépourvus de cryptoprocesseur. À chaque fois qu'un terminal équipé d'un cryptoprocesseur est victime d'une tentative de cryptanalyse, cette information est transmise au serveur 60 et l'indice de confiance de ce terminal est diminué. L'indice de confiance d'un terminal est également diminué à chaque fois qu'il est utilisé pour frauder ou pour tenter de frauder.

[0066] La liste L_{Rev} comporte une liste des certificats cryptographiques révoqués. Ces certificats cryptographiques révoqués ne sont donc plus utilisables.

[0067] Le certificat C_{y20} contient la clef publique K_{pub20} et une signature obtenue en chiffrant des informations contenues dans ce certificat à l'aide de la clef privée du fabricant de microprocesseur 28. Ainsi, il est possible de vérifier que le cryptoprocesseur 28 est un cryptoprocesseur authentique fabriqué par un fabricant connu.

[0068] La liste L_u contient les données de transactions déjà utilisées pour générer et réaliser une transaction hors-lignes. Cette liste L_u est ici utilisée pour empêcher la double-dépense pour les transactions hors-lignes.

[0069] La figure 3 représente plus en détail le contenu de la mémoire 46 du terminal 22. La mémoire 46 comporte ici notamment :

- la liste L_{Rep},
- la liste L_{Rev},
- un certificat cryptographique C_{y22} contenant une clef publique K_{pub22}, et
- une clef privée K_{pr22} correspondant à la clef publique K_{pub22}.

[0070] Par exemple, le certificat cryptographique C_{y22} est signé à l'aide d'une clef privée du fabricant du terminal 22 ou du microprocesseur 44.

[0071] Le fonctionnement du système 2 va maintenant être décrit en référence au procédé de la figure 4. Par la suite, pour simplifier les explications, les taxes prélevées sur chaque transaction notamment par les serveurs de l'ensemble 6 sont considérées comme nulle et ignorées. Toutefois, l'homme du métier est capable d'appliquer l'enseignement donnée ci-dessus dans le cas où de telles taxes ne seraient pas nulles, comme c'est le cas en pratique actuellement.

[0072] Initialement, lors d'une étape 100, le cryptoprocesseur 28 génère une paire de clefs publique/privée contenant une clef publique K_{pub1} et une clef privée K_{pr1}. Par exemple, le cryptoprocesseur 28 génère d'abord la clef K_{pr1} de façon aléatoire ou pseudo-aléatoire ou par tout autre moyen puis il construit la clef K_{pub1} à partir de la clef K_{pr1}. Ensuite, le cryptoprocesseur 28 construit, seulement à partir de la clef K_{pub1}, une adresse d'arrivée $@_1$ apte à recevoir des Bitcoins.

[0073] Lors d'une étape 101, une transaction T_1 signée pour transférer X Bitcoins vers l'adresse $@_1$ est construite et transmise à l'ensemble 6. Cette transaction T_1 peut être construite par un terminal tiers. La transaction T_1 peut aussi être construite par le terminal 20 lui-même qui transfert alors en-ligne X Bitcoins qu'il possédait déjà vers l'adresse $@_1$.

[0074] Par la suite, pour simplifier la description, on suppose que la transaction T_1 comporte une seule entrée et une seule sortie connue sous l'acronyme UTXO («Unspent Transaction output »).

[0075] Lors d'une étape 102, en réponse à la réception de la transaction T_1, les serveurs informatiques de l'ensemble 6 la vérifient en exécutant les premier et second scripts de cette transaction T_1. En particulier, les serveurs de l'ensemble 6 s'assurent que l'adresse de départ contenue dans cette transaction T_1 n'a pas déjà été utilisée pour une autre transaction déjà enregistrée dans la chaîne de blocs. Si la transaction T_1 est valide, alors les serveurs informatiques de l'ensemble 6 l'enregistrent dans la chaîne de blocs. Dans le cas contraire, c'est-à-dire si les serveurs informatiques déterminent que la transaction T_1 n'est pas valide, celle-ci n'est pas enregistrée dans la chaîne de blocs.

[0076] Lors d'une étape 104, quel que soit le terminal qui a créé la transaction T_1, le cryptoprocesseur 28 se connecte à l'ensemble 6 et vérifie que la transaction T_1

a bien été enregistrée dans la chaîne de blocs de sorte que le cryptoprocesseur 28 est sûr de disposer de X Bitcoins sur l'adresse $@_1$. Dans l'affirmative, le cryptoprocesseur 28 mémorise dans la mémoire 36 que le transfert des X Bitcoins de l'adresse $@_1$ est maintenant autorisé aussi bien en-ligne que hors-ligne. Dans le cas contraire, cette autorisation n'est pas enregistrée dans la mémoire 36 de sorte que le cryptoprocesseur 28 empêche la création de toute transaction hors-ligne ayant pour objet le transfert des X Bitcoins de l'adresse $@_1$.

[0077] Les étapes 100, 101, 102 et 104 sont réalisées, par exemple, comme décrit dans le système Bitcoin. Elles ne sont donc pas décrites plus en détail.

[0078] Lors d'une étape 106, le cryptoprocesseur 28 se connecte au serveur 60 et met à jour la liste L_{Rep} à partir des informations téléchargées depuis ce serveur 60. Lors de cette étape, le cryptoprocesseur 28 se connecte également au serveur 62 et met à jour la liste L_{Rev} à partir des informations téléchargées depuis le serveur 62. De façon similaire, le terminal 22 peut lui aussi mettre à jour les listes L_{Rep} et L_{Rev} enregistrées dans sa mémoire 46.

[0079] Lors d'une étape 108, le terminal 20 se déconnecte de l'ensemble 6. Par exemple, ici, on considère que le terminal 20 est déconnecté du réseau 4. À partir de ce moment, on considère également que le terminal 22 est également déconnecté de l'ensemble 6. Par exemple, les terminaux 20 et 22 sont sur une île déserte dépourvue de moyens d'accès au réseau 4. On suppose également que, dans ces conditions, le terminal 20 souhaite transférer au terminal 22, la totalité des X Bitcoins encore disponibles à l'adresse $@_1$ par l'intermédiaire d'une transaction hors-ligne.

[0080] A cet effet, lors d'une étape 110, les terminaux 20 et 22 établissent la liaison 40 entre eux par l'intermédiaire des émetteur/récepteur 30 et 50. Lors de cette étape, les terminaux 20 et 22 échangent leurs certificats cryptographiques C_{y20} et C_{y22}. Le cryptoprocesseur 28 et le terminal 22 vérifient alors si les certificats, respectivement, C_{y22} et C_{y20} reçus appartiennent à la liste L_{Rev}. Dans l'affirmative, c'est-à-dire si l'un des terminaux 20 et 22 détermine que le certificat reçu a été révoqué, le procédé s'arrête et, par exemple, la liaison 40 est interrompue. Aucune transaction hors-ligne entre les terminaux 20 et 22 n'est alors réalisée.

[0081] Dans le cas contraire, c'est-à-dire s'il a été déterminé que les certificats C_{y22} et C_{y20} n'ont pas été révoqués alors, à partir de maintenant, tous les messages transmis du terminal 20 au terminal 22 sont chiffrés avec la clef K_{pub22} et tous les messages transmis du terminal 22 vers le terminal 20 sont chiffrés avec la clef K_{pub20}. De plus, à chaque fois que le terminal 20 envoie un message au terminal 22, ce message est signé avec la clef K_{pr20} de sorte que le terminal 22 peut aussi vérifier l'authenticité de ce message avec la clef K_{pub20} reçue. À l'inverse, les messages transmis du terminal 22 vers le terminal 20 sont aussi signés avec la clef K_{pr22} de sorte que le terminal 20 peut en vérifier l'authenticité avec la clef K_{pub20} reçue. Si jamais l'authenticité d'un message échangé entre les terminaux 20 et 22 ne peut pas être vérifiée, alors le procédé est interrompu et aucune transaction entre les terminaux 20 et 22 n'est réalisée. À ce stade, la liaison 40 entre les terminaux 20 et

22 est dite sécurisée car, d'une part, elle est chiffrée et d'autre part, les messages échangés sont authentifiés.

[0082] Lors d'une étape 112, le terminal 22 transmet son identifiant au terminal 20 et le cryptoprocesseur 28 vérifie, à l'aide de l'identifiant reçu, l'indice de confiance associé à ce terminal 22 dans la liste L_{Rep}. Si l'indice de confiance associé au terminal 22 est inférieur à un seuil prédéterminé S_1, aucune transaction n'est réalisée entre les terminaux 20 et 22. Par exemple, le procédé s'arrête et la liaison 40 est interrompue. Dans le cas contraire, le procédé se poursuit par une étape 116.

[0083] Par la suite, on considère que le terminal 22 est associé à un indice de confiance supérieur au seuil S_1. De plus, pour simplifier les explications, on fait l'hypothèse que la totalité des X Bitcoins sont transférés du terminal 20 vers le terminal 22. A partir de maintenant, tous les échanges de messages entre les terminaux 20 et 22 se font par l'intermédiaire de la liaison sécurisée 40.

[0084] Lors de l'étape 116, le cryptoprocesseur 28 acquiert par l'intermédiaire de l'interface homme-machine 32 le choix de l'utilisateur entre deux options, respectivement, « transaction anonyme » et « transaction non-anonyme ».

[0085] Lors d'une étape 118, le cryptoprocesseur 28 génère un nombre N_{a1} tiré de façon aléatoire ou pseudo-aléatoire et transmet ce nombre N_{a1} au terminal 22. De préférence, ce nombre N_{a1} est transmis au terminal 22 par un autre moyen de communication que la liaison 40 pour éviter l'attaque connue sous le nom de « man in the middle ». Par exemple, le nombre N_{a1} est affiché sur l'écran du terminal 20 sous la forme d'un QR-code et le terminal 22 est utilisé pour scanner ce QR-Code et ainsi acquérir le nombre N_{a1}. Le nombre N_{a1} est ensuite inclus dans tous les messages échangés entre les terminaux 20 et 22 par l'intermédiaire de la liaison 40. Cela permet d'éviter la mise en œuvre d'une attaque par « rejeu ».

[0086] Ensuite, lors d'une étape 120, le cryptoprocesseur 28 acquiert par l'intermédiaire de l'interface homme-machine 32 une confirmation que les X Bitcoins associés à l'adresse $@_1$ doivent être transférés au terminal 22.

[0087] En réponse, lors d'une étape 124, si l'utilisateur a sélectionné l'option « transaction anonyme », le cryptoprocesseur 28 compare la clef K_{pr1} au contenu de la liste L_u. Si la clef K_{pr1} appartient à la liste L_u le procédé s'arrête et la transaction entre les terminaux 20 et 22 n'est pas réalisée. En effet, si la clef K_{pr1} appartient déjà à la liste L_u cela signifie que les X Bitcoins associés à l'adresse $@_1$ ont déjà été dépensés. L'interruption de la transaction dans ce cas empêche donc la double-dépense. Si la clef K_{pr1} n'appartient pas à la liste L_u, le procédé se poursuit par une étape 126.

[0088] Lors de l'étape 126, le cryptoprocesseur 28 transmet au terminal 22 la clef K_{pr1}, la clef K_{pub1}, le montant en Bitcoins transféré vers le terminal 22, et l'adresse @1. Dès lors, le terminal 22 est capable de construire une transaction signée et valide depuis l'adresse de départ l'adresse $@_1$ vers une autre adresse d'arrivée. Ainsi, les X Bitcoins associés à l'adresse $@_1$ ont été transférés du terminal 20 vers le terminal 22. De plus, dans ce cas, cette transaction est anonyme car elle ne laisse

aucune trace dans la chaîne de blocs même après qu'une transaction valide de l'adresse $@_1$ vers une autre adresse $@_2$ ait été construite par le terminal 22, puis enregistrée dans la chaîne de blocs lorsque le terminal 22 est de nouveau connecté à l'ensemble 6.

[0089] Ensuite, lors d'une étape 128, une fois que ces informations ont été transmises au terminal 22, le cryptoprocesseur 28 interdit la réalisation de toutes nouvelles transactions ayant pour objet le transfert des mêmes X Bitcoins que ceux transférés lors de l'étape 126. A cet effet, le cryptoprocesseur 28 enregistre dans la liste L_u un identifiant de l'objet transféré, c'est-à-dire ici des X Bitcoins transférés. Dans ce mode de réalisation, la clef K_{pr1}, la clef K_{pub1} ou l'adresse $@_1$ permet d'identifier sans ambiguïté ces X Bitcoins transférés. Ainsi, ici, le cryptoprocesseur 28 enregistre ces clefs K_{pr1}, K_{pub1} et l'adresse $@_1$ dans la liste L_u contenue dans la mémoire 36.

[0090] À l'issue de l'étape 120, si l'utilisateur avait sélectionné l'option « transaction non-anonyme », lors d'une étape 130, le terminal 22 obtient une paire de clefs publique/privée K_{pub2}/K_{pr2} et une adresse $@_2$ construite à partir de la clef K_{pub2}. Par exemple, ici, le terminal 22 génère et enregistre ces données comme décrit en référence à l'étape 100.

[0091] A la fin de l'étape 130, l'adresse $@_2$ est alors transmise au terminal 20.

[0092] En réponse à la réception de l'adresse $@_2$, lors d'une étape 132, le cryptoprocesseur 28 compare les clefs K_{pr1}, K_{pub1} et l'adresse $@_1$ au contenu de la liste L_u. Si l'une des clefs K_{pr1}, K_{pub1} et de l'adresse $@_1$ appartient à la liste L_u, le procédé s'arrête et la transaction entre les terminaux 20 et 22 n'est pas réalisée. En effet, comme déjà expliqué en regard de l'étape 124, cela signifie que les X Bitcoins associés à l'adresse $@_1$ ont déjà été dépensés. L'interruption de la transaction dans ce cas empêche donc la double-dépense. Si les clefs K_{pr1}, K_{pub1} et l'adresse $@_1$ n'appartiennent pas à la liste L_u, le procédé se poursuit par une étape 136.

[0093] Lors de l'étape 136, le cryptoprocesseur 28 construit une transaction T_2 ayant comme adresse de départ l'adresse $@_1$ et comme adresse d'arrivée l'adresse $@_2$ et un montant de X Bitcoins. Cette transaction est construite de la même façon que pour une transaction conventionnelle en-ligne. En particulier, elle est signée avec la clef K_{pr1}.

[0094] A la fin de l'étape 136, le cryptoprocesseur 28 transmet la transaction T_2 au terminal 22.

[0095] Lors d'une étape 138, une fois que la transaction T_2 a été transmise au terminal 22, le cryptoprocesseur 28 interdit la réalisation de toutes nouvelles transactions ayant pour objet le transfert des mêmes X Bitcoins que ceux transférés lors de l'étape 136. A cet effet, le cryptoprocesseur 28 procède comme décrit en référence à l'étape 128. Ainsi, l'adresse $@_1$, les clefs K_{pr1} et K_{pub1} et le montant transféré sont enregistrés dans la liste L_u de la mémoire 36.

[0096] Une fois que la transaction a été réalisée entre les terminaux 20 et 22, lors d'une étape 150, la liaison 40 est interrompue. Typiquement, lors de l'étape 150,

les terminaux 20 et 22 affichent chacun un message sur les écrans, respectivement, des interfaces homme-machine 32 et 52 pour indiquer aux utilisateurs de ces terminaux que la transaction a correctement été réalisée et que celle-ci est maintenant terminée.

[0097] Par la suite, dans le cas où l'option « transaction anonyme » avait été sélectionnée, lors d'une étape 152, le terminal 22 construit une transaction T_3 depuis l'adresse $@_1$ vers une autre adresse $@_3$. La transaction T_3 est signée avec la clef K_{pr1} reçue lors de l'étape 126. Ensuite, lorsque le terminal 22 se connecte à nouveau à l'ensemble 6 de serveurs, il transmet la transaction T_3 aux serveurs informatiques de l'ensemble 6 pour qu'elle soit enregistrée dans la chaîne de blocs. Le terminal 22 peut construire une telle transaction valide car il est en possession de la clef K_{pr1}.

[0098] Dans le cas où l'option « transaction non-anonyme » avait été sélectionnée, lors d'une étape 154, le terminal 22 se connecte aux serveurs informatiques de l'ensemble 6 et leur transmet la transaction T_2 reçue lors de l'étape 136. La transaction T_2 est alors vérifiée par ces serveurs informatiques puis enregistrée dans la chaîne de blocs.

[0099] En parallèle, après l'étape 150, à chaque fois que le terminal 20 souhaite réaliser une nouvelle transaction hors-ligne ou en-ligne avec un terminal d'encaissement, lors d'une étape 156, le cryptoprocesseur 28 vérifie si les Bitcoins que l'utilisateur souhaite dépenser ne correspondent pas à ceux déjà dépensés lors de la transaction hors-lignes. Par exemple, pour cela, avant de construire chaque nouvelle transaction, le cryptoprocesseur 28 procède d'abord comme décrit aux étapes 124 et 132. Ensuite, seulement si l'adresse de départ et les clefs publique/privée utilisées pour construire cette nouvelle transaction ne se trouvent pas déjà dans la liste L_u, alors cette nouvelle transaction est autorisée.

[00100] <u>Chapitre III : Variantes :</u>

[00101] <u>Variantes de la structure du système sécurisé de transactions :</u>

[00102] En variante, le cryptoprocesseur 28 n'est pas intégré à l'intérieur du terminal 20 mais relié à ce terminal 20 par l'intermédiaire d'une liaison d'échange d'informations. Dans ce cas, le cryptoprocesseur 28 peut être situé à plusieurs dizaines de centimètres ou dizaines de mètres ou centaines de mètres du terminal 20.

[00103] Le certificat C_{y22} du terminal 22 peut être signé par le terminal 22 et non pas par un fabricant connu du terminal 22 ou du microprocesseur 44.

[00104] Le terminal 22 peut lui aussi comporter un cryptoprocesseur tel que le cryptoprocesseur 28. Dans ce cas, les étapes décrites précédemment sont réalisées par le cryptoprocesseur du terminal 22 plutôt que par le microprocesseur 44.

[00105] De préférence, le terminal 22 vérifie l'indice de confiance du terminal 20 et/ou le certificat C_{y20} transmis par ce terminal 20. Dans le cas où l'indice de confiance du terminal 20 serait inférieur au seuil S_1 ou dans le cas où le certificat C_{y20} serait révoqué, le terminal 22 peut interrompre le procédé de la figure 4 de sorte que la transaction entre les terminaux 20 et 22 n'est pas réalisée. Par exemple, pour cela, le

cryptoprocesseur du terminal 22 réalise les mêmes opérations que celles décrites aux étapes 110 et 112 dans le cas particulier du cryptoprocesseur 28. De plus, si le terminal 22 est équipé d'un cryptoprocesseur il peut aussi être utilisé en tant que terminal de paiement. Par exemple, il est programmé pour être capable de fonctionner comme le terminal 20. Dans ce cas, il peut à son tour réaliser une transaction hors-ligne en utilisant comme moyen de paiement les X Bitcoins qu'il a précédemment reçu du terminal 20. A cet effet, dans le cas d'une transaction anonyme, il transmet hors-ligne la clef K_{pr1}, qu'il a précédemment reçue du terminal 20, vers un autre terminal d'encaissement. Dans le cas d'une transaction non-anonyme, le terminal 22 peut générer une autre transaction signée avec cette clé K_{pr1} associée à l'adresse $@_1$ encore non utilisée.

[00106] Le terminal de paiement 20 n'est pas obligé de vérifier que le certificat du terminal d'encaissement 22 n'est pas révoqué.

[00107] Le terminal de paiement 20 n'est pas obligé de vérifier l'indice de confiance du terminal 22.

[00108] La mémoire 36 peut être remplacée par une mémoire volatile sécurisée, par exemple, du type INTEL® SGK. Dans ce cas, le niveau de sécurité est moindre mais reste quand même acceptable. En effet, en cas de coupure de l'alimentation de la mémoire 36, la liste L_u, la clef K_{pr1} et la clef K_{pub1} sont effacées. Dès lors, le terminal 20 est dans l'incapacité de construire une nouvelle transaction signée utilisant comme adresse de départ l'adresse $@_1$ car la clef K_{pr1} a été effacée. Ainsi, la perte de la liste L_u suite à une coupure d'alimentation ne remet pas en cause complètement la sécurité du système de transactions.

[00109] De façon similaire, la mémoire 46 peut être remplacée par une mémoire volatile.

[00110] Le serveur 60 peut être omis. Dans ce cas, l'étape 112 est également omise.

[00111] Le serveur 62 peut être omis. Dans ce cas, l'étape 110 est omise.

[00112] Dans une variante simplifiée, la signature des messages échangés entre les terminaux 20 et 22 est omise.

[00113] Les messages échangés entre les terminaux 20 et 22 par l'intermédiaire de la liaison 40 peuvent être protégés par d'autres procédés de chiffrement comme, par exemple, un chiffrement symétrique au lieu d'un chiffrement asymétrique comme précédemment décrit.

[00114] Les messages échangés entre les terminaux 20 et 22 peuvent également comporter des informations additionnelles telle qu'une date courante, un prix, des informations sur un produit acheté ou autres.

[00115] Dans un mode de réalisation simplifié, les messages échangés entre les terminaux 20 et 22 par l'intermédiaire de la liaison 40 ne sont pas chiffrés. L'absence de chiffrement de la liaison 40 est surtout envisageable lorsque c'est la transaction T_2 qui est transmise du terminal 20 au terminal 22.

[00116] En variante, la liaison 40 n'est pas une liaison point-à-point mais une liaison point-à-multipoints. Par exemple, la liaison 40 est alors établies conformément au standard WIFI ou autre.

[00117] La liaison 40 peut aussi être établie par l'intermédiaire d'un réseau grande distance de transmission d'informations comme le réseau 4. Dans ce cas, les terminaux 20 et 22 n'ont pas besoin d'être situés à proximité l'un de l'autre pour mettre en œuvre le procédé décrit en référence à la figure 4.

[00118] Le terminal d'encaissement ou le terminal d'encaissement peut être composé de plusieurs parties mécaniquement indépendantes les unes des autres et raccordées les unes aux autres par l'intermédiaire de liaisons d'échanges d'information. Par exemple, seul l'émetteur/récepteur 50 est situé à proximité du terminal 20. Les autres composants du terminal 22 sont déportés à plusieurs mètres ou plusieurs dizaines de mètres de distance de l'émetteur/récepteur 50.

[00119] Variantes du procédé :

[00120] En variante, initialement, les étapes 100 à 104 sont réitérées plusieurs fois pour transférer des Bitcoins sur plusieurs adresses d'arrivée différentes. Dès lors, le cryptoprocesseur 28 peut construire hors-ligne :

- une transaction entre plusieurs de ces adresses d'arrivée et l'adresse $@_2$ fournie par le terminal 22 lors de l'étape 136, ou
- transmettre les clefs publique/privée et les adresses associées au terminal 20 lors de l'étape 126.

[00121] Dans une autre variante, lors de l'étape 102, la transaction T_1 construite a plusieurs sorties et donc plusieurs adresses d'arrivée. Dans ce cas, lors de l'étape 136, le cryptoprocesseur 28 peut construire une transaction T_2 ayant une seule ou plusieurs adresses de départ identifiant chacune une adresse d'arrivée respective de cette transaction T_1 et l'adresse d'arrivée $@_2$ fournie par le terminal 22. Dans ce mode de réalisation, les adresses de départ utilisées par la transaction T_2 sont enregistrées dans la liste L_u et une nouvelle utilisation de la clé K_{pr1} est interdite seulement si cette nouvelle utilisation consiste à construire une nouvelle transaction entre au moins une des adresses de départs déjà utilisées et une nouvelle adresse d'arrivée.

[00122] Lorsque la transaction T_1 a plusieurs adresses d'arrivée $@_i$, il est aussi possible d'associer à chacune de ces adresses d'arrivée une clé privée K_{pri} respectives. Dans ce cas, le transfert de la clé privée K_{pri} du terminal 20 vers le terminal 22 permet de transférer, de façon anonyme, le montant en Bitcoins associé à l'adresse $@_i$ correspondant à la clé K_{pri}. Ce qui a été décrit précédemment pour empêcher la double-dépense dans le cas du transfert de la clé K_{pr1} s'applique au transfert de chacune des clés K_{pri}.

[00123] Lors de l'étape 136, le cryptoprocesseur 28 peut également construire une transaction T_2 qui a plusieurs adresses d'arrivée. Par exemple, si l'adresse $@_1$ de départ est associée à X Bitcoins et que la transaction T_2 doit seulement transférer X-Y Bitcoins au terminal 22, alors le cryptoprocesseur 28 construit une transaction T_2 qui transfert X-Y Bitcoins depuis l'adresse $@_1$ vers l'adresse d'arrivée $@_2$ et qui

transfert Y Bitcoins depuis l'adresse $@_1$ vers l'adresse d'arrivée $@_1$. Lorsque cette transaction T_2 sera enregistrée ultérieurement dans la chaîne de blocs, les montants associés aux adresses $@_1$ et $@_2$ dans la chaîne de blocs deviendront, respectivement, Y et X-Y Bitcoins. Par exemple, tant que cette transaction T_2 n'a pas été enregistrée dans la chaîne de blocs et vérifiée par le cryptoprocesseur 28 lors d'une nouvelle itération de l'étape 104, les Y Bitcoins associés à l'adresse $@_1$ ne peuvent pas être dépensés hors-ligne car la transaction T_2 correspondante n'a pas encore été vérifiée. Toutefois, en variante, si le terminal d'encaissement est prêt à accepter le risque que le terminal de paiement ne se reconnecte jamais à l'ensemble 6 et donc que la transaction T_2 ne soit jamais enregistrée dans la chaîne de blocs, les Y Bitcoins associés à l'adresse $@_1$ peuvent être dépensés hors-ligne. Les Y Bitcoins peuvent aussi être transférés vers une adresse $@_4$ associé à une clé privée K_{pr4} générée au préalable par le terminal 20. Dans ce dernier cas, les Y Bitcoins associés à l'adresse $@_4$ peuvent être transférés hors-ligne à un autre terminal d'encaissement par l'intermédiaire d'une transaction anonyme comme précédemment décrit si le terminal d'encaissement est prêt à accepter le risque que le terminal de paiement ne se reconnecte jamais à l'ensemble 6 et donc que la transaction T_2 ne soit jamais enregistrée dans la chaîne de blocs. Si le terminal de paiement ne se reconnecte pas, les Y Bitcoins restant seront aussi perdus pour le propriétaire du terminal de paiement. La double-dépense est bien interdite y compris dans ce cas.

[00124] La transmission initiale d'une clef publique K_{pub20} peut aussi être réalisée par d'autres moyens que par l'intermédiaire de la liaison 40. Par exemple, le terminal 20 affiche sur son écran un QR-code contenant la clef K_{pub20} et le nombre N_{a1}. Le terminal 22 scanne alors ce QR-code pour acquérir ces données.

[00125] Le nombre N_{a1} peut aussi être transmis par téléphone. Par exemple, le nombre N_{a1} est affiché sur l'écran du terminal 20. Ensuite, l'utilisateur du terminal 20 appelle l'utilisateur du terminal 22 et lui communique par téléphone le numéro affiché sur l'écran du terminal 20. En réponse, pour que la transaction hors-ligne soit réalisée, l'utilisateur du terminal 22 saisit dans le terminal 22 le nombre N_{a1} qui lui a été communiqué verbalement. Dans ce dernier cas, les terminaux 20 et 22 n'ont pas besoin d'être à proximité l'un de l'autre pour réaliser une transaction hors-ligne. Il est aussi possible de transmettre le nombre N_{a1} par SMS (« Small Message Service») ou autres.

[00126] L'étape 118 peut être omise. Dans ce cas, les messages échangés entre les terminaux 20 et 22 ne comportent pas le nombre N_{a1}.

[00127] L'étape 126 peut être simplifiée. Par exemple, lors de cette étape, seule la clef K_{pr1} est transmise au terminal 22. Puis, à partir de la clef K_{pr1} reçue, le terminal 22 génère la clef publique K_{pub1} et l'adresse $@_1$. En effet, dans le système Bitcoin, les algorithmes de génération, à partir d'une clef privée, de la clef publique correspondante et de l'adresse Bitcoin sont des algorithmes publiques connus de tous les terminaux. En particulier, il n'est pas absolument nécessaire de transférer, en plus de la clé K_{pr1}, le montant transféré de Bitcoins.

[00128] Pour identifier les X Bitcoins transférés, d'autres informations que celles précédemment décrites peuvent être enregistrées dans la liste L_u. Par exemple, selon une première variante, seule la clé K_{pr1} ou la clé K_{pub1} ou l'adresse $@_1$ est enregistrée dans la liste L_u. Selon une seconde variante, l'identifiant des X Bitcoins transférés est obtenu en appliquant une transformation bijective à au moins l'un des éléments choisis dans le groupe composé de la clé K_{pr1}, la clé K_{pub1} et l'adresse $@_1$. Dans le cas d'une transaction non-anonyme, la transaction T_2 peut être enregistrée dans la liste L_u. A l'inverse, il est aussi possible d'enregistrer dans la liste L_u moins d'éléments que ce qui a été décrit précédemment. Par exemple, en variante, le montant de la transaction n'est pas enregistré dans la liste L_u. Ainsi, il existe de nombreux modes de réalisation possibles des étapes 128 et 138. Les étapes, telles que les étapes 124, 132 et 156 de vérification, doivent être adaptées au mode de réalisation implémenté des étapes 128 et 138.

[00129] En variante, les adresses d'arrivée, comme les adresses $@_1$ et $@_2$, peuvent être générées uniquement à partir de la clef privée de la paire de clefs publique/privée ou uniquement à partir de la clef publique de cette paire ou à la fois à partir de la clef privée et de la clef publique de cette paire.

[00130] Le terminal 22 peut générer les clefs K_{pub2} et K_{pr2} et l'adresse $@_2$ avant l'étape 130. Par exemple, les clefs K_{pub2} et K_{pr2} et l'adresse $@_2$ sont générées avant que le terminal 22 se déconnecte de l'ensemble 6 de serveur ou avant que la liaison 40 soit établie. Dans ce cas, l'étape 130 consiste simplement à obtenir, depuis la mémoire 46, les clefs K_{pub2} et K_{pr2} et l'adresse $@_2$ précédemment générées.

[00131] L'étape 154 peut être déclenchée automatiquement dès que le terminal 22 se connecte pour la première fois aux serveurs informatiques de l'ensemble 6 après l'achèvement de l'étape 138.

[00132] Dans une autre variante, la transaction T_2 construite par le terminal 20 est aussi enregistrée dans la mémoire 36. Ensuite, dès que le terminal 20 se connecte à un serveur informatique de l'ensemble 6, le cryptoprocesseur 28 transmet automatiquement lui aussi cette transaction T_2 à cet ensemble 6 de serveurs informatiques. On notera que cela ne pose aucun problème de transmettre deux fois la même transaction T_2 à l'ensemble 6 de serveurs informatiques. En effet, seule la transaction T_2 reçue en premier sera alors enregistrée dans la chaîne de blocs et toutes les copies de cette transaction T_2 reçues ultérieurement seront ignorées.

[00133] En variante, une transaction hors-ligne peut aussi être réalisée entre les terminaux 20 et 22 même si, au même instant, ces terminaux sont connectés ou ont la possibilité de se connecter à l'ensemble 6 de serveurs. Par exemple, dans ce cas, lors de l'étape 116, l'utilisateur a le choix entre trois options :

- réaliser une transaction hors-ligne « anonyme » comme précédemment décrit,
- réaliser une transaction hors-ligne « non-anonyme » comme précédemment décrit, et
- réaliser une transaction en-ligne de façon conventionnelle.

[00134] Dans une autre variante, l'option « transaction anonyme » est omise. Dans ce cas, les étapes 116, 124, 126, 128 et 152 sont omises. À l'inverse, l'option « transaction non-anonyme » peut être omise. Dans ce cas, les étapes 116, 130 à 138 et 154 sont omises.

[00135] Pour libérer de l'espace dans la mémoire 36, les clefs K_{pr1}, K_{pub1} et l'adresse $@_1$ peuvent être effacées, par exemple, lorsque le cryptoprocesseur 28 constate que la transaction T_2 a été enregistrée dans la chaîne de blocs. A partir de ce moment là, le cryptoprocesseur 28 peut aussi effacer les clefs K_{pr1}, K_{pub1} et l'adresse $@_1$ et/ou la transaction T_2 de la liste L_u.

[00136] Les modes de réalisation précédents ont été décrits dans le cas particulier d'un système conforme aux spécifications du système Bitcoin. Toutefois, l'enseignement donné ici peut être transposé sans difficulté à tout système de transactions entre terminaux utilisant une chaîne de blocs de manière similaire à ce qui est décrit dans le cas du système Bitcoin. En particulier, ce qui a été décrit ici s'applique à tout système de transactions utilisant une chaîne de blocs dans laquelle sont enregistrées les transactions, les transactions ayant chacune une adresse de départ et une adresse d'arrivée et un identifiant d'objet transféré entre ces adresses. Par exemple, on connaît le système de transfert de la crypto-monnaie connue sous le terme de « Ether ». Ce système est appelé par la suite système « Ethereum ». Les principes de fonctionnement du système Ethereum sont proches de ceux mis en œuvre dans le système Bitcoin. Ainsi, ce qui a été décrit ici peut aussi être implémenté dans le système Ethereum. Dans le système Ethereum, les adresses d'arrivée et de départ sont des adresses de comptes qui peuvent être crédités ou débités. Le montant d'Ethers disponible sur un compte est égal à la différence entre les entrées et les sorties d'Ethers sur ce compte. Dans le système Ethereum, pour empêcher la double-dépense, chaque transaction comporte un numéro d'ordre. Il n'est pas possible d'enregistrer dans la chaîne de blocs deux transactions depuis une même adresse de départ et ayant chacune le même numéro d'ordre. De plus, une transaction est invalide si le montant à débiter sur un compte est supérieur au montant disponible sur ce compte. Dans le système Ethereum, l'objet transféré est identifié, par exemple, par la combinaison de l'adresse de départ et du montant transféré et, éventuellement, du montant disponible après le transfert. C'est donc ces données qui sont enregistrées dans la liste L_u du terminal de paiement une fois que la transaction hors-ligne entre les terminaux a été réalisée. De préférence, le numéro d'ordre de la transaction hors-ligne réalisée est également enregistré dans la mémoire 36. Le montant transféré est utile pour vérifier, même dans le cas de transactions hors-lignes, que le cumul des transactions réalisées ne dépasse pas le montant d'Ether disponible sur le compte. Ensuite, pour empêcher la double-dépense, la transmission de la transaction T_2 au terminal 22 est autorisée uniquement si le montant disponible à cette adresse de départ est supérieur ou égal au montant transféré par cette transaction T_2, le montant disponible à l'adresse de départ étant obtenu à partir de la liste L_u. Pour cela, par exemple, la transaction T_2 est d'abord

construite et la vérification qu'elle ne conduit pas à dépenser plus que le montant disponible est réalisée après. En particulier, on notera que dans le système Ethereum la même clé privée K_{pr1} peut être utilisée pour signer plusieurs transactions valides à partir de la même adresse $@_1$ de départ à partir du moment où :

- les numéros de transactions affectés à chacune de ces transactions ne sont pas identiques, et que
- le montant cumulé de ces transactions ne dépasse pas le montant disponible sur le compte identifié par l'adresse $@_1$.

C'est pourquoi lorsque l'option anonyme est choisie dans le cas d'Ethereum, il est utile de transférer non seulement la clé privée mais aussi le numéro de transaction et le montant disponible afin de les enregistrer dans le terminal de paiement pour pouvoir réaliser des transactions hors-lignes après réception.

[00137] Jusqu'à présent, les systèmes sécurisés de transactions ont été décrits dans le cas particulier où l'objet de la transaction est de la crypto-monnaie. Toutefois, ce qui a été décrit fonctionne quel que soit l'objet de la transaction. Par exemple, l'identifiant d'une somme en crypto-monnaie décrit jusqu'à présent peut être remplacé par l'identifiant d'un titre de propriété sur un bien matériel ou immatériel tel qu'un brevet ou l'identifiant d'un contrat ou l'identifiant de tout autre objet pouvant faire l'objet d'une transaction entre deux terminaux.

[00138] Chapitre IV : Avantages des modes de réalisation décrits ici :

[00139] Le système 2 permet de sécuriser une transaction hors-ligne réalisée entre les terminaux 20 et 22 sans qu'il soit nécessaire pour cela que l'identité ou une adresse d'arrivée du terminal 22 soit connue avant que les terminaux 20 et 22 se déconnectent de l'ensemble 6 de serveurs.

[00140] Ce système 2 permet aussi bien de réaliser des transactions en-lignes que hors-lignes, sans qu'il soit pour cela nécessaire de modifier les procédés exécutés par l'ensemble 6 de serveurs. En d'autres termes, ce système 2 conserve les avantages des systèmes existants. En particulier, les transactions hors-lignes sont aussi sécurisées que les transactions en-lignes et la double-dépense est rendue impossible. On notera aussi que le système fonctionne même si le terminal d'encaissement est dépourvu de cryptoprocesseur et cela sans remettre en cause la sécurité des transactions réalisées entre les terminaux 20 et 22.

[00141] Le fait de transférer la clé privée K_{pr1} et, si nécessaire l'adresse $@_1$, permet de réaliser une transaction anonyme entre les terminaux 20 et 22 et cela tout en restant compatible avec le système Bitcoin actuel.

[00142] Le fait de chiffrer les messages échangés entre les terminaux 20 et 22 par l'intermédiaire de la liaison 40 augmente la sécurité du système.

[00143] Le fait de signer les messages échangés entre les terminaux 20 et 22 augmente également la sécurité du système.

REVENDICATIONS

1. Système sécurisé de transactions entre terminaux, ce système comportant :

- un ensemble (6) de serveurs programmés pour vérifier et, après vérification, enregistrer des transactions dans une chaîne de blocs, chaque bloc comportant plusieurs transactions et une empreinte numérique d'un bloc précédent dans la chaîne de blocs, chaque transaction étant un message numérique qui contient au moins les informations suivantes :
  - au moins une adresse d'arrivée de cette transaction, cette adresse d'arrivée ayant été construite à partir d'une paire de clefs publique/privée contenant une clef privée et une clef publique correspondant à cette clef privée,
  - au moins une adresse de départ qui identifie sans ambiguïté une adresse d'arrivée d'une transaction précédemment enregistrée dans la chaîne de blocs,
  - un identifiant d'objet transféré depuis l'adresse de départ vers l'adresse d'arrivée,
  - une signature numérique de la transaction obtenue en signant au moins les adresses de départ et d'arrivée avec la clef privée, cette signature permettant à l'ensemble de serveurs de vérifier que cette transaction a bien été générée par un terminal ayant accès à cette clef privée,
- un terminal (20) de paiement apte à communiquer avec l'ensemble de serveurs, ce terminal de paiement étant équipé :
  - d'un cryptoprocesseur (28), ce cryptoprocesseur comportant une mémoire sécurisée (36) uniquement accessible par le cryptoprocesseur,
  - d'un émetteur/récepteur (30) apte à établir une liaison d'échange d'informations avec un autre terminal,
- un terminal (22) d'encaissement apte à communiquer avec l'ensemble de serveurs, ce terminal d'encaissement étant équipé :
  - d'un microprocesseur (44),
  - d'une mémoire (46), et
  - d'un émetteur/récepteur (50) apte à établir une liaison d'échange d'informations avec un autre terminal,
- l'ensemble (6) de serveurs est apte à enregistrer une première transaction contenant une première adresse d'arrivée et un premier identifiant d'objet transféré vers cette première adresse d'arrivée,
- la mémoire sécurisée (36) comporte une première paire de clefs publique/privée à partir de laquelle la première adresse d'arrivée a été générée, cette première paire de clefs publique/privée comportant :
  - une première clef privée qui est la seule à pouvoir valablement signer une seconde transaction ayant une adresse de départ qui identifie sans ambiguïté la première adresse d'arrivée, et
  - une première clef publique correspondant à la première clef privée,

- le terminal (22) d'encaissement est apte à obtenir et à enregistrer dans sa mémoire (46), une seconde adresse d'arrivée et une seconde paire de clefs publique/privée à partir de laquelle la seconde adresse d'arrivée a été générée, cette seconde paire de clefs publique/privée comportant une seconde clef privée et une seconde clef publique correspondant à cette seconde clef privée,
- le cryptoprocesseur (28) est apte, en réponse à la réception de la seconde adresse d'arrivée, à construire la seconde transaction, signée avec la première clef privée, entre une adresse de départ identifiant sans ambiguïté la première adresse d'arrivée et cette seconde adresse d'arrivée,
- les terminaux (20, 22) de paiement et d'encaissement sont aptes à transmettre toutes transactions construites ou reçues à l'ensemble (6) de serveurs lorsqu'ils sont connectés à cet ensemble de serveurs,
- l'ensemble (6) de serveurs est apte, en réponse à la réception de toute transaction :
    - à vérifier la validité de cette transaction reçue à l'aide de la signature qu'elle contient,
    - à vérifier l'absence de double-dépense en vérifiant qu'il n'existe pas une transaction du même objet depuis la même adresse de départ déjà enregistrée dans la chaîne de blocs, et
    - seulement si ces vérifications confirment la validité de la transaction reçue et l'absence de double-dépense, à enregistrer la transaction reçue dans la chaîne de blocs,
- même en absence de connexion avec l'ensemble de serveurs, les terminaux (20, 22) de paiement et d'encaissement sont aptes à établir, par l'intermédiaire de leurs émetteurs/récepteurs respectifs, une liaison (40) d'échange d'informations entre eux,

caractérisé en ce que, même en absence de connexion avec l'ensemble de serveurs, le cryptoprocesseur (28) est apte :

- à transmettre au terminal (22) d'encaissement, par l'intermédiaire de la liaison d'échange d'informations, la première clef privée ou la seconde transaction signée construite par le cryptoprocesseur, puis
- à enregistrer, dans la mémoire sécurisée, un second identifiant de l'objet transféré par la seconde transaction propre à empêcher tout nouveau transfert du même objet à partir de la première adresse d'arrivée, et ensuite
- lorsqu'une nouvelle transaction est construite par le cryptoprocesseur, à comparer l'identifiant de l'objet transféré par cette nouvelle transaction au second identifiant de l'objet transféré enregistré dans la mémoire sécurisée, et
- seulement dans le cas où l'identifiant de l'objet transféré par cette nouvelle transaction correspond au second identifiant de l'objet transféré enregistré dans la mémoire sécurisée, à interdire la transmission de cette nouvelle transaction vers le terminal d'encaissement.

2. Système selon la revendication 1, dans lequel :

- le microprocesseur du terminal d'encaissement est un cryptoprocesseur, ce cryptoprocesseur comportant une mémoire sécurisée uniquement accessible par le cryptoprocesseur,
- la mémoire du terminal d'encaissement est la mémoire sécurisée du cryptoprocesseur du terminal d'encaissement.

3. Système selon la revendication 1 ou 2, dans lequel l'ensemble de serveurs est apte à vérifier l'absence de double-dépense en vérifiant seulement que l'adresse de départ de la transaction reçue n'est pas égale à une adresse de départ déjà enregistrée dans la chaîne de blocs.

4. Terminal (20) de paiement pour la réalisation d'un système conforme à l'une quelconque des revendications 1 à 3, dans lequel le terminal (20) de paiement est apte à communiquer avec l'ensemble de serveurs, ce terminal de paiement étant équipé :

- d'un cryptoprocesseur (28), ce cryptoprocesseur comportant une mémoire sécurisée (36) uniquement accessible par le cryptoprocesseur,
- d'un émetteur/récepteur (30) apte à établir une liaison d'échange d'informations avec un autre terminal,

- la mémoire sécurisée (36) est apte à comporter une première paire de clefs publique/privée à partir de laquelle la première adresse d'arrivée a été générée, cette première paire de clefs publique/privée comportant :
  - une première clef privée qui est la seule à pouvoir valablement signer une seconde transaction ayant une adresse de départ qui identifie sans ambiguïté la première adresse d'arrivée, et
  - une première clef publique correspondant à la première clef privée,
- le cryptoprocesseur (28) est apte, en réponse à la réception de la seconde adresse d'arrivée, à construire la seconde transaction, signée avec la première clef privée, entre une adresse de départ identifiant sans ambiguïté la première adresse d'arrivée et cette seconde adresse d'arrivée,
- le terminal (20) de paiement est apte à transmettre toutes transactions construites à l'ensemble (6) de serveurs lorsqu'il est connecté à cet ensemble de serveurs,
- même en absence de connexion avec l'ensemble de serveurs, le terminal (20) de paiement est apte à établir, par l'intermédiaire de son émetteur/récepteur, une liaison (40) d'échange d'informations avec le terminal (22) d'encaissement,

caractérisé en ce que, même en absence de connexion avec l'ensemble de serveurs, le cryptoprocesseur (28) est apte :

- à transmettre au terminal (22) d'encaissement, par l'intermédiaire de la liaison (40) d'échange d'informations, la première clef privée ou la seconde transaction signée construite par le cryptoprocesseur, puis

- à enregistrer, dans le mémoire sécurisée (36), un second identifiant de l'objet transféré par la seconde transaction propre à empêcher tout nouveau transfert du même objet à partir de la première adresse d'arrivée, et ensuite
- lorsqu'une nouvelle transaction est construite par le cryptoprocesseur (28), à comparer l'identifiant de l'objet transféré par cette nouvelle transaction au second identifiant de l'objet transféré enregistré dans la mémoire sécurisée, et
- seulement dans le cas où l'identifiant de l'objet transféré par cette nouvelle transaction correspond au second identifiant de l'objet transféré enregistré dans la mémoire sécurisée, à interdire la transmission de cette nouvelle transaction vers le terminal d'encaissement.

5. Terminal selon la revendication 4, dans lequel le cryptoprocesseur (28) est apte à transmettre au terminal (22) d'encaissement, par l'intermédiaire de la liaison (40) d'échange d'informations, la première clef privée.

6. Terminal selon l'une quelconque des revendications 4 à 5, dans lequel le terminal de paiement est apte à chiffrer la transmission, par l'intermédiaire de la liaison d'échange d'informations, de la première clef privée ou de la seconde transaction signée.

7. Terminal selon l'une quelconque des revendications 4 à 6, dans lequel le terminal de paiement est apte à signer la transmission, par l'intermédiaire de la liaison d'échange d'informations, de la première clef privée ou de la seconde transaction signée.

8. Terminal selon l'une quelconque des revendications 4 à 7, dans lequel le cryptoprocesseur (28) est apte :
- en absence de connexion avec l'ensemble de serveurs, à construire et à enregistrer dans sa mémoire sécurisée la première transaction, puis
- après la transmission de la première clef privée ou de la seconde transaction signée, en réponse à une connexion avec l'ensemble de serveurs, à transmettre automatiquement la première transaction enregistrée à l'ensemble de serveurs.

9. Terminal selon l'une quelconque des revendications 4 à 8, dans lequel la mémoire sécurisée est une mémoire non-volatile.

10. Procédé sécurisé de transactions entre terminaux destiné à être mis en œuvre dans un système conforme à l'une quelconque des revendications 1 à 3, dans lequel :
- l'ensemble de serveurs enregistre (104) une première transaction contenant une première adresse d'arrivée et un premier identifiant d'objet transféré vers cette première adresse d'arrivée,

- le cryptoprocesseur du terminal de paiement enregistre (102) dans sa mémoire sécurisée une première paire de clefs publique/privée à partir de laquelle la première adresse d'arrivée a été générée, cette première paire de clefs publique/privée comportant :
  - une première clef privée qui est la seule à pouvoir valablement signer une seconde transaction ayant une adresse de départ qui identifie sans ambiguïté la première adresse d'arrivée, et
  - une première clef publique correspondant à la première clef privée,
- le terminal d'encaissement obtient et enregistre (130) dans sa mémoire, une seconde adresse d'arrivée et une seconde paire de clefs publique/privée à partir de laquelle la seconde adresse d'arrivée a été générée, cette seconde paire de clefs publique/privée comportant une seconde clef privée et une seconde clef publique correspondant à cette seconde clef privée,
- en absence de connexion avec l'ensemble de serveurs, les terminaux (20, 22) de paiement et d'encaissement établissent (110), par l'intermédiaire de leurs émetteurs/récepteurs respectifs, une liaison d'échange d'informations entre eux,

caractérisé en ce que, même en absence de connexion avec l'ensemble de serveurs, :

- le cryptoprocesseur transmet (126, 136) au terminal d'encaissement, par l'intermédiaire de la liaison d'échange d'informations, la première clef privée ou la seconde transaction signée construite par le cryptoprocesseur, puis
- le cryptoprocesseur enregistre (128, 138), dans la mémoire sécurisée, un second identifiant de l'objet transféré par la seconde transaction propre à empêcher tout nouveau transfert du même objet à partir de la première adresse d'arrivée, et ensuite
- lorsqu'une nouvelle transaction est construite par le cryptoprocesseur, le cryptoprocesseur compare (124, 132, 156) l'identifiant de l'objet transféré par cette nouvelle transaction au second identifiant de l'objet transféré enregistré dans la mémoire sécurisée, et
- seulement dans le cas où l'identifiant de l'objet transféré par cette nouvelle transaction correspond au second identifiant de l'objet transféré enregistré dans la mémoire sécurisée, le cryptoprocesseur interdit (124, 132, 156) la transmission de cette nouvelle transaction vers le terminal d'encaissement.

11. Support (36) d'enregistrement d'informations, lisible par un cryptoprocesseur, caractérisé en ce que ce support d'enregistrement d'informations comporte des instructions pour la mise en œuvre d'un procédé conforme à la revendication 10 lorsque ces instructions sont exécutées par le cryptoprocesseur.

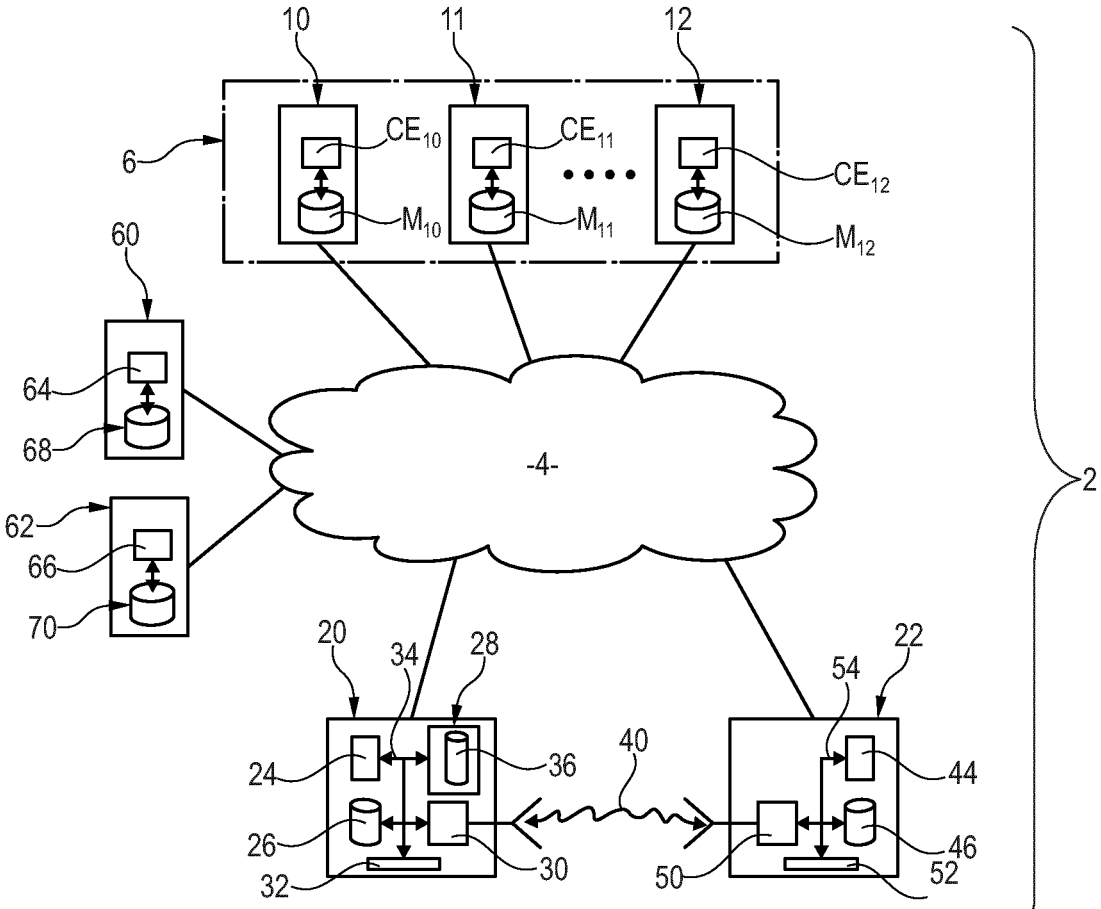

Fig. 1

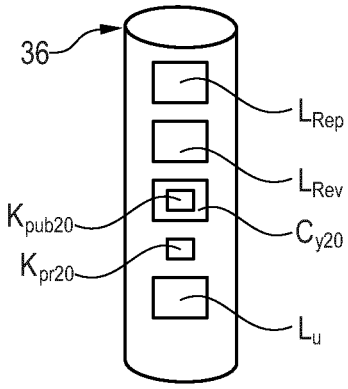

Fig. 2

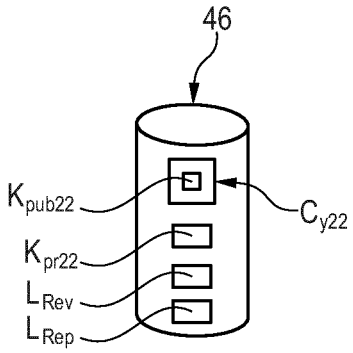

Fig. 3

Fig. 4

RAPPORT DE RECHERCHE

articles L.612-14, L.612-53 à 69 du code de la propriété intellectuelle

OBJET DU RAPPORT DE RECHERCHE

L'I.N.P.I. annexe à chaque brevet un "RAPPORT DE RECHERCHE" citant les éléments de l'état de la technique qui peuvent être pris en considération pour apprécier la brevetabilité de l'invention, au sens des articles L. 611-11 (nouveauté) et L. 611-14 (activité inventive) du code de la propriété intellectuelle. Ce rapport porte sur les revendications du brevet qui définissent l'objet de l'invention et délimitent l'étendue de la protection.
Après délivrance, l'I.N.P.I. peut, à la requête de toute personne intéressée, formuler un "AVIS DOCUMENTAIRE" sur la base des documents cités dans ce rapport de recherche et de tout autre document que le requérant souhaite voir prendre en considération.

CONDITIONS D'ETABLISSEMENT DU PRESENT RAPPORT DE RECHERCHE

[x] Le demandeur a présenté des observations en réponse au rapport de recherche préliminaire.

[x] Le demandeur a maintenu les revendications.

☐ Le demandeur a modifié les revendications.

☐ Le demandeur a modifié la description pour en éliminer les éléments qui n'étaient plus en concordance avec les nouvelles revendications.

☐ Les tiers ont présenté des observations après publication du rapport de recherche préliminaire.

☐ Un rapport de recherche préliminaire complémentaire a été établi.

DOCUMENTS CITES DANS LE PRESENT RAPPORT DE RECHERCHE

La répartition des documents entre les rubriques 1, 2 et 3 tient compte, le cas échéant, des revendications déposées en dernier lieu et/ou des observations présentées.

[x] Les documents énumérés à la rubrique 1 ci-après sont susceptibles d'être pris en considération pour apprécier la brevetabilité de l'invention.

☐ Les documents énumérés à la rubrique 2 ci-après illustrent l'arrière-plan technologique général.

☐ Les documents énumérés à la rubrique 3 ci-après ont été cités en cours de procédure, mais leur pertinence dépend de la validité des priorités revendiquées.

☐ Aucun document n'a été cité en cours de procédure.

1. ELEMENTS DE L'ETAT DE LA TECHNIQUE SUSCEPTIBLES D'ETRE PRIS EN CONSIDERATION POUR APPRECIER LA BREVETABILITE DE L'INVENTION

Andreas M. Antonopoulos: "Mastering Bitcoin - Unlocking Digital Cryptocurrencies" In: "Mastering bitcoin : [unlocking digital cryptocurrencies]", 20 décembre 2014 (2014-12-20), O'Reilly Media, Beijing Cambridge Farnham Köln Sebastopol Tokyo, XP055306939, ISBN: 978-1-4493-7404-4

Melanie Swan: "Blockchain: Blueprint for a New Economy" In: "Blockchain: Blueprint for a New Economy", 8 février 2015 (2015-02-08), O'Reilly, XP055279098, ISBN: 978-1-4919-2049-7

WO 2017/145008 A1 (NCHAIN HOLDINGS LTD [AG]) 31 août 2017 (2017-08-31)

EP 2 953 076 A1 (MONI LTD [GB]) 9 décembre 2015 (2015-12-09)

EBERHARDT JACOB ET AL: "On or Off the Blockchain? Insights on Off-Chaining Computation and Data", 1 septembre 2017 (2017-09-01), MEDICAL IMAGE COMPUTING AND COMPUTER-ASSISTED INTERVENTION - MICCAI 2015 : 18TH INTERNATIONAL CONFERENCE, MUNICH, GERMANY, OCTOBER 5-9, 2015; PROCEEDINGS; [LECTURE NOTES IN COMPUTER SCIENCE; LECT.NOTES COMPUTER], SPRINGER INTERNATIONAL PUBLISHING, CH, XP047446752, ISSN: 0302-9743 ISBN: 978-3-642-38287-1 [extrait le 2017-09-01]

2. ELEMENTS DE L'ETAT DE LA TECHNIQUE ILLUSTRANT L'ARRIERE-PLAN TECHNOLOGIQUE GENERAL

NEANT

3. ELEMENTS DE L'ETAT DE LA TECHNIQUE DONT LA PERTINENCE DEPEND DE LA VALIDITE DES PRIORITES

NEANT