



(72) MARQUET, BERTRAND, FR

(72) FOUQUET, GUY, FR

(72) BALLESTER, LAURENT, FR

(71) ALCATEL, FR

(51) Int.Cl.⁷ H04L 12/22

(30) 1999/05/06 (99 05 763) FR

(54) **SYSTEME DE TRAITEMENT DE L'INFORMATION**

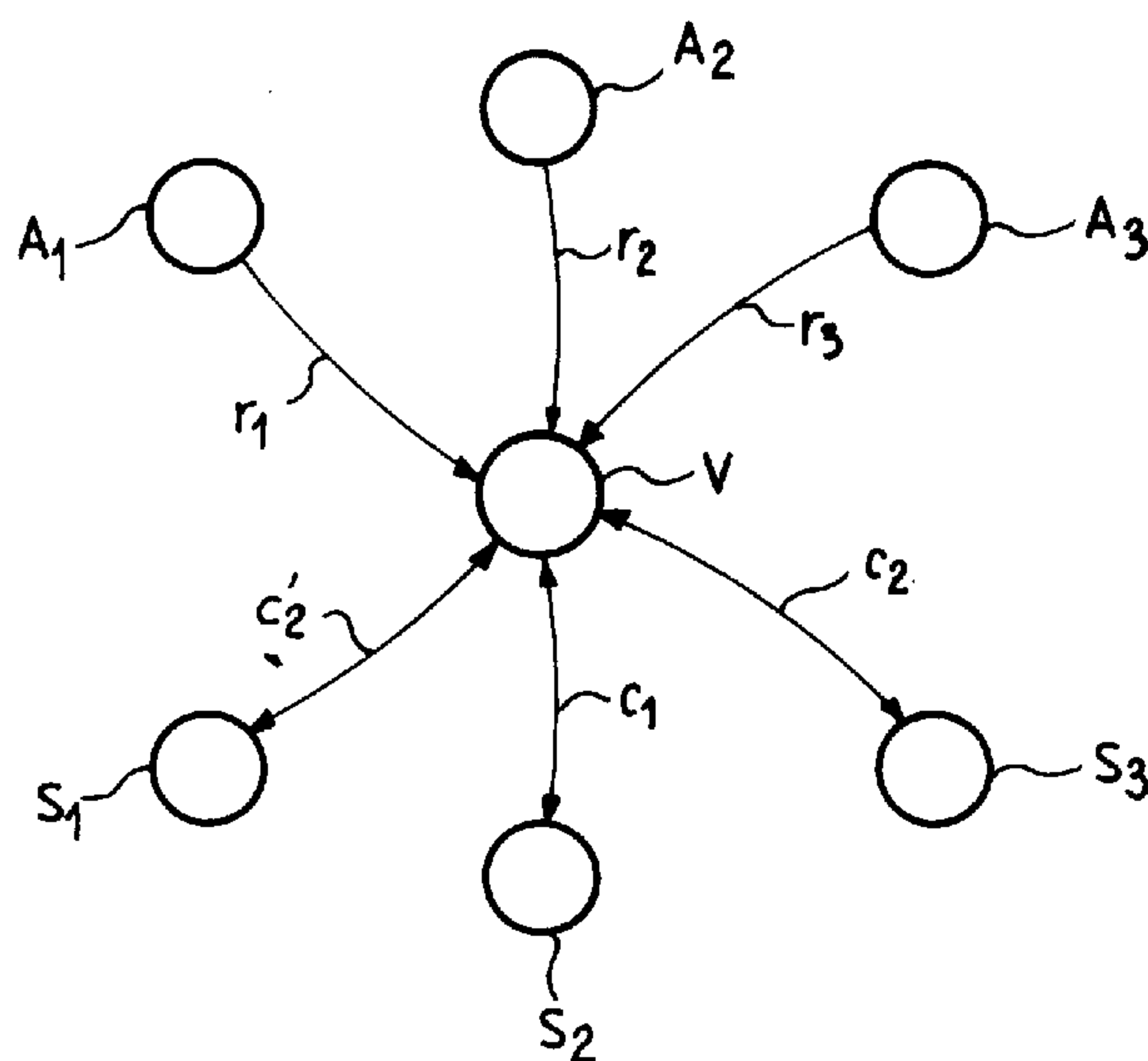
PERMETTANT LA SECURISATION DES

COMMUNICATIONS ENTRE COMPOSANTS LOGICIELS

(54) **INFORMATION PROCESSING SYSTEM PERMITTING**

SECURE COMMUNICATIONS BETWEEN SOFTWARE

COMPONENTS



(57) Serveur virtuel de sécurité permettant à un ensemble d'applications d'accéder à un ensemble de services de sécurité, caractérisé en ce qu'il possède · des moyens pour recevoir de la part des applications logicielles des requêtes de services, · des moyens de communication avec des serveurs de sécurité mettant en oeuvre ces services de sécurité, · des moyens pour choisir au moins un serveur de sécurité destinataire pour chaque requête de service reçue et pour lui transmettre des informations lui permettant de fournir le service de sécurité correspondant à la requête de service.

ABREGE

Serveur virtuel de sécurité permettant à un ensemble d'applications d'accéder à un ensemble de services de sécurité, caractérisé en ce qu'il possède :

- des moyens pour recevoir de la part des applications logicielles des requêtes de services,
- des moyens de communication avec des serveurs de sécurité mettant en œuvre ces services de sécurité,
- des moyens pour choisir au moins un serveur de sécurité destinataire pour chaque requête de service reçue et pour lui transmettre des informations lui permettant de fournir le service de sécurité correspondant à la requête de service.

Système de traitement de l'information permettant la sécurisation des communications entre composants logiciels

La présente invention est relative à un système informatique
5 nécessitant la sécurisation des communications entre certains des composants logiciels le composant.

L'invention s'applique particulièrement bien aux systèmes de gestion de réseau, notamment de réseau de télécommunication. Cet exemple d'application sera plus particulièrement développé par la suite, mais
10 l'invention est toutefois susceptible de s'appliquer à d'autres types d'applications, par exemple aux applications pour le commerce électronique.

De façon classique, les systèmes de gestion de réseaux de télécommunication comportent un ensemble d'applications logicielles de
15 gestion. Ces applications peuvent être réparties au sein d'un système distribué, et peuvent avoir besoin de communiquer entre elles afin d'échanger des informations.

Un besoin peut exister de sécuriser ces communications. En fonction
20 de la menace que l'on craint ainsi que du niveau de sensibilité des informations transmises, différents services de sécurisation peuvent être mis en œuvre, comme par exemple :

- Identification et authentification : par cette technique, le destinataire d'un message est assuré de l'origine de ce message. On a ainsi la
25 garantie qu'il n'existe pas de messages émis par un tiers malveillant dans le système.
- Contrôle d'accès : une application ne réagira aux commandes contenues dans les messages, qu'en fonction de règles définies dans une politique de sécurité. Par exemple, une application peut

ne communiquer qu'avec un ensemble déterminé de correspondants.

- Non-répudiation : un certain nombre d'informations sur les messages échangés sont mémorisées afin qu'aucune des parties prenantes ne puisse renier le fait d'avoir participé à la communication.
- Confidentialité : les messages sont chiffrés de sorte qu'un tiers ne puisse être en mesure d'en interpréter le contenu.

10 Typiquement, ces services, ainsi que d'autres non mentionnés, sont mis en œuvre par des applications logicielles spécifiques que l'on appelle serveurs de sécurité. Plusieurs serveurs de sécurité peuvent exister, chacun fournissant un ou plusieurs des services de sécurisation. De même chaque service de sécurisation peut être fourni à différents niveaux de qualité.

15 La figure 1 présente une architecture conforme à l'état de la technique dans ce domaine. L'application A_1 souhaite sécuriser une communication avec l'application A_2 . Pour cela, elle peut utiliser le serveur de sécurité S_1 qui fournit un support bas-niveau de cryptographie, ou bien le serveur de sécurité S_2 qui
20 fournit un service de cryptographie de haut-niveau.

Dans le cas où l'application A_1 utilise le serveur S_1 , elle envoie dans un premier temps un message m_1 de requête de service au serveur S_1 . Ce serveur S_1 retourne une clé dans un message de réponse m_2 . Ensuite, l'application A_1 peut envoyer son message m_3 à destination de l'application A_2 après l'avoir
25 chiffré avec la clé.

Dans le cas où l'application A_1 fait appel au serveur de sécurité S_2 , elle confie son message à transmettre m'_1 au serveur de sécurité S_2 . Ce serveur S_2 se charge alors de mettre en œuvre les techniques de cryptographie puis transmet le message crypté m'_2 à l'application destinatrice A_2 .

Ces deux exemples triviaux permettent de mettre en exergue le fait que pour un même service de sécurisation, différents niveaux de services peuvent être disponibles. De la même façon pour un même service et un même niveau de service, il peut exister différents protocoles de négociation
5 entre l'application initiatrice (A_1) et le serveur de sécurité. C'est par exemple le cas pour la négociation de la clé de chiffrement dans le cas d'un service de cryptographie. On peut citer, à titre d'exemples de tels protocoles, les méthodes de Diffie-Hellman ou de Needham-Schröder. Ces méthodes sont par exemples présentées dans l'ouvrage intitulé "*Practical Intranet Security*" de
10 Paul Ashley et Mark Vandenwauver, publié aux éditions Kluwer Academic Publishers en 1999.

Pour résumer, une application désirant sécuriser une communication avec une autre application, devra être capable de mettre en œuvre différents
15 protocoles de négociation en fonction du service et du niveau de service qu'elle désire.

Cela implique aussi que si l'on désire remplacer un serveur de sécurité correspondant à un service et un niveau de service donné, par un autre présentant par exemple une meilleure qualité de service mais mettant en
20 œuvre un protocole de négociation différent, les applications devront être modifiées si elles n'étaient pas prévues dès le départ pour ce nouveau protocole.

De plus, les moyens de communication à utiliser pour atteindre chacun des serveurs de sécurité peuvent être différents. Par exemple, chacune
25 des applications doit donc pouvoir gérer un accès direct, ou à travers un bus logiciel comme CORBA (*Common Object Request Broker Architecture*) de l'OMG (*Open Management Group*) ou DCOM (*Distributed Common Object Management*) de la société Microsoft, ou encore à travers un réseau.

Le but de la présente invention est, entre autres, de pallier ces différents inconvénients. Pour cela, elle a pour objet un serveur virtuel de sécurité permettant à un ensemble d'applications logicielles d'accéder à un ensemble de services de sécurité. Ce serveur virtuel de sécurité se caractérise

5 en ce qu'il possède :

- des moyens pour recevoir de la part des applications logicielles, des requêtes de services,
- des moyens de communication avec des serveurs de mettant en œuvre les services de sécurité en question,
- 10 • des moyens pour choisir au moins un serveur de sécurité destinataire pour chaque requête de service reçue et pour lui transmettre, via ces moyens de communication, des informations lui permettant de fournir le service de sécurité correspondant à la requête de service, certaines au moins de ces informations étant
- 15 contenues dans la requête de service.

Selon une mise en œuvre de l'invention, les applications logicielles peuvent accéder au serveur virtuel de sécurité au travers d'une interface de programmation ou API (pour *Application Programming Interface*, en anglais).

20 Selon une mise en œuvre de l'invention, le serveur virtuel de sécurité peut être développé en un langage de programmation tel Java.

L'invention a aussi pour objet le procédé susceptible d'être mis en œuvre par les applications logicielles pour utiliser ce serveur virtuel de sécurité. Ce procédé se caractérise en ce qu'il comporte les étapes de :

- émission de requêtes de service par une application logicielle à destination d'un serveur virtuel de sécurité,
- choix, par le serveur virtuel de sécurité, d'un serveur de sécurité destinataire, à même de fournir le service de sécurité
- 30 correspondant à cette requête de service,

- émission d'informations à destination du serveur de sécurité destinataire afin qu'il puisse fournir le service de sécurité, certaines au moins de ces informations étant contenues dans la requête de service.

5

La figure 1, déjà commentée, représente une architecture conforme à l'état de l'art permettant la sécurisation de communications entre applications.

La figure 2 représente une architecture générale conforme à l'invention.

10

La figure 2 schématise un exemple d'architecture conforme à l'invention. Les références A_1 , A_2 et A_3 représentent trois applications logicielles. Ces applications peuvent émettre trois requêtes de service, r_1 , r_2 et r_3 respectivement, au serveur virtuel de sécurité V afin de sécuriser les communications qu'elles désirent initier.

15

Selon une mise en œuvre de l'invention, ces requêtes contiennent uniquement le type de services désiré (confidentialité, authentification...).

Le serveur virtuel de sécurité a alors pour tâche de déterminer le ou les serveurs de sécurités parmi l'ensemble disponible S_1 , S_2 et S_3 , qui permettent de fournir le service demandé. Pour cela, le serveur virtuel V peut posséder un moyen de mémorisation permettant d'associer les services de sécurité demandés par les applications et les serveurs de sécurité fournissant ces services.

25

Dans l'exemple de la figure 1, la requête de service r_1 engendre une requête c_1 entre le serveur virtuel V et le serveur de sécurité S_2 . Cette requête peut être composée d'un échange de messages entre les deux participants. Cet échange dépend du serveur de sécurité S_2 et du protocole de négociation qu'il met en œuvre. On voit que cette architecture masque le protocole de

5 négociation en question pour l'application qui n'a donc pas besoin de s'en préoccuper.

Par ailleurs l'application A_2 émet au serveur virtuel de sécurité V une requête de service r_2 . Cette requête va être interprétée par le serveur virtuel de
5 la même façon que précédemment, mais ici la réalisation du service demandé nécessite deux serveurs de sécurité, S_1 et S_3 , avec lesquels le serveur virtuel initie deux requêtes c_2 et c'_2 .

Selon l'architecture conforme à l'invention, les applications (A_1 , A_2 et
10 A_3) peuvent n'avoir aucune information sur les différents serveurs de sécurités (S_1 , S_2 , S_3). Comme évoqué précédemment, cela simplifie les communications entre les applications puisque celles-ci sont déchargées de la gestion des différents protocoles de négociation, ainsi que du choix du ou des serveurs de sécurité appropriés au service requis. Il résulte de cela une simplification du
15 développement des applications, et donc une réduction du coût associé.

Par ailleurs, il est courant d'ajouter un serveur de sécurité au sein d'un système de gestion de réseaux, ou d'en remplacer un par un autre. Dans une architecture selon l'état de l'art, cet ajout ou ce remplacement peut résulter sur
20 un nouveau protocole à prendre en compte par les applications, ou en tout cas par un nouveau serveur à prendre en compte pour remplir les services de sécurité pouvant être requis. Cela implique donc nécessairement la modification de toutes les applications concernées.

Au contraire, dans une architecture selon l'invention, seul le serveur
25 virtuel doit être adapté. Bien évidemment, le coût engendré par la modification d'un unique composant logiciel bien identifié est bien moindre que celui qui serait nécessaire pour modifier tout un ensemble de composants logiciels hétérogènes.

De plus, l'architecture selon l'invention permet l'utilisation de plusieurs serveurs de sécurité pour réaliser un même service de sécurité. Cette utilisation est totalement transparente pour les applications. Il est ainsi aisément possible de rendre disponibles des services de sécurité de haut niveau, et de tirer profit
5 des capacités des serveurs de sécurité pour rendre un service qu'ils ne peuvent pas offrir individuellement.

Par ailleurs, c'est au serveur virtuel que revient la charge de gérer la localisation physique des différents serveurs de sécurité ainsi que le moyen
10 d'accès correspondant : accès direct ou par un bus logiciel par exemple.

Enfin, les applications peuvent n'avoir aucune information sur les serveurs de sécurité autres que le serveur virtuel V. Par conséquent, elles peuvent n'avoir aucune information sur les protocoles de négociation mis en
15 œuvre, et les attaques malveillantes du système n'en seront que plus difficiles.

Selon une mise en œuvre de l'invention, les applications logicielles peuvent accéder aux fonctions du serveur virtuel au travers d'une interface de programmation ou API (*Application Programming Interface*).
20

Selon une mise en œuvre particulière de l'invention, le serveur virtuel de sécurité est développé en un langage de programmation tel que Java, afin de le rendre indépendant du système d'exploitation sous-jacent. En effet, ainsi développé, le serveur virtuel de sécurité devient à même de fonctionner sur
25 n'importe quel type de système d'exploitation à condition qu'une machine virtuelle Java soit insérée entre ce système d'exploitation et ce serveur.

REVENDEICATIONS

5 **1)** Serveur virtuel de sécurité permettant à un ensemble d'applications logicielles (A_1, A_2, A_3) d'accéder à un ensemble de services de sécurité, caractérisé en ce qu'il possède :

- des moyens pour recevoir de la part desdites applications logicielles des requêtes de services,
- 10 • des moyens de communication avec des serveurs de sécurité (S_1, S_2, S_3) mettant en œuvre lesdits services de sécurité,
- des moyens pour choisir au moins un serveur de sécurité destinataire pour chaque requête de service reçue et pour lui transmettre, via lesdits moyens de communication, des informations lui permettant de fournir le service de sécurité
- 15 correspondant à ladite requête de service, certaines au moins de ces informations étant contenues dans ladite requête de service.

2) Système selon la revendication précédente, dans lequel lesdites applications logicielles peuvent accéder audit serveur virtuel au travers d'une

20 interface de programmation ou API.

3) Système selon l'une des revendications précédentes, dans lequel ledit serveur virtuel de sécurité est développé en un langage de programmation tel Java.

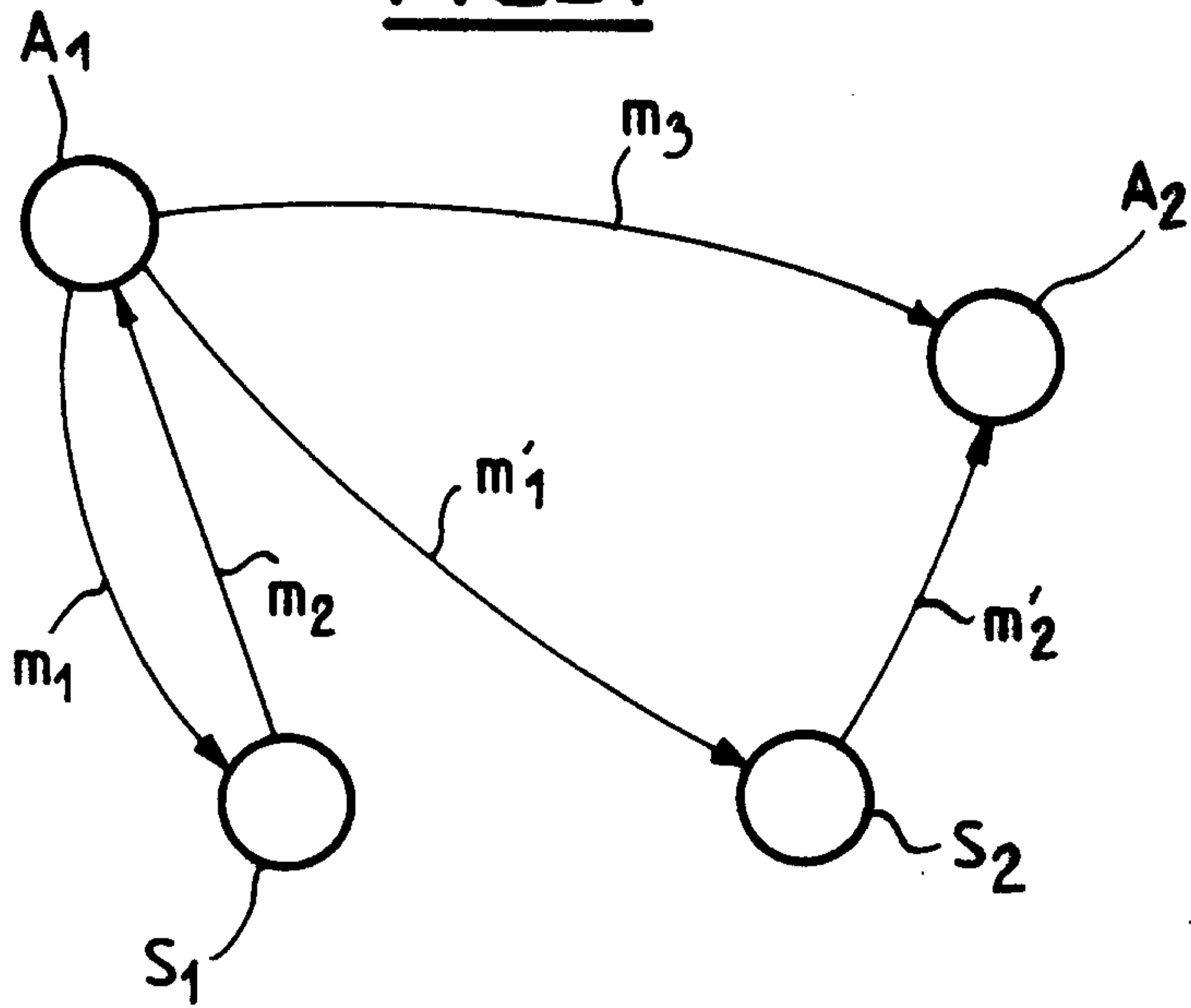
25

4) Procédé pour permettre à une application logicielle d'accéder à un service de sécurité, caractérisé en ce qu'il comporte des étapes de :

- émission de requêtes de service par ladite application logicielle à destination d'un serveur virtuel de sécurité,

- 5
- choix, par ledit serveur virtuel de sécurité, d'un serveur de sécurité destinataire, à même de fournir le service de sécurité correspondant à ladite requête de service,
 - émission d'informations à destination dudit serveur de sécurité destinataire afin qu'il puisse fournir ledit service de sécurité, certaines au moins de ces informations étant contenues dans ladite requête de service.

FIG_1



FIG_2

