(19) RÉPUBLIQUE FRANÇAISE

INSTITUT NATIONAL
DE LA PROPRIÉTÉ INDUSTRIELLE

PARIS

(11) **N° de publication :** **2 910 202**

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

(21) **N° d'enregistrement national :** **06 55633**

(51) Int Cl[8] : ***H 04 L 9/00*** *(2006.01), G 06 F 21/00, G 06 Q 90/00*

(12) DEMANDE DE BREVET D'INVENTION A1

(22) **Date de dépôt :** 19.12.06.

(30) **Priorité :**

(43) **Date de mise à la disposition du public de la demande :** 20.06.08 Bulletin 08/25.

(56) **Liste des documents cités dans le rapport de recherche préliminaire :** *Se reporter à la fin du présent fascicule*

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

(71) **Demandeur(s) :** *FRANCE TELECOM Société anonyme* — FR.

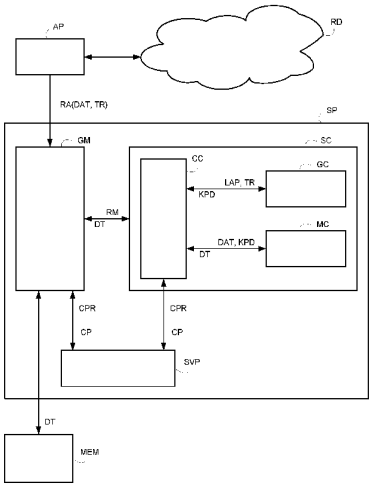
(72) **Inventeur(s) :** LOBRY OLIVIER, GERMAIN FLORENCE et ANNIC ETIENNE.

(73) **Titulaire(s) :**

(74) **Mandataire(s) :** MARTINET ET LAPOUX.

(54) TRAITEMENT DE DONNEE RELATIVE A UN RESEAU DE DONNEES.

(57) Pour traiter, par exemple chiffrer ou déchiffrer, une donnée numérique (DAT) crée, stockée, utilisée, ou destinée à être utilisée, par une application (AP) dédiée à un réseau de données (RD) et exécutée dans un terminal, une requête d'accès mémoire (RA) incluant la donnée numérique (DAT) ayant été émise par l'application (AP) à destination d'un système d'exploitation du terminal, un système de protection (SP) intercepte la requête d'accès mémoire (RA) destinée au système d'exploitation. Ce dernier identifie l'application (AP) à l'origine de ladite requête, un serveur de paramètres (SVP) détermine un identificateur (IRD) du réseau de données associé à l'application (AP), et un générateur de clé (GC) génère une clé (KPD) en fonction du réseau de données identifié par cet identificateur (IRD) afin de traiter la donnée numérique en fonction de la clé (KPD) générée.



FR 2 910 202 - A1

Traitement de donnée relative à un réseau de données

La présente invention concerne un traitement d'une donnée numérique qui est relative à un réseau de données.

Plus particulièrement, elle a trait à la protection par chiffrement/déchiffrement d'une donnée numérique relative à un réseau de données.

Dans le cadre de la présente invention, on appellera "réseau de données" un ensemble de ressources constitué par tout ou partie d'un équipement informatique et/ou un réseau de communication pouvant être une pluralité de réseaux de communication interconnectés, et désigné de manière unique par un identificateur appelé "identificateur de réseau de données". Le terme de réseau de données désignera par la suite indifféremment un ensemble de ressources et l'ensemble des données stockées ou transférées au sein de cet ensemble de ressources. Par ailleurs, deux réseaux de données quelconques sont supposés disjoints.

Les réseaux de communication concernés sont de tout type connu. Un exemple d'un réseau de communication, emprunté à la norme GPRS/UMTS ("General Packet Radio System"/"Universal Mobile Telecommunications System" en anglais), est un espace d'adressage identifié par un nom de point d'accès APN ("Access Point Name" en anglais).

Un réseau de données peut être, par exemple :

un réseau de communication accessible via un réseau d'accès cellulaire numérique du type UMTS, ou

le réseau Internet rendu accessible depuis un réseau d'accès WiFi ("Wireless Fidelity" en anglais), ou

un réseau de services bancaires accessible depuis un terminal bancaire avec un réseau d'accès sans contact, ou

une carte mémoire amovible, telle qu'une carte à puce, insérable dans un terminal et contenant un logiciel informatique, ou bien encore

tout ou partie d'un terminal.

Une "application" est classiquement définie comme une unité d'exécution d'un programme exécutable par un terminal. Un même programme peut être exécuté plusieurs fois de manière à ce que différentes applications requises par l'exécution du programme soient respectivement dédiées à des réseaux de données différents. Par exemple, un lecteur de média peut être exécuté à la fois pour lire un fichier appartenant au réseau Internet, et pour lire un fichier appartenant à un réseau de données proposant des services de musique à la demande.

Une application est dite "dédiée à un réseau de données" lorsque son exécution est lancée dans le contexte du réseau de données, c'est-à-dire sur les données stockées ou transférées au sein du réseau de données.

Une donnée numérique est dite "relative à un réseau de données" si cette donnée numérique est créée, mémorisée, utilisée, ou destinée à être utilisée, par une application dédiée à ce réseau de données. Dans ce cas, le réseau de données est dit "à l'origine de la donnée numérique" en question. Par exemple, des données mémorisées dans un terminal peuvent comprendre, d'une part, des données de

l'interface utilisateur du terminal relativement à un réseau de données interne au terminal, et d'autre part des données d'un porte-monnaie électronique inclus dans le terminal relativement à un réseau de données fournissant un service bancaire.

Dans la suite de la description, un "réseau de services" désigne un réseau de données offrant un ou plusieurs services, les services étant considérés comme des données du point de vue du réseau de données. Actuellement, certains réseaux de services offrent divers types de services dont la sécurité des données doit être garantie. Par exemple, des données numériques fournies par un service issu d'un réseau de services payants de téléchargement d'enregistrements vidéo ne doivent être exploitables que dans le cadre de ce réseau, et non pas dans le cadre d'un réseau de services gratuits. De même, des informations personnelles telles que des informations de nature bancaire relatives à un utilisateur doivent être accessibles seulement dans le cadre du réseau bancaire qui en est à l'origine.

Au travers d'un réseau d'accès cellulaire, un terminal mobile actuel peut accéder consécutivement à plusieurs réseaux de données, mais n'est pas capable de confiner les données à leurs réseaux de données d'origine. Avec un tel terminal, il est par exemple possible de télécharger des données issues d'un réseau de données dans une mémoire amovible installée dans un terminal mobile et de les copier dans un autre réseau de données après avoir installé la mémoire amovible sur un autre terminal. Ainsi, des données confidentielles issues d'un réseau bancaire peuvent être copiées sur le réseau Internet au moyen de la mémoire amovible du terminal.

Pour pallier ce problème, une solution consiste à interdire la mémorisation de données sur un terminal. De ce fait, des services nécessitant la mémorisation d'un nombre élevé de données sur le terminal sont pénalisés commercialement. Par exemple, un service de jeu en réseau nécessite le téléchargement d'un jeu qui peut requérir plus d'une heure. Dans ce cas, le téléchargement systématique du jeu à chaque utilisation de ce dernier est une contrainte rédhibitoire à la commercialisation du jeu.

Une autre solution consiste à contrôler l'usage des ressources mémoire d'un terminal. Plusieurs approches existent actuellement. Une première approche consiste à contrôler le droit d'accès aux différentes ressources mémoire d'un terminal par le système d'exploitation gérant ces ressources, notamment les systèmes de fichiers. Le système d'exploitation vérifie qu'une application exécutée pour le compte d'un utilisateur donné possède bien les droits d'accès en écriture ou en lecture au fichier concerné avant de satisfaire la requête. Ainsi, à chaque fichier peuvent être associés des droits différents en fonction de l'utilisateur. Une seconde approche consiste à protéger directement les données destinées à être mémorisées, en fonction de l'utilisateur de ces données. Plus précisément, cela consiste à rendre les données inintelligibles en l'absence d'un décodeur approprié activable par un ayant-droit, et ce, via un mécanisme de chiffrement. Ce dernier est mis en œuvre par le système d'exploitation d'un terminal qui peut chiffrer un fichier donné, avec une clé attribuée à l'utilisateur du terminal pour le compte duquel le fichier est

manipulé. Ces deux dernières approches ont pour inconvénient majeur d'être "orientées utilisateur". Ainsi, aucune d'elle n'interdit à un utilisateur quelconque d'échanger des données entre deux applications dédiées respectivement à deux réseaux de données distincts et s'exécutant sur un même système d'exploitation.

Pour remédier aux inconvénients évoqués ci-dessus, un procédé selon l'invention pour traiter une donnée numérique créée, mémorisée, utilisée, ou destinée à être utilisée par une application dédiée à un réseau de données et exécutée dans un terminal, une requête d'accès mémoire incluant la donnée numérique ayant été émise par l'application à destination d'un système d'exploitation du terminal, est caractérisé en ce qu'il comprend les étapes suivantes :

après interception de la requête d'accès mémoire destinée au système d'exploitation, identifier l'application à l'origine de ladite requête,

déterminer un identificateur du réseau de données associé à l'application identifiée,

générer une clé en fonction de l'identificateur du réseau de données déterminé, et

traiter la donnée numérique en fonction de la clé générée.

Dans le cadre de la présente invention, il est supposé qu'une application est toujours lancée dans le contexte d'un et un seul réseau de données, généralement à l'initiative de l'utilisateur. Le lien entre une application et le réseau de données peut par exemple être effectué par le système d'exploitation du terminal lors du lancement de

l'exécution de l'application, éventuellement à l'insu de cette dernière.

L'invention suppose par ailleurs que le système d'exploitation du terminal gère explicitement, c'est-à-dire mémorise, protège et rend accessible, d'une part la correspondance entre un processus système et une application, et d'autre part la correspondance entre une application et un identificateur de réseau de données associé.

L'invention garantit l'absence totale d'échange de données entre deux réseaux de données distincts quelconques via un support mémoire, y compris via un support mémoire amovible. L'invention confine les données dans leurs réseaux de données d'origine, y compris lorsque ces données résultent de l'exécution d'applications, en prolongeant ce confinement jusqu'à leur stockage en mémoire, y compris pour un stockage sur un support d'enregistrement du type mémoire persistante amovible. De ce fait, nous dirons de l'invention qu'elle réalise un "confinement à l'exécution" des réseaux de données. Cela signifie en premier lieu que toute donnée résultant de l'exécution d'une application dédiée à un réseau de données est nécessairement "attribuée" à ce seul réseau de données, et en second lieu qu'une application qui est lancée dans le contexte du réseau de données n'a accès à aucune donnée appartenant à un autre réseau de données ou résultant de l'exécution d'une application lancée dans le contexte d'un autre réseau de données.

Par ailleurs, un terminal est dit "compatible avec un réseau de données" lorsque ce terminal garantit un confinement à l'exécution de ce réseau de données. Par conséquent, l'accès à un réseau de

données est réservé aux terminaux compatibles avec ce réseau de données.

Ainsi, dans un terminal compatible avec des premier et deuxième réseaux de données, une première application relative au premier réseau de données ne peut accéder à une donnée élaborée par une deuxième application relative au deuxième réseau de données. De même, si une donnée est stockée dans une mémoire amovible par une première application dédiée à un premier réseau de données et exécutée dans un premier terminal, une deuxième application dédiée à un deuxième réseau de données et exécutée dans un deuxième terminal ne peut accéder à la donnée lorsque la mémoire amovible est installée dans le deuxième terminal.

En proposant une solution au problème de sécurisation du stockage des données issues de réseaux de données, l'invention offre à l'utilisateur d'un terminal un gain de temps et d'argent résultant d'une économie de chargements. Pour un opérateur de réseau de télécommunications, l'invention apporte un élargissement des opportunités de services, et un gain en exploitation en évitant de congestionner le réseau par le rechargement, à chaque utilisation, des services auxquels l'utilisateur est déjà abonné.

L'invention optimise de plus le téléchargement de données dans un terminal en facilitant la gestion des reprises sur incident, telle qu'une extinction du terminal. En effet, un téléchargement interrompu peut être repris dès la fin de l'incident, et les données déjà téléchargées sont mémorisées dans une mémoire persistante du terminal pour ne pas perdre le bénéfice du téléchargement déjà réalisé.

Selon une autre caractéristique de l'invention, la donnée numérique peut être un identificateur d'un fichier mémorisé dans le terminal et des première et deuxième clés sont générées afin de traiter l'identificateur de fichier et le fichier mémorisé respectivement en fonction des première et deuxième clés générées.

Le traitement d'un identificateur d'un fichier par exemple téléchargé et mémorisé dans une mémoire volatile du terminal assure un premier niveau de protection quant à la localisation du fichier mémorisé.

L'invention concerne également un système pour traiter une donnée numérique créée, mémorisée, utilisée, ou destinée à être utilisée par une application dédiée à un réseau de données et exécutée dans un terminal, une requête d'accès mémoire incluant la donnée numérique ayant été émise par l'application à destination du système d'exploitation du terminal. Le système est caractérisé en ce qu'il comprend :

- un moyen pour identifier l'application à l'origine de la requête d'accès mémoire destinée au système d'exploitation, après interception de ladite requête,
- un moyen pour déterminer un identificateur du réseau de données associé à l'application identifiée,
- un moyen pour générer une clé en fonction de l'identificateur du réseau de données déterminé, et
- un moyen pour traiter la donnée numérique en fonction de la clé générée.

Selon une autre caractéristique de l'invention, le système peut comprendre un moyen interceptant la

requête d'accès mémoire destinée au système d'exploitation pour introduire des champs de paramètres de contexte, associés notamment au contexte de lancement et d'exécution de l'application courante, dans la requête interceptée, et au moins un moyen pour renseigner les champs de paramètres de contexte par des valeurs dont une correspond à l'identificateur du réseau de données.

Le système de protection selon l'invention intercepte systématiquement tout appel à une mémoire du terminal par une application afin d'assurer une protection des accès en lecture ou en écriture à la mémoire. En particulier, des champs de paramètres de contexte sont renseignés de manière à paramétrer l'appel à une mémoire par au moins un identificateur du réseau de données dans le contexte duquel s'exécute l'application.

Enfin, l'invention se rapporte à un programme d'ordinateur apte à être mis en œuvre dans un système pour traiter une donnée numérique créée, mémorisée, utilisée, ou destinée à être utilisée par une application dédiée à un réseau de données et exécutée dans un terminal, ledit programme comprenant des instructions qui, lorsque le programme est exécuté dans ledit système, réalisent les étapes selon le procédé de l'invention.

D'autres caractéristiques et avantages de la présente invention apparaîtront plus clairement à la lecture de la description suivante de plusieurs réalisations de l'invention données à titre d'exemples non limitatifs, en référence aux dessins annexés correspondants dans lesquels :

- la figure 1 est un bloc-diagramme schématique d'un système de protection selon l'invention pour traiter une donnée relative à un réseau de données ; et

- la figure 2 est l'algorithme d'un procédé de traitement de donnée selon l'invention mis en œuvre dans le système de protection.

En référence à la figure 1, le système de protection SP selon l'invention est inclus dans un terminal informatique capable de gérer des données numériques.

Le terminal informatique peut être de tout type et peut gérer tout type de données. Par exemple, le terminal est un ordinateur personnel, un terminal radio mobile, un terminal bancaire, un serveur applicatif ou encore une caméra de surveillance.

Le système de protection SP comprend un gestionnaire de mémoire GM, un système de chiffrement SC et au moins un serveur de paramètres SVP. Le système de chiffrement SC comprend un contrôleur de chiffrement CC, un générateur de clé GC et un module de chiffrement MC. Le serveur de paramètres SVP communique directement avec le gestionnaire de mémoire GM et/ou le contrôleur de chiffrement CC. Les communications entre les différentes entités comprises dans le système de protection SP sont sécurisées, de manière à ce qu'aucun échange d'information ne puisse être intercepté pendant une communication entre deux de ces entités.

Le gestionnaire de mémoire GM gère à travers le système d'exploitation du terminal l'accès en écriture et en lecture à des mémoires MEM du terminal, telles qu'une mémoire persistante ou une mémoire volatile.

Le serveur de paramètres SVP gère une base de données qui comprend notamment des informations nécessaires au système de protection selon l'invention telles que des paramètres requis par le générateur de clé GC.

Toujours en référence à la figure 1, le terminal comprend des mémoires MEM qui regroupent une ou plusieurs mémoires volatiles et une ou plusieurs mémoires persistantes accessibles à travers le système d'exploitation depuis le gestionnaire de mémoire GM du système de protection. En outre, des mémoires persistantes peuvent être implantées sur un support amovible du terminal.

Le système d'exploitation du terminal exécute des applications AP et gère le partage des ressources du terminal entre les applications.

Une application AP qui est exécutée par le système d'exploitation du terminal dans le contexte d'un réseau de données RD peut demander la mémorisation d'une donnée dans l'une des mémoires MEM du terminal.

En particulier, le système de protection SP peut être mis en œuvre dans un composant électronique de type microcontrôleur, circuit à microprocesseur ou carte à puce, afin que le système ne soit pas piraté, espionné ou modifié. Ce mode de réalisation est particulièrement intéressant lorsque le composant électronique inclut, en plus du système de protection SP, une mémoire volatile, une mémoire persistante, stockant par exemple les codes applicatifs permettant au système d'exploitation de s'exécuter, et un processeur dans lequel s'exécute le code source du système de protection. En effet, une faiblesse du système de protection est que son exécution réclame

que son code source soit écrit en "clair". Si le code source est écrit en mémoire volatile, il est possible de copier le code source de la mémoire, d'en comprendre le fonctionnement et de le détourner de ce fonctionnement. En revanche, si le code source du système de protection est exécuté au sein du composant électronique, il est beaucoup plus difficile de le pirater.

La mémoire persistante du composant électronique contient des applications critiques comme, par exemple, un système d'exploitation certifié mettant en œuvre le gestionnaire de mémoire GM, ou un navigateur certifié.

Quand un service nécessitant un haut niveau de sécurité doit être exécuté sur le terminal, il est alors exécuté au moyen du processeur et des mémoires volatiles et persistantes du composant électronique. Les capacités en mémoire volatile et persistante du composant électronique étant, par essence, limitées, le composant électronique peut stocker des données relatives au service sur les mémoires volatiles ou persistantes du terminal. Toutes les données relatives au service nécessitant d'être sécurisées sont préalablement traitées par le système de protection SP du composant électronique avant d'être stockées sur lesdites mémoires volatiles ou persistantes du terminal.

Ainsi, même si une application dite "pirate" possède des moyens matériels d'espionner la mémoire volatile du terminal pour extraire des données confidentielles provenant d'un service, l'application ne pourra pas accéder à ces données car ces dernières auront été préalablement chiffrées par le système de chiffrement SC. Par conséquent, il n'est pas possible d'extraire des données provenant d'un réseau de

données, d'y accéder et de les copier dans un autre réseau de données de manière illégitime.

Par ailleurs, le système de chiffrement SC peut être réalisé en logique câblée plutôt que sous forme de logiciel. En effet, la logique câblée optimise le fonctionnement en augmentant considérablement les performances. Puisque le temps de traitement de données du système de chiffrement SC constitue le principal délai pour le stockage de données en mémoire, si ce traitement est trop long, l'exécution des services peut être aussi perçue comme trop longue par l'utilisateur du terminal. Pour que le fonctionnement du terminal ne soit pas pénalisé par le temps de traitement du système de chiffrement SC, ce dernier doit être de préférence optimisé, par exemple en ayant recours à une réalisation en logique câblée.

Enfin, cette réalisation permet de faire appel à des applications certifiées qui seront stockées et exécutées dans le composant électronique de manière à garantir que ces applications ne seront pas modifiées ou piratées, ce qui permet de prolonger leur certification jusqu'à l'exécution. On notera que l'usage d'applications certifiées est un pré-requis exigé par la plupart des services nécessitant un haut degré de sécurité tels qu'un service bancaire, le contrôle d'accès à un site, ou la gestion des droits numériques DRM ("Digital Rights Management" en anglais).

En référence à la figure 2, le procédé selon l'invention comprend des étapes E1 à E7 exécutées automatiquement sous le contrôle du système de protection SP inclus dans le terminal et mises en œuvre par des instructions d'un programme

d'ordinateur enregistré sur un support d'enregistrement lisible par le système de protection SP.

Au préalable, un utilisateur du terminal a souscrit un abonnement auprès d'un opérateur gérant un réseau de données et devient ainsi un abonné à au moins un réseau de données.

A l'étape E1, une application AP est exécutée dans le terminal dans le contexte d'un réseau de données RD. Par exemple, l'utilisateur du terminal a sélectionné le lancement de l'application via une interface homme-machine du terminal indiquant à l'utilisateur les réseaux de données auquel ce dernier est abonné et les applications disponibles pour ces réseaux de données. A titre d'exemple, l'application est un lecteur de média qui est lancé dans le contexte d'un réseau de données proposant des services de musique à la demande.

L'application exécutée AP transmet une requête d'accès mémoire RA à destination du système d'exploitation du terminal afin d'accéder aux mémoires MEM du terminal. La requête d'accès mémoire RA inclut un type de requête TR indiquant le type d'accès à la mémoire MEM requis par l'application. Par exemple, le type d'accès est une lecture ou une écriture de données. La requête d'accès mémoire RA inclut en outre une donnée numérique DAT à traiter, c'est-à-dire à chiffrer ou à déchiffrer. La donnée numérique DAT peut servir à identifier un contenu numérique dit "simple" ou "structuré". Un contenu numérique simple peut être une variable ou une chaîne de caractères telle qu'un nom ou une date. Un contenu numérique structuré peut être un ensemble hétérogène de données simples, comme un fichier déjà mémorisé

dans l'une des mémoires MEM et référencé par un identificateur inclus dans la donnée à traiter.

A titre d'exemple, la requête d'accès mémoire RA requiert l'écriture d'un fichier dans une mémoire persistante des mémoires MEM du terminal, et inclut un identificateur de ce fichier qui est mémorisé dans une mémoire volatile des mémoires MEM.

A l'étape E2, la requête RA destinée au système d'exploitation est interceptée par le gestionnaire de mémoire GM du système de protection SP qui est à l'écoute de toute requête provenant de toute application à destination du système d'exploitation du terminal. Le gestionnaire de mémoire GM identifie alors l'application AP qui est à l'origine de la requête RA, par exemple au moyen d'une correspondance entre un processus système et un identificateur de l'application.

Le gestionnaire de mémoire GM modifie la requête interceptée RA en une requête modifiée RM en la complétant par un ou plusieurs champs de paramètres de contexte CP, l'un d'eux étant nécessairement destiné à être renseigné par un identificateur IRD du réseau de données dans le contexte duquel l'application a été lançée. Les champs peuvent être renseignés par les valeurs d'autres paramètres tels qu'un identificateur de l'utilisateur ou un identificateur du terminal. Optionnellement, le gestionnaire de mémoire GM mémorise temporairement la requête modifiée RM.

A l'étape E3, le gestionnaire de mémoire GM transmet la requête modifiée RM au contrôleur de chiffrement CC du système de chiffrement SC. Le contrôleur de chiffrement CC mémorise la requête modifiée RM, notamment le type de requête TR et la donnée à traiter DAT inclus dans la requête RM.

A l'étape E4, le contrôleur de chiffrement CC interroge le serveur de paramètres SVP afin de renseigner les champs de paramètres de contexte CP dans la requête modifiée RM.

Le serveur de paramètres SVP détermine les valeurs des paramètres, notamment l'identificateur du réseau de données IRD, associés à l'application AP, afin de renseigner les champs de la requête modifiée par ces valeurs. Avantageusement, si le serveur SVP ne peut pas renseigner la valeur du paramètre de contexte "identificateur du réseau de données", la procédure de cryptage échoue, entraînant ainsi l'échec de l'opération de lecture/écriture en mémoire et le procédé se termine, assurant ainsi la sécurité des données.

Par exemple, dans le cas d'un terminal compatible avec un réseau de données, le système d'exploitation du terminal gère une correspondance entre l'application AP, dont l'exécution est à l'origine de la requête RA dans le terminal, et un identificateur du réseau de données auquel est dédiée l'application. Le serveur de paramètres SVP consulte alors une table de correspondances entre des identificateurs d'application et des identificateurs de réseau de données IRD.

Selon un autre exemple, l'application AP est spécifique à une carte à puce qui est reliée au terminal et considérée comme constituant un réseau de données RD. L'identificateur de réseau de données est alors le numéro de série de la carte à puce.

Le serveur de paramètres SVP transmet les champs de paramètres de contexte renseignés CPR au contrôleur de chiffrement CC qui les mémorise.

En variante, les étapes E3 et E4 sont remplacées respectivement par des étapes E31 et E41.

A l'étape E31, le gestionnaire de mémoire GM interroge le serveur de paramètres SVP afin de renseigner les champs de paramètres de contexte CP dans la requête modifiée RM. Le serveur de paramètres SVP détermine et transmet les paramètres de contexte renseignés CPR au gestionnaire de mémoire GM.

A l'étape E41, le gestionnaire de mémoire GM renseigne les champs de la requête modifiée par les paramètres de contexte associés dans la requête modifiée RM et transmet cette dernière au contrôleur de chiffrement CC du système de chiffrement SC qui la mémorise.

A l'étape E5, le contrôleur de chiffrement CC transmet les champs de paramètres de contexte renseignés CPR et le type de requête TR au générateur de clé GC.

Le générateur de clé GC génère au moins une clé de protection de données KPD en fonction des valeurs des champs de paramètres de contexte, et notamment de l'identificateur du réseau de données IRD, et en fonction du type de requête TR et de la nature des données à traiter DAT incluses dans la requête RA.

Par exemple, dans le cas où la donnée à traiter DAT représente un contenu numérique "simple", le contrôleur de chiffrement requiert au générateur de clé la génération d'une seule clé KPD. La clé KPD est une clé de chiffrement si le type de requête TR est relatif à une écriture, ou est une clé de déchiffrement si le type de requête TR est relatif à une lecture.

Dans un autre exemple, si la donnée à traiter DAT est un identificateur référençant un contenu

numérique "structuré" tel qu'un fichier, et si le type de requête est relatif à une lecture, le contrôleur de chiffrement requiert de la part du générateur de clé la génération d'une première clé pour chiffrer l'identificateur de fichier et d'une deuxième clé pour déchiffrer le contenu du fichier référencé par l'identificateur. Si le type de requête est relatif à une écriture, le contrôleur de chiffrement requiert de la part du générateur de clé la génération d'une seule clé pour chiffrer l'identificateur de fichier et le contenu du fichier référencé par l'identificateur.

Le générateur de clé GC transmet alors la clé générée KPD au contrôleur de chiffrement CC.

La clé générée KPD est unique et irrévocable. Pour une liste de paramètres de contexte donnée est générée une unique clé KPD.

Une donnée chiffrée au moyen d'une liste de paramètres de contexte ne pourra être déchiffrée qu'au moyen d'une clé générée en fonction de cette même liste de paramètres de contexte. Puisque la liste de paramètres de contexte contient au moins un identificateur de réseau de données IRD, une donnée chiffrée à la demande d'une application issue d'un réseau de données ne pourra être déchiffrée qu'à la demande d'une application qui est exécutée dans le contexte du même réseau de données.

En variante, le générateur de clé GC extrait une clé de protection de données KPD parmi une liste de clés pré-calculées en fonction de la liste de paramètres de contexte et du type de requête TR, la clé KPD étant soit une clé de chiffrement KC, soit une clé de déchiffrement KD.

A l'étape E6, le contrôleur de chiffrement CC transmet la donnée à traiter DAT, la clé générée KPD et le type de requête TR au module de chiffrement MC.

Le module de chiffrement MC traite la donnée reçue DAT en fonction du type de requête TR et de la clé générée KPD. Le module de chiffrement MC possède au moins deux types d'algorithme pour le traitement des données, par exemple un algorithme de chiffrement auquel est applicable toute clé de chiffrement et un algorithme de déchiffrement auquel est applicable toute clé de déchiffrement. Par conséquent, l'algorithme de chiffrement ou de déchiffrement utilisé par le module de chiffrement MC est dual de l'algorithme de génération des clés de chiffrement ou de déchiffrement utilisé par le générateur de clé GC.

Le module de chiffrement MC chiffre la donnée reçue DAT avec la clé KPD si le type de requête TR est relatif à une écriture, ou déchiffre la donnée reçue DAT avec la clé KPD si le type de requête TR est relatif à une lecture. La donnée chiffrée ou déchiffrée est alors une donnée traitée DT.

Le module de chiffrement MC transmet alors la donnée traitée DT au contrôleur de chiffrement CC.

En variante, la donnée à traiter DAT est transmise par le contrôleur de chiffrement CC au module de chiffrement MC avant l'étape E5, c'est-à-dire avant la génération d'une clé KPD.

A l'étape E7, le contrôleur de chiffrement CC transmet la donnée traitée DT au gestionnaire de mémoire GM afin que ce dernier mémorise la donnée traitée DT dans l'une des mémoires MEM du terminal selon le type de requête TR.

Par exemple, si le type de requête TR est relatif à une écriture, la donnée traitée DT pourra être mémorisée dans une mémoire persistante, et si le

type de requête TR est relatif à une lecture, la donnée traitée DT pourra être mémorisée dans une mémoire volatile dont l'adresse est transmise à l'application AP pour que cette dernière lise la donnée traitée mémorisée.

Dans une variante, les étapes E6 et E7 sont complétées comme expliqué ci-après.

La donnée à traiter DAT incluse dans la requête est par exemple un identificateur d'un fichier mémorisé dans une mémoire volatile du terminal et le type de requête TR est relatif à une écriture.

Dans cette variante, à l'étape E5, le contrôleur de chiffrement CC requiert de la part du générateur de clé GC la génération d'une clé de chiffrement, et charge le fichier mémorisé dans la mémoire volatile via le gestionnaire de mémoire GM, par exemple au moyen d'une référence de l'adresse mémoire du fichier, telle qu'un pointeur initialement inclus dans la requête d'accès mémoire RA.

Puis à l'étape E6, le contrôleur de chiffrement CC transmet la donnée à traiter DAT, c'est-à-dire l'identificateur de fichier, ainsi que le fichier chargé, la clé générée et le type de requête TR au module de chiffrement MC. Le module de chiffrement MC chiffre l'identificateur de fichier et le fichier avec la clé et transmet l'identificateur chiffré et le fichier chiffré au contrôleur de chiffrement CC.

A l'étape E7, le contrôleur de chiffrement CC commande au gestionnaire de mémoire GM la mémorisation de l'identificateur chiffré et du fichier chiffré dans une mémoire du terminal.

Dans une autre variante, l'étape E6 est complétée comme expliqué ci-après.

La donnée à traiter DAT est par exemple un identificateur d'un fichier mémorisé dans une mémoire persistante du terminal et le type de requête TR est relatif à une lecture.

Dans cette variante, à l'étape E5, le contrôleur de chiffrement CC requiert de la part du générateur de clé GC la génération d'une clé de chiffrement et d'une clé de déchiffrement qui peuvent être identiques.

Puis à l'étape E6, le contrôleur de chiffrement CC transmet la donnée à traiter DAT, c'est-à-dire l'identificateur de fichier, la clé de chiffrement et le type de requête TR au module de chiffrement MC. Le module de chiffrement MC chiffre l'identificateur de fichier avec la clé de chiffrement et transmet l'identificateur chiffré au contrôleur de chiffrement CC.

Le contrôleur de chiffrement CC charge le fichier chiffré mémorisé dans la mémoire volatile via le gestionnaire de mémoire GM au moyen de l'identificateur chiffré. Le contrôleur de chiffrement CC transmet alors le fichier chiffré chargé, la clé de déchiffrement et le type de requête TR au module de chiffrement MC. Le module de chiffrement MC déchiffre le fichier chiffré avec la clé de déchiffrement et transmet le fichier déchiffré au contrôleur de chiffrement CC.

A l'étape E7, le contrôleur de chiffrement CC commande au gestionnaire de mémoire GM la mémorisation du fichier déchiffré dans une mémoire volatile du terminal pour être lisible par l'application AP.

Dans un souci de synchronisation entre les opérations exécutées par le générateur de clé GC et

le module de chiffrement MC, le contrôleur de chiffrement CC reçoit une clé de protection de données KPD générée par le générateur de clé GC et associe la clé KPD à la donnée à traiter préalablement chargée avant de les transmettre au module de chiffrement MC. Cette association garantit que la clé et la donnée à traiter transmises au module de chiffrement MC correspondent à une même requête. Ainsi, plusieurs requêtes peuvent être traitées simultanément et indépendamment les unes des autres par le contrôleur de chiffrement CC.

En variante, les fonctionnalités du contrôleur de chiffrement CC sont intégrées partiellement ou totalement dans le gestionnaire de mémoire GM et/ou le générateur de clé GC et/ou le module de chiffrement MC. Dans ce cas, le gestionnaire de mémoire GM communique directement avec le générateur de clé GC et le module de chiffrement MC et ces deux derniers communiquent également directement entre eux.

L'invention décrite ici concerne un procédé et un système pour traiter une donnée numérique créée, mémorisée, utilisée, ou destinée à être utilisée par une application AP dédiée à un réseau de données RD et exécutée dans un terminal. Selon une implémentation, les étapes du procédé de l'invention sont déterminées par les instructions d'un programme d'ordinateur incorporé dans le système de protection selon l'invention. Le programme comporte des instructions de programme qui, lorsque ledit programme est exécuté dans le système dont le fonctionnement est alors commandé par l'exécution du

programme, réalisent les étapes du procédé selon l'invention.

En conséquence, l'invention s'applique également à un programme d'ordinateur, notamment un programme d'ordinateur enregistré sur ou dans un support d'informations lisible par un ordinateur et tout dispositif de traitements de données, adapté à mettre en œuvre l'invention. Ce programme peut utiliser n'importe quel langage de programmation, et être sous la forme de code source, code objet, ou de code intermédiaire entre code source et code objet tel que dans une forme partiellement compilée, ou dans n'importe quelle autre forme souhaitable pour implémenter le procédé selon l'invention.

Le support d'informations peut être n'importe quelle entité ou dispositif capable de stocker le programme. Par exemple, le support peut comporter un moyen de stockage ou support d'enregistrement sur lequel est enregistré le programme d'ordinateur selon l'invention, tel qu'une ROM, par exemple un CD ROM ou une ROM de circuit microélectronique, ou encore une clé USB, ou un moyen d'enregistrement magnétique, par exemple une disquette ("floppy disc") ou un disque dur.

D'autre part, le support d'informations peut être un support transmissible tel qu'un signal électrique ou optique, qui peut être acheminé via un câble électrique ou optique, par radio ou par d'autres moyens. Le programme selon l'invention peut être en particulier téléchargé sur un réseau de type internet.

Alternativement, le support d'informations peut être un circuit intégré dans lequel le programme est incorporé, le circuit étant adapté pour exécuter ou

pour être utilisé dans l'exécution du procédé selon l'invention.

REVENDICATIONS

1 - Procédé pour traiter une donnée numérique (DAT) créée, mémorisée, utilisée, ou destinée à être utilisée par une application (AP) dédiée à un réseau de données (RD) et exécutée dans un terminal, une requête d'accès mémoire (RA) incluant la donnée numérique (DAT) ayant été émise par l'application (AP) à destination d'un système d'exploitation du terminal, caractérisé en ce qu'il comprend les étapes suivantes :

après interception (E2) de la requête d'accès mémoire (RA) destinée au système d'exploitation, identifier l'application (AP) à l'origine de ladite requête,

déterminer (E4) un identificateur (IRD) du réseau de données associé à l'application identifiée (AP),

générer (E5) une clé (KPD) en fonction de l'identificateur du réseau de données déterminé, et

traiter (E6) la donnée numérique en fonction de la clé générée (KPD).

2 - Procédé conforme à la revendication 1, selon lequel la clé (KPD) est générée en outre en fonction d'un type de requête (TR) inclus dans la requête d'accès mémoire (RA).

3 - Procédé conforme à la revendication 1 ou 2, selon lequel la clé générée (KPD) est une clé de chiffrement et la donnée numérique (DAT) est chiffrée en fonction de la clé de chiffrement générée (KPD).

4 - Procédé conforme à la revendication 1 ou 2, selon lequel la clé générée (KPD) est une clé de

déchiffrement et la donnée numérique (DAT) est déchiffrée en fonction de la clé de déchiffrement générée (KPD).

5 – Procédé conforme à l'une des revendications 1 à 4, selon lequel la donnée numérique (DAT) est un identificateur d'un fichier mémorisé dans le terminal et des première et deuxième clés (KPD) sont générées afin de traiter la donnée numérique et le fichier mémorisé respectivement en fonction des première et deuxième clés générées.

6 – Système pour traiter une donnée numérique (DAT) créée, mémorisée, utilisée, ou destinée à être utilisée par une application (AP) dédiée à un réseau de données (RD) et exécutée dans un terminal, une requête d'accès mémoire (RA) incluant la donnée numérique (DAT) ayant été émise par l'application (AP) à destination du système d'exploitation du terminal, caractérisé en ce qu'il comprend :

- un moyen (GM) pour identifier l'application (AP) à l'origine la requête d'accès mémoire (RA) destinée au système d'exploitation, après interception de ladite requête,
- un moyen (SVP) pour déterminer un identificateur (IRD) du réseau de données associé à l'application identifiée (AP),
- un moyen (GC) pour générer une clé (KPD) en fonction de l'identificateur du réseau de données déterminé, et
- un moyen (MC) pour traiter la donnée numérique (DAT) en fonction de la clé générée (KPD).

7 – Système conforme à la revendication 6, comprenant un moyen (GM) interceptant la requête

d'accès mémoire (RA) destinée au système d'exploitation pour introduire des champs de paramètres de contexte (CP) dans la requête interceptée (RA), et au moins un moyen (SVP) pour renseigner les champs de paramètres de contexte par des valeurs dont une correspond à l'identificateur (IRD) du réseau de données.

8 - Système conforme à la revendication 6 ou à la revendication 7, inclus dans un composant électronique.

9 - Programme d'ordinateur apte à être mis en œuvre dans un système pour traiter une donnée numérique (DAT) créée, mémorisée, utilisée, ou destinée à être utilisée par une application (AP) dédiée à un réseau de données (RD) et exécutée dans un terminal, une requête d'accès mémoire (RA) incluant la donnée numérique (DAT) ayant été émise par l'application (AP) à destination du système d'exploitation du terminal, ledit programme étant caractérisé en ce qu'il comprend des instructions qui, lorsque le programme est exécuté dans ledit système, réalisent les étapes de :

après interception (E2) de la requête d'accès mémoire (RA) destinée au système d'exploitation, identifier l'application (AP) à l'origine de ladite requête,

déterminer (E4) un identificateur (IRD) du réseau de données associé à l'application identifiée (AP),

générer (E5) une clé (KPD) en fonction de l'identificateur du réseau de données déterminé, et

traiter (E6) la donnée numérique (DAT) en fonction de la clé générée (KPD).

10 - Support d'enregistrement lisible par un système pour traiter une donnée numérique (DAT) créée, mémorisée, utilisée, ou destinée à être utilisée par une application (AP) dédiée à un réseau de données (RD) et exécutée dans un terminal, une requête d'accès mémoire (RA) incluant la donnée numérique (DAT) ayant été émise par l'application (AP) à destination d'un système d'exploitation du terminal, caractérisé en ce qu'il a enregistré un programme d'ordinateur comportant des instructions pour l'exécution des étapes suivantes :

après interception (E2) de la requête d'accès mémoire (RA) destinée au système d'exploitation, identifier l'application (AP) à l'origine de ladite requête,

déterminer (E4) un identificateur (IRD) du réseau de données associé à l'application identifiée (AP),

générer (E5) une clé (KPD) en fonction de l'identificateur du réseau de données déterminé, et

traiter (E6) la donnée numérique (DAT) en fonction de la clé générée (KPD).

FIG. 1

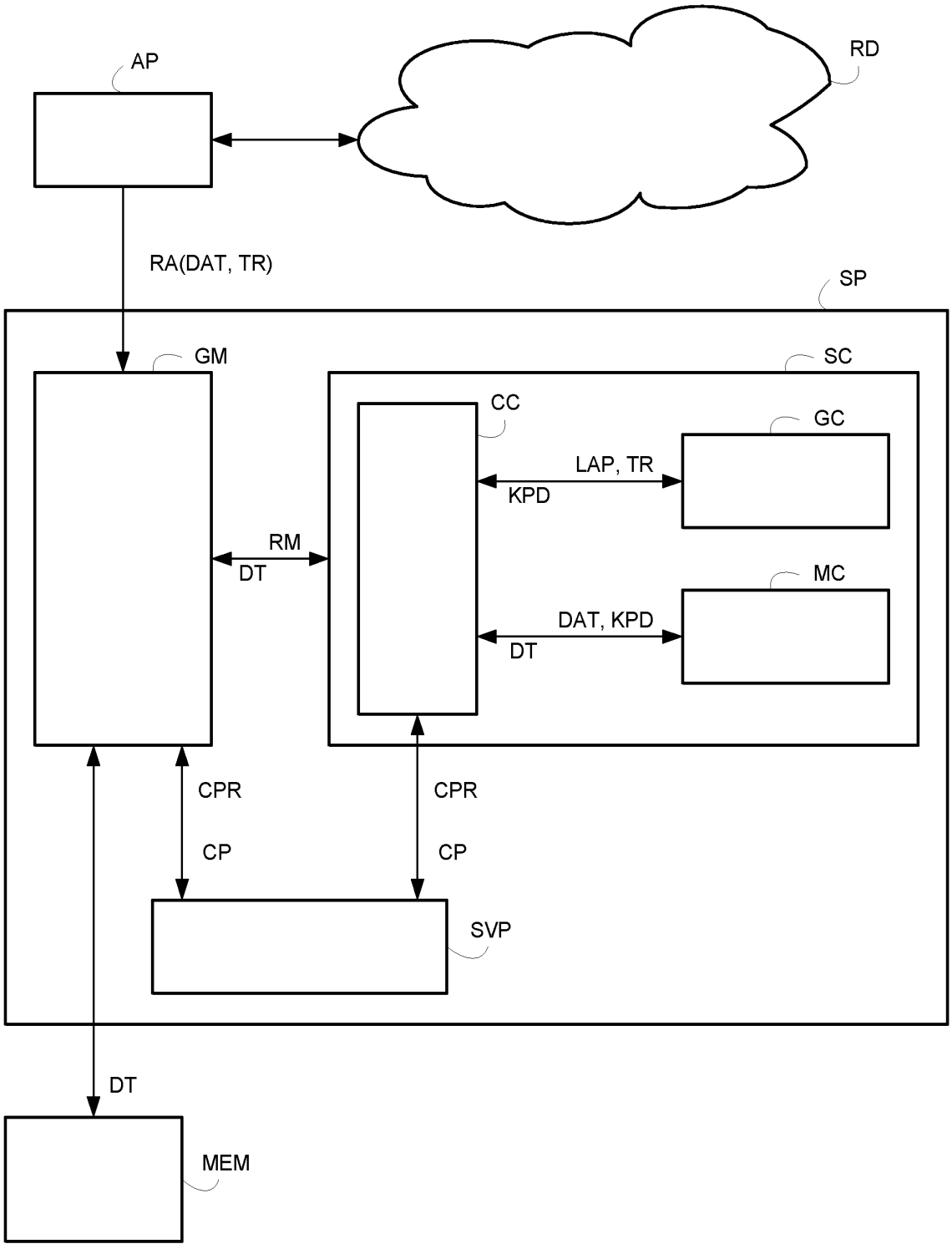
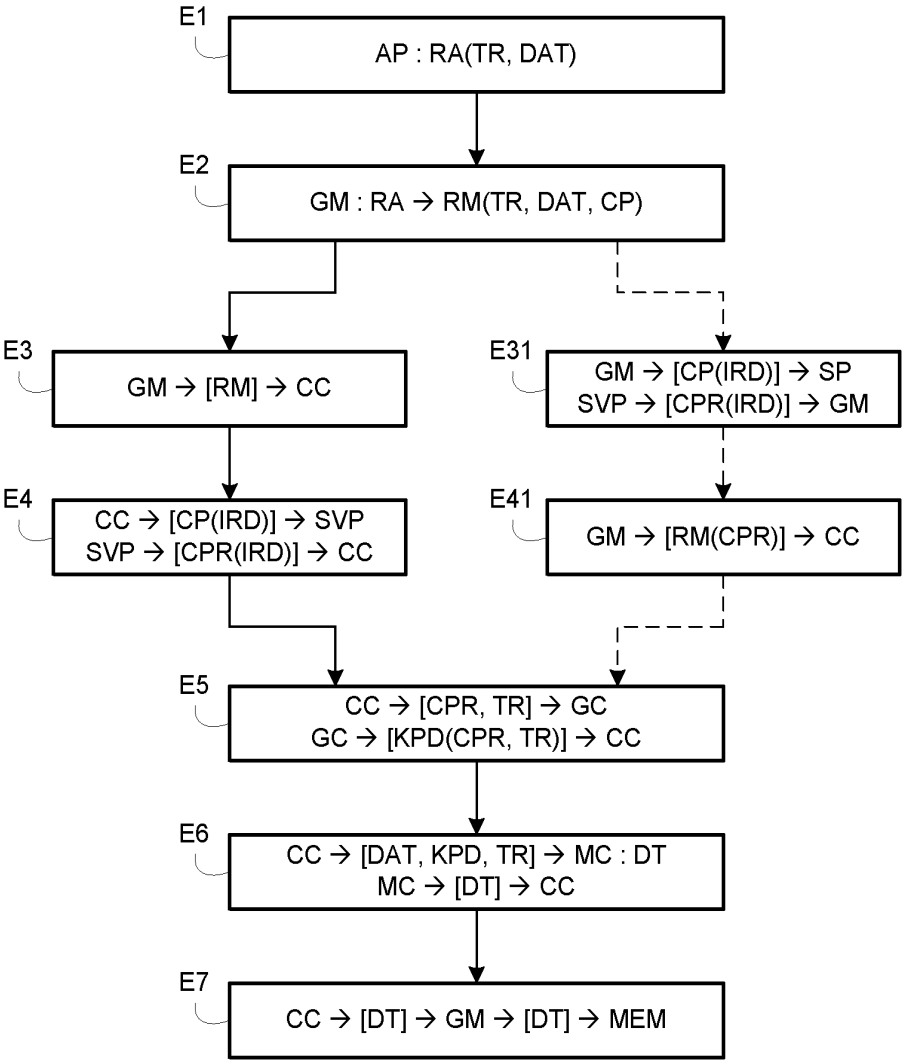

AP
RD
RA(DAT, TR)
SP
GM
SC
CC
GC
LAP, TR
KPD
RM
DT
MC
DAT, KPD
DT
CPR
CP
CPR
CP
SVP
DT
MEM

FIG. 2


E1
AP : RA(TR, DAT)
E2
GM : RA → RM(TR, DAT, CP)
E3
GM → [RM] → CC
E31
GM → [CP(IRD)] → SP
SVP → [CPR(IRD)] → GM
E4
CC → [CP(IRD)] → SVP
SVP → [CPR(IRD)] → CC
E41
GM → [RM(CPR)] → CC
E5
CC → [CPR, TR] → GC
GC → [KPD(CPR, TR)] → CC
E6
CC → [DAT, KPD, TR] → MC : DT
MC → [DT] → CC
E7
CC → [DT] → GM → [DT] → MEM

RÉPUBLIQUE FRANÇAISE

2910202

INSTITUT NATIONAL DE LA PROPRIÉTÉ INDUSTRIELLE

RAPPORT DE RECHERCHE PRÉLIMINAIRE

établi sur la base des dernières revendications déposées avant le commencement de la recherche

N° d'enregistrement national

FA 688823
FR 0655633

DOCUMENTS CONSIDÉRÉS COMME PERTINENTS		Revendication(s) concernée(s)	Classement attribué à l'invention par l'INPI
Catégorie	Citation du document avec indication, en cas de besoin, des parties pertinentes		
A	WO 99/49380 A1 (SYMANTEC CORP [US]) 30 septembre 1999 (1999-09-30) * abrégé * * page 5, ligne 1 - page 7, ligne 27 * -----	1-10	H04L9/00 G06F21/00 G06Q90/00
A	GB 2 425 439 A (MOTOROLA INC [US]) 25 octobre 2006 (2006-10-25) * abrégé * * page 6, ligne 1 - page 12, ligne 10 * -----	1-10	
			DOMAINES TECHNIQUES RECHERCHÉS (IPC)
			H04L

Date d'achèvement de la recherche	Examinateur
21 août 2007	Adkhis, Franck

CATÉGORIE DES DOCUMENTS CITÉS

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 bénéficiant d'une date antérieure à la date de dépôt et qui n'a été publié qu'à cette date de dépôt ou qu'à une date postérieure.
D : cité dans la demande
L : cité pour d'autres raisons

.....

& : membre de la même famille, document correspondant

EPO FORM 1503 12.99 (P04C14)

ANNEXE AU RAPPORT DE RECHERCHE PRÉLIMINAIRE RELATIF A LA DEMANDE DE BREVET FRANÇAIS NO. FR 0655633 FA 688823

La présente annexe indique les membres de la famille de brevets relatifs aux documents brevets cités dans le rapport de recherche préliminaire visé ci-dessus.
Les dits membres sont contenus au fichier informatique de l'Office européen des brevets à la date du 21-08-2007
Les renseignements fournis sont donnés à titre indicatif et n'engagent pas la responsabilité de l'Office européen des brevets, ni de l'Administration française

Document brevet cité au rapport de recherche	Date de publication	Membre(s) de la famille de brevet(s)	Date de publication
WO 9949380 A1	30-09-1999	AT 252248 T	15-11-2003
		CA 2325621 A1	30-09-1999
		DE 69912109 D1	20-11-2003
		EP 1066554 A1	10-01-2001
		US 2001044901 A1	22-11-2001
		US 2004093506 A1	13-05-2004
GB 2425439 A	25-10-2006	WO 2006113058 A1	26-10-2006

EPO FORM P0465