(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) **EP 3 373 509 A1**

(12) **DEMANDE DE BREVET EUROPEEN**

(43) Date de publication:
12.09.2018 Bulletin 2018/37

(51) Int Cl.:
H04L 9/30 *(2006.01)* ***H04L 9/32*** *(2006.01)*

(21) Numéro de dépôt: **18160420.8**

(22) Date de dépôt: **07.03.2018**

(84) Etats contractants désignés:
AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO RS SE SI SK SM TR
Etats d'extension désignés:
BA ME
Etats de validation désignés:
KH MA MD TN

(30) Priorité: **08.03.2017 FR 1751894**

(71) Demandeur: **Idemia Identity & Security France**
92130 Issy-les-Moulineaux (FR)

(72) Inventeurs:
- **CHABANNE, Herve**
  92130 Issy Les Moulineaux (FR)
- **PROUFF, Emmanuel**
  92130 Issy Les Moulineaux (FR)

(74) Mandataire: **Regimbeau**
87 rue de Sèze
69477 Lyon Cedex 06 (FR)

(54) PROCÉDÉ DE SIGNATURE ÉLECTRONIQUE D'UN DOCUMENT AVEC UNE CLÉ SECRÈTE PRÉDÉTERMINÉE

(57) La présente invention concerne un procédé de de signature électronique d'un document avec une clé secrète (x) prédéterminée, le procédé étant caractérisé en ce qu'il comprend la mise en oeuvre d'étapes de :
(a) Tirage d'un couple d'un premier état interne

$$(s_1^i)$$

et d'une implémentation blanchie (WB_i) d'une opération d'arithmétique modulaire parmi un ensemble de couples

$$\left(\{(s_1^i, WB_i)\}_{i\in[\![0,n-1]\!]}\right)$$

prédéterminés chacun pour un nonce (k_i), ledit premier état interne

$$(s_1^i)$$

étant fonction du nonce (k_i) et ladite opération d'arithmétique modulaire étant fonction du premier état interne

$$(s_1^i),$$

du nonce (k_i) et de la clé secrète (x) ;
(b) Détermination d'un deuxième état interne

$$(s_2^i)$$

EP 3 373 509 A1

par application à un condensat du document obtenu par une fonction de hachage donnée, de ladite implémentation blanchie (WB_i) tirée ;
(c) génération d'une signature électronique du document à partir du premier état interne

$$(s_1^i)$$

du couple tiré et du deuxième état interne

$$(s_2^i)$$

déterminé, et suppression du couple tiré dudit ensemble de couples

$$\left(\left\{(s_1^i, WB_i)\right\}_{i\in[\![0,n-1]\!]}\right).$$

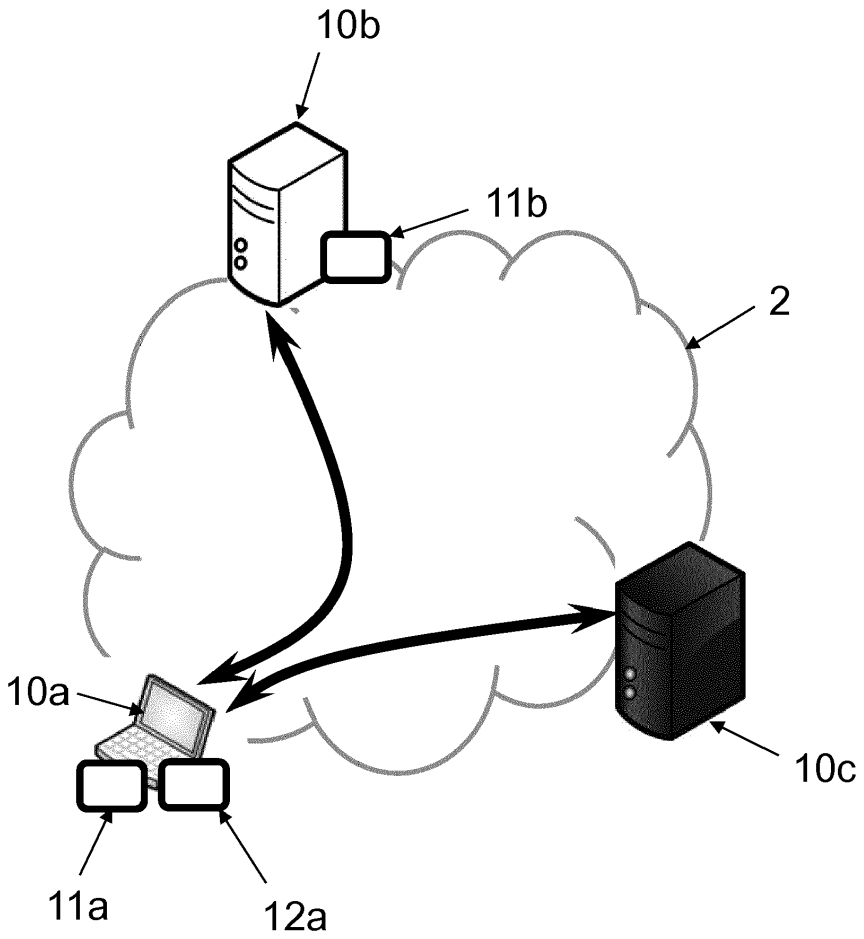

FIG. 1

Description

DOMAINE TECHNIQUE GENERAL

[0001] La présente invention concerne le domaine de la cryptographie, et en particulier un procédé de signature de type « boite blanche ».

ETAT DE L'ART

[0002] Une fonction est considérée comme une « boîte noire » lorsqu'on ne peut pas accéder à son fonctionnement interne, i.e. qu'on peut connaître ses entrées et ses sorties mais pas ses paramètres secrets ou ses états intermédiaire.

[0003] Les algorithmes cryptographiques (par exemple pour du chiffrement ou de la signature) sont ainsi classiquement supposés des boites noires lorsqu'on évalue leur fiabilité (résistance aux attaques).

[0004] L'hypothèse de boite noire impose une contrainte forte sur le stockage et la manipulation de ces paramètres. Cependant des outils ont été récemment publiés pour permettre l'automatisation d'attaques sur implémentation matérielle, attaques dites par canaux auxiliaires ou par fautes.

[0005] Aujourd'hui, pour de nombreux cas d'usages incluant le paiement sur mobile, il est nécessaire de déployer des algorithmes cryptographiques en faisant le moins d'hypothèses possibles sur la sécurité du matériel cible. Le stockage et la manipulation sécurisés des paramètres secrets doivent alors être assurés au niveau applicatif.

[0006] La cryptographie dite boite blanche vise à répondre à ce défi en proposant des implémentations des algorithmes cryptographiques qui sont sensés rendre l'extraction des secrets impossible, même en cas d'attaque permettant à l'attaquant un accès complet à l'implémentation logicielle de l'algorithme. Plus précisément, une fonction est considérée comme une « boîte blanche » lorsque ses mécanismes sont visibles et permettent d'en comprendre le fonctionnement. En d'autres termes, on fait directement l'hypothèse que l'attaquant a accès à tout ce qu'il souhaite (le binaire est complètement visible et modifiable par l'attaquant et celui-ci a le plein contrôle de la plateforme d'exécution). Par conséquent, l'implémentation elle-même est la seule ligne de défense. On parle alors « d'implémentation blanchie » d'un calcul élémentaire de l'algorithme lorsqu'on arrive à représenter ce calcul sous une forme sécurisée évitant d'avoir les clés utilisées en clair, par exemple en représentant le calcul par une table stockée en mémoire.

[0007] Il a par exemple été proposé dans la demande US2016/328543 une méthode visant à cacher les entrées et les sorties d'une fonction d'exponentiation modulaire, ce qui permet d'améliorer la sécurité d'algorithmes cryptographiques tels que RSA (« Rivest, Shamir, Adleman »).

[0008] Toutefois, il a été montré que cette méthode n'était pas suffisante pour assurer une protection satisfaisante de l'algorithme de signature DSA (« Digital Signature Algorithm »).

[0009] Le calcul d'une signature DSA présuppose qu'une clef publique (p,q,g,y) et une clef privée x ont été associées à l'utilisateur. Le procédé de génération de ces clefs se compose des différentes étapes suivantes :

- Choisir un nombre premier p de longueur L telle que $512 \leq L \leq 1024$, et L est divisible par 64
- Choisir un nombre premier q de 160 bits, de telle façon que $p - 1 = qc$, avec c un entier
- Choisir h, avec $1 < h < p - 1$ de manière à construire g tel que $g = h^c \bmod p > 1$
- Générer aléatoirement un x, avec $0 < x < q$
- Calculer $y = g^x \bmod p$

[0010] La signature DSA d'un message m commence par un calcul de haché $H(m)$ (à l'aide d'une fonction de hachage standard comme le SHA256) puis se poursuit par le calcul de deux valeurs s_1 et s_2 telles que :

- $$s_1 = (g^k \bmod p) \bmod q$$
- $$s_2 = (H(m) + s_1 x) k^{-1} \bmod q$$

où k est une donnée appelée nonce (i.e. un nombre arbitraire, c'est-à-dire un aléa, à usage unique, de l'anglais « number used once »), qui doit être tirée aléatoirement pour chaque nouvelle signature.

[0011] La signature est (s_1,s_2).

[0012] Comme expliqué, on peut obtenir des implémentations blanchies de chacune des fonctions de calcul des états internes s_1 et s_2 en cachant l'exponentiation modulaire.

[0013] Toutefois, la même application de l'implémentation blanchie de s_2 à deux données z et z' différentes (par exemple les condensats de deux messages différents) tout en forçant la réutilisation d'un même nonce k (ce qui est normalement impossible sauf dans une attaque boite blanche ou l'on aurait accès au matériel) permet par calcul de $s_2(z)$-$s_2(z')$ de remonter à x. On appelle cela une « restart attack ».

[0014] Il serait par conséquent souhaitable de disposer d'une nouvelle solution de signature « boite blanche » utilisant le mécanisme standard tel que DSA qui soit complètement résistante à toutes attaques connues.

PRESENTATION DE L'INVENTION

[0015] Selon un premier aspect, la présente invention concerne un procédé de signature électronique d'un document avec une clé secrète prédéterminée, le procédé étant caractérisé en ce qu'il comprend la mise en oeuvre par des moyens de traitement de données d'un équipement d'étapes de :

(a) Tirage d'un couple d'un premier état interne et d'une implémentation blanchie d'une opération d'arithmétique modulaire parmi un ensemble de couples prédéterminés chacun pour un nonce, ledit ensemble de couples étant stocké sur des moyens de stockage de données de l'équipement, ledit premier état interne étant fonction du nonce et ladite opération d'arithmétique modulaire étant fonction du premier état interne, du nonce et de la clé secrète ;
(b) Détermination d'un deuxième état interne par application à un condensat du document obtenu par une fonction de hachage donnée, de ladite implémentation blanchie tirée ;
(c) génération d'une signature électronique du document à partir du premier état interne du couple tiré et du deuxième état interne déterminé, et suppression du couple tiré dudit ensemble de couples.

[0016] Selon d'autres caractéristiques avantageuses et non limitatives :

- ladite opération d'arithmétique modulaire est $z \mapsto \left(z + s_1^i x\right) k_i^{-1} \, mod \, q,$ où s_1^i est le premier état interne, k_i le nonce, x la clé secrète et q une constante ;
- $s_1^i = (g^{k_i} \, mod \, p) \, mod \, q,$ où g et p sont des constantes ;
- la signature est le couple $\left(s_1^i, s_2^i\right)$ des premier et deuxième états internes ;
- le procédé comprend une étape préalable (a0) de génération dudit ensemble de couples par des moyens de traitement de données d'un serveur, et sa transmission à l'équipement ;
- l'étape (a0) comprend la génération d'une pluralité de nonces, puis la génération du couple pour chaque nonce ;
- l'étape (a0) comprend la génération préalable des constantes p,q,g conformément à l'algorithme DSA ;
- l'étape (a0) comprend également la génération préalable de la clé secrète et d'une clé publique associée en fonction des constantes p,q,g ;
- lesdites implémentations blanchies utilisent un Système de Représentation Modulaire, RNS, pour effectuer ladite opération d'arithmétique modulaire ;
- le procédé comprend une étape subséquente (d) d'association par les moyens de traitement de données de l'équipement de la signature électronique générée au document de sorte à former le document signé

[0017] Selon un deuxième et un troisième aspect, l'invention propose un produit programme d'ordinateur comprenant des instructions de code pour l'exécution d'un procédé selon le premier aspect de signature électronique d'un document avec une clé secrète prédéterminée ; et un moyen de stockage lisible par un équipement informatique sur lequel un produit programme d'ordinateur comprend des instructions de code pour l'exécution d'un procédé selon le premier aspect de signature électronique d'un document avec une clé secrète prédéterminée.

PRESENTATION DES FIGURES

[0018] D'autres caractéristiques et avantages de la présente invention apparaîtront à la lecture de la description qui va suivre d'un mode de réalisation préférentiel. Cette description sera donnée en référence aux dessins annexés dans lesquels :

- la figure 1 est un schéma d'une architecture pour la mise en oeuvre du procédé selon l'invention.

DESCRIPTION DETAILLEE

Architecture

[0019] En référence à la **figure 1,** est proposé un procédé de signature « boite blanche » d'un document mis en oeuvre au sein d'un équipement 10a tel qu'un terminal mobile (smartphone, tablette tactile, etc.), i.e. un équipement ne disposant pas particulièrement d'un matériel sécurisé et qui peut faire l'objet d'attaques sur implémentation matérielle, et pour lequel l'approche boite blanche prend tout son intérêt.

[0020] L'équipement 10a comprend des moyens de traitement de données 11a (un processeur) et des moyens de stockage de données 12a (une mémoire, par exemple flash).

[0021] L'équipement 10a est par exemple relié à un serveur 10b par exemple via le réseau internet 20. Il peut être amené à recevoir depuis ce serveur 10b (par exemple celui d'un fournisseur de solutions de sécurité) des objets cryptographiques (qu'on décrira plus loin) contenant des secrets qui vont être stockées dans la mémoire 12a et utilisées pour la mise en oeuvre du présent procédé

[0022] L'équipement 10a peut lui-même être connecté à d'autres serveurs 10c de tiers auxquels il pourra transmettre le document signé une fois qu'il aura généré la signature électronique.

Procédé de signature

[0023] Le présent procédé est bien un procédé de « génération de signature électronique de document », et non « de signature d'un document ». Cela signifie qu'il permet seulement d'obtenir la signature électronique du document et pas encore le « document signé », i.e. l'association du document original et de la signature, généralement dans un conteneur quelconque.

[0024] Par « signature électronique » d'un document,

on entend la définition classique de ce terme, à savoir une primitive cryptographique permettant d'identifier le signataire et garantissant que le document n'a pas été altéré depuis l'instant où la signature a été produite et est bien le document original (dans la suite de la présente description, on désignera comme « original » le document dont est réellement issu le condensat). Cet objet cryptographique consiste généralement en un chiffré du condensat du document grâce à une fonction de chiffrement asymétrique : le signataire utilise une clé privé, et tout le monde peut vérifier la signature grâce à une clé publique (en comparant le condensat contenu dans la signature et un condensat recalculé).

[0025] On comprendra que le présent procédé est une nouvelle implémentation d'algorithmes connus utilisant des opérations d'arithmétique modulaire (i.e. des opérations comprenant des calculs de modulo, notamment des exponentiations modulaires), en particulier DSA qui est un standard actuel et dont on prendra l'exemple dans la suite de la description. Plus précisément, il ne propose pas une nouvelle stratégie de signature, mais seulement une nouvelle façon de manipuler les données au sein de l'algorithme qui soit résistante à toutes les attaques matérielles en « boite blanche ».

[0026] Le document est associé à un condensat obtenu par une fonction de hachage donnée.

[0027] Comme expliqué, une fonction de hachage prend en entrée un message de taille arbitraire (le document original) et produit un condensat de taille fixe associé à ce message. Ici, ladite fonction de hachage donnée est avantageusement une fonction dite cryptographique, c'est-à-dire avec des propriétés supplémentaires : le condensat est statistiquement bien réparti dans l'ensemble des valeurs d'arrivées, il est impossible en temps raisonnable de trouver deux messages qui ont le même condensat (résistance aux collisions) et on ne peut pas à partir du condensat retrouver un message qui a permis d'obtenir cette valeur (résistance au calcul de pré-image).

[0028] On prendra l'exemple des fonctions de la famille SHA (« Secure Hash Algorithm »), standardisées par le NIST (« National Institute of Standards and Technology »), en particulier les sous-familles SHA-1 ou SHA-2 (notamment SHA-256).

Principe

[0029] La présente invention propose, pour une même paire de clés publique/privée, de faire précalculer plusieurs valeurs d'un premier état interne s_1^i chacune pour un nonce k_i, puis une implémentation blanchie WB_i d'une opération d'arithmétique modulaire utilisée par l'algorithme de signature pour chacun des nonces k_i, ladite opération d'arithmétique modulaire étant en effet fonction du premier état interne s_1^i, du nonce k_i et de la clé secrète x. On rappelle que par implémentation blanchie, ou « white box implementation », d'une opération, on entend une représentation de l'opération qui ne permette pas de remonter aux états internes ou aux paramètres lorsque l'opération est exécutée (par application de l'implémentation blanchie aux données d'entrée).

[0030] Ainsi, un ensemble de couples

$$\left\{(s_1^i, WB_i)\right\}_{i\in[\![0,n-1]\!]}$$

du premier état interne s_1^i et de l'implémentation blanchie WB_i de l'opération d'arithmétique modulaire sont prédéfinis et stockés sur les stocké sur les moyens de stockage de données 12a de l'équipement 10a.

[0031] En d'autres termes, à clés publique/privée fixée, chaque couple (s_1^i, WB_i) est entièrement déterminé par le nonce k_i : tirer un nonce équivaut à tirer un couple. Le nonce k_i n'est donc pas une donnée d'entrée de l'implémentation blanchie WB_i mais bien un paramètre « encapsulé » auquel un attaquant ne peut pas remonter.

[0032] De façon préférée, ledit ensemble de couples

$$\left\{(s_1^i, WB_i)\right\}_{i\in[\![0,n-1]\!]}$$

est générée par les moyens de traitement de données 11b du serveur 1b et sa transmis pour stockage à l'équipement 10a dans une étape préalable (a0).

[0033] Cette génération peut comprendre la génération de la pluralité de nonces

$$\{\mathrm{k}_i\}_{i\in[\![0,n-1]\!]},$$

puis la génération du couple (s_1^i, WB_i) pour chaque nonce k_i de sorte à définir l'ensemble

$$\left\{(s_1^i, WB_i)\right\}_{i\in[\![0,n-1]\!]}.$$

[0034] Les propriétés attendues du blanchiment d'application font que l'observation de l'exécution de l'implémentation blanchie WB_i ne doit pas permettre de retrouver les valeurs de la clef secrète x et du nonce k_i enfouies dans le calcul. Il suffit donc de n'autoriser l'utilisation de chaque WB_i qu'une seule fois pour empêcher de remonter à k_i et ainsi éviter les restart attack. Le procédé de signature proposé permet donc de résister à un attaquant qui observerait les états intermédiaires (par exemple à

l'aide d'une application malveillante tournant sur le mobile).

[0035] Les implémentations blanchies WB_i (et donc les couples d'un premier état et d'une implémentation blanchie) sont ainsi supprimés après utilisation pour éviter toute attaque possible.

Mode de réalisation préféré

[0036] Dans l'exemple où l'algorithme de signature est conforme à DSA, alors s_1^i peut être égal à ($g^{k_i} \bmod p$) $\bmod q$, où g,p et q sont des constantes (en particulier des nombres premiers pour p et q).

[0037] Similairement, ladite opération d'arithmétique modulaire représentée par l'implémentation blanchie WB_i peut être $z \mapsto \left(z + s_1^i x\right) k_i^{-1} \bmod q$. On peut facilement vérifier que ce calcul correspond à celui du s_2^i associé au s_1^i généré précédemment, pour une valeur de condensat z du message.

[0038] Les valeurs p,q,g sont préférentiellement prédéterminés conformément à l'algorithme DSA, en particulier par les moyens de traitement de données 11b du serveur 10b lors de l'étape préalable (a0).

[0039] On rappelle que dans DSA, (p,q,g,y) forme une clé publique générée avec une clef privée x de la façon suivante :

- Choisir un nombre premier p de longueur L telle que $512 \leq L \leq 1024$, et L est divisible par 64
- Choisir un nombre premier q de 160 bits, de telle façon que $p - 1 = qc$, avec c un entier
- Choisir h, avec $1 < h < p - 1$ de manière que $g = h^c \bmod p > 1$
- Générer aléatoirement un x, avec $0 < x < q$
- Calculer $y = g^x \bmod p$

[0040] L'homme du métier comprendra toutefois que le présent procédé n'est pas limité à DSA et apporte satisfaction pour tout algorithme de signature comportant une opération d'arithmétique modulaire fonction d'un premier état interne, d'un nonce et d'une clé secrète.

Mise en oeuvre

[0041] Le présent procédé est mis en oeuvre par les moyens de traitement de données 11a de l'équipement 10a et commence par une étape (a) de tirage d'un couple d'un premier état interne s_1^i et d'une implémentation blanchie WB_i d'une opération d'arithmétique modulaire parmi ledit ensemble de couples

$$\left\{\left(s_1^i, WB_i\right)\right\}_{i \in [\![0, n-1]\!]}$$

prédéterminés stocké sur des moyens de stockage de données 12a de l'équipement 10a.

[0042] Ce tirage peut être aléatoire ou séquentiel, il revient au tirage d'un nonce k_i puisque chacun couple (s_1^i, WB_i) est entièrement déterminé par le nonce k_i à partir duquel il a été généré.

[0043] Dans une étape (b) est déterminé un deuxième état interne s_2^i par application à un condensat $H(m)$ du document m (obtenu par une fonction de hachage donnée, telle que SHA256), de ladite implémentation blanchie WB_i tirée. En d'autres termes, on prend $z = H(m)$.

[0044] A partir de là, dans une troisième étape (c) peut être générée la signature électronique du document à partir du premier état interne s_1^i du couple tiré et du deuxième état interne s_2^i déterminé à l'étape (b). Dans l'exemple préféré de DSA, la signature est simplement $\left(s_1^i, s_2^i\right)$.

[0045] L'étape (c) comprend également comme expliqué la suppression du couple tiré dudit ensemble de couples

$$\left\{\left(s_1^i, WB_i\right)\right\}_{i \in [\![0, n-1]\!]}$$

(avant ou après la génération de la signature) de sorte à empêcher les restart attack.

[0046] Additionnellement, le procédé peut comprendre une étape subséquente (d) d'association par les moyens de traitement de données 11a de l'équipement 10a de la signature électronique au document de sorte à former le document signé. L'équipement 10a peut alors se prévaloir juridiquement de cette signature électronique auprès d'autres entités (serveurs 10c).

Implémentation blanchie

[0047] La réalisation d'implémentations blanchies d'opérations d'arithmétique modulaire est connue de l'homme du métier.

[0048] Toutefois, l'application de l'implémentation blanchie pour exécuter l'opération représentée peut être longue, en particulier si ladite opération d'arithmétique modulaire comprend des exponentiations modulaire.

[0049] A ce titre, de façon préférée lesdites implémentations blanchies WB_i utilisent un Système de Représentation Modulaire (« Residue Number System », RNS) pour effectuer ladite opération d'arithmétique modulaire.

[0050] Plus précisément, le principe d'implémentation RNS permet de décomposer des calculs modulo une valeur donnée en des calculs modules des petits nombres premiers dont le produit est plus grand que ladite valeur donnée (grâce au théorème chinois).

[0051] L'homme du métier pourra procéder par exemple conformément à la demande US2016/239267.

Produit programme d'ordinateur

[0052] Selon un deuxième et un troisième aspects, l'invention concerne un produit programme d'ordinateur comprenant des instructions de code pour l'exécution (en particulier sur les moyens de traitement de données 11a de l'équipement 10a) d'un procédé selon le premier aspect de l'invention de signature électronique d'un document avec une clé secrète x prédéterminée, ainsi que des moyens de stockage lisibles par un équipement informatique (une mémoire 12a de l'équipement 10a) sur lequel on trouve ce produit programme d'ordinateur.

Revendications

1. Procédé de signature électronique d'un document avec une clé secrète (*x*) prédéterminée, le procédé étant **caractérisé en ce qu'**il comprend la mise en oeuvre par des moyens de traitement de données (11a) d'un équipement (10a) d'étapes de :

(a) Tirage d'un couple d'un premier état interne $\left(s_1^i\right)$ et d'une implémentation blanchie (WB_i) d'une opération d'arithmétique modulaire parmi un ensemble de couples

$$\left(\left\{(s_1^i, WB_i)\right\}_{i\in[\![0,n-1]\!]}\right)$$

prédéterminés chacun pour un nonce (k_i), ledit ensemble de couples

$$\left(\left\{(s_1^i, WB_i)\right\}_{i\in[\![0,n-1]\!]}\right)$$

étant stocké sur des moyens de stockage de données (12a) de l'équipement (10a), ledit premier état interne $\left(s_1^i\right)$ étant fonction du nonce (k_i) et ladite opération d'arithmétique modulaire étant fonction du premier état interne $\left(s_1^i\right)$, du nonce (k_i) et de la clé secrète (*x*) ;

(b) Détermination d'un deuxième état interne $\left(s_2^i\right)$ par application à un condensat du document obtenu par une fonction de hachage donnée, de ladite implémentation blanchie (WB_i) tirée ;

(c) génération d'une signature électronique du document à partir du premier état interne $\left(s_1^i\right)$ du couple tiré et du deuxième état interne $\left(s_2^i\right)$ déterminé, et suppression du couple tiré dudit ensemble de couples

$$\left(\left\{(s_1^i, WB_i)\right\}_{i\in[\![0,n-1]\!]}\right).$$

2. Procédé selon la revendication 1, dans lequel ladite opération d'arithmétique modulaire est $z \mapsto \left(z + s_1^i x\right)k_i^{-1}\, mod\ q,$ où s_1^i est le premier état interne, k_i le nonce, *x* la clé secrète et *q* une constante.

3. Procédé selon la revendication 2, dans lequel s_1^i = (g^{k_i} *mod p*) *mod q,* où *g* et *p* sont des constantes.

4. Procédé selon l'une des revendications 1 à 3, dans lequel la signature est le couple $\left(s_1^i, s_2^i\right)$ des premier et deuxième états internes.

5. Procédé selon l'une des revendications 1 à 4, comprenant une étape préalable (a0) de génération dudit ensemble de couples

$$\left(\left\{(s_1^i, WB_i)\right\}_{i\in[\![0,n-1]\!]}\right)$$

par des moyens de traitement de données (11b) d'un serveur (1b), et sa transmission à l'équipement (10a).

6. Procédé selon la revendication 5, dans lequel l'étape (a0) comprend la génération d'une pluralité de nonces

$$\left(\{\mathrm{k}_i\}_{i\in[\![0,n-1]\!]}\right),$$

puis la génération du couple (s_1^i, WB_i) pour chaque nonce (k_i).

7. Procédé selon les revendications 3 et 6 en combinaison, dans lequel l'étape (a0) comprend la génération préalable des constantes *p*, *q, g* conformément à l'algorithme DSA.

8. Procédé selon la revendication 7, dans lequel l'étape (a0) comprend également la génération préalable de la clé secrète (*x*) et d'une clé publique associée en fonction des constantes *p, q, g.*

9. Procédé selon l'une des revendications 1 à 8, dans lequel lesdites implémentations blanchies (WB_i) utilisent un Système de Représentation Modulaire, RNS, pour effectuer ladite opération d'arithmétique modulaire.

10. Procédé selon l'une des revendications 1 à 9, comprenant une étape subséquente (d) d'association par les moyens de traitement de données (11a) de l'équipement (10a) de la signature électronique générée au document de sorte à former le document signé

11. Produit programme d'ordinateur comprenant des instructions de code pour l'exécution d'un procédé selon l'une des revendications 1 à 10 de signature électronique d'un document avec une clé secrète (*x*) prédéterminée, lorsque ledit programme est exécuté par un ordinateur.

12. Moyen de stockage lisible par un équipement informatique sur lequel un produit programme d'ordinateur comprend des instructions de code pour l'exécution d'un procédé selon l'une des revendications 1 à 10 de de signature électronique d'un document avec une clé secrète (x) prédéterminée.

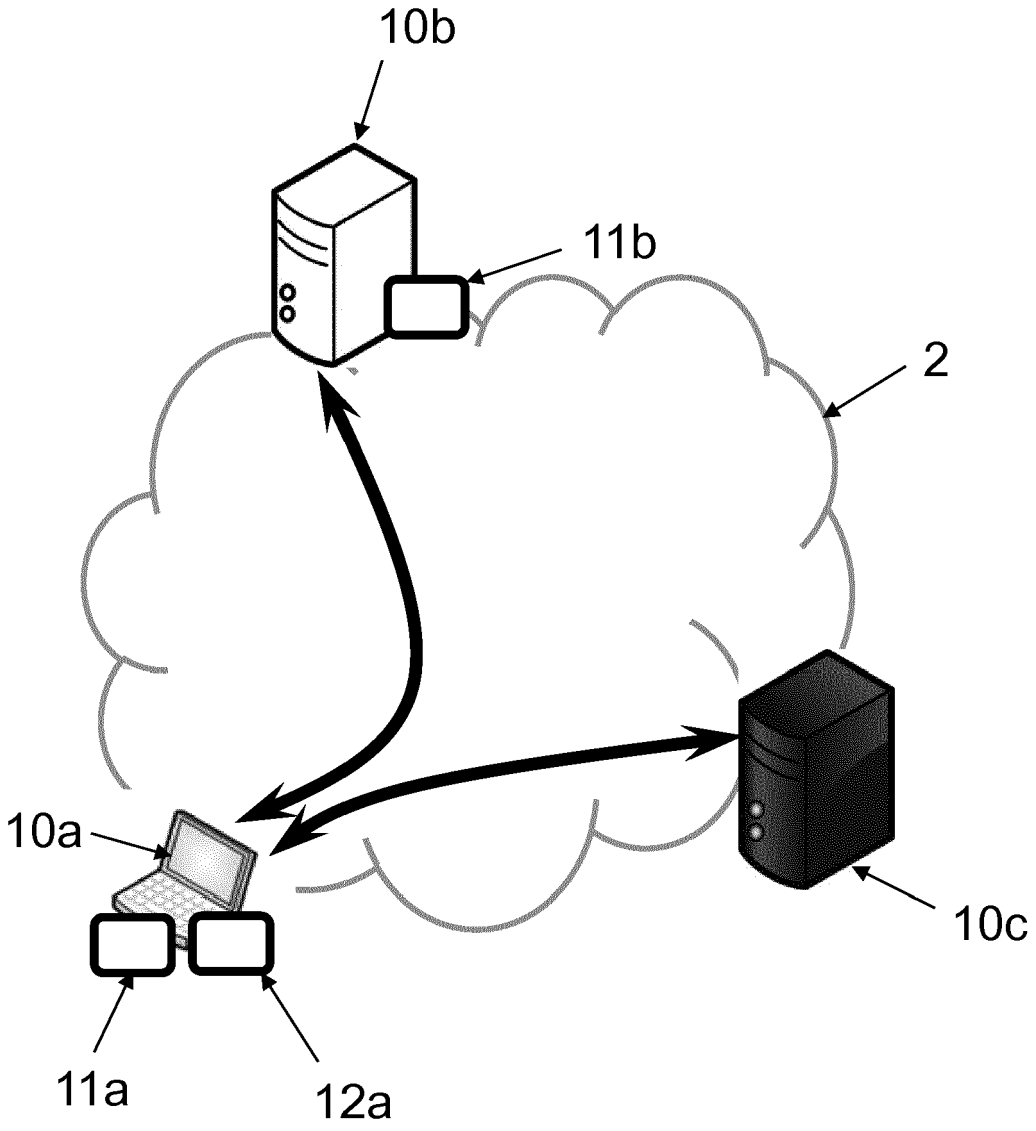

FIG. 1

Europäisches Patentamt
European Patent Office
Office européen des brevets

RAPPORT DE RECHERCHE EUROPEENNE

Numéro de la demande
EP 18 16 0420

DOCUMENTS CONSIDERES COMME PERTINENTS

Catégorie	Citation du document avec indication, en cas de besoin, des parties pertinentes	Revendication concernée	CLASSEMENT DE LA DEMANDE (IPC)
X	Jung Yeon Hwang ET AL: "Digital Signature Schemes with Restriction on Signing Capability" In: "Network and Parallel Computing", 1 janvier 2003 (2003-01-01), Springer International Publishing, Cham 032548, XP055424426, ISSN: 0302-9743 ISBN: 978-3-642-24784-2 vol. 2727, pages 324-335, DOI: 10.1007/3-540-45067-X_28, * section 5.2 *	1-12	INV. H04L9/30 H04L9/32
A,D	EP 3 059 894 A1 (NXP BV [NL]) 24 août 2016 (2016-08-24) * alinéas [0028], [0044] *	1-12	
A	WO 2009/136361 A1 (KONINKL PHILIPS ELECTRONICS NV [NL]; MICHIELS WILHELMUS P A J [NL]; GO) 12 novembre 2009 (2009-11-12) * page 5, lignes 18-25 *	1-12	DOMAINES TECHNIQUES RECHERCHES (IPC) H04L

Le présent rapport a été établi pour toutes les revendications

Lieu de la recherche	Date d'achèvement de la recherche	Examinateur
Munich	19 juillet 2018	Manet, Pascal

CATEGORIE DES DOCUMENTS CITES

X : particulièrement pertinent à lui seul
Y : particulièrement pertinent en combinaison avec un autre document de la même catégorie
A : arrière-plan technologique
O : divulgation non-écrite
P : document intercalaire

T : théorie ou principe à la base de l'invention
E : document de brevet antérieur, mais publié à la date de dépôt ou après cette date
D : cité dans la demande
L : cité pour d'autres raisons

& : membre de la même famille, document correspondant

EPO FORM 1503 03.82 (P04C02)

ANNEXE AU RAPPORT DE RECHERCHE EUROPEENNE RELATIF A LA DEMANDE DE BREVET EUROPEEN NO. EP 18 16 0420

La présente annexe indique les membres de la famille de brevets relatifs aux documents brevets cités dans le rapport de recherche européenne visé ci-dessus.
Lesdits members sont contenus au fichier informatique de l'Office européen des brevets à la date du
Les renseignements fournis sont donnés à titre indicatif et n'engagent pas la responsabilité de l'Office européen des brevets.

19-07-2018

Document brevet cité au rapport de recherche	Date de publication	Membre(s) de la famille de brevet(s)	Date de publication
EP 3059894 A1	24-08-2016	CN 105892991 A	24-08-2016
		EP 3059894 A1	24-08-2016
		US 2016239267 A1	18-08-2016
WO 2009136361 A1	12-11-2009	CA 2736898 A1	12-11-2009
		CN 102099780 A	15-06-2011
		EP 2286330 A1	23-02-2011
		EP 2669789 A2	04-12-2013
		JP 2011520150 A	14-07-2011
		JP 2014207717 A	30-10-2014
		KR 20110014630 A	11-02-2011
		US 2011064215 A1	17-03-2011
		WO 2009136361 A1	12-11-2009

EPO FORM P0460

Pour tout renseignement concernant cette annexe : voir Journal Officiel de l'Office européen des brevets, No.12/82

RÉFÉRENCES CITÉES DANS LA DESCRIPTION

Cette liste de références citées par le demandeur vise uniquement à aider le lecteur et ne fait pas partie du document de brevet européen. Même si le plus grand soin a été accordé à sa conception, des erreurs ou des omissions ne peuvent être exclues et l'OEB décline toute responsabilité à cet égard.

Documents brevets cités dans la description

- US 2016328543 A **[0007]**
- US 2016239267 A **[0051]**