

(12) DEMANDE INTERNATIONALE PUBLIÉE EN VERTU DU TRAITÉ DE COOPÉRATION  
EN MATIÈRE DE BREVETS (PCT)

(19) Organisation Mondiale de la Propriété  
Intellectuelle  
Bureau international



(43) Date de la publication internationale  
20 février 2003 (20.02.2003)

PCT

(10) Numéro de publication internationale  
**WO 03/014888 A1**

(51) Classification internationale des brevets<sup>7</sup> : G06F 1/00,  
17/30

(21) Numéro de la demande internationale :

PCT/FR02/02824

(22) Date de dépôt international : 7 août 2002 (07.08.2002)

(25) Langue de dépôt : français

(26) Langue de publication : français

(30) Données relatives à la priorité :

01/10552

7 août 2001 (07.08.2001)

FR

(71) Déposant (pour tous les États désignés sauf US) :  
**CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE - CNRS** - [FR/FR]; 3, rue Michel-Ange,  
F-75794 Paris Cedex 16 (FR).

(72) Inventeurs; et

(75) Inventeurs/Déposants (pour US seulement) :  
**PUCHERAL, Philippe** [FR/FR]; 88, rue Sacco et  
Vanzetti, F-94460 Valenton (FR). **BOUGANIM, Luc**  
[FR/FR]; 29, rue Boussingault, F-75013 Paris (FR).

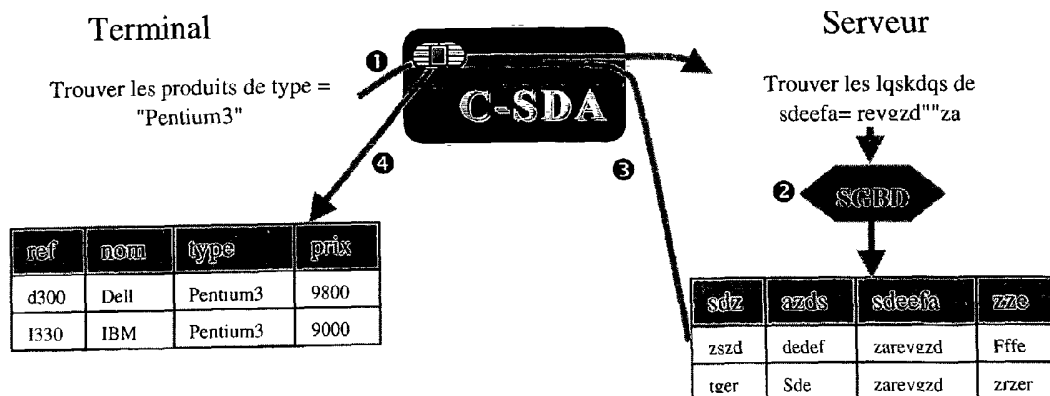
(74) Mandataires : **BREESÉ, Pierre** etc.; Breesé-Majerowicz,  
3, avenue de l'Opéra, F-75001 Paris (FR).

(81) États désignés (national) : CA, JP, US.

[Suite sur la page suivante]

(54) Title: METHOD FOR MAKING DATABASES SECURE

(54) Titre : PROCEDE DE SECURISATION DE BASES DE DONNEES



(57) **Abstract:** The invention concerns a secure management system of confidential databases comprising a server consisting of at least a computer equipped with means for recording databases and communication means, by at least a host computer equipment comprising means for communicating with the server and means for constructing requests and processing results of the requests, as well as means for making secure exchanges between the client equipment (1) and the server. The invention is characterised in that said securing means consists of a secure hardware support connected to the client equipment (1) and comprising a microprocessor for encrypting attributes of requests transmitted by the client equipment (1) and for decrypting replies transmitted by the server, a storage for recording intermediate results, a storage for recording the operating system and said server records encrypted data. The invention also concerns the method implemented by said system.

(57) **Abrégé :** La présente invention concerne un système de gestion sécurisée de bases de données confidentielles comprenant un serveur constitué par au moins un ordinateur équipé de moyens d'enregistrement de bases de données et de moyens de communication, par au moins un équipement informatique hôte comprenant des moyens de communication avec le serveur et des moyens pour la construction de requêtes et le traitement des résultats des requêtes, ainsi qu'un moyen de sécurisation des échanges entre l'équipement client (1) et le serveur, caractérisé en ce que ledit moyen de sécurisation est constitué par un support matériel sécurisé relié à l'équipement client (1) et comportant un microprocesseur pour le chiffrement des attributs des requêtes émises par l'équipement

[Suite sur la page suivante]

WO 03/014888 A1



(84) **États désignés (régional)** : brevet européen (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, SK, TR).

*En ce qui concerne les codes à deux lettres et autres abréviations, se référer aux "Notes explicatives relatives aux codes et abréviations" figurant au début de chaque numéro ordinaire de la Gazette du PCT.*

**Publiée :**

- avec rapport de recherche internationale
- avant l'expiration du délai prévu pour la modification des revendications, sera republiée si des modifications sont reçues

---

client (1) et le déchiffrement des réponses émises par le serveur, une mémoire pour l'enregistrement de résultats intermédiaires, une mémoire pour l'enregistrement du système d'exploitation et que ledit serveur enregistre des données chiffrées. Elle concerne également le procédé mis en oeuvre par un tel système.

## PROCEDE DE SECURISATION DE BASES DE DONNEES

La présente invention concerne le domaine des systèmes d'information sécurisés, et plus particulièrement des systèmes et procédés de sécurisation de bases de données.

La sécurisation des données est devenue un des enjeux majeurs des systèmes informatiques, compte tenu de la prolifération des données mises en ligne sur l'Internet (sites marchands, hébergement de données personnelles ou professionnelles, accès distants à des serveurs d'entreprise par des employés mobiles) et de l'interconnexion croissante de bases de données enrichies et consultées par de multiples acteurs (banques de données scientifiques, techniques, médicales). Le besoin de sécurité est lié au caractère confidentiel d'un sous-ensemble de ces données. La motivation de pirates s'attaquant aux données peut être multiple : tentative de fraude (vol de n° de cartes bancaires sur des sites marchands ...), atteinte à la vie privée des particuliers (enquêtes policières, politiques, fiscales, financières, médicales ...), guerres commerciales, diplomatiques ou militaires (système Echelon, violation de secrets industriels ou commerciaux, violation de la propriété intellectuelle ...).

Le problème ne se réduit pas à se protéger contre des pirates externes s'introduisant sur le système informatique attaqué. En effet, les attaques internes, c'est-à-dire conduites par des employés ayant légalement accès à tout ou parties des données d'une entreprise, sont les plus fréquentes et les plus coûteuses.

Trois classes de pirates susceptibles d'attaquer des bases de données peuvent être distinguées :

- Le pirate externe est un intrus qui s'infiltré sur un système informatique et récupère les

fichiers de données produits par le SGBD. Pour se prémunir contre ce type d'attaque, une solution naturelle est de chiffrer les données afin de rendre leur empreinte disque illisible. Le pirate externe va alors tenter de casser la  
5 clé de chiffrement. Les attaques sur clés sont facilitées dans un contexte bases de données du fait du volume important de données chiffrées avec la même clé (attaques statistiques). De plus, le chiffrement des données est statique (les clés ne changent pas d'une session à  
10 l'autre), augmentant par là-même la vulnérabilité de la base.

- Le pirate utilisateur est un usager courant du système informatique reconnu par le système d'exploitation et le SGBD. Il possède des droits sur une  
15 partie d'une base de données partagée et veut accéder à des données outrepassant ses droits. Ce pirate a potentiellement le même pouvoir qu'un pirate externe et peut en plus exploiter ses droits restreints. Si la base est chiffrée, il possède en outre un accès à certaines clés  
20 de déchiffrement.

- Le pirate administrateur ou DBA est un employé peu scrupuleux de l'entreprise dont la fonction est d'administrer les ressources informatiques de l'entreprise ou la base de données (DBA : Data Base Administrator). A ce  
25 titre, il possède tous les droits système, il dispose d'informations inaccessibles à tout autre (ex : journaux mémorisant les opérations effectuées sur la base) et peut espionner le comportement du SGBD pendant son exécution. Notons qu'un pirate externe ou utilisateur arrivant à  
30 s'octroyer les droits d'administration possède les mêmes pouvoirs que le pirate administrateur.

On connaît dans l'état de la technique plusieurs solutions pour sécuriser l'accès à des bases de données.

Une première famille de solutions est basée sur l'identification et authentification des utilisateurs.

Pour pouvoir utiliser un système de gestion de bases de données (SGBD), tout utilisateur doit être  
5 identifié et authentifié. L'authentification se fait classiquement par le biais d'un mot de passe stocké de façon chiffrée dans un catalogue du SGBD, ce dernier contrôlant la correspondance entre identité et mot de passe de l'utilisateur à chaque nouvelle connexion. La procédure  
10 d'authentification peut être renforcée par l'utilisation d'un service dédié [Oracle Advanced Security Administrator Guide, Release 8.1.7, September 2000]. Ce service dédié évite le cheminement en clair du mot de passe sur le réseau. On connaît une variante qui permet d'authentifier  
15 un utilisateur par différents biais tels que carte à puce ou TokenCard. En plus de l'authentification par un code personnel (PIN), l'élément matériel carte doit lui-même être reconnu par le serveur, augmentant ainsi le degré de sécurité. Ces mécanismes sont inopérants contre une attaque  
20 dirigée sur les fichiers contenant la base de données par un pirate externe (car le SGBD est alors court-circuité) ou contre une attaque menée par un pirate utilisateur ou DBA (qui n'auront aucun mal à s'authentifier).

Une autre famille de solutions est basée sur la  
25 gestion des droits des utilisateurs. Dans un SGBD, les droits portant sur les données (consultation, insertion, mise à jour, destruction de données) peuvent être affectés aux utilisateurs, soit sur les tables de base (on obtient ainsi une gestion de droits primitive proche de celle  
30 offerte par un système d'exploitation), soit sur des vues (tables virtuelles calculées par une requête SQL), soit encore sur des procédures stockées (le droit porte alors sur le déclenchement d'un traitement). Ces deux derniers cas permettent une gestion de droits sophistiquée et très  
35 sélective. Cependant, cette gestion de droit est inopérante

contre une attaque dirigée sur les fichiers contenant la base de données (le SGBD est à nouveau court-circuité) ou contre une attaque menée par un DBA (car il possède tous les droits, y compris le droit de changer les droits).

5           Une autre solution est basée sur le chiffrement des communications. Le chiffrement des communications peut être utilisé en complément d'autres mécanismes de sécurité afin d'assurer notamment la confidentialité et l'intégrité des messages émis et reçus via un réseau. Leur utilisation  
10 dépasse largement le cadre des bases de données. Le protocole le plus connu dans ce domaine est SSL (Secure Socket Layer). C-SET (Chip-Secured Transaction), utilisé pour les paiements électroniques sur Internet, exploite des cartes à puce pour identifier et authentifier les  
15 utilisateurs et assurer le chiffrement des messages. L'identification est de ce fait bien associée au porteur de carte et non au terminal sur lequel il est connecté, la confidentialité des clés de chiffrement est assurée et les algorithmes de chiffrement sont exécutés dans un  
20 environnement sécurisé, même si le terminal est lui-même non sécurisé. Bien évidemment, le chiffrement des communications ne permet pas de se prémunir contre une attaque sur les fichiers contenant la base de données.

D'autres solutions, complémentaire aux précédentes,  
25 consistent à chiffrer le contenu même de la base de données. Cette solution est la seule permettant de se prémunir contre des attaques dirigées sur les fichiers contenant la base de données. Elle n'est exploitée que depuis peu car sa mise en œuvre pose des problèmes  
30 complexes, comme mentionné ci-dessous.

La société Oracle (nom commercial) propose un package PL/SQL (ensemble de procédures stockées) permettant d'effectuer du chiffrement à clé secrète DES (Data Encryption Standard) [Oracle Technical White Paper.  
35 Database Security in Oracle 8i, November 1999]. La clé de

chiffrement doit être passée en paramètre à la procédure PL/SQL de chiffrement/déchiffrement. La gestion des clés (notamment leur confidentialité) est laissée à la charge de l'application utilisant ce package. Cette gestion est  
5 d'autant plus délicate que les mêmes clés doivent être partagées par les utilisateurs partageant les mêmes données. D'autre part, cette technique de chiffrement ne permet pas de résister aux attaques d'un DBA car celui-ci possède le moyen d'intercepter les clés ou de substituer le  
10 package de chiffrement par un autre. Ce dernier problème paraît sans solution car restreindre les droits du DBA revient à l'empêcher d'accomplir sa tâche d'administration.

Un article de recherche [J. He, M. Wang. Cryptography and Relational Database Management Systems. Int. Database and Engineering Symposium (IDEAS), 2001]  
15 propose une solution de chiffrement plus fortement intégrée au SGBD censée (i) faciliter l'utilisation du chiffrement par les applications et (ii) résister aux attaques du DBA. Le premier objectif est atteint du fait que le SGBD assure  
20 lui-même la confidentialité des clés de chiffrement (celles-ci sont stockées dans un catalogue sécurisé). Pour permettre le partage entre utilisateurs, les données sont chiffrées avec une clé secrète, celle-ci étant à son tour chiffrée avec la clé de chaque utilisateur. Le SGBD  
25 effectuant lui-même ce chiffrement, l'utilisateur n'a jamais accès aux clés. Une solution élégante est également proposée pour chiffrer la base de données avec un grand nombre de clés générées dynamiquement par le SGBD, rendant plus difficile les attaques statistiques conduites pour  
30 casser la (les) clé(s) de chiffrement. L'atteinte du second objectif (résister aux attaques du DBA) passe par la réalisation du catalogue sécurisé dans lequel sont stockées les clés de chiffrement. Celui-ci est implanté de sorte à constituer un SOE (Secure Operating Environment) dans  
35 lequel le pouvoir du DBA est considérablement réduit.

Transformer un catalogue en SOE ne rend pas l'intégralité du SGBD SOE. Dans cette méthode, comme dans celle d'Oracle, le chiffrement et le déchiffrement sont toujours assurés par le serveur. Il paraît alors illusoire d'assurer que le  
5 DBA ne pourra jamais accéder aux données en clair (par exemple pendant l'exécution d'une requête ou dans les journaux), à moins de restreindre ses droits à un point tel que toute tâche d'administration s'avérera impossible. Pour cette raison, la solution semble peu convaincante, d'autant  
10 plus qu'elle impose la réécriture d'une partie (dont les contours restent à préciser) du noyau du SGBD.

La société Protegrity propose une solution de chiffrement de données, appelée Secure.Data [P. Nilsson, U. Mattsson. Secure.Data Functional Overview, Doc. TWP-0011,  
15 Protegrity Inc., January 2000], adaptable à différents SGBD. Cette solution, dont le principe est proche de celui proposé par IBM, repose sur deux modules distincts. Secure.Manager est un serveur de sécurité indépendant du SGBD hôte qui prend en charge la gestion des utilisateurs,  
20 des droits d'accès et des clés de chiffrement. Il joue donc le rôle du catalogue sécurisé introduit précédemment. Secure.Server est un module intégré au SGBD hôte qui s'interface avec Secure.Manager et exécute les primitives de chiffrement/déchiffrement de données à la volée, pendant  
25 l'exécution des requêtes. Selon Protegrity, la sécurité de la solution repose sur une séparation stricte des rôles de DBA (Database Administrator) et de SA (Security Administrator gérant le Secure.Manager), ces rôles étant joués par des personnes physiques différentes. Au-delà du  
30 fait que ce principe de séparation physique puisse être difficile à garantir dans la pratique, chiffrement et déchiffrement sont toujours assurés par le SGBD. Ce dernier n'étant toujours pas un SOE, la protection vis-à-vis du DBA reste toute relative.

Les solutions de l'art antérieur ne sont pas totalement satisfaisantes, car :

- Seul le chiffrement des données permet de résister aux attaques conduites sur les fichiers de la base
- 5       - Pour résister aux attaques du DBA ou d'un pirate ayant usurpé les droits de ce dernier, le noyau de sécurité (incluant le chiffrement/déchiffrement) doit être géré dans un SOE.
- Le noyau de sécurité et le SOE l'englobant
- 10       doivent être suffisamment simples pour être auto-administrables.
- Un SGBD n'est pas auto-administrable.

Le but de la présente invention est de remédier à ces inconvénients en proposant un système et un procédé

15       garantissant un degré de protection maximal contre tout type d'attaque (interne ou externe) visant des données mises en ligne sur un réseau quelconque et gérées par un Système de Gestion de Bases de Données (SGBD) traditionnel.

Les buts de l'invention sont les suivants :

- 20       - assurer une confidentialité parfaite des données gérées par un SGBD,
- offrir aux utilisateurs un accès sécurisé à toutes données auxquelles ils ont légalement accès depuis n'importe quel terminal connecté à Internet,
- 25       - permettre à chaque utilisateur de partager ses données avec d'autres utilisateurs,
- être compatible avec les outils logiciels (SGBD) et matériels (cartes à puce) existants sur le marché.

A cet effet, l'invention concerne selon son

30       acception la plus générale un système de gestion sécurisée de bases de données confidentielles comprenant un serveur constitué par au moins un ordinateur équipé de moyens d'enregistrement de bases de données et de moyens de communication, par au moins équipement informatique hôte

35       comprenant des moyens de communication avec le serveur et

des moyens pour la construction de requêtes et le traitement des résultats des requêtes, ainsi qu'un moyen de sécurisation des échanges entre l'équipement client et le serveur (désigné également dans la suite sous le terme C-SDA, pour "Chip-Secured Data Access"), caractérisé en ce que ledit moyen de sécurisation est constitué par un support matériel sécurisé s'interfaçant entre l'équipement client et le serveur et comportant un microprocesseur pour le chiffrement des attributs des requêtes émises par l'équipement client et le déchiffrement des réponses émises par le serveur, une mémoire vive pour l'enregistrement de résultats intermédiaires, une mémoire non inscriptible pour l'enregistrement du système d'exploitation et une mémoire pour l'enregistrement des informations de personnalisation de l'utilisateur et que ledit serveur enregistre des données chiffrées.

Selon un mode de réalisation préféré de l'invention, le(s) client(s) et le serveur sont reliés par des moyens de communication connectés à un réseau de télécommunication (Internet, intranet, ...). Selon une variante, le client et le serveur sont installés sur la même machine physique et les moyens de communication correspondent à des moyens de communication internes à un ordinateur, voire à des moyens de communication logiciels.

Ce système répond aux objectifs visés en assurant une confidentialité parfaite des données gérées par un SGBD, vis-à-vis des trois classes de pirates citées précédemment, en offrant aux utilisateurs un accès (interrogation / modification) sécurisé à toutes données auxquelles ils ont légalement accès depuis n'importe quel terminal connecté à Internet, et en permettant à chaque utilisateur de partager ses données avec d'autres utilisateurs.

L'invention prend en compte les contraintes suivantes :

- la base de données peut être gérée par tout SGBD commercial traditionnel,

- le support matériel sécurisé utilisé est de technologie courante (par exemple, une carte à puce conventionnelle),

- les algorithmes de chiffrement utilisés sont connus.

Les applications de l'invention sont diverses, et prennent en compte le développement de l'informatique mobile qui traduit une tendance croissante des utilisateurs à accéder à leurs données à tout moment, de n'importe où et surtout à partir de n'importe quel terminal (PC, assistant personnel, téléphone cellulaire, borne banalisée ...). Ceci suppose que les données de l'utilisateur sont hébergées par un prestataire de service garantissant un stockage et un accès permanent, tolérant aux pannes et sécurisé à ces données. Les données hébergées dans ce contexte sont généralement personnelles, par exemple :

- le dossier médical, qui contient des données hautement confidentielles et qu'un patient veut pouvoir accéder chez le médecin, à la pharmacie, chez lui, en voyage, etc.

- le courrier électronique qui, de plus en plus, devient un outil de travail et contient bien sûr des informations confidentielles.

- l'environnement virtuel ou virtual home, qui répertorie des informations privées de diverses natures telles que : données bancaires, agenda, carnet d'adresses, mots de passes, bookmarks, cookies ..., et qui doivent pouvoir être accédées de n'importe où.

Ces applications personnelles se caractérisent par les besoins suivants :

- données devant être accessibles de n'importe où et à tout moment,

- données hautement confidentielles car personnelles,

- données partagées par plusieurs acteurs humains (ex : dossier médical) ou logiciels (virtual home),

5 - petit volume de données,

- interrogations complexes (inégalités sur dates ou valeurs, recherche textuelle sur libellé, calculs d'agrégats et de groupement ...).

Elle concerne également les situations où un  
10 utilisateur fait appel aux services d'un hébergeur de données pour stocker des données comptables, des données liées à des procédés de fabrication ou des informations confidentielles sur ses clients et fournisseurs. En effet, une PME dispose rarement de l'ensemble des moyens  
15 techniques et humains nécessaires à la sécurisation de ces données contre des attaques directes (piratage, vol) ou indirectes (virus informatique). De plus, la motivation initiale d'une PME pour recourir aux services d'un hébergeur peut être en priorité d'assurer la durabilité de  
20 ses données (tolérance aux pannes), le besoin de confidentialité étant alors induit par le fait que les données de l'utilisateur sont externalisées. La confidentialité des données est liée à la protection de l'entreprise vis-à-vis de la concurrence (ex : secret de  
25 fabrication, stratégie commerciale ...) et à sa crédibilité vis-à-vis de ses clients (ex : n° de cartes bancaires). Cette classe d'application introduit le même type de besoin que la précédente, avec la contrainte supplémentaire que le volume de données est potentiellement beaucoup plus grand.

30 L'invention concerne également un procédé mettant en œuvre une application de sécurité qui s'interface entre un client et un serveur de données et il respecte les règles suivantes :

- Chiffrement des données à protéger  
35 stockées sur le serveur,

- Gestion des requêtes portant sur les données chiffrées et le déchiffrement des résultats,

- Gestion des privilèges (droits d'accès) de chaque utilisateur,

5            Cette application de sécurité est auto-administrable, et s'exécute sur une plateforme matérielle et logicielle sécurisée (SOE). Tout manquement à l'une de ces règles introduit un trou de sécurité tels que ceux identifiés dans la section précédente. Ces cinq règles  
10            déterminent les éléments techniques de la solution proposée et en font probablement l'unique solution au problème posé.

          Selon une variante préférée, le moyen de sécurisation est constitué par une carte à microprocesseur.

          Les cartes à puce de dernière génération  
15            comprennent sur une puce monolithique, un processeur RISC 32 bits à environ 30 MIPS, des modules mémoire (d'environ 96 Ko de ROM, 4 Ko de RAM et 128 Ko d'EE-PROM), un composant d'I/O gérant les communications avec le terminal et enfin des modules de sécurité protégeant la carte contre  
20            des attaques physiques. L'EEPROM permet de stocker des informations persistantes avec un temps d'accès très rapide en lecture, comparable à celui de la RAM (60 à 100 ns/mot) mais très lent en écriture (1 à 10 ms/mot). L'intérêt principal de la carte à puce est son très haut niveau de  
25            sécurité. Cette sécurité est obtenue grâce aux modules de sécurité matériels présents dans la puce et grâce également à la très faible taille des puces utilisées (< 25mm<sub>2</sub>) rendant les attaques physiques très complexes et coûteuses. La carte à puce est sans aucun doute le plus petit et le  
30            plus sûr des "ordinateurs". Cependant, la faible taille des puces a une incidence immédiate sur la capacité de stockage des composants mémoire. Une carte à microprocesseur se traduit par les caractéristiques suivantes :

- sécurité maximale des données et du code hébergés  
35            dans la puce,

- processeur puissant (notamment pour les fonctions de chiffrement/déchiffrement),

- très faible capacité de la RAM,

- capacité du stockage persistant en constante extension mais temps d'écriture très lent.

Selon une variante préférée de l'invention, le moyen de sécurisation est constitué par une carte à microprocesseur.

Selon une autre variante, le moyen de sécurisation est constitué par une clé raccordable sur un port d'un équipement client.

De préférence, le moyen de sécurisation comporte en outre au moins un compteur pour l'exécution de requêtes statistiques.

De préférence, le moyen de sécurisation comporte en outre un registre pour l'enregistrement temporaire des droits téléchargés lors de l'ouverture d'une session avec le serveur.

Selon une mode de mise en œuvre avantageux, le moyen de sécurisation comporte en outre une mémoire pour l'enregistrement d'une partie de la base de données.

Cette solution permet d'éviter l'enregistrement sur le serveur de certaines données sensibles. La base de données est alors divisée en deux parties, l'une enregistrée sur le serveur, l'autre enregistrée sur le support de sécurisation.

L'invention concerne également un moyen de sécurisation des échanges entre un équipement client et un serveur constitué par un support matériel sécurisé relié à l'équipement client et comportant un microprocesseur pour le chiffrement des attributs des requêtes émises par l'équipement client et le déchiffrement des réponses émises par le serveur, une mémoire vive pour l'enregistrement de résultats intermédiaires, une mémoire non inscriptible pour l'enregistrement du système d'exploitation et une mémoire

pour l'enregistrement des informations de personnalisation de l'utilisateur.

Selon une variante, il comporte en outre au moins un compteur pour l'exécution de requêtes statistiques.

5 L'invention concerne également un procédé de gestion sécurisée d'une base de données comportant une étape de construction d'une requête comprenant au moins un attribut caractérisé en ce qu'il comporte une étape de chiffrement des attributs par un calculateur solidaire d'un  
10 dispositif individuel de sécurité relié à un équipement client, le procédé consistant à interroger ensuite une base de données contenant des données chiffrées avec les mêmes moyens de chiffrement que ceux utilisés lors de l'étape précédente, à retourner une réponse contenant les données  
15 correspondant aux attributs de la requêtes et à procéder au déchiffrement desdites données par ledit calculateur d'un dispositif individuel de sécurité avant de les transmettre à l'équipement hôte.

Selon un mode de mise en œuvre particulier, le  
20 chiffrement des attributs de la requête et le déchiffrement de la réponse est réalisé selon un algorithme DES.

Selon une variante simplifiée, la traduction de requêtes est limitée aux prédicats d'égalité pour permettre l'exécution desdites requêtes exécutables directement par  
25 le serveur sur des données chiffrées.

Selon une variante préférée, la traduction de requêtes comportant des prédicats d'inégalité ou fonctions de calcul comporte une étape de décomposition de la requête  $Q$  en un plan  $Q_t(Q_c(Q_s))$ , d'interrogation de la base de  
30 données chiffrées (3) par des requêtes d'inégalités à partir d'attributs chiffrés, une étape d'enregistrement des résultats des requêtes dans une mémoire temporaire du moyen de sécurisation individuel après déchiffrement des données, et de recomposition du résultat ;  $Q_s$  symbolise la portion  
35 de requête évaluable par le serveur directement sur les

données chiffrées,  $Q_c$  symbolise le complément de cette requête évaluable par le moyen de sécurisation après déchiffrement et  $Q_t$  symbolise la portion de requête liée à la présentation et pouvant être déportée sur le terminal  
5 sans réduire le degré de sécurité.

Selon une autre variante, la traduction de requêtes comportant des prédicats d'inégalité ou fonctions de calcul comporte une étape de décomposition d'une requête en un plan  $Q = Q_t (Q_c(Q_s (*[Q_c^p (Q_s^p)])))$ , où  $Q_c^p$  et  $Q_s^p$  sont des  
10 requêtes de préparation, d'interrogation de la base de données chiffrées (3) par des requêtes d'inégalités à partir d'attributs chiffrés, une étape d'enregistrement des résultats des requêtes dans une mémoire temporaire du moyen de sécurisation individuel après déchiffrement des données,  
15 et de recomposition du résultat.

Selon un mode de mise en œuvre particulier, une partie de la base des données est enregistrée dans une mémoire du moyen individuel de sécurisation.

L'invention sera mieux comprise à la lecture de la description d'un exemple non limitatif de réalisation, se  
20 référant aux dessins annexés où :

- la figure 1 représente une vue schématique d'un système selon l'invention ;
- la figure 2 représente une vue schématique du  
25 chiffrement d'une table ;
- la figure 3 représente un diagramme des flux d'informations entre l'équipement client et le serveur ;
- la figure 4 représente un diagramme des flux d'informations entre l'équipement client et le serveur,  
30 dans une version évoluée de l'invention ;
- Les figures 5 et 6 représentent les schémas d'exécution de requête 'complexe'
- La figure 7 représente une vue schématique des traitements ;

- la figure 8 représente l'architecture de la carte de sécurisation.

La description qui suit concerne un exemple non limitatif de mise en œuvre de l'invention. Elle associe des technologies logicielles (gestion de requêtes bases de données et algorithmes de chiffrement) et matérielles (code embarqué sur une carte à puce ou un support matériel assurant le même niveau de sécurité).

La figure 1 représente une vue schématique d'un système selon l'invention.

Il comporte un équipement client (1) qui peut être constitué par un micro-ordinateur, ou encore un équipement portable tel qu'un PDA, un téléphone cellulaire ou tout autre équipement permettant de consulter une base de données. Il est équipé avec un calculateur et des applications pour la gestion d'une base de données (insertion de données, consultations, modifications des données) et pour l'établissement d'une session avec un serveur (2) auquel il est relié par un réseau de télécommunication (Internet, extranet, intranet, réseau GSM,...).

Ce serveur (2) comporte une base de données (3) de type connu. La base de données contient exclusivement des données chiffrées par l'utilisateur, comme cela sera exposé dans la suite de la description. Le serveur (2) gère les données sans pour autant y avoir accès. Le serveur de données est de technologie relationnelle (Oracle, DB2, SQL Server, Sybase, Informix ...). La prise en compte d'autres types de serveurs de données et d'autres langages de requêtes peut amener quelques modifications dans les algorithmes présentés, sans remettre en cause les principes décrits dans la description qui suit.

La communication entre l'équipement client (1) et le serveur (2) transite par une carte à microprocesseur (4) exploitée par un périphérique de l'équipement client (1),

par exemple un terminal raccordé sur un port USB de l'équipement client.

Le chiffrement de la base de données est assuré par la carte (4) de la façon suivante.

5           La carte (4) intercepte toutes les requêtes SQL de création/modification de schéma ainsi que toutes les requêtes SQL d'insertion/modification/destruction de tuples dans les tables. Le schéma de la base de données ainsi que les données sont chiffrées par la carte, par exemple avec  
10 un algorithme à clé secrète de type DES. Dans la suite, l'algorithme de chiffrement utilisé est appelé Chiffre. Ces requêtes, dont les paramètres sont désormais chiffrés, sont ensuite envoyées au serveur de données (2) comme des requêtes classiques. Cette technologie ne demande donc  
15 aucune modification du SGBD utilisé.

Les données enregistrées dans le serveur (2) sont enregistrées sous forme chiffrée. Comme illustré sur la figure 2, la table Produit (ref, nom, type, prix) est stockée comme table  $P(r, n, t, x)$  où  $P =$   
20 Chiffre("produit"),  $r = \text{Chiffre(ref)}$ ,  $n = \text{Chiffre(nom)}$ , etc... Dans cet exemple, l'intégralité des données est chiffrée, sachant qu'un chiffrement partiel des données peut également être envisagé.

Le chiffrement est appliqué aux données ainsi  
25 qu'aux informations globales telles que le nom de la table et les noms des rubriques.

La figure 3 représente les étapes d'une procédure de consultation de la base de données chiffrées.

On procède à une première étape consistant à  
30 préparer une requête en clair sur l'équipement client (étape 1).

Cette requête est transmise par l'équipement client à la carte (4) où les attributs de la requête sont chiffrés avant d'être transmis au serveur (2). Cette requête est  
35 alors traitée de façon habituelle, la recherche

s'effectuant sur critères d'égalité entre le contenu chiffré de la requête et le contenu chiffré avec le même chiffre, enregistré dans la base de données (3). (étape 2). Le serveur ne nécessite donc aucune adaptation, la seule  
5 différence avec une application traditionnelle venant du fait que la recherche s'effectue sur des attributs chiffrés.

La réponse à cette requête transite à nouveau par la carte (4), qui procède à une conversion du contenu pour  
10 restituer les données correspondant à la requête en clair (étape 3). Ces données en clair sont ensuite retransmises à l'équipement client, pour la visualisation ou l'enregistrement dans une table locale (étape 4).

Cette technique permet d'exécuter toute requête sur  
15 des données chiffrées ne faisant intervenir que des prédicats d'égalité (attribut = valeur ou attribut = attribut) sans fonction de calcul (agrégat, ...). En effet, les propriétés des algorithmes de chiffrement ne permettent pas, de façon générale, d'évaluer des prédicats d'inégalité  
20 ni des fonctions de calcul sur des données chiffrées. Cette contrainte est forte car :

- Elle limite les capacités d'interrogation des utilisateurs,
- Elle limite de la même façon la puissance  
25 d'expression des droits d'accès.

La résolution d'une requête SQL simple du type "select\* from produit where type = 'pentium3'", permettant de retrouver tous les produits de type pentium3, sera effectuée comme suit :

- 30 1. Terminal (1) : saisie de la requête en clair
2. Carte (4) : chiffrement et envoi de la requête chiffrée au serveur : select \* from Chiffre(Produit) where Chiffre(type) = Chiffre("pentium3")

3.           Serveur (2) :       Résolution de la requête  
sur les données chiffrées (transparent pour le SGBD)
4.           Carte (4):       Déchiffrement du résultat
5.           Terminal :       Affichage du résultat en  
5 clair.

Cette technique d'évaluation de requêtes est possible lorsque tous les utilisateurs partageant les mêmes données possèdent la même clé de chiffrement. Ceci amène deux remarques importantes :

- 10           Le chiffrement ne doit en aucun cas interférer avec les droits d'accès. Vis-à-vis des droits, il est envisageable de chiffrer l'intégralité de la base de données avec une seule clé. Chaque utilisateur n'a accès qu'aux données sur lesquelles il possède un droit, la clé  
15 étant connue uniquement par la carte (4) et non par chaque utilisateur. Par exemple, dans le cas où l'application de sécurité s'exécute dans une carte à puce, la clé permettant de déchiffrer l'intégralité de la base de données est présente dans la carte à puce de chaque utilisateur sans  
20 que celui-ci n'y ait accès.

- Plusieurs clés de chiffrement peuvent être utilisées pour augmenter la difficulté de cassage des clés et en réduire l'impact. La gestion des droits faisant partie intégrante du mécanisme de sécurité, celle-ci ne  
25 peut en aucun cas être déléguée au serveur de données. Autrement, le DBA n'aura aucune difficulté à s'octroyer tous les droits sur l'ensemble des données. La gestion des droits doit donc être assurée par l'application de sécurité de la carte (4). Cette gestion pose un problème particulier  
30 lorsque cette application s'exécute dans une carte à puce propre à chaque utilisateur. En effet, les droits expriment un partage de données entre plusieurs utilisateurs et ces droits sont dynamiques (de nouveaux droits peuvent être définis et certains existants peuvent être supprimés). Si  
35 la carte à puce est responsable de la vérification des

droits, elle ne peut par contre pas stocker elle-même la définition de ces droits du fait du partage entre les utilisateurs. Par conséquent, la définition des droits est stockée sur le serveur et est chargée dynamiquement (ou  
5 simplement rafraîchie) sur la carte lors de la connexion de l'utilisateur au serveur. Afin de prévenir toute altération, pouvant introduire un trou de sécurité, la définition des droits doit être stockée chiffrée sur le serveur.

10 La puissance du mécanisme de droits des SGBD provient de la capacité d'affecter un droit à un utilisateur sur des vues, c'est-à-dire des tables virtuelles calculées par une requête SQL. Il devient alors possible d'exprimer des droits d'une grande finesse, par  
15 exemple : donner le droit à un utilisateur de consulter une vue définie comme la moyenne des salaires des employés de plus de 40 ans travaillant dans le service informatique sans lui donner aucun droit sur les données permettant de réaliser ce calcul. Soit une vue V définie par la requête  
20  $Q_{vue}$  sur laquelle l'utilisateur possède le droit D. Lorsque l'utilisateur exprime une requête  $Q(V)$ , le gestionnaire de droits vérifie tout d'abord que cette requête est compatible avec le droit D. Si oui, la requête  $Q(V)$  est transformée par le gestionnaire de vue en une requête  
25  $Q(Q_{vue}) = Q'(\{Ti\})$ , où  $\{Ti\}$  dénote la liste des tables de base sur laquelle la vue V est définie.  $Q'$  est ensuite évaluée comme une requête classique selon la méthode introduite précédemment. La conséquence en est que C-SDA doit intégrer la gestion des vues au même titre que la  
30 gestion des droits (la définition des vues est stockée chiffrée sur le serveur et chargée dynamiquement sur la carte).

La description qui suit concerne une version améliorée de l'invention, permettant de traiter tout type  
35 de requête SQL. Cette solution étendue implique que

l'application de sécurité participe de façon active au traitement de la requête afin d'évaluer les prédicats non évaluables par le serveur. Cette interaction de l'application de sécurité de la carte (4) dans le traitement a deux conséquences. La première porte sur la performance de l'exécution globale et la seconde sur l'apparition de nouvelles perspectives pour renforcer la protection des données de la base.

Dans le cas d'interrogations complexes (prédicats d'inégalité, fonctions de calcul, agrégats ...), la requête  $Q$  doit être décomposée en une partie  $Q_S$  évaluable par le serveur sur les données chiffrées et son complément  $Q_C$  à évaluer par l'application de sécurité sur le résultat partiel, après déchiffrement. Il est fondamental que  $Q_C$  soit exécutée par l'application de sécurité et non par le terminal car seul C-SDA s'exécute dans un environnement sécurisé. Comme il a été montré dans la section 5.4, les droits des utilisateurs portent généralement sur des vues dont le contenu virtuel est calculé par une requête. Donc, seul un traitement de  $Q_C$  par l'application de sécurité, sans externalisation de données confidentielles, peut garantir le niveau de sécurité escompté par une gestion de droits. Par exemple, un utilisateur peut avoir le droit de consulter le résultat d'un calcul d'agrégat sans pour autant posséder le droit de consulter les données élémentaires à partir desquelles cet agrégat est calculé.

Afin d'évaluer  $Q_C$ , C-SDA doit contenir des fonctions de filtrage (pour traiter les prédicats d'inégalité) et des fonctions de calcul. C'est l'objet du deuxième exemple.

La requête "select Avg(prix) from produit where type = 'pentium3'" sera effectuée comme suit :

1. Terminal : Saisie de la requête en clair

2. Carte : Chiffrement et envoi d'une sous-requête chiffrée  $Q_S$  : `select Chiffre(prix) from Chiffre(Produit) where Chiffre(type) = Chiffre("pentium3")`

3. Serveur : Résolution de la requête sur les  
5 données chiffrées : Résultat (`Chiffre(prix1)`,  
`Chiffre(prix2)`, `Chiffre(prix3)`, ...)

4. Carte : Déchiffrement du résultat de  $Q_S$  et évaluation de  $Q_C$  (calcul de la moyenne).

5. Terminal : Affichage du résultat en clair.

10 Pour que ce principe soit applicable sur une carte à puce, il faut que l'évaluation de  $Q_C$  par la carte (4) puisse se faire en respectant les contraintes inhérentes à la carte. Ces contraintes imposent d'évaluer  $Q_C$  sans jamais matérialiser de résultats temporaires car la très faible  
15 capacité de la RAM ne permet pas une telle matérialisation en mémoire volatile et le coût d'écriture de l'EEPROM rend également cette matérialisation impossible en mémoire stable. La solution proposée consiste à exécuter l'intégralité de  $Q_C$  en mode pipeline (c.à.d, tuple à  
20 tuple). Bien que ce mode d'évaluation ne soit pas traditionnel, surtout pour les requêtes faisant intervenir des groupements et calculs d'agrégats, nous pouvons montrer que toute requête  $Q_C$  est évaluable en pipeline en suivant le principe d'exécution suivant.

25 La forme générale d'une requête SQL est :

- **Select** liste d'attributs, fonctions
- **From** liste de relations
- **Where** qualification
- **Group by** liste d'attributs
- 30 - **Having** qualification sur fonction agrégat
- **Order by** critère de tri

La sémantique opérationnelle de SQL précise que le résultat d'une requête SQL doit être équivalent au traitement suivant :

1. Calcul du produit cartésien des relations impliquées dans la clause **From**
2. Restriction aux seuls tuples produits en (1) satisfaisant la qualification de la clause **Where**
- 5 3. **Groupe**ment des tuples issus de (2) ayant même valeur pour la liste d'attributs spécifiée
4. Calcul des fonctions agrégats apparaissant dans les clauses **Select**, **Having** et **Order by**
5. Restriction aux seuls groupes produits en
- 10 (4) satisfaisant la qualification de la clause **Having**
6. Projection des tuples issus de (5) sur la liste d'attributs et fonctions de la clause **Select**
7. Tri des tuples issus de (6) en fonction du critère de tri spécifié par la clause **Order by**
- 15 Le serveur est capable d'exécuter sur des données chiffrées les étapes suivantes, soit  $Q_S$  :
  1. Calcul du produit cartésien des relations impliquées dans la clause **From**
  - 2s. Restriction aux seuls tuples produits en (1)
  - 20 satisfaisant les prédicats d'égalité présents dans la qualification de la clause **Where**
  3. **Groupe**ment des tuples issus de (2s) ayant même valeur pour la liste d'attributs spécifiée
  - 6s. Projection des tuples issus de (3) sur la
  - 25 liste d'attributs résultat de l'union des listes d'attributs des clauses **Select**, **Group by**, **Order by** ainsi que de la liste d'attributs nécessaires à l'évaluation des prédicats non traités en 2s de la clause **Where** et des prédicats de la clause **Having**.
- 30 L'application de sécurisation est pour sa part contrainte d'exécuter sur la carte à puce en pipeline les étapes suivantes, soit  $Q_C$  :
  - Pour chaque tuple  $t$  résultat de  $Q_S$ , faire :
    - 2c. Vérifier les prédicats de la clause **Where**
    - 35 non traités en 2s

4. Agréger dans une variable tampon la valeur de(s) attribut(s) de  $t$  impliqué(s) dans l'évaluation d'une fonction agrégat (apparaissant dans les clauses **Select**, **Having** et **Order by**) et ce, tant que  $t$  appartient au même groupe que le tuple précédent (par exemple, le calcul de la moyenne d'un attribut se réalise en sommant la valeur de l'attribut de chaque tuple du groupe. Lorsque le dernier tuple du groupe est détecté, cette somme est divisée par le nombre de tuples du groupe).

10 5. Lorsque la fin de groupe est détectée et que les fonctions agrégats sont calculées, évaluer la qualification de la clause **Having**.

15 6c. S'il est sélectionné, projeter le tuple issu de (5) sur la liste d'attributs et de fonctions spécifiée dans la clause **Select**

A noter que la clause **Order by** peut être systématiquement déportée sur le terminal et constitue donc une troisième requête notée  $Q_t$ . En effet, la clause **Order by** n'influe pas sur la valeur des tuples résultat mais simplement sur leur ordonnancement. Cette clause est donc sans effet vis-à-vis de la gestion des droits et peut de ce fait être exécutée à l'extérieur de C-SDA.

En résumé, et comme illustré par l'exemple de la figure 5, l'évaluation d'une requête SQL quelconque  $Q$  se fait de la façon suivante :  $Q = Q_t(Q_c(Q_s))$ . Ce principe d'évaluation respecte bien les contraintes de la carte puisque l'espace mémoire consommé se réduit à la mémorisation d'un tuple courant et de quelques variables tampons destinées au calcul des fonctions agrégats.

30 La figure 5 représente le schéma d'exécution de requête 'complexe'. Ce mode d'évaluation présenté ci-dessus peut être amélioré pour deux raisons :

- L'évaluation de  $Q_s$ , telle que présentée précédemment, laisse à penser que les produits cartésiens sont exécutés en premier (étape 1). En fait, l'évaluation

de tout ou parties de 2s et de 6s peut être réalisée avant l'étape 1 en suivant les techniques d'optimisation de requêtes traditionnelles (remplacement des produits cartésiens par des jointures, application des restrictions et des projections au plus tôt).  $Q_S$  étant transmise au serveur comme une requête SQL classique, elle suivra le processus d'optimisation traditionnel.

- Les prédicats d'inégalité (restrictions, inéqui-jointures) ne pouvant être évalués directement par le serveur, le volume de données transféré vers la carte peut être très supérieur au volume du résultat final. Dans l'exemple de la figure 5, toutes les commandes des clients sont transférées à C-SDA, sans distinction de date. La solution à ce problème est détaillée ci-dessous.

Il est nécessaire d'évaluer les prédicats d'inégalité au plus tôt afin de profiter de leur sélectivité pour simplifier l'ensemble de la chaîne de traitements réalisés sur l'équipement client, la carte et le serveur. Cela nécessite un pré-traitement coopératif entre la carte et le serveur. Soit une requête  $Q$  contenant un prédicat de la forme  $\sigma_{a_i \theta \text{ valeur}}(T)$ , où  $\sigma$  dénote l'opérateur de restriction,  $\theta \in \{<, >, \leq, \geq\}$  et  $a_i$  est un attribut quelconque de la table  $T$ . Le principe d'optimisation consiste pour l'application de sécurisation à envoyer au serveur une requête de pré-traitement  $Q_S^P = \Pi_{\text{clé}, a_i}(T)$ ,  $\Pi$  dénotant l'opérateur de projection. C-SDA exécute alors  $Q_C^P = \Pi_{\text{clé}}(\sigma_{a_i \theta \text{ valeur}}(\Pi_{\text{clé}, \text{Déchiffre}(a_i)}(Q_S^P)))$  et renvoie le résultat  $R$  au serveur. C-SDA transforme ensuite la requête  $Q$  en remplaçant le prédicat initial  $\sigma_{a_i \theta \text{ valeur}}(T)$  par le prédicat  $\propto(T, R)$ , où  $\propto$  dénote l'opérateur de semi-jointure sur clé. Sur l'exemple de la figure 5, ce principe donne l'enchaînement suivant exécuté en pipe-line sur la carte :

- l'application de sécurité de la carte rapatrie les tuples de la table Commandes après avoir fait une projection sur l'attribut clé et l'attribut Date;

5       - l'application de sécurité de la carte déchiffre l'attribut Date de ces tuples, sélectionne uniquement les tuples dont la date est supérieure à 1996 et renvoie leur clé au serveur sous la forme d'une table temporaire R;

10       - l'application de sécurité de la carte transforme enfin la requête initiale Q en remplaçant le prédicat (Q.Date > 1996) par (Q.clé = R.clé) et envoie cette requête au serveur pour exécution. Le volume de données transféré est très fortement réduit. En effet, le surcoût par rapport à une exécution traditionnelle se limite à  $\text{taille}(\Pi_{\text{clé}, \text{ai}}(T)) + \text{taille}(\Pi_{\text{clé}}(\sigma_{\text{ai} \neq \text{valeur}}(T)))$ . Ce principe s'applique de  
15       façon similaire pour les inéqui-jointures.

Comme illustré par la figure 6, le principe général d'évaluation d'une requête contenant des prédicats d'inégalité devient :  $Q = Q_t(Q_c(Q_s(*[Q_c^p(Q_s^p)])))$ , la composition  $Q_c^p(Q_s^p)$  se répétant pour chaque prédicat  
20       d'inégalité.

L'utilisation de plusieurs clés de chiffrement pour une même base de données est une idée simple et efficace pour réduire le volume de données dévoilées en cas de cassage d'une clé. Dans notre contexte, l'utilisation de  
25       plusieurs clés doit respecter la contrainte suivante :  $\forall a, \forall b, a = b \Leftrightarrow \text{Chiffre}(a) = \text{Chiffre}(b)$ . Cette contrainte est nécessaire pour permettre au serveur d'exécuter tous les prédicats d'égalité présents dans les requêtes de type  $Q_s$ . Toutefois, cette contrainte n'a pas lieu d'être si a et b  
30       ne sont jamais comparés, soit qu'ils ne sont pas de types comparables (ex : un entier et une chaîne de caractère), soit que leur comparaison est dépourvue de toute sémantique (ex : l'âge d'un client et le numéro de client), soit

encore que ces données appartiennent à des utilisateurs ou groupes d'utilisateurs différents ne désirant pas les partager. Dans tous ces cas, il est possible d'utiliser des clés de chiffrement différentes selon une technique que  
5 nous appellerons chiffrement à fragmentation verticale.

Cette technique peut se combiner avec une technique de chiffrement à fragmentation horizontale qui consiste à utiliser plusieurs clés de chiffrement pour différentes valeurs d'un même attribut. Dans cette technique, la clé de  
10 chiffrement appliquée à chaque valeur est déterminée par l'application d'une fonction de hachage à cette valeur. Ainsi, pour chaque valeur  $v$ , le résultat de la fonction de hachage  $h(v)$  donne l'indice de la clé de chiffrement à utiliser. Le couple  $(h(v), \text{Chiffre}_{h(v)}(v))$  est stocké dans  
15 la base en lieu et place de chaque valeur. Il est facile de montrer que la contrainte précédente est respectée. Dans ces deux techniques, l'ensemble des clés de chiffrement utilisées est géré par la carte de sécurisation.

Les données renvoyées par le serveur à la carte de  
20 sécurisation étant déjà chiffrées, l'opportunité de chiffrer les communications n'est pas indispensable. Ce chiffrement s'avère cependant nécessaire, notamment pour résister à des attaques menées par un pirate utilisateur. En effet, celui-ci peut obtenir une paire (texte chiffré /  
25 texte clair) en écoutant les communications lors de l'exécution d'une requête autorisée (i.e., sur laquelle il possède les droits). Cette paire de texte est une arme importante pour casser la clé de chiffrement. Les communications entre le serveur et la carte de sécurisation  
30 doivent donc être chiffrées avec une des techniques de l'état de l'art.

La carte de sécurisation (4) participant de façon active à l'exécution des requêtes, il devient envisageable de stocker une partie de la base de données, considérée  
35 comme très sensible, directement sous le contrôle de la

carte. Par exemple, dans le cas où l'application de sécurité s'exécute dans une carte à puce, ces données très sensibles peuvent être stockées dans la mémoire stable (EEPROM) de la carte à puce. Ceci permet de rendre ces données définitivement inaccessibles à tout type de pirate (y compris un pirate utilisateur). Par exemple, si les données stockées dans la carte correspondent aux données d'identification de l'utilisateur, même un pirate arrivant à déchiffrer l'ensemble de la base ne saura pas à qui appartiennent les données correspondantes. Cette approche offre un degré de protection inégalable mais pose trois problèmes importants : (i) augmentation de la complexité de l'évaluation de requêtes ; (ii) durabilité des données sensibles ; et (iii) partage éventuel de ces données entre plusieurs utilisateurs.

Pour évaluer une requête sur des données dont une partie réside dans une carte à puce, l'application de sécurisations devra générer un plan d'exécution réparti entre deux sites. Cette tâche, complexe pour une carte à puce, peut être simplifiée en imposant une façon particulière d'isoler les données sensibles. L'idée consiste à supprimer les données sensibles de la base stockée sur le serveur, en les remplaçant par des indices dans un domaine sensible (ensemble de données sans doublons), stocké lui, sur la carte à puce. Par exemple, les codes de cartes de crédit d'un ensemble de clients pourront être stockés comme des indices dans le domaine 'codes de cartes de crédit' stocké dans la carte à puce. Ce 'codage' des données sensibles peut être vu comme un chiffrement très particulier de ces données, chiffrement dont la particularité est d'être définitivement incassable sans la carte à puce. L'intérêt de cette modélisation est qu'elle permet de conserver exactement les mêmes techniques d'évaluation de requêtes, une partie de l'évaluation

pouvant se faire sur le serveur puisque la propriété  $\forall a, \forall b, a = b \Leftrightarrow \text{Chiffre}(a) = \text{Chiffre}(b)$  est préservée.

Pour ce qui concerne la durabilité et le partage, considérons tout d'abord le cas des données sensibles statiques (i.e., sans mises à jour). Ce cas particulier est susceptible de représenter une majorité de cas pratiques (par exemple, les données d'identification de particuliers ou de PME). La durabilité des données statiques peut être assurée par le stockage redondant de ces données dans une carte à puce de backup. Cette carte de backup n'est utilisée qu'en cas de perte, vol ou destruction de la carte initiale. Le partage des données sensibles statiques est assuré par leur copie dans les cartes à puce de chaque utilisateur. Pour ce type de données, le niveau de sécurité atteint est maximum puisque aucune copie n'est accessible sur aucun serveur et que le contenu de la carte est considéré comme inviolable.

Considérons maintenant le cas des données dynamiques. Le partage et la durabilité de ces données doivent être assurés par un serveur connu de l'ensemble des utilisateurs partageant ces données. Ce serveur doit conserver l'historique des mises à jour effectuées sur les données dynamiques et diffuser, sur demande, ces mises à jour aux utilisateurs y ayant accès. Il est préférable que ce serveur soit distinct du serveur de données pour se prémunir contre des attaques internes (outre le cassage des clés de chiffrement, une coalition entre pirates internes des deux sites devient alors nécessaire pour mener une attaque). Les données stockées sur le serveur de partage et de durabilité sont évidemment chiffrées. Notons qu'il est plus facile d'obtenir un chiffrement efficace sur ces données historiques car ici, aucune contrainte relative à l'exécution de requêtes n'est imposée sur les méthodes de

chiffrement. A l'extrême, on peut envisager une clé différente pour chaque donnée historique.

La figure 7 résume une utilisation conjointe de l'ensemble des techniques exposées dans les sections  
5 précédentes, avec C-SDA s'exécutant dans une carte à puce. Les étapes de l'exécution d'une requête utilisateur sont les suivantes :

1. Lors de la connexion, l'application de sécurisation contacte le serveur de droits et de vues pour  
10 mettre à jour la liste des droits et des vues de l'utilisateur. Il les déchiffre avec sa base de clés.

2. l'application de sécurisation contacte le serveur de partage/durabilité afin de mettre à jour les données sensibles dynamiques hébergées par la carte à puce.

15 3. L'utilisateur émet une requête  $Q$ .

4. Les droits sont vérifiés par l'application de sécurisation avec la base de droits et de vues.

5. La requête, si elle met en jeu des vues, est traduite en une requête  $Q'$  portant sur les relations de  
20 base.

6. La requête  $Q'$  est transformée en un plan d'exécution de la forme  $Q' = Q_t(Q_c(Q_s))$ , éventuellement précédé de requêtes de préparation  $(*[Q_c^p(Q_s^p)])$  en cas de prédicats d'inégalité dont la sélectivité est intéressante  
25 à exploiter.

7.  $Q_s$  est envoyée au serveur qui l'exécute sur les données chiffrées et renvoie les tuples résultats à l'application de sécurisation. Le résultat est déchiffré par l'application de sécurité en utilisant la base de clés.

30 8. La requête complément  $Q_c$  est exécutée par l'application de sécurité.

9. Le résultat est éventuellement complété avec les valeurs des données sensibles (DS) lors des projections.

10. Le résultat final déchiffré est renvoyé au terminal qui exécute  $Q_t$ , c.à.d., le tri final s'il est demandé.

La figure 8 représente une vue de l'architecture de la carte de sécurité (4). Le circuit électronique comporte un circuit d'interface entrée-sortie (14) pour les échanges avec l'équipement client (1) et le serveur (2). La sortie de ce circuit (14) est relié à une mémoire tampon (15).

La carte comporte également un microprocesseur (10) et une mémoire EEPROM (11) pour l'enregistrement du système d'exploitation et de l'application de sécurité et une mémoire (12) pour l'enregistrement des données personnelles de l'utilisateur et en particulier la clé de chiffrement ou les paramètres de chiffrement de l'utilisateur, ainsi que les données relatives aux droits de l'utilisateur.

La carte comporte encore une mémoire vive (15) comportant des registres (16) pour l'enregistrement des données intermédiaires des calculs, et un compteur (17) pour la réalisation des calculs correspondant à des requêtes contenant des prédicats statistiques tels que des moyennes.

L'invention est décrite dans ce qui précède de façon non limitative. L'architecture du moyen de sécurisation peut être réalisé sous différentes formes sans pour autant sortir du cadre de l'invention.

## REVENDICATIONS

1 - Système de gestion sécurisée de bases de données confidentielles comprenant un serveur constitué par  
5 au moins un ordinateur équipé de moyens d'enregistrement de bases de données et de moyens de communication, par au moins un équipement informatique hôte comprenant des moyens de communication avec le serveur et des moyens pour la construction de requêtes et le traitement des résultats des  
10 requêtes, ainsi qu'un moyen de sécurisation des échanges entre l'équipement client (1) et le serveur, caractérisé en ce que ledit moyen de sécurisation est constitué par un support matériel sécurisé relié à l'équipement client (1) et comportant un microprocesseur pour le chiffrement des  
15 attributs des requêtes émises par l'équipement client (1) et le déchiffrement des réponses émises par le serveur, une mémoire pour l'enregistrement de résultats intermédiaires, une mémoire pour l'enregistrement du système d'exploitation et que ledit serveur enregistre des données chiffrées.

20

2 - Système de gestion sécurisée de bases de données selon la revendication 1, caractérisé en ce que le moyen de sécurisation est constitué par une carte à microprocesseur.

25

3 - Système de gestion sécurisée de bases de données selon la revendication 1, caractérisé en ce que le moyen de sécurisation est constitué par une clé raccordable sur un port d'un équipement client (1).

30

4 - Système de gestion sécurisée de bases de données selon l'une quelconque des revendications précédentes, caractérisé en ce que le moyen de sécurisation comprend en outre une mémoire pour l'enregistrement des  
35 informations de personnalisation de l'utilisateur.

5 - Système de gestion sécurisée de bases de données selon l'une quelconque des revendications précédentes, caractérisé en ce que le moyen de sécurisation  
5 comporte en outre au moins un compteur pour l'exécution de requêtes statistiques.

6 - Système de gestion sécurisée de bases de données, selon l'une quelconque des revendications  
10 précédentes caractérisé en ce que le moyen de sécurisation comporte en outre un registre pour la gestion des droits téléchargés lors de l'ouverture d'une session avec le serveur.

7 - Système de gestion sécurisée de bases de données selon l'une quelconque des revendications précédentes, caractérisé en ce que le moyen de sécurisation  
15 comporte en outre une mémoire pour l'enregistrement d'une partie de la base de données.

8 - Procédé de gestion sécurisée d'une base de données comportant une étape de construction d'une requête comprenant au moins un attribut, caractérisé en ce qu'il  
20 comporte une étape de chiffrement des attributs par un calculateur solidaire d'un dispositif individuel de sécurité relié à un équipement client, le procédé consistant à interroger ensuite une base de données contenant des données chiffrées avec les mêmes moyens de  
25 chiffrement que ceux utilisés lors de l'étape précédente, à retourner une réponse contenant les données correspondant aux attributs de la requête et à procéder au déchiffrement desdites données par ledit calculateur d'un dispositif individuel de sécurité avant de les transmettre à  
30 l'équipement hôte.

9 - Procédé de gestion sécurisée d'une base de données selon la revendication 8, caractérisé en ce que le chiffrement des attributs de la requête et le déchiffrement de la réponse est réalisé par une application de sécurité exploitée par une carte à microprocesseur.

10 - Procédé de gestion sécurisée d'une base de données selon la revendication 8, caractérisé en ce que le chiffrement des attributs de la requête et le déchiffrement de la réponse est réalisé par une application de sécurité exploitée par une clé raccordable sur un port d'un équipement client.

11 - Procédé de gestion sécurisée d'une base de données selon la revendication 8, caractérisé en ce que le chiffrement des attributs de la requête et le déchiffrement de la réponse est réalisé selon un algorithme à clé secrète tel que DES.

12 - Procédé de gestion sécurisée d'une base de données selon la revendication 8, caractérisé en ce que la traduction de requêtes est limitée aux prédicats d'égalité pour permettre l'exécution desdites requêtes exécutables directement par le serveur sur des données chiffrées.

13 - Procédé de gestion sécurisée d'une base de données selon la revendication 8, caractérisé en ce que la traduction de requêtes comportant des prédicats d'inégalité ou fonctions de calcul comporte une étape de décomposition de la requête  $Q$  en un plan  $Q_t(Q_c(Q_s))$ , d'interrogation de la base de données chiffrées (3) par des requêtes d'inégalités à partir d'attributs chiffrés, une étape d'enregistrement des résultats des requêtes dans une mémoire temporaire du moyen de sécurisation individuel

après déchiffrement des données, et de recomposition du résultat.

14 - Procédé de gestion sécurisée d'une base de données selon la revendication 8, caractérisé en ce que la traduction de requêtes comportant des prédicats d'inégalité ou fonctions de calcul comporte une étape de décomposition d'une requête en un plan  $Q = Q_t (Q_c (Q_s (*[Q_c^P (Q_s^P)])))$ , où  $Q_c^P$  et  $Q_s^P$  sont des requêtes de préparation, d'interrogation de la base de données chiffrées (3) par des requêtes d'inégalités à partir d'attributs chiffrés, une étape d'enregistrement des résultats des requêtes dans une mémoire temporaire du moyen de sécurisation individuel après déchiffrement des données, et de recomposition du résultat.

15 - Procédé de gestion sécurisée d'une base de données selon la revendication 8, caractérisé en ce qu'une partie de la base des données est enregistrée dans une mémoire du moyen individuel de sécurisation.

16 - Procédé de gestion sécurisée d'une base de données selon la revendication 8, caractérisé en ce que la résolution d'une requête SQL est réalisée en effectuant les étapes de saisie de la requête en clair sur l'équipement client (1), de chiffrement par l'application de sécurité (4) de l'attribut de la requête et l'envoi de la requête chiffrée au serveur, de résolution de la requête sur les données chiffrées par le serveur (2), déchiffrement du résultat sur la carte (4) et de restitution du résultat en clair sur l'équipement client (1).

17 - Procédé de gestion sécurisée d'une base de données selon la revendication 8, caractérisé en ce que la résolution d'une requête SQL comportant des prédicats

d'inégalité ou des fonctions de calcul est réalisée en effectuant les étapes de saisie de la requête en clair sur l'équipement client (1), de chiffrement par l'application de sécurité (4) et d'envoi au serveur (2) d'une sous-  
 5 requête chiffrée  $Q_S$ , contenant uniquement des prédicats d'égalité, de résolution de la requête sur les données chiffrées par le serveur, de déchiffrement du résultat de  $Q_S$  et évaluation de  $Q_C$  et de reconstruction de la réponse à la requête initiale par l'application de sécurisation (4).

10

18 - Procédé de gestion sécurisée d'une base de données selon la revendication 8, caractérisé en ce qu'il comporte un pré-traitement coopératif entre la carte et le serveur, consistant, pour une requête  $Q$  contenant un  
 15 prédicat de la forme  $\sigma_{a_i \theta \text{ valeur}}(T)$ , où  $\sigma$  dénote l'opérateur de restriction,  $\theta \in \{<, >, \leq, \geq\}$  et  $a_i$  est un attribut quelconque de la table  $T$ , à envoyer au serveur une requête de pré-traitement  $Q_S^P = \Pi_{\text{clé}, a_i}(T)$ ,  $\Pi$  dénotant l'opérateur de projection, à exécuter sur le moyen de sécurisation (4)  
 20 la fonction  $Q_C^P = \Pi_{\text{clé}}(\sigma_{a_i \theta \text{ valeur}}(\Pi_{\text{clé}, \text{Déchiffre}(a_i)}(Q_S^P)))$  de renvoyer le résultat  $R$  au serveur (2), l'application de sécurisation (4) transformant ensuite la requête  $Q$  en remplaçant le prédicat initial  $\sigma_{a_i \theta \text{ valeur}}(T)$  par le prédicat  $\propto(T, R)$ , où  $\propto$  dénote l'opérateur de semi-jointure sur clé.

25

19 - Procédé de gestion sécurisée d'une base de données selon la revendication 8, caractérisé en ce qu'il comporte les étapes suivantes :

- Lors de la connexion, l'application de  
 30 sécurisation contacte le serveur de droits et de vues pour mettre à jour la liste des droits et des vues de l'utilisateur. Il les déchiffre avec sa base de clés.

- L'application de sécurisation contacte le serveur de partage/durabilité afin de mettre à jour les données sensibles dynamiques hébergées par la carte (4),
- 5       - L'utilisateur émet une requête  $Q$ ,
- Les droits sont vérifiés par l'application de sécurisation avec la base de droits et de vues,
- La requête, si elle met en jeu des vues, est traduite en une requête  $Q'$  portant sur les relations
- 10       de base,
- La requête  $Q'$  est transformée en un plan d'exécution de la forme  $Q' = Q_t(Q_c(Q_s))$ , éventuellement précédé de requêtes de préparation  $(*[Q_c^p(Q_s^p)])$  en cas de prédicats d'inégalité dont
- 15       la sélectivité est intéressante à exploiter,
- $Q_s$  est envoyée au serveur qui l'exécute sur les données chiffrées et renvoie les tuples résultats à l'application de sécurisation, le résultat est déchiffré par l'application de sécurité en utilisant
- 20       la base de clés,
- La requête complément  $Q_c$  est exécutée par l'application de sécurité,
- Le résultat est éventuellement complété avec les valeurs des données sensibles (DS) lors des
- 25       projections,
- Le résultat final déchiffré est renvoyé au terminal qui exécute  $Q_t$ , c.à.d., le tri final s'il est demandé.

Fig. 1

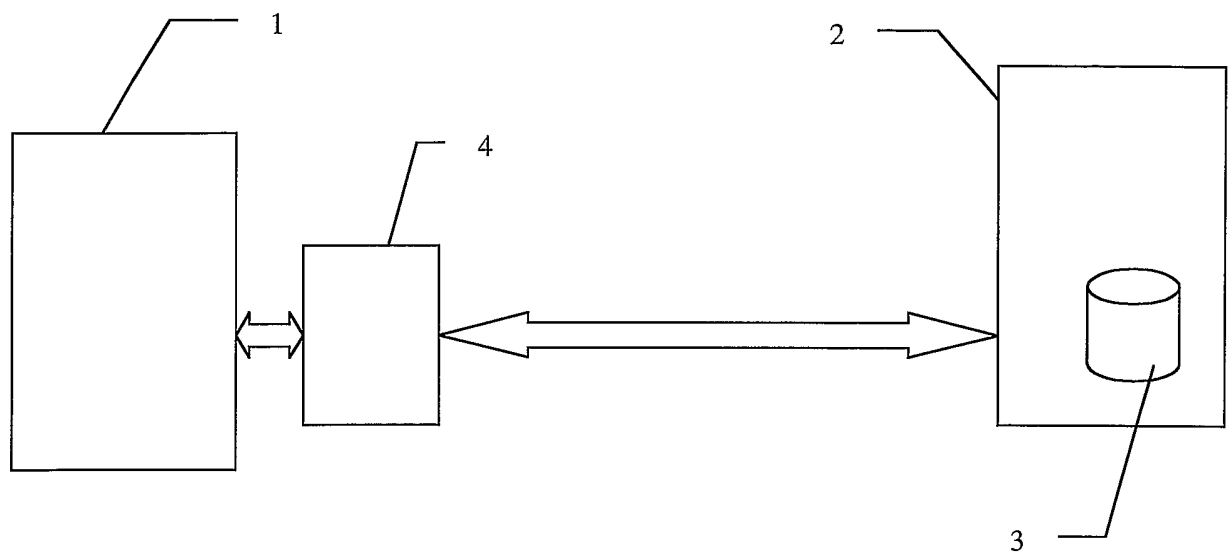


Fig. 2

| Produits |      |          |      |
|----------|------|----------|------|
| ref      | nom  | type     | prix |
| d300     | Dell | Pentium3 | 9800 |
| I260     | IBM  | Pentium2 | 6400 |
|          |      |          |      |

| lqskdqs |       |          |      |
|---------|-------|----------|------|
| sdz     | azds  | sdeefa   | zze  |
| zsxd    | dedef | zarevgzd | Fffe |
| df'g    | Sde   | iukèfsa  | dgss |
|         |       |          |      |

Fig. 3

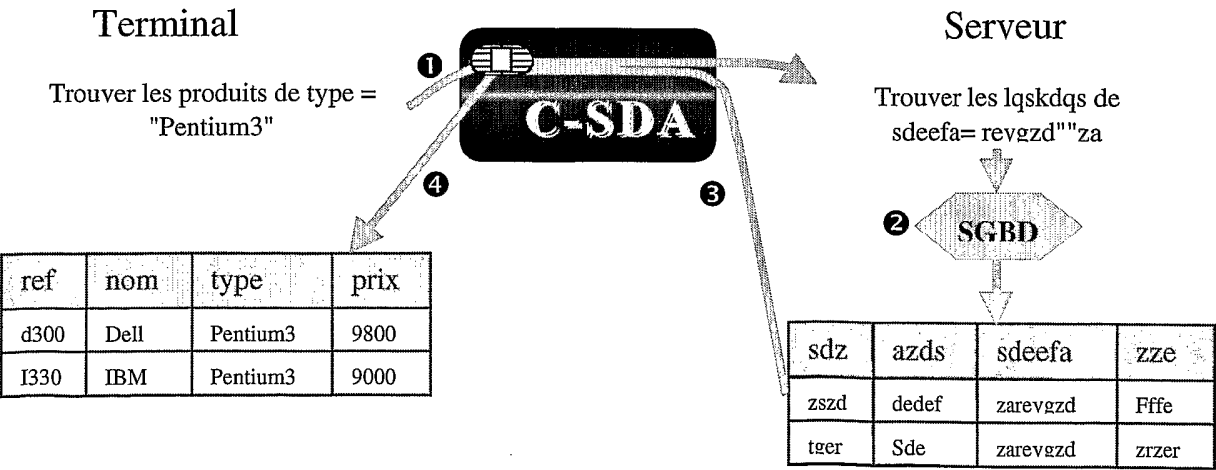


Fig. 4

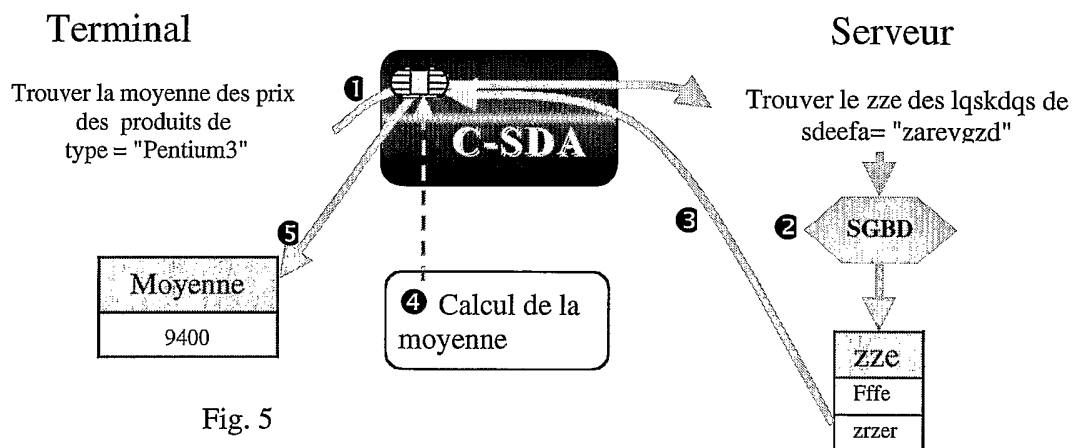


Fig. 5

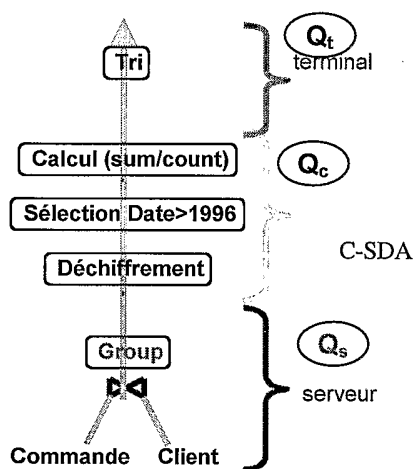
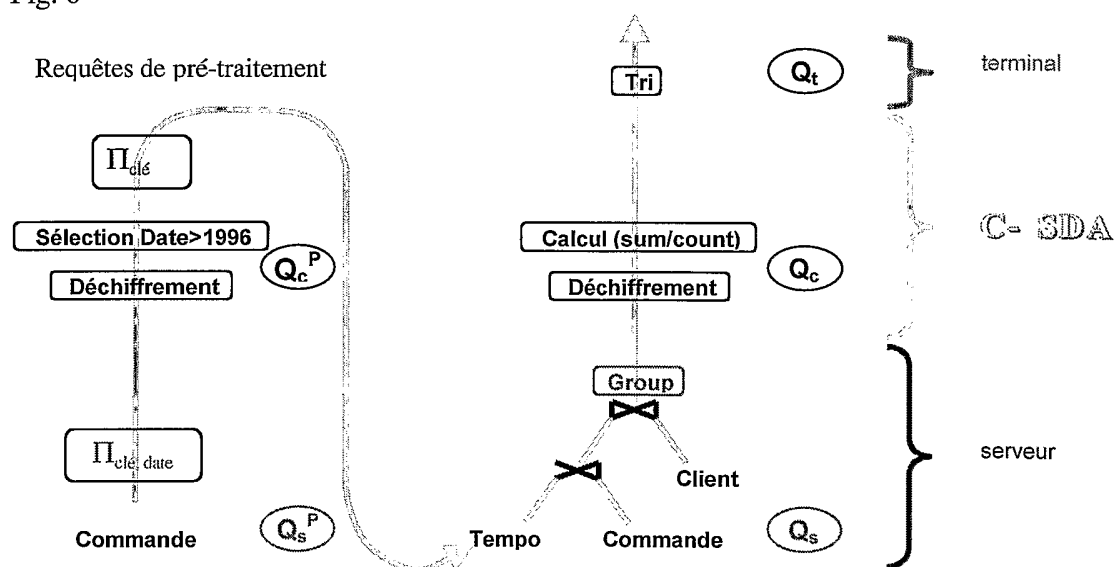


Fig. 6



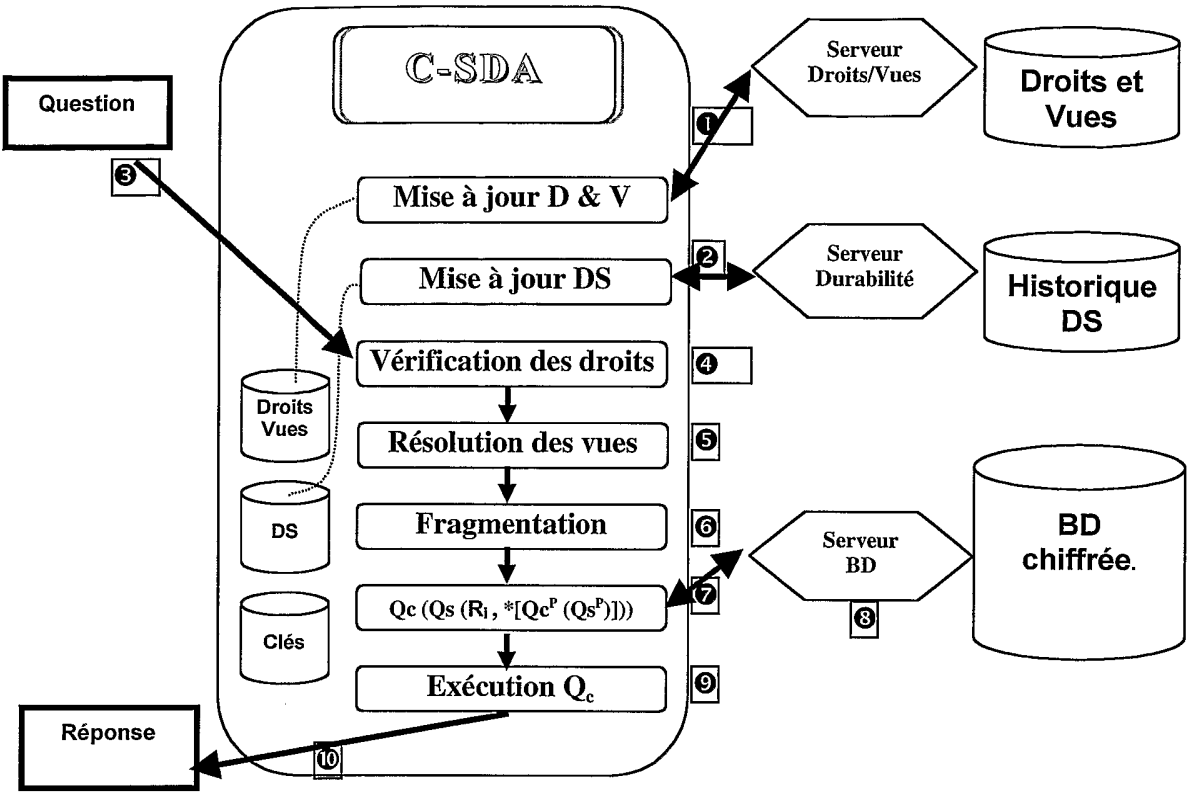


Fig. 7

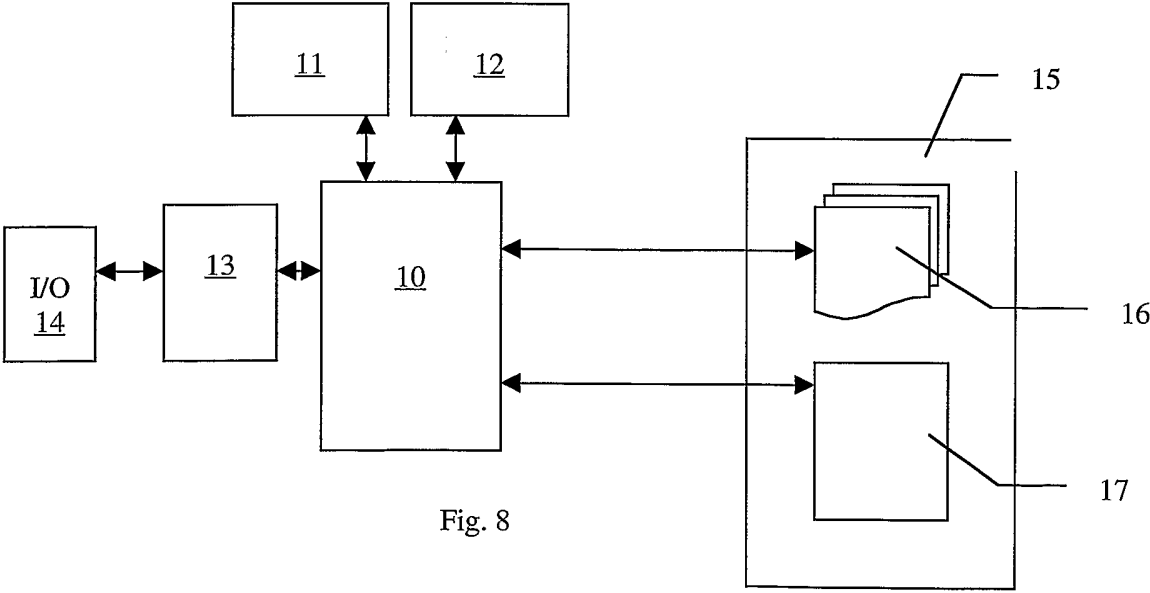


Fig. 8

## INTERNATIONAL SEARCH REPORT

International Application No

PCT/FR 02/02824

A. CLASSIFICATION OF SUBJECT MATTER  
 IPC 7 G06F1/00 G06F17/30

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC 7 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

WPI Data, EPO-Internal

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

| Category * | Citation of document, with indication, where appropriate, of the relevant passages                                                                                                                                                                  | Relevant to claim No. |
|------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| Y          | US 2001/007975 A1 (NYBERG JAMES RONALD ET AL) 12 July 2001 (2001-07-12)<br>abstract<br>page 1, paragraph 16 -page 3, paragraph 28<br>page 3, paragraph 35<br>figures 1,2<br>---                                                                     | 1                     |
| Y          | US 5 559 888 A (JAIN AJAY ET AL)<br>24 September 1996 (1996-09-24)<br>column 1, line 58 - line 67<br>---                                                                                                                                            | 1                     |
| Y          | US 5 963 642 A (GOLDSTEIN BENJAMIN D)<br>5 October 1999 (1999-10-05)<br>column 11, line 58 -column 12, line 36<br>column 13, line 27 - line 61<br>column 14, line 38 -column 15, line 31<br>column 26, line 43 - line 56<br>figure 5<br>---<br>-/-- | 1,2,7-9,<br>12,15     |



Further documents are listed in the continuation of box C.



Patent family members are listed in annex.

\* Special categories of cited documents :

- \*A\* document defining the general state of the art which is not considered to be of particular relevance
- \*E\* earlier document but published on or after the international filing date
- \*L\* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- \*O\* document referring to an oral disclosure, use, exhibition or other means
- \*P\* document published prior to the international filing date but later than the priority date claimed

- \*T\* later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
- \*X\* document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
- \*Y\* document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.
- \* & \* document member of the same patent family

Date of the actual completion of the international search

19 December 2002

Date of mailing of the international search report

02/01/2003

Name and mailing address of the ISA

European Patent Office, P.B. 5818 Patentlaan 2  
 NL - 2280 HV Rijswijk  
 Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,  
 Fax: (+31-70) 340-3016

Authorized officer

Arbutina, L

## INTERNATIONAL SEARCH REPORT

International Application No

PCT/FR 02/02824

## C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

| Category * | Citation of document, with indication, where appropriate, of the relevant passages                                                                                                                                                                                                                                                                                                                                                                                   | Relevant to claim No. |
|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| Y          | <p>DOMINGO-FERRER J: "Multi-application smart cards and encrypted data, processing"<br/> FUTURE GENERATIONS COMPUTER SYSTEMS,<br/> ELSEVIER SCIENCE PUBLISHERS. AMSTERDAM,<br/> NL,<br/> vol. 13, no. 1, 1 July 1997 (1997-07-01),<br/> pages 65-74, XP004081710<br/> ISSN: 0167-739X<br/> page 65, left-hand column, line 1 -page<br/> 69, left-hand column, line 17<br/> figures 1,2</p> <p>----</p>                                                               | 1,2,7-9,<br>12,15     |
| A          | <p>US 5 721 777 A (BLAZE MATTHEW A)<br/> 24 February 1998 (1998-02-24)<br/> column 2, line 10 - line 35</p> <p>----</p>                                                                                                                                                                                                                                                                                                                                              | 5                     |
| A          | <p>FERREIRA R: "THE PRACTICAL APPLICATION OF<br/> STATE OF THE ART SECURITY IN REAL<br/> ENVIRONMENTS"<br/> ADVANCES IN CRYPTOLOGY - AUSCRYPT. SYDNEY,<br/> JAN. 8 - 11, 1990, PROCEEDINGS OF THE<br/> INTERNATIONAL CONFERENCE ON CRYPTOLOGY -<br/> AUSCRYPT, BERLIN, SPRINGER, DE,<br/> vol. CONF. 1, 8 January 1990 (1990-01-08),<br/> pages 334-355, XP000145211<br/> ISBN: 3-540-53000-2<br/> page 351, paragraph 3 -page 352, paragraph<br/> 2</p> <p>----</p> | 6                     |
| A          | <p>EP 0 766 184 A (SUN MICROSYSTEMS INC)<br/> 2 April 1997 (1997-04-02)<br/> abstract</p> <p>-----</p>                                                                                                                                                                                                                                                                                                                                                               | 16                    |

## INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No

PCT/FR 02/02824

| Patent document<br>cited in search report |    | Publication<br>date | Patent family<br>member(s)                                        | Publication<br>date                                  |
|-------------------------------------------|----|---------------------|-------------------------------------------------------------------|------------------------------------------------------|
| US 2001007975                             | A1 | 12-07-2001          | AU 1230900 A<br>WO 0025247 A1                                     | 15-05-2000<br>04-05-2000                             |
| US 5559888                                | A  | 24-09-1996          | NONE                                                              |                                                      |
| US 5963642                                | A  | 05-10-1999          | AU 5723198 A<br>EP 1013030 A1<br>JP 2001507837 T<br>WO 9829981 A1 | 31-07-1998<br>28-06-2000<br>12-06-2001<br>09-07-1998 |
| US 5721777                                | A  | 24-02-1998          | NONE                                                              |                                                      |
| EP 0766184                                | A  | 02-04-1997          | US 5713018 A<br>EP 0766184 A2<br>JP 9179871 A                     | 27-01-1998<br>02-04-1997<br>11-07-1997               |

## RAPPORT DE RECHERCHE INTERNATIONALE

Demande Internationale No

PCT/FR 02/02824

**A. CLASSEMENT DE L'OBJET DE LA DEMANDE**  
 CIB 7 G06F1/00 G06F17/30

Selon la classification internationale des brevets (CIB) ou à la fois selon la classification nationale et la CIB

**B. DOMAINES SUR LESQUELS LA RECHERCHE A PORTE**

Documentation minimale consultée (système de classification suivi des symboles de classement)

CIB 7 G06F

Documentation consultée autre que la documentation minimale dans la mesure où ces documents relèvent des domaines sur lesquels a porté la recherche

Base de données électronique consultée au cours de la recherche internationale (nom de la base de données, et si réalisable, termes de recherche utilisés)

WPI Data, EPO-Internal

**C. DOCUMENTS CONSIDERES COMME PERTINENTS**

| Catégorie ° | Identification des documents cités, avec, le cas échéant, l'indication des passages pertinents                                                                                                                                                                      | no. des revendications visées |
|-------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------|
| Y           | US 2001/007975 A1 (NYBERG JAMES RONALD ET AL) 12 juillet 2001 (2001-07-12)<br>abrége<br>page 1, alinéa 16 -page 3, alinéa 28<br>page 3, alinéa 35<br>figures 1,2<br>---                                                                                             | 1                             |
| Y           | US 5 559 888 A (JAIN AJAY ET AL)<br>24 septembre 1996 (1996-09-24)<br>colonne 1, ligne 58 - ligne 67<br>---                                                                                                                                                         | 1                             |
| Y           | US 5 963 642 A (GOLDSTEIN BENJAMIN D)<br>5 octobre 1999 (1999-10-05)<br>colonne 11, ligne 58 -colonne 12, ligne 36<br>colonne 13, ligne 27 - ligne 61<br>colonne 14, ligne 38 -colonne 15, ligne 31<br>colonne 26, ligne 43 - ligne 56<br>figure 5<br>-----<br>-/-- | 1,2,7-9,<br>12,15             |



Voir la suite du cadre C pour la fin de la liste des documents



Les documents de familles de brevets sont indiqués en annexe

° Catégories spéciales de documents cités:

- \*A\* document définissant l'état général de la technique, non considéré comme particulièrement pertinent
- \*E\* document antérieur, mais publié à la date de dépôt international ou après cette date
- \*L\* document pouvant jeter un doute sur une revendication de priorité ou cité pour déterminer la date de publication d'une autre citation ou pour une raison spéciale (telle qu'indiquée)
- \*O\* document se référant à une divulgation orale, à un usage, à une exposition ou tous autres moyens
- \*P\* document publié avant la date de dépôt international, mais postérieurement à la date de priorité revendiquée

- \*T\* document ultérieur publié après la date de dépôt international ou la date de priorité et n'appartenant pas à l'état de la technique pertinent, mais cité pour comprendre le principe ou la théorie constituant la base de l'invention
- \*X\* document particulièrement pertinent; l'invention revendiquée ne peut être considérée comme nouvelle ou comme impliquant une activité inventive par rapport au document considéré isolément
- \*Y\* document particulièrement pertinent; l'invention revendiquée ne peut être considérée comme impliquant une activité inventive lorsque le document est associé à un ou plusieurs autres documents de même nature, cette combinaison étant évidente pour une personne du métier
- \*Z\* document qui fait partie de la même famille de brevets

Date à laquelle la recherche internationale a été effectivement achevée

19 décembre 2002

Date d'expédition du présent rapport de recherche internationale

02/01/2003

Nom et adresse postale de l'administration chargée de la recherche internationale

Office Européen des Brevets, P.B. 5818 Patentlaan 2  
 NL - 2280 HV Rijswijk  
 Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,  
 Fax: (+31-70) 340-3016

Fonctionnaire autorisé

Arbutina, L

## C.(suite) DOCUMENTS CONSIDERES COMME PERTINENTS

| Catégorie | Identification des documents cités, avec, le cas échéant, l'indication des passages pertinents                                                                                                                                                                                                                                                                                                                                                           | no. des revendications visées |
|-----------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------|
| Y         | <p>DOMINGO-FERRER J: "Multi-application smart cards and encrypted data, processing"<br/> FUTURE GENERATIONS COMPUTER SYSTEMS,<br/> ELSEVIER SCIENCE PUBLISHERS. AMSTERDAM,<br/> NL,<br/> vol. 13, no. 1,<br/> 1 juillet 1997 (1997-07-01), pages 65-74,<br/> XP004081710<br/> ISSN: 0167-739X<br/> page 65, colonne de gauche, ligne 1 -page<br/> 69, colonne de gauche, ligne 17<br/> figures 1,2</p> <p>---</p>                                        | 1,2,7-9,<br>12,15             |
| A         | <p>US 5 721 777 A (BLAZE MATTHEW A)<br/> 24 février 1998 (1998-02-24)<br/> colonne 2, ligne 10 - ligne 35</p> <p>---</p>                                                                                                                                                                                                                                                                                                                                 | 5                             |
| A         | <p>FERREIRA R: "THE PRACTICAL APPLICATION OF<br/> STATE OF THE ART SECURITY IN REAL<br/> ENVIRONMENTS"<br/> ADVANCES IN CRYPTOLOGY - AUSCRYPT. SYDNEY,<br/> JAN. 8 - 11, 1990, PROCEEDINGS OF THE<br/> INTERNATIONAL CONFERENCE ON CRYPTOLOGY -<br/> AUSCRYPT, BERLIN, SPRINGER, DE,<br/> vol. CONF. 1, 8 janvier 1990 (1990-01-08),<br/> pages 334-355, XP000145211<br/> ISBN: 3-540-53000-2<br/> page 351, alinéa 3 -page 352, alinéa 2</p> <p>---</p> | 6                             |
| A         | <p>EP 0 766 184 A (SUN MICROSYSTEMS INC)<br/> 2 avril 1997 (1997-04-02)<br/> abrégé</p> <p>-----</p>                                                                                                                                                                                                                                                                                                                                                     | 16                            |

# RAPPORT DE RECHERCHE INTERNATIONALE

Renseignements relatifs aux membres de familles de brevets

Deposition internationale No

PCT/FR 02/02824

| Document brevet cité<br>au rapport de recherche | Date de<br>publication | Membre(s) de la<br>famille de brevet(s)                           | Date de<br>publication                               |
|-------------------------------------------------|------------------------|-------------------------------------------------------------------|------------------------------------------------------|
| US 2001007975 A1                                | 12-07-2001             | AU 1230900 A<br>WO 0025247 A1                                     | 15-05-2000<br>04-05-2000                             |
| US 5559888 A                                    | 24-09-1996             | AUCUN                                                             |                                                      |
| US 5963642 A                                    | 05-10-1999             | AU 5723198 A<br>EP 1013030 A1<br>JP 2001507837 T<br>WO 9829981 A1 | 31-07-1998<br>28-06-2000<br>12-06-2001<br>09-07-1998 |
| US 5721777 A                                    | 24-02-1998             | AUCUN                                                             |                                                      |
| EP 0766184 A                                    | 02-04-1997             | US 5713018 A<br>EP 0766184 A2<br>JP 9179871 A                     | 27-01-1998<br>02-04-1997<br>11-07-1997               |