(19) CONFÉDÉRATION SUISSE
INSTITUT FÉDÉRAL DE LA PROPRIÉTÉ INTELLECTUELLE

(11) **CH 716 287 A2**

(51) Int. Cl.: ***G06F 21/62*** *(2013.01)*
G06F 21/57 *(2013.01)*
H04L 9/06 *(2006.01)*
H04L 9/08 *(2006.01)*
H04L 29/08 *(2006.01)*
G06K 9/00 *(2006.01)*

Demande de brevet pour la Suisse et le Liechtenstein
Traité sur les brevets, du 22 décembre 1978, entre la Suisse et le Liechtenstein

(12) **DEMANDE DE BREVET**

(21) Numéro de la demande: 00759/19

(22) Date de dépôt: 07.06.2019

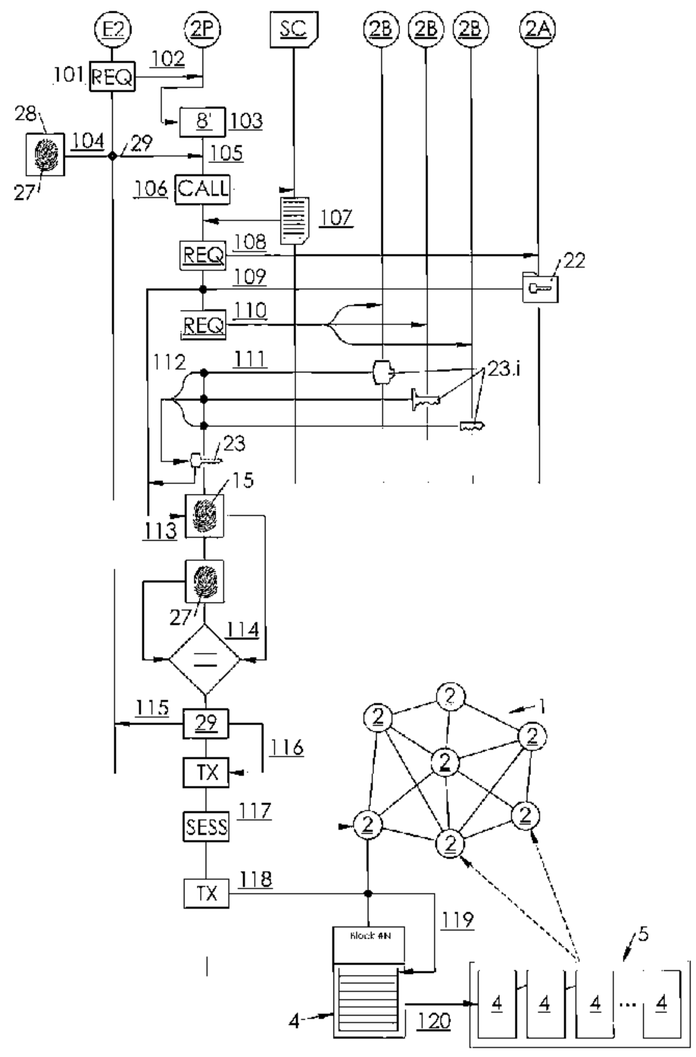
(43) Demande publiée: 15.12.2020

(71) Requérant:
Lapsechain SA C/O Leax Avocats,
Faubourg de l'Hôpital 18
2000 Neuchâtel (CH)

(72) Inventeur(s):
Jonathan Attia, 2000 Neuchâtel (CH)
Raphaël Louiset, 2072 Saint-Blaise (CH)

(54) **Procédé de traitement de données biométriques d'un individu, avec phases de calibrage et de contrôle et inscription, dans une blockchain, d'un résultat d'analyse.**

(57) L'invention concerne un procédé de traitement des données biométriques d'un individu, à partir d'un émetteur (**E2**) relié à un réseau (**1**) pair-à-pair sur lequel est déployée une blockchain (**5**), ce procédé comprenant la distribution préalable, sur le réseau (**1**), d'un conteneur (**22**) crypté contenant des données (**15**) biométriques de référence puis, ultérieurement, la comparaison de données (**27**) scannées au niveau de l'émetteur (**E2**) avec les données (**15**) de référence contenues dans le conteneur (**22**) crypté ; le résultat de la comparaison est inscrit dans la blockchain (**5**).



Description

DOMAINE TECHNIQUE

[0001] L'invention a trait au domaine de l'informatique, et plus précisément au traitement des données biométriques d'un individu.

ART ANTERIEUR

[0002] Les données biométriques des individus (typiquement une empreinte digitale ou de la paume, une empreinte rétinienne ou de l'iris, le réseau veineux de la main, une image du visage), tendent à se généraliser en tant que données d'authentification, notamment à l'usage des systèmes de contrôle d'accès à un environnement physique (par ex. local, coffre-fort) ou virtuel (typiquement une session sur un ordinateur, une tablette ou encore un smartphone).

[0003] Classiquement, des données biométriques de référence, capturées lors d'une session de paramétrage, sont stockées dans une base de données, puis, sur requête, sont appelées pour être comparées à des données biométriques instantanées, capturées localement à partir d'un terminal sur un individu souhaitant accéder à un environnement donné.

[0004] Lorsque plusieurs individus („utilisateurs“) sont présumés autorisés à accéder à un même environnement, une technique classique consiste à mémoriser, dans la base de données, autant de données biométriques que d'individus bénéficiant d'une autorisation d'accès.

[0005] Dans une première version, dite locale, la base de données est locale, c'est-à-dire qu'elle est stockée dans un espace mémoire équipant le (ou directement relié au) terminal à partir duquel est réalisée la capture. Cette technique peut paraître à l'abri des intrusions (et donc des usurpations d'identité) en raison de son caractère local, mais il est le plus souvent nécessaire de prévoir un accès réseau à la base de données, aux fins d'administration (y compris l'ajout ou le retrait d'utilisateurs. Il en résulte que la base de données peut être piratée.

[0006] Dans une deuxième version, dite délocalisée, la base de données est stockée dans un espace mémoire réservé dans un serveur distant auquel, à chaque sollicitation par un utilisateur, le terminal se connecte pour comparer les données biométriques capturées aux données biométriques de référence.

[0007] Cette deuxième version présente l'avantage de permettre une administration à distance des autorisations associées aux utilisateurs. Cependant, elle repose, d'une part, sur la politique de sécurité informatique à laquelle est soumis le serveur distant sur lequel est stockée la base de données ; d'autre part, sur la confiance que l'on peut accorder à l'administrateur dudit serveur.

[0008] Il existe par ailleurs de nombreux services pour lesquels les données biométriques des individus sont stockées sur des serveurs distants pour être utilisées en tant que moyen d'authentification sur des comptes utilisateurs.

[0009] Dans tous les cas, il n'est jamais certain que les données biométriques des utilisateurs soient protégées des intrusions, de la copie, d'une éventuelle exploitation commerciale, ou encore de l'effacement.

[0010] Il existe par conséquent un besoin d'améliorer la confidentialité avec laquelle sont traitées les données biométriques des individus.

RESUME DE L'INVENTION

[0011] Il est proposé un procédé de traitement de données biométriques d'un individu, à partir d'une unité de traitement informatique, dite émetteur, reliée ou intégrée à un réseau pair-à-pair composé d'une pluralité de nœuds formant une base de données distribuée sur laquelle est mémorisée, par réplication sur chaque nœud, une chaîne de blocs, ce procédé comprenant :

A) Une phase de calibrage, qui comprend les opérations suivantes :

- Au moyen d'un dispositif de capture, réaliser une première capture de données biométriques, dites de référence, de l'individu au niveau d'un membre ou d'un organe de celui-ci ;

- Chiffrer les données biométriques issues de cette première capture pour former un conteneur crypté de données biométriques ;

- Mémoriser séparément le conteneur crypté de données biométriques et une clé cryptographique de déchiffrement de celui-ci ;

- Inscrire, dans un bloc de la chaîne de blocs, une transaction contenant une trace de cette capture et/ou de cette mémorisation ;

A) Une phase de contrôle, qui comprend les opérations suivantes :

- Etablir une session de communication entre l'émetteur et le réseau ;

- Sélectionner parmi le réseau un nœud dit de calcul, équipé d'une unité de traitement dans laquelle est implémenté un environnement d'exécution sécurisé par cryptographie, dit enclave ;

- Instancier l'enclave ;

- Au moyen d'un dispositif de capture équipant ou relié à l'émetteur, réaliser une capture de données biométriques de l'individu au niveau d'un membre ou d'un organe de cet individu ;

- Charger dans l'enclave, via une ligne de communication sécurisée, les données biométriques capturées ;

- Sélectionner parmi le réseau un nœud de stockage sur lequel est mémorisé un exemplaire du conteneur crypté contenant les données biométriques de référence ;

- Charger le conteneur crypté dans l'enclave ;

- Déchiffrer dans l'enclave les données biométriques de référence, au moyen d'une clé de déchiffrement associée au conteneur crypté ;

- Dans l'enclave, comparer les données biométriques capturées et les données de référence ;

- Inscrire dans un bloc de la chaîne de blocs une transaction comprenant une trace du résultat de cette comparaison.

[0012] Selon un mode préféré de réalisation, la phase de calibrage comprend les opérations suivantes :

– Fragmenter la clé cryptographique de déchiffrement ;
– Désigner, parmi le réseau, un groupe de nœuds de stockage en nombre égal aux fragments ;
– Distribuer chaque fragment de la clé cryptographique de déchiffrement vers un nœud de stockage, chaque nœud de stockage recevant un unique fragment.

[0013] Dans ce cas, la phase de contrôle comprend avantageusement les opérations suivantes :

- A partir de l'enclave du nœud de calcul, sélectionner parmi le réseau des nœuds sur lesquels sont stockés des fragments de la clé cryptographique de déchiffrement associée au conteneur ;

- Charger lesdits fragments dans l'enclave du nœud de calcul ;

- Dans l'enclave :

o Reconstituer la clé cryptographique de déchiffrement associée au conteneur, à partir des fragments ainsi chargés ;

o Déchiffrer les données biométriques du conteneur au moyen de la clé ainsi reconstituée.

BREVE DESCRIPTION DES FIGURES

[0014] D'autres objets et avantages de l'invention apparaîtront à la lumière de la description d'un mode de réalisation, faite ci-après en référence aux dessins annexés dans lesquels :

La **FIG.1** est un schéma fonctionnel simplifié illustrant un réseau pair-à-pair sur lequel est distribuée une chaîne de blocs ;

La **FIG.2** est un schéma fonctionnel simplifié illustrant différents composants d'une unité de traitement informatique impliqués dans la création et l'exploitation d'un environnement d'exécution sécurisé appelé enclave ;

La **FIG.3** est un schéma fonctionnel illustrant pour partie une architecture réseau, pour partie des étapes d'une phase de calibrage, et pour partie des fichiers produits, échangés ou stockés au sein du réseau pour les besoins (ou en application) de cette phase ;

La **FIG.4** est un schéma fonctionnel illustrant des étapes d'un procédé de traitement des données biométriques d'un individu.

DESCRIPTION DETAILLEE DE L'INVENTION

[0015] Le procédé proposé vise à traiter, de manière confidentielle, des données biométriques d'un individu.

[0016] Sans s'y restreindre, le procédé de traitement proposé exploite, en les combinant, des fonctionnalités offertes par deux technologies relativement récentes dont il paraît utile de faire une description préalable avant d'entrer dans les détails du procédé, à savoir :

– La technologie de la chaîne de blocs ou, en terminologie anglo-saxonne, blockchain (dans ce qui suit, on préférera la terminologie anglo-saxonne, en raison de son emploi courant dans la plupart des langues, y compris en langue française) ;
– La technologie de l'environnement d'exécution sécurisé ou, en terminologie anglo-saxonne, du trusted execution environment (TEE).

[0017] La technologie blockchain est organisée en couches. Elle comprend :

– Une couche d'infrastructure matérielle, appelée „réseau blockchain“ ;
– Une couche protocolaire appelée „protocole blockchain“ ;
– Une couche informationnelle, appelée „registre blockchain“.

[0018] Le réseau blockchain est un réseau informatique décentralisé, dit réseau pair-à-pair (en terminologie anglo-saxonne Peer-to-Peer ou P2P), constitué d'une pluralité d'ordinateurs (au sens fonctionnel du terme : il s'agit d'un appareil pourvu d'une unité de traitement informatique programmable, qui peut se présenter sous forme d'un smartphone, d'une tablette, d'un ordinateur de bureau, d'une station de travail, d'un serveur physique ou virtuel, c'est-à-dire un espace de calcul et de mémoire alloué au sein d'un serveur physique et sur lequel tourne un système d'exploitation ou une émulation de système d'exploitation), appelés „nœuds“ en référence à la théorie des graphes, capables de communiquer entre eux (c'est-à-dire de s'échanger des données informatiques), deux à deux, au moyen de liaisons filaires ou sans fil.

[0019] Un réseau **1** blockchain comprenant des nœuds **2** communiquant par des liaisons **3** est illustré sur la **FIG.1**. Par souci de simplification et de conformité à la théorie des graphes, sur la **FIG.1**, les nœuds **2** du réseau **1** sont représentés par des cercles ; les liaisons **3**, par des arêtes reliant les cercles. Pour ne pas surcharger de traits le dessin, seules certaines liaisons **3** entre les nœuds **2** sont représentées.

[0020] Les nœuds **2** peuvent être disséminés sur de larges régions géographiques ; ils peuvent également être regroupés dans des régions géographiques plus restreintes.

[0021] Le protocole blockchain se présente sous forme d'un programme informatique implémenté dans chaque nœud **2** du réseau **1** blockchain, et qui inclut, outre des fonctions de dialogue - c'est-à-dire d'échange des données informatiques - avec les autres nœuds **2** du réseau **1**, un algorithme de calcul qui, à partir de données d'entrée appelées „transactions“ (qui sont des transcriptions d'interactions entre un ou plusieurs terminaux informatiques émetteurs et un ou plusieurs terminaux informatiques destinataires) :

- Élabore des fichiers **4** de données structurées appelés „blocs“, chaque bloc **4** comprenant un corps **4A** contenant des empreintes numériques de transactions, et un en-tête **4B** contenant :

o Un numéro d'ordre, ou rang, ou encore hauteur (height en anglais), sous forme d'un nombre entier qui désigne la position du bloc **4** au sein d'une chaîne dans l'ordre croissant à partir d'un bloc initial (Genesis block en anglais) ;

o Une empreinte numérique unique des données du corps **4A** ;

o Une empreinte numérique unique, appelée pointeur, de l'en-tête du bloc **4** précédent,

o Une donnée d'horodatage (timestamp en anglais) ;

- Met en œuvre un mécanisme de validation des blocs **4** par consensus entre tout ou partie des nœuds **2** ;

- Concatène les blocs **4** validés pour former un registre **5** (le registre blockchain) sous forme d'un agrégat dans lequel chaque bloc **4** est relié mathématiquement au précédent par son pointeur.

[0022] La moindre modification des données du corps **4A** ou de l'en-tête **4B** d'un bloc **4** affecte la valeur de son empreinte numérique et rompt par conséquent le lien existant entre ce bloc **4** ainsi modifié et le bloc **4** suivant dont le pointeur ne correspond plus.

[0023] Selon un mode particulier de réalisation, l'empreinte numérique de chaque bloc **4** est un condensé (ou condensat, en anglais hash) des données du bloc **4**, c'est-à-dire le résultat d'une fonction de hachage appliquée aux données du bloc **4** (y compris le corps **4A** et l'en-tête **4B** à l'exception de l'empreinte numérique elle-même). La fonction de hachage est typiquement SHA-256.

[0024] Pour un bloc **4** donné de rang N (N un entier), le pointeur assure avec le bloc **4** précédent de rang N-1 une liaison inaltérable. En effet, toute modification des données du bloc **4** de rang N-1 aboutirait à la modification de son empreinte, et donc à un défaut de correspondance entre cette empreinte (modifiée) du bloc **4** de rang N-1 et le pointeur mémorisé parmi les métadonnées du bloc **4** de rang N.

[0025] La succession des blocs **4** reliés entre eux deux à deux par correspondance du pointeur d'un bloc **4** donné de rang N avec l'empreinte numérique du bloc précédent de rang N-1 constitue par conséquent le registre **45** blockchain sous forme d'un agrégat de blocs **4** corrélés, dans lequel la moindre modification des données d'un bloc **4** de rang N-1 se traduit par une rupture du lien avec le bloc **4** suivant de rang N - et donc la rupture du registre blockchain.

[0026] C'est cette structure particulière qui procure aux données contenues dans le registre **5** blockchain une réputation d'immuabilité, garantie par le fait que le registre **5** blockchain est répliqué sur tous les nœuds **2** du réseau **1**, obligeant tout attaquant, non seulement à modifier tous les blocs **4** de rang supérieur au bloc **4** modifié, mais à déployer ces modifications (alors même que le registre **5** blockchain continue de se constituer par les nœuds **2** appliquant le protocole blockchain) à l'ensemble des nœuds **2**.

[0027] Quel que soit le type de consensus appliqué par le mécanisme de validation des blocs **4**, la plupart des technologies blockchain ont pour fonction primaire d'enregistrer, dans leur registre **5** blockchain, des transactions passées entre un ou plusieurs terminaux émetteurs, et un ou plusieurs terminaux récepteurs, indifféremment appelés „utilisateurs“.

[0028] A chaque utilisateur est associé un compte, appelé de manière simplificatrice „portefeuille électronique“ (en anglais digital wallet), qui contient une zone mémoire et une interface programmatique ayant des fonctions d'interaction avec le réseau **1** blockchain pour lui soumettre des transactions, et des fonctions de synchronisation avec le registre **5** blockchain pour inscrire, dans la zone mémoire, les transactions validées par inscription dans le registre **5** blockchain.

[0029] Sauf mention contraire, et par souci de simplification, l'expression simple „chaîne de blocs“ ou „blockchain“ désigne le registre **5** blockchain lui-même.

[0030] Certaines technologies blockchain récentes (Ethereum, typiquement) ajoutent aux trois couches matérielle (réseau blockchain), protocolaire (protocole blockchain) et informationnelle (registre blockchain) une couche applicative qui se présente sous forme d'un environnement de développement permettant de programmer des applications, appelées „contrats intelligents“ (en anglais Smart contracts), qui peuvent être déployées sur le registre **5** blockchain à partir des nœuds **2**.

[0031] On décrit à présent succinctement la technologie des contrats intelligents.

[0032] Un contrat intelligent comprend deux éléments :

- Un compte, appelé „compte de contrat“ (en anglais Contract account), dans la zone mémoire duquel est inscrit un code source contenant des instructions informatiques implémentant les fonctions attribuées au contrat intelligent ;
- Un code exécutable (en anglais Executable Bytecode) résultant d'une compilation du code source, ce code exécutable étant mémorisé ou déployé au sein du registre **5** blockchain, c'est-à-dire inséré en tant que transaction dans un bloc **4** du registre **5** blockchain.

[0033] Dans la technologie blockchain proposée par Ethereum, un smart contrat est activé par un appel (en anglais Call) adressé par un autre compte, dit compte initiateur (qui peut être un compte utilisateur ou un compte de contrat), cet appel se présentant sous forme d'une transaction contenant, d'une part, un fonds de réserve à transférer (c'est-à-dire un paiement) depuis le compte initiateur au compte de contrat et, d'autre part, des conditions initiales.

[0034] Cet appel est inscrit en tant que transaction dans le registre **5** blockchain. Il déclenche :

- Le transfert du fonds de réserve du compte initiateur au compte de contrat ;
- La désignation, parmi le réseau **1** blockchain, d'un nœud d'exécution associé à un compte utilisateur ;
- L'activation, dans une unité de traitement informatique du nœud d'exécution, d'un environnement d'exécution ou machine virtuelle (appelé Ethereum Virtual Machine ou EVM dans le cas d'Ethereum) ;
- L'exécution pas-à-pas des étapes de calcul du code exécutable par la machine virtuelle à partir des conditions initiales, chaque étape de calcul étant accompagnée d'un transfert d'une fraction (appelée gas dans le cas d'Ethereum) du fonds de réserve depuis le compte de contrat vers le compte utilisateur du nœud d'exécution, et ce jusqu'à épuisement des étapes de calcul, au terme desquelles est obtenu un résultat ;
- L'inscription (éventuellement sous forme d'une empreinte numérique) de ce résultat en tant que transaction dans le registre **5** blockchain.

[0035] Le compte initiateur récupère (c'est-à-dire, en pratique, télécharge) le résultat lors de sa synchronisation au registre **5** blockchain.

[0036] On introduit à présent brièvement les environnements d'exécution sécurisé.

[0037] Un environnement d'exécution sécurisé (Trusted execution environment ou TEE) est, au sein d'une unité **6** de traitement informatique pourvue d'un processeur ou CPU (Central Processing Unit) **7**, un espace temporaire de calcul et de stockage de données, appelé (par convention) enclave, ou encore enclave cryptographique, qui se trouve isolé, par des moyens cryptographiques, de toute action non autorisée résultant de l'exécution d'une application hors de cet espace, typiquement du système d'exploitation.

[0038] Intel® a, par exemple, revu à partir de 2013 la structure et les interfaces de ses processeurs pour y inclure des fonctions d'enclave, sous la dénomination Software Guard Extension, plus connue sous l'acronyme SGX. SGX équipe la plupart des processeurs de type XX86 commercialisés par Intel® depuis 2015, et plus précisément à partir de la sixième

génération incorporant la microarchitecture dite Skylake. Les fonctions d'enclave proposées par SGX ne sont pas accessibles d'office : il convient de les activer via le système élémentaire d'entrée/sortie (Basic Input Output System ou BIOS).

[0039] Il n'entre pas dans les nécessités de la présente description de détailler l'architecture des enclaves, dans la mesure où :

- En dépit de sa relative jeunesse, cette architecture est relativement bien documentée, notamment par Intel® qui a déposé de nombreux brevets, cf. par ex., parmi les plus récents, la demande de brevet américain US 2019/0058696 ;
- Des processeurs permettant de les implémenter sont disponibles sur le marché - notamment les processeur Intel® précités ;
- Seules les fonctionnalités permises par l'enclave nous intéressent ici, ces fonctionnalités pouvant être mises en œuvre via des lignes de commande spécifiques. A ce titre, l'homme du métier pourra se référer au guide édité en 2016 par Intel® : Software Guard Extensions, Developer Guide.

[0040] Pour une description plus accessible des enclaves, et plus particulièrement d'Intel® SGX, l'homme du métier peut également se référer à A. Adamski, Overview of Intel SGX - Part 1, SGX Internai, ou à D. Boneh, Surnaming Schemes, Fast Verification, and Applications to SGX Technology, in Topics in Cryptology, CT - RSA 2017, The Cryptographers' Track at the RSA Conférence 2017, San Francisco, CA, USA, Feb.14-17, 2017, Proceedings, pp.149-164, ou encore à K. Severinsen, Secure Programming with Intel SGX and Novel Applications, Thesis submitted for the Degree of Master in Programming and Networks, Dept. Of Informatics, Faculty of Mathematics and Natural Science, University of Oslo, Autumn 2017.

[0041] Pour résumer, en référence à la **FIG.2**, une enclave **8** comprend, en premier lieu, une zone **9** mémoire sécurisée (dénommée Page Cache d'enclave, en anglais Enclave Page Cache ou EPC), qui contient du code et des données relatives à l'enclave elle-même, et dont le contenu est chiffré et déchiffré en temps réel par une puce dédiée dénommée Moteur de Chiffrement de Mémoire (en anglais Memory Encryption Engine ou MEE). L'EPC **9** est implémentée au sein d'une partie de la mémoire vive dynamique (DRAM) **10** allouée au processeur **7**, et à laquelle les applications ordinaires (notamment le système d'exploitation) n'ont pas accès.

[0042] L'enclave **8** comprend, en deuxième lieu, des clés cryptographiques employées pour chiffrer ou signer à la volée les données sortant de l'EPC **9**, ce grâce à quoi l'enclave **8** peut être identifiée (notamment par d'autres enclaves), et les données qu'elle génère peuvent être chiffrées pour être stockées dans des zones de mémoire non protégées (c'est-à-dire hors de l'EPC **9**).

[0043] Pour pouvoir exploiter une telle enclave **8**, une application **11** doit être segmentée en, d'une part, une ou plusieurs parties **12** non sécurisées (en anglais untrusted part(s)), et, d'autre part, une ou plusieurs parties **13** sécurisées (en anglais trusted part(s)).

[0044] Seuls les processus induits par la (les) partie(s) **13** sécurisée(s) de l'application **11** peuvent accéder à l'enclave **8**. Les processus induits par la (les) partie(s) **12** non sécurisée(s) ne peuvent pas accéder à l'enclave **8**, c'est-à-dire qu'ils ne peuvent pas dialoguer avec les processus induits par la (les) partie(s) **13** sécurisée(s).

[0045] La création (également dénommée instanciation) de l'enclave **8** et le déroulement de processus en son sein sont commandés via un jeu **14** d'instructions particulières exécutables par le processeur **7** et appelées par la (les) partie(s) **13** sécurisée(s) de l'application **11**.

[0046] Parmi ces instructions :

- ECREATE commande la création d'une enclave **8** ;
- EINIT commande l'initialisation de l'enclave **8** ;
- EADD commande le chargement de code dans l'enclave **8** ;
- EENTER commande l'exécution de code dans l'enclave **8** ;
- ERESUME commande une nouvelle exécution de code dans l'enclave **8** ;
- EEXIT commande la sortie de l'enclave **8**, typiquement à la fin d'un processus exécuté dans l'enclave **8**.

[0047] On a, sur la **FIG.2**, représenté de manière fonctionnelle l'enclave 8 sous la forme d'un bloc (en traits pointillés) englobant la partie **13** sécurisée de l'application **11**, le jeu **14** d'instructions du processeur **7**, et l'EPC **9**. Cette représentation n'est pas réaliste ; elle vise simplement à regrouper visuellement les éléments qui composent ou exploitent l'enclave **8**.

[0048] Nous expliquerons ci-après comment sont exploitées les enclaves.

[0049] Le procédé proposé vise à effectuer un traitement décentralisé de données biométriques d'un individu personne physique. Ce traitement comprend deux phases :

- Une phase préalable de calibrage ;
- Une phase de contrôle.

[0050] On décrit tout d'abord la phase de calibrage, qui vise à élaborer des données **15** biométriques de référence de l'individu, à partir d'une première unité **E1** de traitement informatique (équipant par ex. un ordinateur personnel fixe ou portable, une tablette, un smartphone, ou encore un dispositif de contrôle d'accès), reliée ou intégrée au réseau 1 blockchain et appelée „premier émetteur“. Sur la **FIG.3**, le premier émetteur se présente sous la forme d'un smartphone. Cette forme d'exécution est purement illustrative.

[0051] A cet effet, le premier émetteur **E1** est équipé de (ou relié à) un dispositif **16** de capture biométrique ou scanneur. Le scanneur **16** est configuré pour réaliser une capture des données **15** biométriques de référence de l'utilisateur au niveau d'un membre (par ex. un ou plusieurs doigt(s), une main) ou d'un organe (par ex. un oeil, le visage, une oreille, une partie du réseau veineux) de cet utilisateur.

[0052] Ces données **15** biométriques de référence sont destinées à être stockées non pas localement dans le premier émetteur **E1**, mais sur le réseau **1** blockchain. En effet, les nœuds **2** du réseau **1** blockchain étant tous équipés de zones mémoires, celles-ci peuvent être exploitées en tant qu'espace de stockage pour les données **15** biométriques de référence. Pour minimiser le risque de perte des données **15**, il est avantageux de procéder à une réplication de celles-ci, c'est-à-dire d'effectuer une copie des données **15**, et de distribuer une copie à plusieurs nœuds **2** du réseau **1** blockchain. La traçabilité des données **15** peut être réalisée par inscription, dans le registre **5** blockchain, d'une ou plusieurs traces (ou empreintes numériques) des mémorisations ainsi effectuées.

[0053] Ce stockage distribué des données **15** est avantageusement piloté par un contrat intelligent, activé par exemple par le premier émetteur **E1** qui, tout en transmettant les données **15** à stocker au réseau **1**, transmet au contrat intelligent un appel par la procédure décrite précédemment.

[0054] Cependant, si la réplication des données **15** au sein du réseau **1** blockchain résout le problème de la pérennité des données **15**, il ne résout pas le problème de leur confidentialité.

[0055] Il est donc proposé de distribuer sur le réseau **1** non seulement les données **15** (sous forme cryptée), mais également une clé de déchiffrement de celles-ci, via une enclave **8** instanciée sur un nœud **2E**, dit nœud d'entrée, du réseau **1** (**FIG.3**).

[0056] A cet effet, une première opération préliminaire consiste, au niveau du premier émetteur **E1**, à réaliser, au moyen de son scanneur **16**, une capture des données **15** biométriques de l'individu, considérées ensuite en tant que données de référence. Dans l'exemple illustré, ces données **15** biométriques de référence sont issues d'une empreinte digitale.

[0057] Une deuxième opération préliminaire consiste à transmettre, depuis le premier émetteur **E1**, une requête de stockage des données **15** au réseau **1**.

[0058] A réception de la requête de stockage par au moins un nœud **2** du réseau **1**, il est sélectionné, au sein du réseau **1**, au moins un nœud **2E** d'entrée, équipé d'une unité de traitement informatique dans laquelle est implémentée une enclave **8**.

[0059] Cette enclave **8** est alors instanciée, et les données **15** biométriques de référence y sont chargées à partir du premier émetteur **E1** via une ligne **17** de communication sécurisée (par ex. utilisant le protocole Transport Layer Security ou TLS). A cet effet, l'enclave **8** peut être pourvue, dès son instanciation, d'une émulation d'interface **18** de communication supportant le protocole choisi (ici TLS) pour l'échange des données sécurisées.

[0060] Le protocole blockchain est avantageusement chargé dans l'enclave **8**, pour former un module **19** blockchain apte à interroger le registre **5** blockchain et/ou à participer au processus de création et validation des blocs **4**.

[0061] Une transaction **20** contenant une empreinte numérique du chargement des données ainsi effectué est inscrite dans un bloc **4** de la blockchain **5**, aux fins de traçabilité.

[0062] Cette transaction **20** peut être transmise au module **19** blockchain par le premier émetteur **E1**, pour être inscrite par un ou plusieurs nœuds **2** du réseau (par le processus décrit plus haut) dans un nouveau bloc **4** du registre **5** blockchain. En variante, le module **19** blockchain émet de lui-même une transaction **20** pour inscription dans un nouveau bloc **4**.

[0063] Dans l'enclave **8** se déroule alors un processus qui comprend les opérations suivantes.

[0064] Une première opération consiste à chiffrer les données **15** au moyen d'une clé **21** cryptographique de chiffrement pour former un conteneur **22** crypté, en lui associant une clé **23** cryptographique de déchiffrement. A cet effet, les données **15** reçues du premier émetteur **E1** sont relayées par l'interface **18** de communication à un module **24** de chiffrement implémenté dans l'enclave **8**.

[0065] Selon un mode préféré de réalisation, le chiffrement est symétrique. Dans ce cas, la clé **21** de chiffrement et la clé **23** de déchiffrement sont une seule et même clé.

[0066] Une deuxième opération consiste à fragmenter la clé **23** de déchiffrement. Plus précisément, la clé **23** de déchiffrement est fragmentée en un nombre N prédéterminé de fragments **23.i**, (i un entier, $2 \leq i \leq N$).

[0067] Selon un mode préféré de réalisation, N est tel que $N>3$ et la fragmentation est réalisée en application des règles de Shamir (dite du Partage de Secret de Shamir, en anglais Shamir's Secret Sharing), et plus précisément du schéma seuil, où un nombre entier prédéterminé K ($1<K<N$) est choisi tel qu'un nombre K de fragments **23.i** sont suffisants pour reconstituer la clé **23** de déchiffrement.

[0068] Une troisième opération consiste à désigner, parmi le réseau **1**, un ou plusieurs nœuds **2A** primaires de stockage.

[0069] A cet effet, l'enclave **8** est avantageusement pourvue d'un module **25** de distribution de données, auquel le module **24** de chiffrement communique le conteneur **22** crypté pour distribution aux nœuds **2A** primaires de stockage.

[0070] Une quatrième opération consiste à désigner, parmi le réseau **1**, un groupe de plusieurs nœuds **2B** secondaires de stockage, en nombre N égal au nombre de fragments **23.i** de la clé **23** de déchiffrement.

[0071] Ces nœuds **2B** secondaires de stockage sont de préférence distinct(s) des nœuds **2A** primaires de stockage - en d'autres termes, les nœuds **2A** primaires de stockage et les nœuds **2B** secondaires de stockage forment deux groupes disjoints (représentés en pointillés sur la **FIG.3**).

[0072] Une cinquième opération consiste à distribuer le conteneur **22** crypté vers le ou les nœuds **2A** primaires de stockage.

[0073] Une sixième opération consiste à distribuer les fragments **23.i** de la clé **23** cryptographique de déchiffrement vers les nœuds **2B** secondaires de stockage, chaque nœud **2B** secondaire de stockage recevant un unique fragment **23.i**.

[0074] Selon un mode préféré de réalisation, l'enclave **8** est pourvue d'un module **26** de fragmentation et de distribution de clé, auquel le module **24** de chiffrement communique la clé **23** de déchiffrement pour fragmentation et distribution aux nœuds **2B** secondaires de stockage.

[0075] Ces opérations achevées, l'enclave **8** peut être refermée.

[0076] Hors de l'enclave **8**, les opérations suivantes sont réalisées :

- o Le conteneur **22** crypté est mémorisé au sein de chaque nœud **2A** primaire de stockage ;
- o Chaque fragment **23.i** de la clé **23** cryptographique de déchiffrement est mémorisé au sein de chaque nœud **2B** respectif secondaire de stockage.

[0077] Pour assurer la traçabilité de ces opérations, au moins une transaction contenant une empreinte numérique des distributions ou des mémorisations ainsi effectuées est inscrite dans un bloc **4** de la chaîne **5** de blocs, par un ou plusieurs nœuds **2** du réseau **1**. Cette transaction peut être initiée par les nœuds **2A** primaires et **2B** secondaires de stockage, mais elle peut également être initiée par l'enclave **8** elle-même (et plus précisément par le module **19** blockchain) avant sa fermeture.

[0078] Les données **15** biométriques de référence, ainsi encapsulées dans un conteneur **22** crypté, peuvent être stockées pour une durée déterminée ou indéterminée sur les nœuds **2A** de stockage du réseau **1**.

[0079] Ces données **15** biométriques de référence servent, autant que de besoin, à authentifier l'utilisateur dès lors que celui-ci a besoin d'accéder à un lieu ou un service : tel est l'objectif de la phase de contrôle, qui est appliquée à de nouvelles données **27** biométriques de l'individu, qui doivent être comparées aux données **15** de référence.

[0080] On décrit à présent cette phase de contrôle, qui est initiée à partir d'une deuxième unité **E2** de traitement informatique (équipant par ex. un ordinateur personnel fixe ou portable, une tablette, un smartphone, ou encore un dispositif de contrôle d'accès), reliée ou intégrée au réseau 1 blockchain et appelée deuxième émetteur.

[0081] Le deuxième émetteur **E2** peut être confondu avec le premier émetteur **E1** ; néanmoins les émetteurs **E1**, **E2** peuvent être différents.

[0082] Le contrôle des données **27** biométriques est typiquement effectué aux fins de déverrouiller, pour l'utilisateur, un accès à un environnement physique (un logement, une pièce, un coffre-fort) ou virtuel (un environnement de travail dans un ordinateur, une tablette ou un smartphone).

[0083] Le contrôle est effectué de manière décentralisée, sur le réseau **1**.

[0084] Pour cette phase de contrôle, le deuxième émetteur **E2** est équipé de (ou relié à) un dispositif **28** de capture biométrique ou scanneur. Le scanneur **28** est configuré pour réaliser une capture des données **27** biométriques de l'utilisateur au niveau d'un membre (par ex. un ou plusieurs doigt(s), une main) ou d'un organe (par ex. un oeil, le visage, une oreille, une partie du réseau veineux) de cet utilisateur.

[0085] Une première opération **101** consiste à établir, sur requête (REQ), une session de communication entre le deuxième émetteur **E2** et le réseau **1**.

[0086] Cette session est activée à partir du deuxième émetteur **E2**, par exemple à la suite d'une action prédéterminée comme l'appui sur un bouton ou la détection (par ex. au moyen d'un capteur de proximité) de l'approche du membre ou de l'organe de l'utilisateur.

[0087] Une deuxième opération **102** consiste, parmi le réseau **1**, à sélectionner un nœud **2P** de calcul, équipé d'une unité de traitement dans laquelle est implémentée une enclave **8'**.

[0088] Une troisième opération **103** consiste, au sein du nœud **2P** de calcul, à instancier l'enclave **8'**.

[0089] Une quatrième opération **104** consiste, au niveau du deuxième émetteur **E2**, à réaliser, au moyen de son scanneur **28**, une capture des données **27** biométriques de l'individu. Dans l'exemple illustré, ces données **27** biométriques sont issues d'une empreinte digitale.

[0090] Une cinquième opération **105** consiste, à partir du deuxième émetteur **E2**, à charger, dans l'enclave **8'** du nœud **2P** de calcul, via une ligne **29** de communication sécurisée (par ex. utilisant le protocole Transport Layer Security ou TLS), les données **27** biométriques ainsi capturées.

[0091] Selon un mode préféré de réalisation, le contrôle des données **27**, conduit au sein de l'enclave **8'**, est effectué suivant les instructions d'un contrat intelligent **SC** déployé sur la blockchain **5**.

[0092] Comme illustré sur la **FIG.4**, une sixième opération **106** consiste, pour l'enclave **8'**, à appeler (**CALL**) le contrat intelligent **SC**.

[0093] Une septième opération **107** consiste, en retour, à charger le code du contrat intelligent **SC** dans l'enclave **8'** pour exécution. Est également chargée dans l'enclave **8'** une machine virtuelle (EVM lorsque le contrat intelligent **SC** est programmé selon les spécifications d'Ethereum) destinée à permettre l'exécution du code du contrat intelligent **SC**.

[0094] Une huitième opération **108** consiste, pour l'enclave **8'**, à sélectionner parmi le réseau **1** un nœud **2A** de stockage sur lequel est stocké un exemplaire du conteneur **22** crypté contenant les données **15** biométriques de référence, et à transmettre à ce nœud **2B** de stockage une requête (**REQ**) de communication de ce conteneur **22**.

[0095] Pour faciliter cette sélection, le conteneur **22** crypté peut être couplé à un identifiant associé au deuxième émetteur **E2** (transmis à l'enclave **8'** avec les données **27** biométriques scannées), ou à l'utilisateur (et qui peut être saisi par celui-ci sur une interface équipant ou reliée au deuxième émetteur **E2**).

[0096] Une neuvième opération **109** consiste, à partir du nœud **2B** de stockage, à charger dans l'enclave **8'** le conteneur **22** crypté.

[0097] Le déchiffrement des données **15** biométriques de référence du conteneur **22** crypté nécessite la clé **23** cryptographique de déchiffrement, également stockée sur le réseau **1**.

[0098] Une dixième opération **110** consiste par conséquent, pour l'enclave **8'**, à sélectionner parmi le réseau **1** K nœuds **2B** de stockage sur lesquels sont stockés K fragments **23.i** respectifs, et à transmettre à chaque nœud **2B** de stockage une requête (**REQ**) de communication de son fragment **23.i**.

[0099] Comme illustré sur la **FIG.3**, une onzième opération **111** consiste, à partir des nœuds **2B** de stockage ainsi sélectionnés, à charger dans l'enclave **8'** les K fragments **23.i**.

[0100] Une douzième opération **112** consiste, pour l'enclave **8'**, à reconstituer la clé **23** à partir des K fragments **23.i** ainsi chargés.

[0101] Une treizième opération **113** consiste, pour l'enclave **8'**, à déchiffrer les données **15** biométriques de référence du conteneur **22** en lui appliquant la clé **23** ainsi reconstituée.

[0102] Une quatorzième opération **114** consiste, pour l'enclave **8'**, à comparer les données **27** biométriques issues de la capture effectuée par le scanneur **28** et les données **15** biométriques de référence ainsi déchiffrées. La comparaison peut être effectuée par une technique classique (typiquement par mesure des distances entre minuties dans le cas de l'empreinte digitale).

[0103] Si cette comparaison est un échec, les données **15**, **27** sont décrétées ne pas correspondre, et l'action pour laquelle le contrôle des données biométriques de l'utilisateur était requis (typiquement un déverrouillage) n'est pas autorisée. Le deuxième émetteur **E2** en est informé. Les opérations de capture et de comparaison des données biométriques peuvent cependant être répétées, pour minimiser le risque de faux négatif.

[0104] Si au contraire la comparaison est un succès, les données **15**, **27** sont décrétées correspondre, et l'action pour laquelle le contrôle des données biométriques de l'utilisateur était requis est autorisée.

[0105] Quoiqu'il en soit, la comparaison produit un résultat **30** (sous forme, par ex., d'un bit de valeur 0 en cas d'échec, et de valeur 1 en cas de succès).

[0106] Lorsque l'action (typiquement un déverrouillage) doit être appliquée au niveau du deuxième émetteur **E2**, celui-ci doit être informé du résultat.

[0107] Dans ce cas, une quinzième opération **115** consiste, pour l'enclave **8'**, à transmettre le résultat **30** au premier émetteur **E2**.

[0108] Par ailleurs, aux fins de traçabilité, le résultat **30** (ou une trace de ce résultat **29**) est de préférence inscrit dans la blockchain **5**.

[0109] Dans ce cas, une seizième opération **116** consiste, pour l'enclave **8**, à initier une transaction **TX** à destination de la blockchain **5**. Cette transaction peut être signée au moyen d'une clé privée associée à l'enclave **8'**.

[0110] A cet effet, et selon un mode de réalisation illustré sur la **FIG.3**, une dix-septième opération **117** consiste, pour l'enclave **8'**, à établir une session de communication (**SESS**) avec au moins un nœud **2** du réseau **1**, une dix-huitième opération **118** consistant alors à transmettre à ce nœud **2** la transaction **TX** signée, en vue de son inscription dans la blockchain **5**. La transaction **TX** signée est alors distribuée sur plusieurs nœuds **2** validateurs du réseau **1**, aux fins de vérification préalable à l'inscription.

[0111] Après que la transaction **TX** signée a été vérifiée, une dixneuvième opération **119** consiste, pour un ou plusieurs nœuds **2** validateurs, à l'inscrire dans un nouveau bloc **4** destiné à la blockchain **5**. Une vingtième opération **120** consiste, pour l'un des nœuds **2** validateurs, à ajouter le nouveau bloc **4** contenant la transaction **TX** signée à la blockchain **5**, après l'achèvement d'un mécanisme de consensus tel que preuve de travail (en anglais proof-of-work ou PoW), preuve d'autorité (en anglais proof-of-authority ou PoA) ou preuve d'enjeu (en anglais proof-of-stake ou PoS).

[0112] Le procédé qui vient d'être décrit présente les avantages suivants.

[0113] Premièrement, les données **15** biométriques de référence, cryptées, ne sont exploitables par aucun tiers, y compris celui qui en assure la mémorisation sur un nœud **2A** de stockage.

[0114] Aucune utilisation (en particulier commerciale) ne peut donc en être faite, au bénéfice de la confidentialité.

[0115] Deuxièmement, la réplication du conteneur **22** sur plusieurs nœuds **2A** de stockage limite, par ailleurs, le risque d'effacement tout en augmentant la fiabilité du procédé.

[0116] Troisièmement, le caractère décentralisé du contrôle évite une éventuelle prise en main frauduleuse sur le deuxième émetteur **E2** par un tiers non autorisé.

Revendications

1. Procédé de traitement de données biométriques d'un individu, à partir d'une unité de traitement informatique, dite émetteur (**E2**), reliée ou intégrée à un réseau (**1**) pair-à-pair composé d'une pluralité de nœuds (**2**) formant une base de données distribuée sur laquelle est mémorisée, par réplication sur chaque nœud (**2**), une chaîne (**5**) de blocs, ce procédé comprenant :
   A) Une phase de calibrage, qui comprend les opérations suivantes :
   - Au moyen d'un dispositif (**16**) de capture, réaliser une première capture de données (**15**) biométriques, dites de référence, de l'individu au niveau d'un membre ou d'un organe de celui-ci ;
   - Chiffrer les données (**15**) biométriques issues de cette première capture pour former un conteneur (**22**) crypté de données biométriques ;
   - Mémoriser séparément le conteneur (**22**) crypté de données biométriques et une clé (**22**) cryptographique de déchiffrement de celui-ci ;
   - Inscrire, dans un bloc (**4**) de la chaîne (**5**) de blocs, une transaction contenant une trace de cette capture et/ou de cette mémorisation ;
   B) Une phase de contrôle, qui comprend les opérations suivantes :
   - Etablir une session de communication entre l'émetteur (**E2**) et le réseau (**1**) ;
   - Sélectionner parmi le réseau (**1**) un nœud (**2P**) dit de calcul, équipé d'une unité de traitement dans laquelle est implémenté un environnement d'exécution sécurisé par cryptographie, dit enclave (**8'**) ;
   - Instancier l'enclave (**8'**) ;
   - Au moyen d'un dispositif (**28**) de capture équipant ou relié à l'émetteur (**E2**), réaliser une capture de données (**27**) biométriques de l'individu au niveau d'un membre ou d'un organe de cet individu ;
   - Charger dans l'enclave (**8'**), via une ligne (**29**) de communication sécurisée, les données (**27**) biométriques capturées ;
   - Sélectionner parmi le réseau (**1**) un nœud (**2B**) de stockage sur lequel est mémorisé un exemplaire du conteneur (**22**) crypté contenant les données (**15**) biométriques de référence ;
   - Charger le conteneur (**22**) crypté dans l'enclave (**8'**) ;
   - Déchiffrer dans l'enclave (**8'**) les données (**15**) biométriques de référence, au moyen d'une clé (**23**) de déchiffrement associée au conteneur (**22**) crypté ;
   - Dans l'enclave (**8'**), comparer les données (**27**) biométriques capturées et les données (**15**) de référence ;
   - Inscrire dans un bloc (**4**) de la chaîne (**5**) de blocs une transaction comprenant une trace du résultat (**30**) de cette comparaison.
2. Procédé de traitement selon la revendication 1, dont la phase de calibrage comprend les opérations suivantes :
   - Fragmenter la clé (**23**) cryptographique de déchiffrement ;
   - Désigner, parmi le réseau, un groupe de nœuds (**2B**) de stockage en nombre égal aux fragments (**23.i**) ;
   - Distribuer chaque fragment (**23.i**) de la clé cryptographique de déchiffrement vers un nœud (**2B**) de stockage, chaque nœud (**2B**) de stockage recevant un unique fragment (**23.i**).
3. Procédé de traitement selon la revendication **2**, dont la phase de contrôle comprend les opérations suivantes :
   - A partir de l'enclave (**8'**) du nœud (**2P**) de calcul, sélectionner parmi le réseau (**1**) des nœuds (**2B**) sur lesquels sont stockés des fragments (**23.i**) de la clé (**22**) cryptographique de déchiffrement associée au conteneur (**22**) ;
   - Charger lesdits fragments (**23.i**) dans l'enclave (**8'**) du nœud (**2C**) de calcul ;
   - Dans l'enclave (**8'**) :
   o Reconstituer la clé (**23**) cryptographique de déchiffrement associée au conteneur (**22**), à partir des fragments (**23.i**) ainsi chargés ;
   o Déchiffrer les données (**15**) biométriques du conteneur (**22**) au moyen de la clé (**23**) ainsi reconstituée.

FIG.1

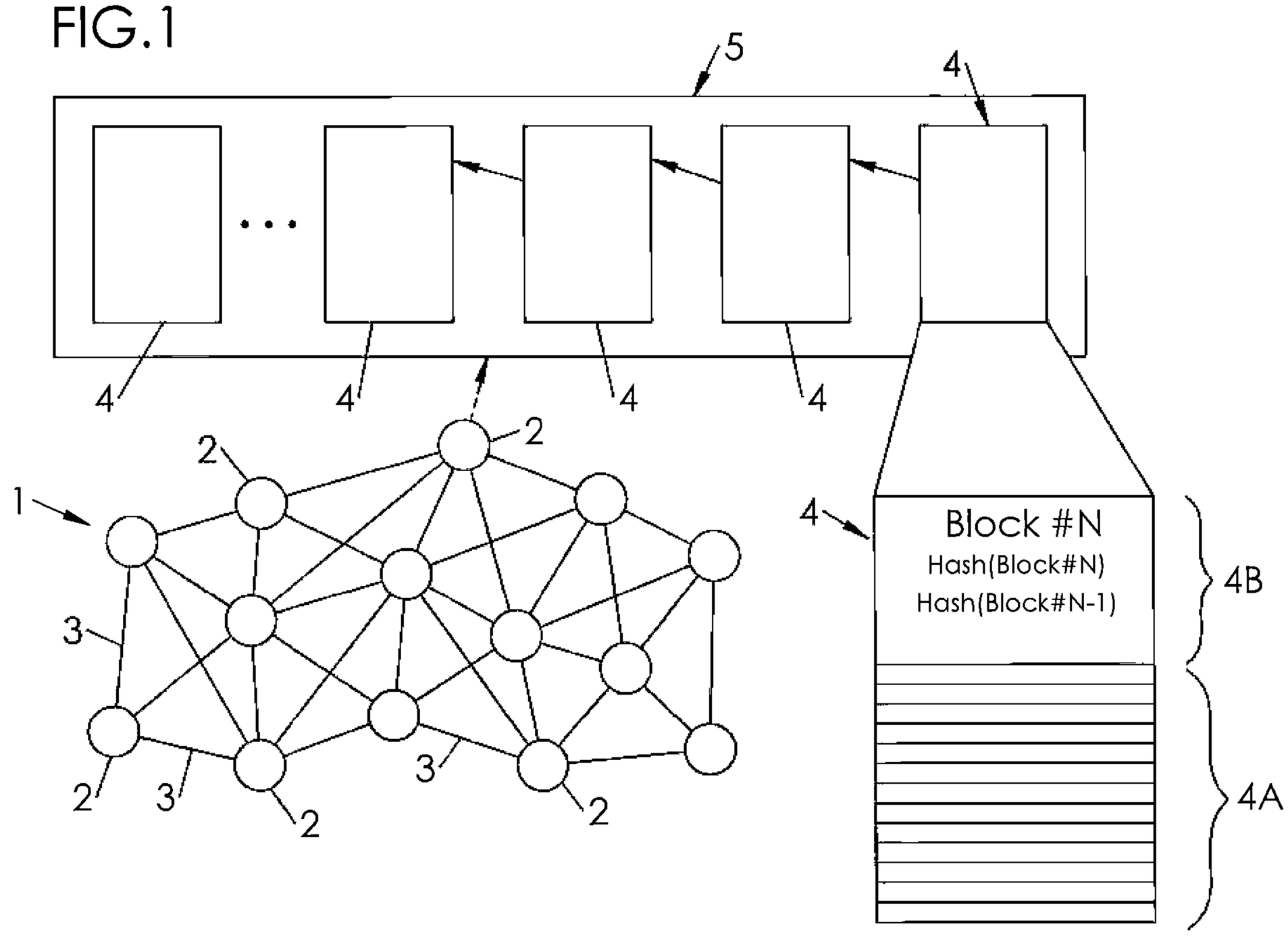
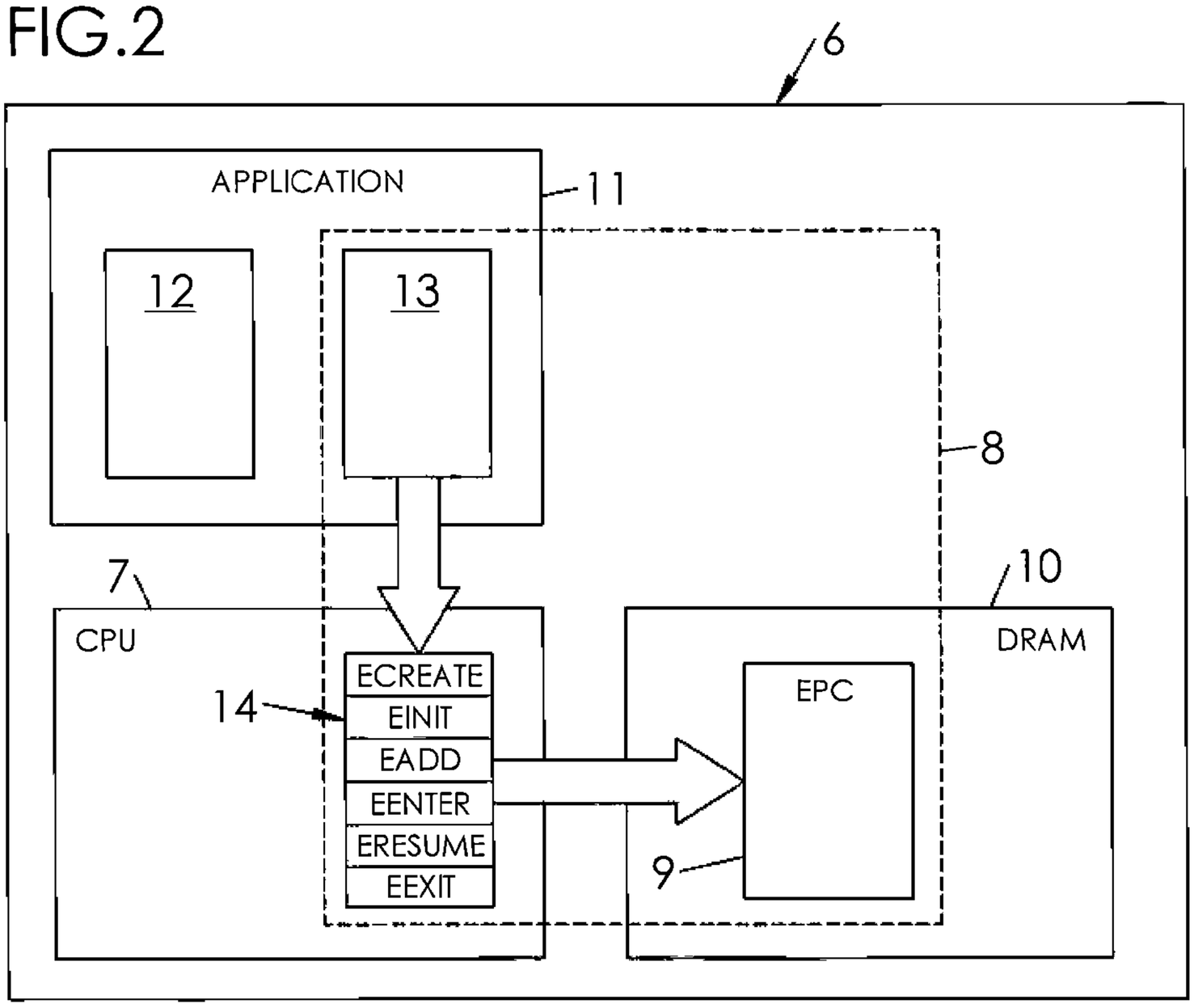

5
4
4
4
4
4
2
2
1
3
3
2
3
2
2
4
Block #N
Hash(Block#N)
Hash(Block#N-1)
4B
4A

FIG.2


6
APPLICATION
11
12
13
8
7
10
CPU
DRAM
ECREATE
EINIT
EADD
EENTER
ERESUME
EEXIT
14
EPC
9

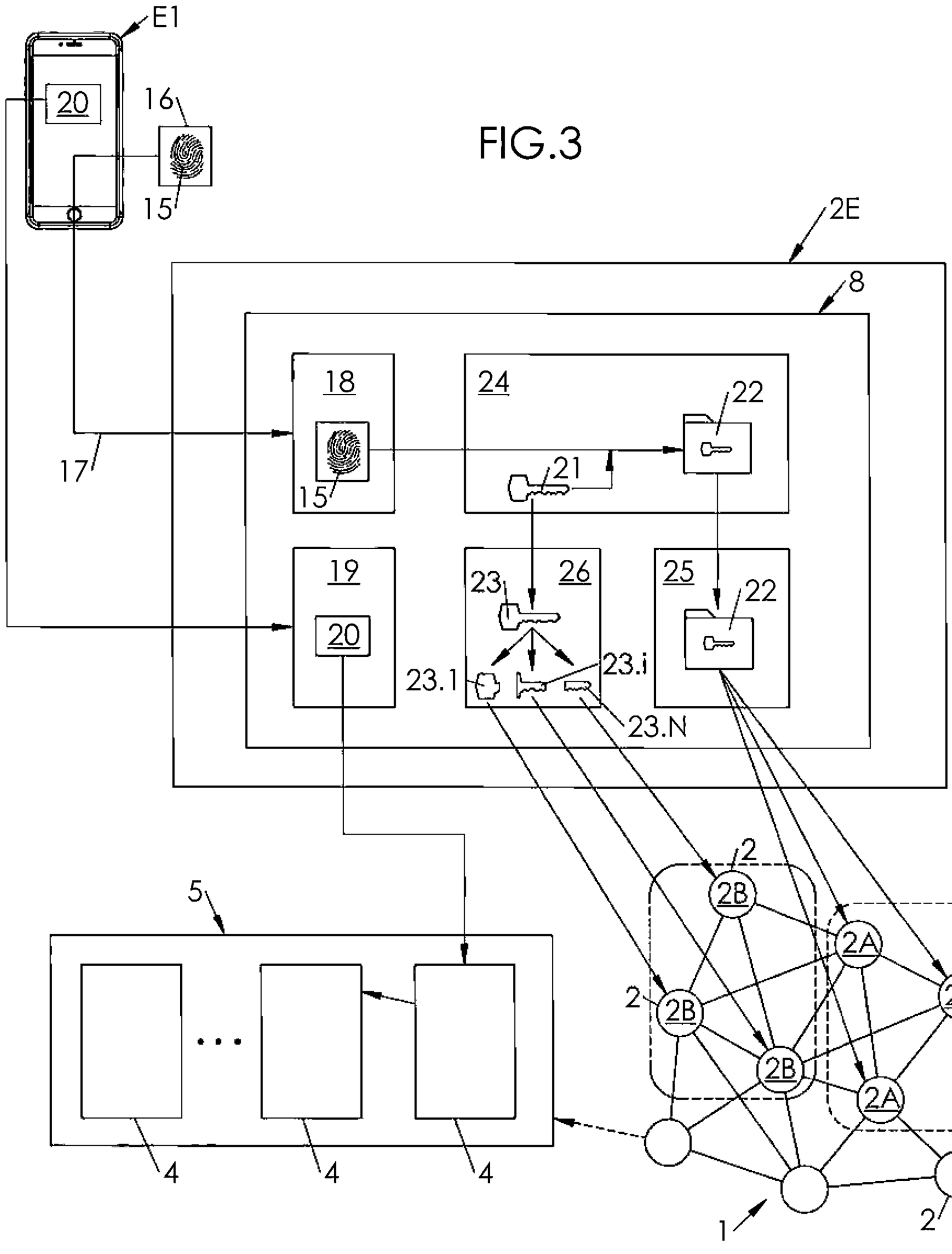

E1
20
16
15
2E
8
18
15
17
24
22
21
19
20
26
23
23.1
23.i
23.N
25
22
5
4
4
4
2
2B
2A
2B
2A
2B
2A
2
1

FIG.3

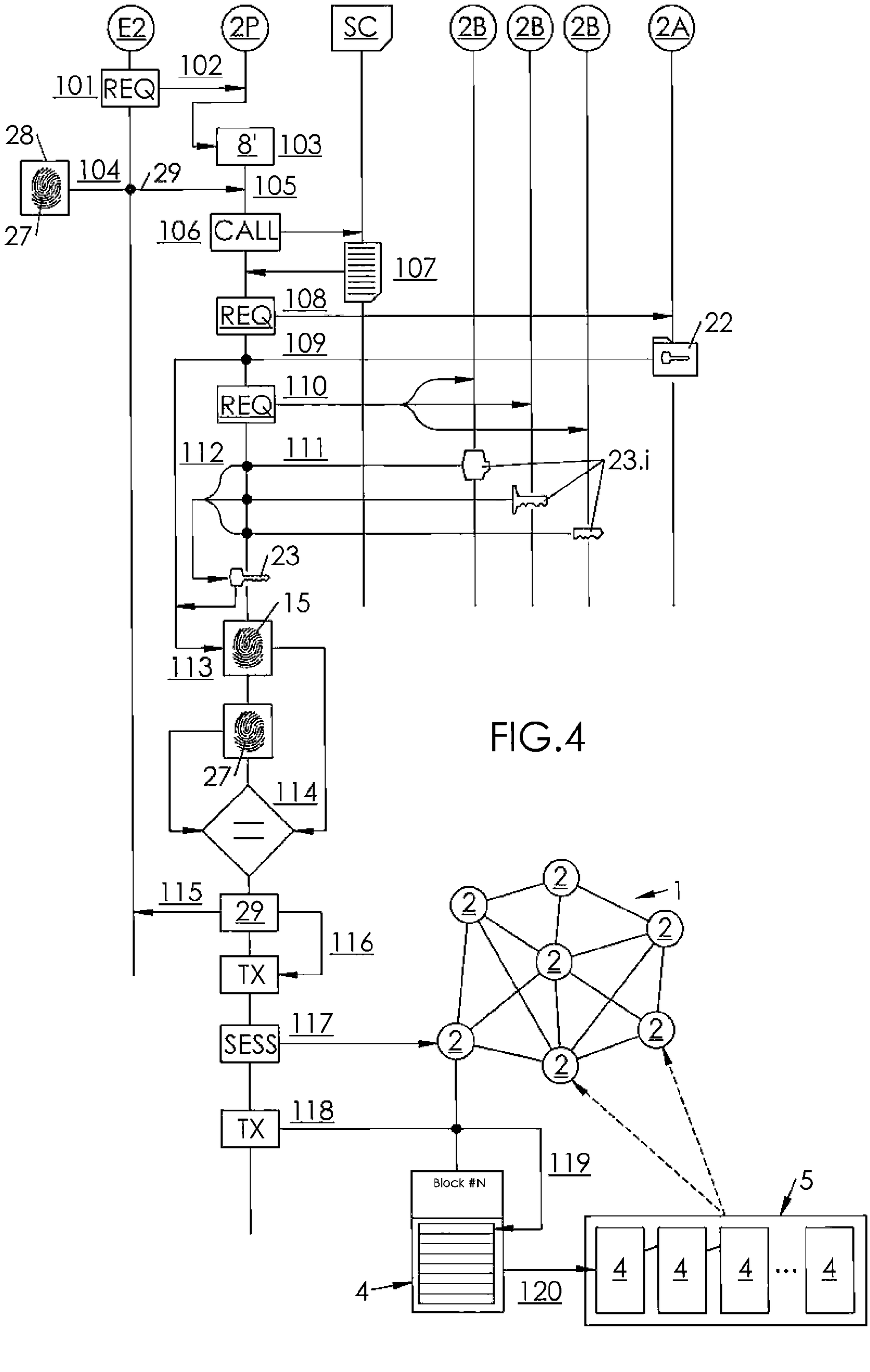

E2
2P
SC
2B
2B
2B
2A
101
REQ
102
8'
103
28
104
29
105
27
106
CALL
107
REQ
108
109
22
REQ
110
112
111
23.i
23
15
113
27
114
115
29
116
TX
117
SESS
118
TX
119
Block #N
4
120
2
1
5
4

FIG.4