
Une architecture de gestion efficace du contrôle d'accès

Olivier PAUL^{*}, Maryline LAURENT, Sylvain GOMBAULT

*Département RSM, ENST de Bretagne, 2 rue de la châtaigneraie - BP 78, 35512 CESSON Cedex - France
Tel: (33) (0) 299127051, Fax: (33) (0) 299127030
{paul/mlaurent/gombault}@rennes.enst-bretagne.fr*

Résumé. Le développement d'architectures de contrôle d'accès distribuées soulève le problème de leur gestion. Dans cet article, nous décrivons une architecture de configuration du contrôle d'accès pour les réseaux de type IP sur ATM. Cette architecture est basée sur des agents distribués sur les équipements de sécurité du réseau. Le problème principal que pose la gestion du contrôle d'accès est l'efficacité de la configuration des mécanismes de contrôle d'accès. Ce problème est présenté et nous donnons des règles permettant d'apporter une réponse.

Mots clef. Agents, Gestion, Distribution, Sécurité, Contrôle d'accès.

1 INTRODUCTION

Avec le développement d'Internet, la sécurité en général et le contrôle d'accès en particulier deviennent des problèmes essentiels dans le développement des réseaux de communication. C'est la raison pour laquelle beaucoup de travaux ont été réalisés afin de produire des outils permettant de fournir le service de contrôle d'accès pour la plupart des types de réseaux. Ces travaux, principalement effectués au sein de groupes de travail sur la sécurité dans certaines entreprises et dans le monde académique ont conduit au développement de l'outil de contrôle d'accès le plus courant de nos jours: le firewall.

En comparaison avec les efforts fournis pour développer les mécanismes de contrôle d'accès, peu de travaux concernant la gestion de ces mécanismes ont été présentés. Le paragraphe 2 montre que cette gestion devient de plus en plus complexe lorsque ce type d'équipement se multiplie sur un réseau et montre les solutions qui ont été proposées afin de résoudre ce problème.

Nous décrivons ensuite une architecture conçue pour gérer de manière automatique des équipements de contrôle d'accès distribués dans un réseau. Bien que notre approche s'adresse à un type de réseau particulier (IP sur ATM) elle pourrait facilement s'appliquer, avec des modifications mineures, à d'autres types de réseaux. Cette architecture se base sur l'utilisation d'agents distribués sur les équipements de contrôle d'accès. Afin de configurer ces agents, deux problèmes principaux doivent être résolus. Le premier est l'expression de la politique de contrôle d'accès. Une solution à ce problème est exposée dans le paragraphe 3 où nous définissons un langage de description du contrôle d'accès. Le second problème est l'application efficace des règles de contrôle d'accès. Nous expliquons dans le para-

*. Ce travail est financé par une bourse DRET.

graphe 4 comment ce problème peut être résolu au moyen d'informations locales et d'interactions entre les agents. Pour conclure, nous présentons dans le paragraphe 5 une comparaison de notre architecture avec les autres approches du domaine et montrons quelles pistes intéressantes s'offrent pour la poursuite de notre travail.

2 LA GESTION DU CONTRÔLE D'ACCÈS

2.1 Méthodes traditionnelles

Les progrès en terme de débit et de qualité de service dans les réseaux provoquent le besoin de nouvelles architectures de contrôle d'accès s'adaptant mieux à ceux-ci. Certaines de ces architectures sont basées sur la distribution des mécanismes de contrôle d'accès dans les réseaux à protéger afin de fournir de meilleures performances et une meilleure capacité à s'adapter aux facteurs d'échelle ([Schu98]). La figure 1 décrit les approches actuellement utilisées pour la gestion de mécanismes de ce type. Dans ces approches, l'officier de sécurité définit une politique de contrôle d'accès pour chacun des modules et configure ensuite les modules. Cette configuration peut se faire:

- De manière directe, en se connectant à chacun des équipements et en configurant chaque module à la main (opération décrite par les flèches en pointillé). Cette méthode est intéressante quand un seul équipement contenant tous les modules doit être configuré mais devient inutilisable lorsque le nombre d'équipements augmente.
- De manière indirecte en utilisant une plate-forme d'administration du contrôle d'accès (opération décrite par les flèches pleines). Cet outil permet de réaliser l'administration de plusieurs modules de contrôle d'accès à partir d'un seul équipement offrant généralement une interface générique à l'officier de sécurité. Cependant la décision de distribuer une partie de la politique de contrôle d'accès sur un module est prise par l'officier de sécurité et non par la plate-forme d'administration. De plus la généricité de l'interface fournie est généralement limitée par le fait qu'elle s'adresse à un ensemble de modules produits par un même constructeur et est donc de ce fait propriétaire.

Dans les deux cas l'officier de sécurité est responsable de la distribution de la politique de contrôle d'accès.

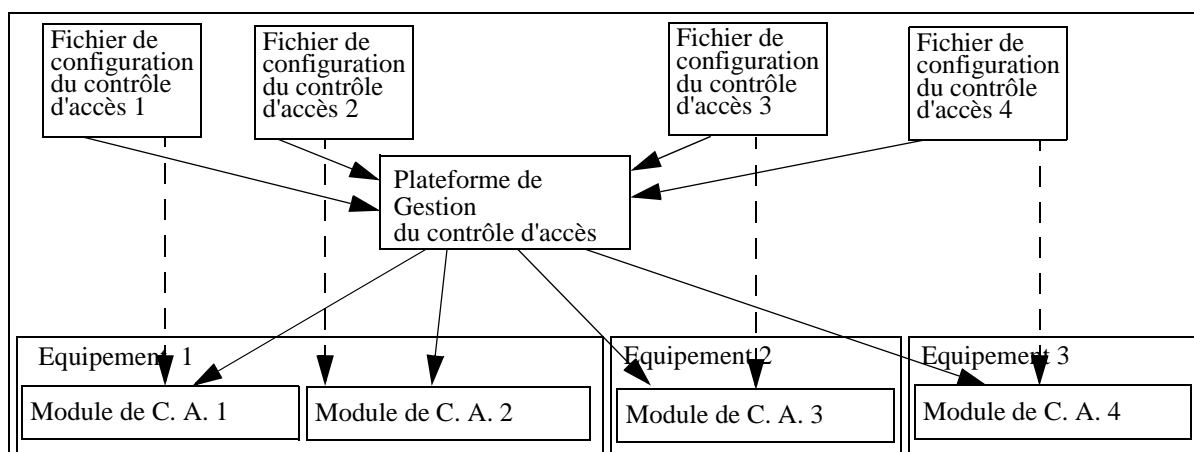


Figure 1 : Architectures traditionnelles de configuration du contrôle d'accès.

Le second problème posé par la gestion d'architectures distribuées est l'hétérogénéité des langages de configuration. Celle-ci est combinée au fait que le terme "firewall" est une expression générique qui désigne des mécanismes divers et variés fournissant plusieurs types de services de contrôle d'accès. La configuration de ces mécanismes se fait géné-

ralement au moyen d'interfaces propriétaires. La figure 2 et la figure 3 montrent comment une simple règle de contrôle d'accès peut être exprimée au moyen de commandes relativement différentes en fonction de l'équipement auquel elle est destinée (Un routeur Cisco et une station Linux dans notre cas).

Dans le cas du routeur on commence par autoriser les demandes de connexion TCP provenant de la station d'adresse 192.165.203.5 avec un port source supérieur à 1023 vers le port WWW (port 80) de n'importe quelle station. On autorise ensuite les paquets de requêtes correspondant à la connexion établie. On autorise enfin dans le sens du retour les paquets correspondant à des réponses. Il faut noter que les paquets de demande de connexion dans la direction serveur vers client sont naturellement bloqués. Cette propriété est due au fait que la politique de contrôle d'accès des routeurs Cisco est de type "Tout ce qui n'est pas explicitement autorisé est interdit."

Dans le cas de la station linux il n'est pas possible de désigner les paquets correspondant à une connexion établie. On est donc amené à interdire les paquets de demande de connexion dans le sens du retour (ligne 2) de manière explicite.

```
access-list 101 permit tcp 192.165.203.5 0.0.0.0 gt 1023 any eq 80
access-list 101 permit tcp 192.165.203.5 0.0.0.0 gt 1023 any eq 80 established
access-list 102 permit tcp any eq 80 192.165.203.5 0.0.0.0 gt 1023 established
```

Figure 2 : Règles de contrôle d'accès pour un routeur Cisco

Ces différences rendent la tâche d'administration de l'officier de sécurité plus difficile par le fait qu'elles peuvent provoquer l'introduction d'erreurs pouvant être exploitées par des attaquants dans les fichiers de configuration. De plus l'hétérogénéité oblige l'officier de sécurité à passer un temps important dans l'apprentissage de ces langages, temps qui pourrait être utilisé de manière plus utile d'un point de vue de la sécurité.

```
ipfwadm -F -a accept -b -P tcp -S 192.165.203.5 1024:65535 -D 0.0.0.0/0 80
ipfwadm -F -a deny -b -P tcp -S 0.0.0.0/0 80 -k -D 192.165.203.5 1024:65535
ipfwadm -F -a accept -b -P tcp -S 0.0.0.0/0 80 -D 192.165.203.5 1024:65535
```

Figure 3 : Règles de contrôle d'accès pour une station Linux utilisant *ipfwadm*

Ces deux aspects montrent qu'une configuration efficace d'outils de contrôle d'accès distribué devient de plus en plus difficile. Il est donc nécessaire de rechercher de nouvelles méthodes rendant cette gestion aussi simple que possible.

2.2 Autres propositions

Les problèmes présentés dans le paragraphe précédent ont donné lieu à un certain nombre de travaux destinés à faciliter la gestion de la sécurité en général et le contrôle d'accès en particulier dans le cadre de mécanismes répartis.

Le projet SAMSON.

Par exemple [Sam95] qui résume les résultats du projet européen SAMSON (Security and Management Services in Open Networks) décrit une architecture de gestion de la sécurité. Cette architecture développe plusieurs idées intéressantes. La première est d'offrir à l'officier de sécurité une interface générique pour la gestion des différents services de sécurité (authentification, contrôle d'accès sur les équipements, audit, gestion des clefs) et des différents mécanismes afin de résoudre le problème d'hétérogénéité. Un autre aspect intéressant est la possibilité d'utiliser plusieurs protocoles de gestion (CMIP et SNMP) de manière simultanée. Cependant le processus de configuration des équipements reste manuel puisque chaque module doit être configuré par l'officier de sécurité au travers de l'interface générique.

Le projet WILMA.

[Ft98] décrivant un travail réalisé au sein du groupe de travail WILMA de l'université technique de Munich propose une formalisation de la politique de contrôle d'accès dans un réseau et présente l'idée de configuration automatique

du contrôle d'accès. Le processus de configuration automatique est basée sur la topologie du réseau et les capacités de contrôle d'accès des équipements. Cependant les questions d'implémentation ne sont pas abordées par ce papier et aucune implémentation, interface ou algorithme n'est présenté.

Le protocole PFIP.

[Hyl98] reprend l'idée d'une interface générique et suggère que les règles de contrôle d'accès soient diffusées dans le réseau à la manière des informations de routage. Les auteurs présentent comme exemple un protocole de distribution des règles de contrôle d'accès pour les routeurs filtrants (PFIP, Packet Filter Information Protocol). Cette approche est intéressante car elle présente une méthode distribuée de configuration d'équipements eux mêmes distribués. Le protocole PFIP permet aux équipements de contrôle d'accès d'adapter les règles de contrôle d'accès qu'ils appliquent en fonction de la topologie du réseau. Les auteurs donnent également des informations sur une implémentation éventuelle du protocole. Cependant la méthode d'optimisation de la configuration des équipements vise à optimiser l'utilisation du réseau et non le processus de contrôle d'accès lui même.

Plusieurs aspects intéressants sont développés dans les travaux précédents. Ceux-ci concernent la distribution automatique de la politique de contrôle d'accès, l'optimisation de la configuration et la notion d'interface générique. Cependant il nous semble que certaines améliorations sont possibles. La suite de cet article porte donc sur la définition d'une architecture de gestion du contrôle d'accès synthétisant ces points intéressants. Nous montrons de quelle manière ils peuvent être implémentés et fournissons certaines améliorations. Ces travaux ont été réalisés dans le cadre de recherches sur le contrôle d'accès dans les réseaux de type IP sur ATM ([PL99]). Ces recherches nous ont amené à concevoir une architecture de gestion de mécanismes distribués de contrôle d'accès. Les aspects de distribution ne peuvent être développés qu'une fois la notion d'interface générique introduite. Nous présentons donc dans le paragraphe suivant l'interface générique que nous avons choisi. Nous décrirons dans le paragraphe 4 comment cette interface peut être utilisée dans notre architecture de configuration du contrôle d'accès.

3 EXPRESSION DE LA POLITIQUE DE CONTRÔLE D'ACCÈS

Afin de permettre l'expression de notre politique de contrôle d'accès nous définissons un Langage de Définition de Politique de Contrôle d'Accès (LDPCA). La définition du LDPCA est basée sur le Langage de Description de Politique (PDL) en cours de définition au sein du groupe de travail travaillant sur les politiques à l'IETF. Dans notre langage une politique est définie par un ensemble de règles, chaque règle étant elle même constituée d'un ensemble de conditions et d'une action qui est exécutée lorsque l'ensemble des conditions est rempli. L'expression suivante (exprimée dans le formalisme Backus Naur) décrit la forme générale d'une règle:

```
Rule ::= IF <Conditions> THEN <Action>
```

Toutes les conditions ont la même structure générique exprimée ci-dessous au moyen du formalisme BNF:

```
Condition ::= <ACCESS CONTROL PARAMETER> <RELATIONAL OPERATOR> <VALUE>
```

En fonction du niveau dans la pile de protocole, plusieurs types de paramètres de contrôle d'accès peuvent être utilisés:

- Au niveau ATM les paramètres intéressants sont décrits dans [PLG98]. Parmi ceux-ci nous avons choisi le type de trafic, les identificateurs de connexion, les informations d'adressage, les descripteurs de QoS et les descripteurs de service.
- Au niveau transport les paramètres que nous avons considérés sont ceux utilisés habituellement afin de réaliser le filtrage des paquets dans les routeurs filtrants (informations d'adressage, les ports source et destination,...).
- Au niveau application nous définissons deux paramètres génériques: l'identificateur de l'utilisateur de l'application ainsi que l'état de l'application.

- Des informations temporelles ont également été incluses afin de spécifier lorsqu'une règle doit être appliquée.

Les actions ont également une structure générique (notation BNF):

Action ::= <ACTION> <ACTION LEVEL> <LOG LEVEL>

Celle-ci se décompose en trois parties. La première indique si la communication décrite par les conditions doit être autorisée ou interdite. Le paramètre <ACTION LEVEL> correspond à la couche protocolaire dans laquelle doit être effectuée l'action. La dernière partie décrit l'importance accordée à l'évènement de contrôle d'accès et permet la classification des résultats.

La figure 4 montre comment notre langage peut être utilisé afin d'exprimer un service de contrôle d'accès comparable à celui décrit dans la figure 2 et la figure 3. Dans cet exemple chaque équipement est identifié par son adresse source et son adresse destination. Le service WWW est identifié par les ports sources et destination. La deuxième ligne de commande permet d'interdire les demandes de connexion sur le port lié au client WWW de notre station interne en n'autorisant que les paquets ne possédant par le drapeau SYN.

```

IF (IP_SRC_ADDRESS = 192.165.203.5 255.255.255.255) AND (IP_DST_ADDRESS = 0.0.0.0
0.0.0.0) AND (SRC_PORT > 1023) AND (DST_PORT = 80) THEN PERMIT TRANSP_CONNECTION
LEVEL1;
IF (IP_SRC_ADDRESS = 0.0.0.0 0.0.0.0) AND (IP_DST_ADDRESS = 192.165.203.5
255.255.255.255) AND (SRC_PORT = 80) AND (DST_PORT > 1023) AND (TCP_FLAG <> SYN) THEN
PERMIT TRANSP_CONNECTION LEVEL1;

```

Figure 4 : Exemple de règles de contrôle d'accès

Une fois définie au moyen du LDPCA, la politique de contrôle d'accès doit être distribuée sur les équipements de gestion au moyen de notre architecture de gestion du contrôle d'accès.

4 DISTRIBUTION DE LA POLITIQUE DE CONTRÔLE D'ACCÈS

4.1 Fonctionnement général

Notre architecture est basée sur l'utilisation d'agents se plaçant sur les équipements fournissant le service de contrôle d'accès. Ces agents interagissent avec les équipements sur lesquels ils sont placés afin de configurer les mécanismes de contrôle d'accès de la manière la plus performante possible. La figure 5 présente ces interactions.

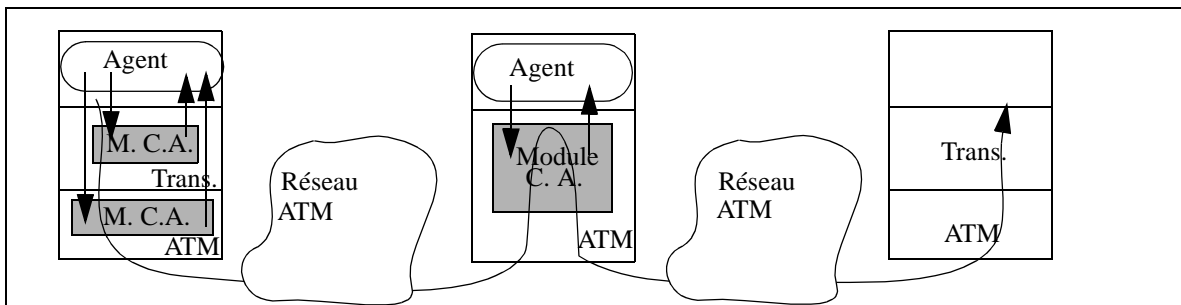


Figure 5 : Architecture de gestion du contrôle d'accès.

L'objectif général de l'application des règles spécifiées par la politique de contrôle d'accès au niveau des équipements

de contrôle d'accès est qu'un nombre minimal de règles de contrôle d'accès soit appliquées sur chaque équipement. Pour cela nous définissons un ensemble d'assertions qui assure cette propriété lorsqu'elles sont appliquées:

- (Assertion 1): Une règle ne peut être attribuée qu'à des modules de contrôle d'accès contenant les objets de contrôle d'accès utilisés par la règle.
- (Assertion 2): Une règle ne peut être attribuée à un module contenu dans un équipement que si celui-ci est situé sur le chemin entre la source et la destination décrit par la règle.

Une meilleure connaissance du type de politique de contrôle d'accès permet de donner des propriétés supplémentaires. Par exemple dans le cas d'une politique de type «Tout ce qui n'est pas explicitement autorisé est interdit», qui est le type de politique le plus répandu, il est possible de ne spécifier les interdictions qu'en un point du chemin qui interconnecte la source et la destination relatifs à une règle. Ceci se justifie par le fait que dans ce type de politique, chaque règle d'interdiction décrit un sous ensemble d'un ensemble décrit par une règle d'autorisation. De plus afin d'assurer une distribution maximale des règles et du fait de la structure généralement arborescente des réseaux, il est important que le placement des règles se fasse sur les équipements les plus proches des équipements source et destination décrits par chaque règle.

- (Assertion 3): Si une règle d'interdiction peut être attribuée à des modules situés sur plusieurs équipements en cascade, la règle ne doit être attribuée qu'au module situé sur l'équipement le plus proche des extrémités.

La traduction de ces lois se fait par la prise en compte de plusieurs paramètres: La topologie du réseau. Celle-ci varie en fonction des informations de routage, les capacités de l'équipement en terme de contrôle d'accès, la place de l'équipement dans la topologie du réseau et enfin la configuration des autres équipements de contrôle d'accès.

Ces paramètres impliquent des interactions entre les agents de configuration et d'autres éléments du réseau. Certaines de celles-ci, détaillées dans la section 4.2 sont internes à un équipement: entre l'agent de configuration et l'agent de routage correspondant ainsi qu'entre l'agent de configuration et les mécanismes de contrôle d'accès.

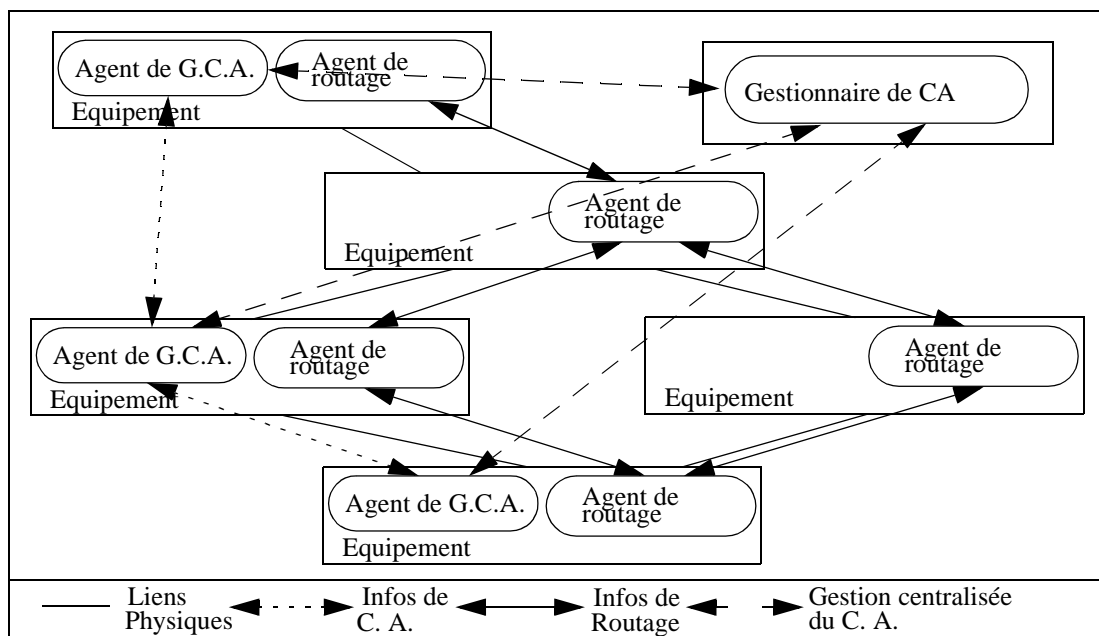


Figure 6 : Exemple de réseau et interfonctionnement des différents éléments.

D'autres interactions, détaillées figure 6, sont externes:

- Entre les agents de gestion du contrôle d'accès (G. C. A.) afin d'assurer une configuration efficace.
- Entre les agents et le gestionnaire centralisé de contrôle d'accès. Celui-ci assure la distribution uniforme de toutes les règles de contrôle d'accès sur les agents et la récupération des résultats de l'application de celles-ci. Le protocole utilisé entre les agents et le gestionnaire doit assurer l'intégrité, l'authentification, la confidentialité et le contrôle d'accès sur les informations transportées. Les protocoles SNMPv2* [Sta98] et SNMPv3 [Ba98] semblent être de bons candidats. Ils exigent cependant de gérer les informations de contrôle d'accès sous forme de bases de données de gestion (MIB) au niveau des agents. Ces informations comportent d'une part les règles de contrôle d'accès et d'autre part les résultats relatifs à l'application de ces règles.

Nous décrivons dans les sections suivantes comment ces règles peuvent être mises en oeuvre au moyen de notre architecture.

4.2 Architecture fonctionnelle

Comme on peut le voir figure 7, notre architecture se base sur l'utilisation de modules interagissant entre eux.

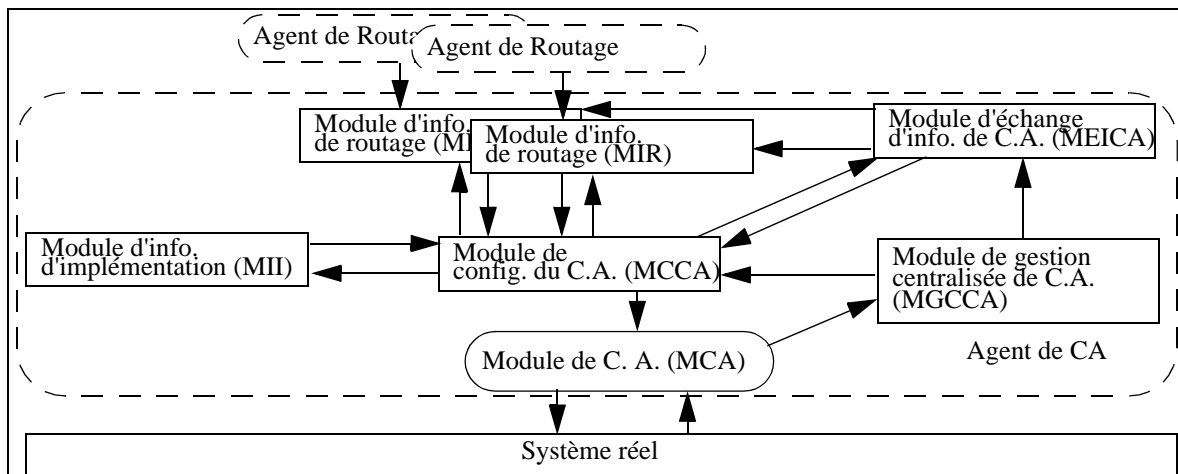


Figure 7 : Architecture fonctionnelle d'un agent de contrôle d'accès

Le module de contrôle d'accès (MCA).

Ce module représente les mécanismes de contrôle d'accès implémentés par l'équipement auprès de l'agent de contrôle d'accès. Ce module reçoit les règles de contrôle d'accès de la part du module de configuration du contrôle d'accès (MCCA). Ces règles sont traduites par le MCA dans les syntaxes réelles des commandes nécessaires à la configuration des mécanismes implémentés. En retour de ces commandes le MCA reçoit des résultats correspondant à l'application des commandes et transmet ces résultats au module de gestion centralisée du contrôle d'accès (MGCCA).

Le module d'information d'implémentation (MII).

Le MII décrit au MCCA les capacités de l'équipement sur lequel se situe l'agent en terme de contrôle d'accès. Pour cela il possède une table de correspondance entre les paramètres présents dans les règles de contrôle d'accès et les mécanismes implémentés au niveau du système. La configuration de cette table se fait par l'officier de sécurité de manière manuelle ou de manière automatique par le logiciel d'installation des outils de contrôle d'accès. Lors de modifications dans cette base de donnée, le MCCA est avertit par le MII.

Le module de gestion centralisée du contrôle d'accès (MGCCA).

Ce module a pour objectif de faire le lien entre l'agent et le gestionnaire centralisé de contrôle d'accès. Pour cela il gère une base de donnée de gestion (MIB) qui est mise à jour par le gestionnaire. Celui-ci distribue au MGCCA toutes les règles de la politique de sécurité. Le MGCCA avertit le MCCA de ces mises à jour en lui fournissant toutes les règles modifiées. En retour de celles-ci, le MGCCA reçoit les résultats correspondants de la part du MCA. Ces résultats sont stockés de manière incrémentale dans une table de la MIB et sont récupérés par le gestionnaire de manière périodique.

Le module d'échange d'informations sur le contrôle d'accès (MEICA).

Le MEICA a pour objectif de répondre à la question suivante du MCCA: «Existe-t'il un équipement mieux positionné qui applique déjà la règle r ?». Afin de répondre à cette question, le MEICA interagit avec les MEICAs des agents voisins au moyen du protocole d'échange d'informations de contrôle d'accès. Ce protocole permet l'élection d'un agent choisi pour appliquer la règle de contrôle d'accès. Pour cela, le MEICA envoie à chaque changement de configuration les informations suivantes à ses voisins: Identificateur de la règle, Adresse de l'agent appliquant la règle, Distance. Pour chaque règle r, le MEICA compare les informations fournies par ses voisins avec les siennes et en déduit si l'agent qu'il représente se trouve en position optimale afin d'appliquer la règle. Une position optimale signifie que la valeur minimale des distances entre la source ou la destination relative à la règle et notre MEICA est la plus faible. Cette distance est donnée par le MIR. Dans le cas de changements de topologie ou dans la configuration des équipements de contrôle d'accès provoquant une modification des informations de contrôle d'accès, le MEICA avertit le MCCA afin que celui-ci revoit sa configuration.

Le module d'informations de routage (MIR).

Le module MIR a un fonctionnement qui peut varier énormément en fonction du protocole de routage qui lui est associé. Cependant les fonctions qu'il remplit sont toujours les mêmes. D'une part il est utilisé par le MCCA afin de savoir si la communication relative à une règle r passe par l'équipement sur lequel est installé l'agent. D'autre part le MIR est chargé par le MEICA de calculer la distance entre l'équipement supportant l'agent et un équipement y. Enfin lors de modifications topologiques, le MIR avertit le MCCA afin que celui-ci prenne en compte celles-ci.

La figure 6 ne propose qu'une vision planaire de notre réseau qui ne correspond pas forcément à la réalité. Dans le cas de réseaux où le routage peut se faire à plusieurs niveaux (Par exemple du type IP sur ATM) il est nécessaire de considérer le fait que les informations de routage se rapportant aux règles peuvent provenir d'agents de routages différents, ce qui explique que le MCCA puisse être en relation avec plusieurs MIRs, chacun étant dédié au routage d'une couche particulière.

Le module de configuration du contrôle d'accès (MCCA).

Le module MCCA est le module central de notre architecture. Il est averti de l'arrivée de chaque nouvelle règle de contrôle d'accès par le MGCCA. Pour chacune de ces règles, il va appliquer les trois assertions définies dans la section précédente. Pour cela il interagit avec les modules MII, MIR, MEICA. L'algorithme de fonctionnement général de ce module est le suivant. Pour chaque règle reçue, le MCCA vérifie si celle-ci peut être appliquée à l'équipement au moyen du MII (assertion 1). Il vérifie ensuite au moyen du MIR que l'équipement peut se situer sur une des routes décrites par la règle (assertion 2). Si la règle de contrôle d'accès est une règle d'interdiction il s'adresse alors au MEICA afin de savoir si un équipement plus proche de la source ou de la destination décrite par la règle l'applique déjà (assertion 3). Les règles n'exprimant pas de notion de source et de destination sont appliquées sans cette vérification. Si toutes ces conditions sont vérifiées, le MCCA passe alors la règle au MCA afin que celui-ci l'implémente.

4.3 Fonctionnement détaillé du MIR

Dans cette partie nous détaillons le fonctionnement du module MIR dans le cas de son utilisation avec le protocole RIP (Routing Information Protocol). Le lecteur intéressé par une description de RIP peut consulter [Tou99].

Configuration initiale.

On suppose que le processus de routage a démarré avant le processus de gestion du contrôle d'accès. Le module MIR contient une table `NRoute` contenant les dernières tables de routage données par les équipements voisins:

- Adresse du voisin (*av*), Adresse IP (*a*), Masque de sous réseau (*m*), Métrique (*d*).

Il contient également une table `Route` contenant la table de routage et ayant le format suivant:

- Adresse IP (*a*), Masque de sous réseau (*m*), Métrique (*d*).

Fonctionnement ultérieur.

- Réception de la commande `comm(rule r)` du MCCA. Par cette demande, le MCCA cherche à savoir si la règle *r* décrit une communication passant par l'équipement sur lequel se place l'agent. Pour répondre on utilise le fait qu'une communication passe par un équipement si celui-ci se trouve sur le chemin le plus court entre la source et la destination.

```
s = Source(r);
d = Destination(r);
Si (s = NIL et d = NIL) alors Renvoyer(Oui);
Sinon
  Soit a/Appartient(s,Route[a].a,Route[a].m);
  Soit b/Appartient(d,Route[b].a,Route[b].m);
  Pour tout i/Appartient(s,NRoute[i].a,NRoute[i].m) faire
    Pour tout j/Appartient(d,NRoute[j].a,NRoute[j].m) faire
      Si (NRoute[i].d + NRoute[j].d) < (Route[a].d + Route[b].d) Alors Renvoyer(Non);
  Renvoyer(Oui);
```

- Réception de la commande `distance(address y)` de la part du MEICA. On renvoie la distance entre l'équipement supportant l'agent et l'équipement d'adresse *y*.

```
Soit i/Appartient(y,Route[i].a,Route[i].m);
Renvoyer(Route[i].d);
```

- Modification des tables `NRoute` ou `Route` par l'Agent de routage. On signale cette modification au MCCA.

```
res = MCCA.rtopologych();
```

Les algorithmes précédents utilisent une fonction externe `Appartient(s, a, m)` qui indique si l'adresse ou l'ensemble d'adresses décrit par *s* est inclus dans le réseau *R* décrit par l'adresse *a* et le netmask *m* et qu'il n'existe pas de réseau *R'*, sous réseau de *R* décrit par *a'* et *m'* tel que `Appartient(s, a', m')`.

Il faut noter que RIP est un protocole simple qui n'utilise que des informations d'adressage. L'utilisation d'un protocole prenant d'autres types d'informations en compte (p.e. PNNI) rendrait le fonctionnement du MIR plus complexe.

5 CONCLUSION

En conclusion, le tableau 1 compare les différentes approches pour la distribution d'une politique de contrôle d'accès.

Propriété/Méthode	A la main	Outil Propr.	Projet Samson	Hyl & al.	Projet Wilma	Paul & al.
Interface Générique	Non	Oui	Oui	Oui	Oui	Oui
Système propriétaire	Oui	Oui	Non	Non	Non	Non
Support de protocoles de gestion multiples	Non	Non	Oui	Non	Non	Non
Gestion multi-couches	Non	Non	Non	Non	Non	Oui
Décision de distribution	Central.	Central.	Central.	Dist.	Central.	Dist.
Méthode de distribution	Manuel	Manuel	Manuel	Les deux	Auto.	Auto.
Optimisation basée sur la topologie et les capacités de contrôle d'accès	Non	Non	Non	Non	Oui	Oui
Optimisation basée sur les règles	Non	Non	Non	Non	Non	Oui
Méthodologie d'optimisation fournie	Non	Non	Non	Non	Non	Oui
Implémentation	Oui	Oui	Oui	Non	Non	Non

Tableau 1 : Comparaison des différentes méthodes de distribution

Comme on peut le voir, notre approche a les avantages suivants: Elle simplifie le processus de gestion de la sécurité en déchargeant l'officier de sécurité de la distribution des règles de la politique de sécurité et en réduisant l'apprentissage à un seul langage de contrôle d'accès. Le modèle utilisé permet l'utilisation d'outils de contrôle d'accès à plusieurs niveaux et prend en compte les possibilités de couches de routage superposées. Enfin elle permet de faire une distribution efficace des règles de contrôle d'accès en utilisant plusieurs méthodes complémentaires d'optimisation du placement des règles.

Ce travail pourrait être poursuivi de manière utile dans deux directions. D'une part il serait intéressant de tester notre architecture soit par simulation soit en l'implémentant. D'autre part il serait intéressant d'évaluer le facteur d'amélioration des performances apporté par notre méthode de distribution.

6 BIBLIOGRAPHIE

- [Ba98]: Basking in Glory-SNMPv3, Dan Backman, Network Computing, Août 1998.
- [Ft98]: Integrated Management of Network and Host Based Security Mechanisms, R. Falk, M. Trommer, 3rd Australasian Conference on Information Security and Privacy ACISP'98, Brisbane, Australia, 13.-15. Juillet 1998.
- [Hyl98]: Management of Network Security Application, P. Hyland, R. Sandhu, 1st NISS Conference, Octobre 1998.
- [PDL98]: Policy Framework Definition Language, draft-ietf-policy-framework-pfdl-00.txt, John Strassner, Stephen Schleimer, Internet Engineering Task Force, 17 Novembre 1998.
- [PL99]: An Alternative Access Control Architecture for IP over ATM Networks, Olivier Paul, Maryline Laurent, IFIP Conference on Communications and Multimedia Security, Leuven, Belgium, Septembre 1999
- [PLG98]: Manageable parameters to improve access control in ATM networks , Olivier Paul, Maryline Laurent, Sylvain Gombault, HP-OVUA Workshop, Rennes, France, Avril 1998.
- [Sam95]: Samson, Final Report, Michael Steinacker, RACE R2058 Project, Janvier 1995.
- [Schu98]: On the modeling , design and implementation of firewall technology, Christoph Schuba, Ph.D. Thesis, Purdue University, Décembre 1997.
- [Sta93]: SNMP, SNMPv2 and CMIP, The practical guide to network management Standards. W. Stallings. Addison-Wesley. 1993.
- [Tou99]: Réseaux locaux et internet, des protocoles à l'interconnexion, 2ème édition, Laurent Toutain, Editions Hermès, 1999.