ANR programme ARPEGE 2008

Systèmes Embarqués et Grandes Infrastructures

Projet SELKIS : Une méthode de développement de systèmes d'information médicaux sécurisés : de l'analyse des besoins à l'implémentation.

ANR-08-SEGI-018

Février 2009 - Décembre 2011 (extension jusqu'à fin août 2012)

Projet SELKIS ANR-08-SEGI-018 Modèles formels pour l'analyse et l'expression des exigences de sécurité

Livrable 1.1 - Mise à jour

Editeur : Institut Mines-Télécom / Télécom Bretagne

T0 + 36

Résumé

Le présent rapport constitue une mise à jour du livrable 1.1 de la tâche WP1 du projet SELKIS.
L'objectif de l'étude menée dans le premier chapitre de la version originale du livrable 1.1 était d'analyser les différentes normes qui peuvent être définies par un législateur. Ces normes s'appliquent à des informations qu'un système d'information peut communiquer à un agent. Pour cela la démarche consistait à analyser les effets qu'un acte de communication réalisé par un système d'information peut avoir sur un agent et à analyser les effets et actions que le législateur peut choisir de permettre ou d'interdire.
La technique qui a été utilisée pour mener cette analyse consistait à exprimer les notions pertinentes en logique modale. Ce travail a été finalisé et est présenté dans la première partie de ce livrable comme prévu et indiqué dans la version précédente.
La deuxième actualisation du livrable 1.1 concerne l'application des travaux menés dans le chapitre 3 au cas d'étude "Echo doppler" de SELKIS. Elle est résumée dans la deuxième partie de ce livrable. L'ensemble des travaux peuvent être retrouvés dans le manuscrit de la thèse menée dans le cadre du projet SELKIS au sein des deux équipes Télécom Bretagne évoluant dans le projet et soutenue en janvier 2012.

Modélisation de la confidentialité et de l'intégrité des systèmes d'information en termes d'agents institutionnels, humains et logiciels

Contents

Abstract

On considère un Système d'Information d'un point de vue global qui intègre les agents logiciels aux agents humains qui les utilisent et aux agents institutionnels qui sont représentés par les agents humains. Les notions de confidentialité et d'intégrité sont définies à l'aide des actions de communication : informer, insérer et supprimer, et des normes qui s'appliquent à ces actions. Les actions et les normes sont formalisées dans le cadre d'une logique multi modale propositionnelle qui permet d'exprimer les croyances explicites et implicites, les actions et les effets dont elles sont la cause, les interdictions et permissions et l'opérateur de "counts as" qui relie les faits bruts et les faits institutionnels.

Le formalisme utilisé vise à faire le lien entre les objectifs du "législateur" et les spécifications définies par les informaticiens qui réalisent les agents logiciels.

1 Introduction

Il y a peu de travaux qui prennent en compte les trois types d'agents qui interviennent dans le contexte des Systèmes d'Informations à savoir : les agents logiciels qui sont des outils informatiques, les agents humains qui utilisent ces outils et les agents institutionnels que représentent certains agents humains [7]. Il nous parait cependant essentiel de prendre en considération ces trois types d'agents lorsqu'on veut définir des normes qui ont pour but de contrôler l'utilisation de ces outils informatiques car ceux-ci sont toujours mis en oeuvre, de façon directe ou indirecte, par des agents humains. C'est en adoptant ce point de vue que nous allons analyser et formaliser les normes portant sur la confidentialité et l'intégrité des Systèmes d'Informations.

Dans la définition des normes nous avons recherché un équilibre entre le point de vue de ceux qui définissent les normes et qui s'intéressent principalement aux effets des actions portant sur la transmission ou la modification d'informations et le point de vue des informaticiens qui s'intéressent en premier à la réalisation des outils informatiques qui doivent satisfaire les normes.

Une façon d'approcher ce point d'équilibre consiste à exprimer les normes dans un langage suffisamment précis pour que les informaticiens puissent vérifier que les agents logiciels satisfont les propriétés désirées et qui soit indépendant des méthodes informatiques pour que ceux qui définissent les normes n'aient pas à rentrer dans des détails techniques. C'est la raison pour laquelle nous avons choisi la logique formelle pour représenter les actions

de communication et leurs effets, ainsi que les normes qui portent sur ces actions.

Pour cela nous présenterons d'abord de façon informelle les notions d'agents institutionnels, humains et logiciels (section 2). Ensuite nous présenterons la logique formelle qui sera utilisée par la suite. Il s'agit d'une logique multi modale dons nous présenterons le langage, la sémantique et une partie de l'axiomatique (section 3). Cette logique sera appliquée pour formaliser les actions de communication entre agents institutionnels du type : informer, insérer et supprimer (section 4), puis pour formaliser ces actions dans le cas des agents humains et logiciels (section 5). Ceci permettra de définir formellement les normes de confidentialité et d'intégrité qui s'appliquent à ces types d'actions (section 6). Les principaux résultats de l'étude seront récapitulés dans la conclusion où nous mentionnerons un certains nombre de questions laissées ouvertes compte tenu des limites de l'étude (section 7).

2 Agents institutionnels, humains et logiciels

Nous allons présenter dans cette section de façon informelle les notions qui seront utilisées par la suite. Les plus importantes par rapport aux questions auxquelles on s'intéresse sont les notions d'agent et d'action de communication.

On distingue trois types d'agents : agents institutionnels, agents humains et agents logiciels. La distinction entre agent institutionnel et agent humain correspond en droit à la distinction entre personne morale et personne physique. Il n'est pas nécessaire de détailler ce qu'on appelle "agent humain" ou "personne physique". Par contre la notion d'agent institutionnel doit être précisée. C'est ce que nous allons faire en utilisant les concepts proposées par John Searl dans [17].

Avant de définir la notion d'agent institutionnel nous allons rappeler la notion d'**institution** telle qu'elle est utilisée par Searl. Une institution est un ensemble de normes considérées comme formant un tout. Ces normes peuvent être, par exemple, des lois, des arrêtés municipaux, ou des réglementations internes d'une entreprise ou d'un hôpital. Ainsi, les réglementations relatives au commerce international constituent une institution, la législation concernant les systèmes de santé constitue une autre institution, les codes de déontologie des médecins constituent encore une autre institution.

Les **agents institutionnels**[1] sont définis en référence à une institution. C'est cette institution qui définit les propriétés d'un agent institutionnel ou d'un type d'agent institutionnel. Si on considère, par exemple, un agent institutionnel du type hôpital, c'est l'institution de référence qui définit, par exemple, comment est créé un hôpital et comment il cesse d'exister, quelles sont ses finalités, comment est constitué son conseil d'adminstration et quels sont ses pouvoirs, quelles sont ses responsabilités,...etc...

On peut dire que les agents institutionnels n'ont d'existence que "conventionnelle", il n'ont pas d'existence matérielle. Par exemple, ils n'ont pas de poids ni de couleur, sauf si ces notions étaient définies par l'institution. Par conséquent un agent institutionnel ne peut pas réaliser d'actions matérielles comme : envoyer une lettre.

Cependant l'institution peut définir une relation entre certaines actions matérielles réalisées par des agents humains et des actions réalisées par un agent institutionnel. Comme pour son existence, les actions que réalise un agent institutionnel ne sont définies que de façon purement conventionnelle. Par exemple, une norme de l'institution pourrait stipuler qu'une lettre envoyée par une personne qui est titulaire de tel rôle dans un hôpital et qui est rédigée sur un papier à en-tête de l'hôpital, "*compte comme*" une lettre envoyée par l'hôpital au regard de cette institution.

Cette relation entre deux actions est appelée "**counts as**" par Searle et sera largement utilisée par la suite. On la traduit en français par "*compte comme*". Elle permet de relier ce que Searle appelle des "*faits bruts*", par exemple, le fait que telle personne a envoyé telle lettre, et des "*faits institutionnels*", par exemple, le fait que l'hôpital a envoyé une lettre. Elle peut aussi relier un fait institutionnel à un autre fait institutionnel[2]. Les normes exprimées par la relation "counts as" sont appelées des "*normes constitutives*" par opposition aux "*normes régulatives*" qui définissent les droits et obligations des agents.

De la même façon que des normes constitutives permettent de définir les actions d'un agent institutionnel, il y a des normes constitutives qui définissent **les connaissances ou les croyances des agents institutionnels**. Comme ceux-ci n'ont pas d'existence matérielle il faut définir quels sont les

[1] Dans le langage courant on confond parfois "institution" et "agent institutionnel". Par exemple, on parle de l'institution du mariage pour parler de l'ensemble des lois qui régissent le mariage, mais on utilise aussi parfois le terme d'institution religieuse pour parler d'un agent institutionnel.

[2] Ce dernier point est contesté par certains auteurs.

faits matériels qui comptent au regard de l'institution comme des connaissances d'un agent institutionnel.

Par exemple, pour les connaissances relatives aux opérations financières on peut avoir une norme qui stipule que les informations qui figurent sur un livre de compte qui satisfait telles et telles caractéristiques comptent comme des connaissances de l'agent institutionnel à propos de ces opérations. Si on se pose la question : "est-ce que l'agent institutionnel sait que telle opération a été réalisée?", pour obtenir la réponse on devra examiner ce qui figure sur le livre de compte. De la même façon le texte d'un contrat d'embauche peut compter comme le fait que l'agent institutionnel sait qu'il a embauché telle personne, ou bien le fait que tel diagnostic figure dans le dossier de tel patient compte comme le fait que l'agent institutionnel sait que tel diagnostic a été établi pour ce patient.

Il se peut que le support matériel au lieu d'être sous forme de papier soit un support informatique. Dans ce cas ce sont des normes constitutives qui définiront de la même manière sur quel support et dans quel fichier figurent les données qui représentent tel type de connaissance de l'agent logiciel. Dans la suite on notera $memory(I, m)$ le fait que sur le support informatique qui, selon les normes, sert à mémoriser les connaissances de l'agent institutionnel I, il y a un message m dont la signification est définie par ces normes. Par exemple, dans un message relatif à un patient, il se peut que la présence d'un 1 signifie qu'il s'agit d'un homme, tandis que la présence d'un 0 signifie qu'il s'agit d'une femme. De façon similaire on suppose que les communications entre l'agent institutionnel I et des agents logiciels se font par l'intermédiaire d'une certaine zone d'un support informatique, défini par les normes, que l'on appelle "mailbox", et on notera $mailbox(I, m)$ le fait que le message m se trouve sur ce support.

On notera que les faits représentés matériellement ne correspondent pas toujours à la réalité. Dans ce cas on parlera de croyances de l'agent institutionnel au lieu de connaissances.

A coté des normes constitutives il peut y avoir des normes régulatives qui s'appliquent à un agent institutionnel. Ces normes peuvent stipuler, par exemple, que l'agent peut embaucher des employés et qu'il doit les payer selon les normes de la législation du travail, ou bien qu'il a obligation de mettre en place certains dispositifs pour garantir la sécurité.

On utilisera le terme "**agent humain**" pour parler de personnes dans le sens ordinaire. Un agent humain peut réaliser des actions physiques et certaines de ses attitudes mentales sont appelées connaissances ou croyances. A

la différence des agents institutionnels on ne peut pas accéder à leur représentation physique. D'autres part on peut avoir des normes constitutives ou régulatives qui s'appliquent à des agents humains.

Les normes constitutives peuvent être utilisées pour définir des **rôles**. Par exemple, on peut dire qu'une personne qui satisfait telles ou telles conditions compte comme un chirurgien, une autre qui satisfait d'autres conditions compte comme un directeur d'hôpital ou comme une infirmière.

A chaque rôle est associé un ensemble de normes régulatives qui définissent les droits et obligations des personnes qui sont titulaires de ce rôle. Par exemple, une personne titulaire du rôle de chirurgien peut avoir obligation de faire un compte rendu après chaque opération et il peut avoir droit à une certaine période de repos entre deux opérations. Les rôles ne sont qu'une commodité pratique pour parler de l'ensemble des normes qui s'appliquent à une personne [13]. En disant que telle personne "est" chirurgien on exprime qu'elle fait l'objet d'un ensemble de droits et obligations qu'il n'est pas nécessaire d'expliciter.[3]

Les relations entre les actions réalisées par des agents humains et celles réalisées par des agents institutionnels sont représentées, comme on l'a vu plus haut, par le "counts as". Par exemple, il se peut que :

le fait que la personne x qui est titulaire du rôle d'agent comptable de la clinique y a envoyé un chèque d'un montant z à la Caisse t

compte pour l'institution S comme

le fait que la clinique y a payé la somme z à la Caisse t

On notera que c'est le fait qu'une action ait été réalisée qui compte comme le fait qu'une autre action ait été réalisée, ce n'est pas, strictement parlant, une relation entre deux actions mais entre deux faits.

D'autre part, il se peut que la réalisation d'une action compte comme la réalisation d'une autre action pour une institution S, alors qu'elle ne compte pas pour cette action pour une autre institution S'. Par exemple, il se peut que si x paye la somme z en argent liquide, cela ne compte pas pour S comme si la clinique avait payé la somme z, alors que pour S' ça compte pour cette action.

[3] Dans le langage ordinaire on utilise parfois le terme "fonction" à la place de "rôle". Par exemple, on peut parler de la fonction de chef de service pour parler du rôle de chef de service. On peut dire aussi que telle personne occupe telle fonction pour dire qu'elle est titulaire de tel rôle.

On appelle **agent logiciel** un support informatique physique qui sert de représentation à un ensemble d'instructions et de données. Les actions que réalise l'agent logiciel sont des actions physiques qui correspondent à l'exécution de ces instructions dans cet environnement physique.

Les normes ne peuvent pas porter sur des agents logiciels parce que ce sont des objets physiques. Les agents logiciels ne sont que des instruments dont les actions sont "déclenchées" de façon directe, ou indirecte, par des agents humains. Ici on utilise le terme "déclenché" pour dire que c'est l'action d'un agent humain qui a eu pour effet d'initialiser l'action d'un agent logiciel, et l'effet de l'action de l'agent logiciel peut être d'initialiser l'action d'un autre agent logiciel.

Pour illustrer le fait que les agents logiciels ne peuvent pas faire l'objet de normes on peut faire l'analogie avec d'autres systèmes physiques qui sont utilisés par des agents humains.

Par exemple, le code de la route interdit aux voitures de rouler à plus de 130 km/h sur autoroute. En réalité cette interdiction s'applique aux conducteurs des voitures, et si la vitesse d'une voiture est déterminée par un régulateur automatique de vitesse, c'est le conducteur, en imposant une vitesse au régulateur, qui est la cause de la vitesse de la voiture, et c'est lui qui en est responsable par rapport au code de la route.

De même, les règles de la navigation aérienne interdisent aux avions de voler en dessous de certaines altitudes dans certaines zones. Cette interdiction s'applique aux pilotes des avions, pas aux avions eux-mêmes. Si un avion est piloté par un pilote automatique, c'est le pilote en affichant une altitude, ou une procédure de descente particulière sur le pilote automatique, qui est responsable de l'altitude de l'avion.

Enfin, si une personne utilise un pigeon voyageur pour porter à quelqu'un d'autre un message qui contient une information qu'il n'a pas la permission de transmettre, ce n'est pas le pigeon qui est responsable...

Par soucis de simplification on ne considère pas ici le cas où l'action d'un agent logiciel peut dépendre des actions de plusieurs agents humains. Par exemple, il se peut que l'action d'un agent humain ne puisse déclencher un agent logiciel pour accéder à certaines données que si un autre agent humain a préalablement déclenché l'action d'un agent logiciel qui donne au premier la possibilité d'accéder à ces données.

Les relations entre les actions réalisées par des agents logiciels et celles réalisées par des agents humains sont représentées elles aussi par le counts

as.

Par exemple, il se peut que :

le fait que la personne x qui est titulaire du rôle d'agent comptable de la clinique y a déclenché l'agent logiciel l qui a envoyé à la banque b l'ordre de payer la somme z à la Caisse t

compte pour l'institution S comme

le fait que la personne x qui est titulaire du rôle d'agent comptable de la clinique y a envoyé un chèque d'un montant z à la Caisse t

On notera qu'ici c'est le fait que l'action réalisée par l'agent logiciel ait été causée par l'agent humain qui justifie le fait que cette action compte comme une autre action de cet agent humain. Par contre, la justification du fait que l'action d'un agent humain compte comme l'action d'un agent institutionnel est justifiée par le fait que cet agent humain est titulaire d'un certain rôle.

Pour décrire les **actions de communication** on distinguera deux niveaux de représentation : un niveau physique qui est le niveau des **actes locutoires**, dans la terminologie de Searle, et un niveau abstrait qui est le niveau des **actes illocutoires**. Le niveau illocutoire définit le sens qui est donné à un acte locutoire.

Par exemple, au niveau locutoire les actions : *dire a un patient qu'il a un cancer*, *envoyer une lettre à un patient dans laquelle il est écrit qu'il a un cancer* et *envoyer un courrier électronique à un patient lui disant qu'il a un cancer* sont trois actions différentes pour réaliser un acte illocutoire qui informe le patient qu'il a un cancer.

De façon plus précise l'acte locutoire : *dire a un patient qu'il a un cancer* consiste à émettre des sons qui sont interprétés par le destinataire comme l'acte illocutoire *informer d'un cancer*. L'acte locutoire : *envoyer une lettre à un patient dans laquelle il est écrit qu'il a un cancer* consiste à envoyer une lettre sur laquelle apparaissent des caractères qui sont interprétés par le destinataire comme le même acte illocutoire. Enfin, l'acte locutoire : *envoyer un courrier électronique à un patient lui disant qu'il a un cancer* consiste à envoyer une suite de bits dans une certaine zone physique où elle est interprétée par le destinataire comme le même acte illocutoire.

On pourrait exprimer les relations entre les actes locutoires et illocutoires par la relation de counts as, mais dans ce cas l'institution à laquelle on ferait référence ne serait pas l'institution qui régit les systèmes de santé ou

le commerce, mais une institution qui établirait la signification des actes de communication locutoires. En fait, en général, cette institution n'a pas d'existence formelle et nous n'y ferons pas référence[4].

En ce qui concerne les actions de communication que l'on veut contrôler quand on s'intéresse à la confidentialité et à l'intégrité on considère que leur force illocutoire, dans la terminologie de Searle, et du type assertif.

Pour la protection de la confidentialité il s'agit d'actions de communication qui donnent au destinataire la possibilité d'avoir connaissance de la description d'une partie du monde. Ce sont des actions qui donnent au destinataire la possibilité de s'informer.

Pour la protection de l'intégrité il peut sembler dans un premier temps que la force illocutoire soit du type directif, si on considère que les actions ont pour objet de demander au destinataire de modifier la représentation qu'il se fait du monde. Mais en fait les actions qui doivent être réglementées sont celles qui changent effectivement cette représentation. Par exemple, il ne s'agit de permettre ou d'interdire à l'agent qui réalise l'acte de communication de *demander* au destinataire de modifier sa représentation du monde mais de permettre ou d'interdire à l'agent qui agit de modifier effectivement la représentation qu'en a le destinataire. Ce sont donc des actions qui informent au sens propre et donc des assertifs.

3 Cadre formel

Nous allons définir dans cette section le cadre formel dans lequel seront exprimées les notions auxquelles nous ferons référence pour définir les normes concernant la confidentialité et l'intégrité. On présentera d'abord la syntaxe du langage utilisé, puis sa sémantique qui explicite le sens que l'on donne aux différentes modalités et enfin l'axiomatique qui permet de dériver les conséquences d'un ensemble d'hypothèses (voir [3]).

3.1 Langage

Le langage est le langage du Calcul des Propositions Modal (LCPM).

On appelle LCP le langage du Calcul des Propositions classique.

[4]On peut considérer que les règles qui permettent de déchiffrer des messages cryptés constituent une amorce d'institution de ce type.

La grammaire de LCPM est définie par les règles suivantes.

1. Si ϕ est un formule de LCP, alors ϕ est dans LCPM.
2. Si M est un opérateur modal et ϕ est dans LCPM, alors $M\phi$ est dans LCPM.
3. Si ϕ et ψ sont dans LCPM, alors $\phi \Rightarrow_S \psi$ est dans LCPM.

Les modalités M du langage et leurs significations intuitives sont les suivantes.

$ExpBel_I\phi$: l'agent institutionnel I croit explicitement ϕ.

On utilisera la notation agt pour dénoter, selon les contextes, un agent i, ou un agent i qui réalise une action α (noté $i : \alpha$), ou un agent i qui réalise une action α en tant que titulaire du rôle r (noté $i : r : \alpha$).

$Bel_{agt}\phi$: l'agent agt croit que ϕ.

$Know_{agt}\phi$: l'agent agt sait que ϕ.

$Does_{agt:\alpha}\phi$: l'agent agt est sur le point de faire l'action α et après la réalisation de α on a ϕ.

$Done_{agt:\alpha}\phi$: l'agent agt vient de réaliser l'action α et on a ϕ.

$E^+_{agt}\phi$: l'agent agt est sur le point, en réalisant une certaine action, de faire en sorte que l'on ait ϕ.

$E_{agt}\phi$: l'agent agt, en réalisant une certaine action, a fait en sorte que l'on ait ϕ.

$Done^{-1}_{agt}\phi$: avant que l'agent agt commence de faire l'action qu'il vient de faire on avait ϕ.

$Int_{agt}\phi$: l'agent agt a l'intention d'être dans une situation où on a ϕ.

$Obg_S\phi$: dans le contexte de l'institution S il est obligatoire que ϕ.

$Perm_S\phi$: dans le contexte de l'institution S il est permis que ϕ.

$Forb_S\phi$: dans le contexte de l'institution S il est interdit que ϕ.

$\phi \Rightarrow_S \psi$: pour l'institution S ϕ compte comme ψ.

$D_S\phi$: pour l'institution S on a ϕ.

Pour les notions d'obligation, permission et interdiction voir [1, 16], pour les notions d'actions et de causalité voir [13, 11, 10, 18, 19, 20], pour les

notions de counts as voir [17, 12, 8, 9], pour les notions de rôle voir [13, 4, 6, 2, 15] et pour les notions d'institution et d'agent institutionnel voir [17, 14, 21, 7, 5].

Projection du Calcul des Prédicats dans le Calcul des Propositions.

Pour éviter la lourdeur de la formalisation des logiques modales dans le Calcul des Prédicats nous avons adopté des conventions de notation qui permettent d'utiliser des variables et des quantificateurs tout en restant dans le Calcul des Propositions.

Pour cela on suppose que le champ des variables quantifiées est fini et qu'il est connu. Dans ce cas les quantificateurs peuvent être considérés comme des abréviations pour des conjonctions ou des disjonctions finies dont les opérandes sont des instances, pour chaque élément du domaine, des formules quantifiées. On a alors :

$\forall\ x\ \phi(x) \stackrel{\text{def}}{=} \bigwedge_{x \in D_x} \phi(x)$

$\exists\ x\ \phi(x) \stackrel{\text{def}}{=} \bigvee_{x \in D_x} \phi(x)$

Dans ces abréviations D_x représente le domaine de définition de la variable x.

En acceptant ces conventions les formules atomiques dans $\phi(x)$ sont des formules sans variables qui peuvent être considérées comme des formules atomiques du Calcul des Propositions et dont la sémantique est définie comme les autres formules de LCP.

La signification intuitive des prédicats qui seront utilisés par la suite est indiquée ci-dessous.

$memory(I, m)$: le message m est dans l'espace physique qui sert de support à la mémoire de l'agent institutionnel I.

$mailbox(J, m)$: le message m est dans la mail box de l'agent institutionnel J.

$counts(I, m, \phi, S)$: le fait qu'il y a dans la mémoire de l'agent institutionnel I le message m compte pour l'institution S comme le fait que I croit explicitement ϕ.

$bel(I, m, \phi, S)$: il y a dans la mémoire de l'agent institutionnel I le message m qui compte pour l'institution S comme le fait que I croit explicitement ϕ.

$sem(i, m, \phi)$: la signification que l'agent humain i attribue au message m est représentée par la formule ϕ.

On utilisera les abréviations suivantes.

$counts(I, m, \phi, S) \stackrel{\text{def}}{=} memory(I, m) \Rightarrow_S ExpBel_I \phi$

$bel(I, m, \phi, S) \stackrel{\text{def}}{=} memory(I, m) \wedge counts(I, m, \phi, S)$

$hbel(i, m, \phi) \stackrel{\text{def}}{=} memory(i, m) \wedge sem(i, m, \phi)$

3.2 Sémantique

Un modèle M du langage LCPM est un n-uple de la forme :

$M = < W, B, K, D, D^{\neg}, I, O, DS, EB, CT, v >$

dans lequel B, K, D, $D^{\neg}$, I, O, DS sont des familles de relations d'accessibilité définies sur $W \times W$, EB et une famille de fonctions de W dans 2^{2^W}, et CT est une famille de fonctions de $W \times 2^{2^W}$ dans 2^{2^W}, et v est une fonction de l'ensemble des formules atomiques de LCPM dans 2^W.

La signification intuitive des relations d'accessibilité et des fonctions est la suivante.

Si B_{agt} est une relation dans B on a :

$wB_{agt}w'$: le monde w' est consistant avec ce que croit l'agent agt dans le monde w.

Si K_{agt} est une relation dans K on a :

$wK_{agt}w'$: le monde w' est consistant avec ce que sait l'agent agt dans le monde w.

Si $D_{agt:\alpha}$ est une relation dans D on a :

$wD_{agt:\alpha}w'$: l'agent agt a commencé en w la réalisation de l'action α, et éventuellement d'autres actions, et il a terminé α en w'.

Si $D_{agt:\neg\alpha}$ est une relation dans $D^{\neg}$ on a :

$wD_{agt:\neg\alpha}w''$: l'agent agt a commencé à réaliser en w les mêmes actions que celles qu'il a réalisées dans le monde w' tel que $wD_{agt:\alpha}w'$, à l'exception de l'action α, et il a terminé ces actions en w''[5].

Si I_{agt} est une relation dans I on a :

[5]Si en w' d'autres agents ont réalisé d'autres actions, on suppose qu'en w'' ces agents ont réalisé les mêmes actions.

$wI_{agt}w'$: le monde w' est un monde dans lequel l'agent agt a l'intention d'être quand il est dans le monde w.

Si O_S est une relation dans O on a :

wO_Sw' : dans le monde w' l'ensemble des normes qui définissent l'institution S dans le monde w sont satisfaites.

Si DS_S est une relation dans DS on a :

wDS_Sw' : le monde w' est reconnu par l'ensemble des normes qui définissent l'institution S dans le monde w.

Si EB_{agt} est une fonction dans EB et X est un sous-ensemble de W on a :

$X \in EB_{agt}(w)$: dans le monde w l'agent agt croit explicitement la proposition représentée par l'ensemble de mondes X.

Si CT_S est une fonction dans CT et X et Y sont des sous-ensembles de W on a :

$Y \in CT_S(w, X)$: pour l'ensemble des normes qui définissent l'institution S dans le monde w la proposition représentée par l'ensemble de mondes X compte comme la proposition représentée par l'ensemble de mondes Y.

Les contraintes imposées aux relations sont celles qui sont imposées aux relations qui interprètent des modalités qui satisfont un système de type KD (voir [3]). De plus on impose aux relations $D_{agt:\alpha}$ la contrainte :

Si $w_1D_{agt:\alpha_1}w$ et $w_2D_{agt:\alpha_2}w$, alors $w_1 = w_2$ et $\alpha_1 = \alpha_2$.

Cette contrainte signifie que le passé est unique et qu'un agent ne peut pas exécuter deux actions simultanément.

Aucune contrainte n'est imposée aux fonctions dans EB.

Les contraintes imposées aux fonctions dans CT sont celles qui ont été fixées dans [12].

Conditions de satisfaisabilité.

Les conditions de satisfaisabilité des modalités qui satisfont un système de type KD sont définies comme d'habitude. La relation $M, w \models \phi$ signifie que ϕ est vraie dans le monde w du modèle M.

$M, w \models Bel_{agt}\phi$ ssi $\forall w'(wB_{agt}w' \quad \Rightarrow \quad M, w' \models \phi)$

$M, w \models Know_{agt}\phi$ ssi $\forall w'(wK_{agt}w' \quad \Rightarrow \quad M, w' \models \phi)$

$M, w \models Obg_S\phi$ ssi $\forall w'(wO_Sw' \quad \Rightarrow \quad M, w' \models \phi)$

Les modalités $Perm_S$ et $Forb_S$ sont définies comme d'habitude en fonction de la modalité Obg_S. On a :

$Perm_S\phi \stackrel{\text{def}}{=} \neg Obg_S\neg\phi$

$Forb_S\phi \stackrel{\text{def}}{=} Obg_S\neg\phi$

Donc leurs conditions de satisfaisabilité se déduisent des conditions de satisafaisabilité de Obg_S.

$M, w \models Does_{agt:\alpha}\phi$ ssi $\forall w'(wD_{agt:\alpha}w' \Rightarrow M, w' \models \phi)$

$M, w \models Done^{-1}_{agt}\phi$ ssi $\exists\alpha\exists w'(w'D_{agt:\alpha}w \textit{ et } M, w' \models \phi)$

La condition de satisfaisabilité de $Does_{agt:\alpha}\phi$ n'est pas ambigu parce qu'en w il n'y a qu'un monde w' tel que $w'D_{agt:\alpha}w$.

$M, w \models E^+_{agt:\alpha}\phi$ ssi il existe α telle que :
(1) $\forall w'(wD_{agt:\alpha}w' \Rightarrow M, w' \models \phi)$ et
(2) $\exists w''(wD_{agt:\neg\alpha}w''$ et $M, w'' \models \neg\phi)$

La condition (1) exprime qu'il suffit de réaliser α pour avoir ϕ et la condition (2) exprime que la réalisation de α est nécessaire pour avoir ϕ car sinon, toutes choses égales par ailleurs, il est possible que l'on n'ait pas ϕ.

Les conditions de satisfaisabilité des modalités *Done* et E sont définies respectivement en fonction de celles des modalités *Does* et E^+. Comme ces définitions sont complexes et tout à fait similaires, pour éviter des répétitions fastidieuses, nous utiliserons la notation A pour désigner *Done* ou E et la notation A^+ pour désigner respectivement *Does* ou E^+. Intuitivement A s'applique quand on se place dans un monde où une action vient d'être réalisée et A^+ quand elle va être réalisée. On a alors :

$M, w \models A_{agt}\phi$ ssi $\exists w_1\exists\alpha_1(\exists w_2(w_1D_{agt:\alpha_1}w_2$ et $Path(\phi, w_2, w))$ et $M, w_1 \models A^+_{agt}T(\phi))$

Les formules dénotées par $Path(\phi, w_2, w)$ et $T(\phi)$ sont définies récursivement de la même manière pour les modalités $Done_{agt}$ ou E_{agt}. Ces modalités peuvent avoir dans leur champ l'une ou l'autre de ces modalités sans aucune limitation sur leur niveau d'imbrication. On a :

- Si ϕ n'est pas de la forme : $A_{agt}\phi_n$, alors
  $Path(\phi, w_n, w) \stackrel{\text{def}}{=} (w_n = w)$ et $T(\phi) \stackrel{\text{def}}{=} \phi$.

- Si ϕ est de la forme : $A_{agt}\phi_n$, alors
  $Path(\phi, w_n, w) \stackrel{\text{def}}{=} \exists w_{n+1}\exists\alpha_n(w_nD_{agt:\alpha_n}w_{n+1}$ et $Path(\phi_n, w_{n+1}, w))$

et $T(\phi) \stackrel{\text{def}}{=} A^+_{agt} T(\phi_n)$.

La signification intuitive de $\exists w_1(\exists w_2(w_1 D_a w_2$ et $Path(\phi, w_2, w))$ est que la séquence des modalités d'action qui son imbriquées a commencé dans le monde w_1, et s'il n'y a pas de modalité imbriquée le monde w_2 où elle s'est terminée est le monde w, tandis que s'il y a des modalités imbriquées la séquence de mondes dans lesquels les actions sont réalisées successivement est définie récursivement par la fonction $Path$.

3.3 Axiomatique

En plus des règles d'inférence et des schémas d'axiomes du Calcul des Propositions (CP) on a l'axiomatique suivante.

On suppose que les modalités Bel_{agt}, $Know_{agt}$, Int_{agt}, Obg_S et D_S satisfont un système de type KD et que les modalités $Does_{agt:\alpha}$, $Done_{agt}$ et $Done^{-1}_{agt}$ satisfont un système de type K. De plus on a :

(DO) $Done_a\phi \rightarrow \phi$

On suppose que l'axiomatique de la modalité $ExpBel_I$ est définie par la seule règle de substitutivité des formules équivalentes.

(EQB) Si $\vdash \phi \leftrightarrow \psi$, alors $\vdash ExpBel_I\phi \leftrightarrow ExpBel_I\psi$.

On suppose que les modalités E^+_{agt} et E_{agt} sont des modalités classiques mais pas normales qui satisfont, entre autres, les règles et schémas d'axiomes suivants.

(EQE^+) Si $\vdash \phi \leftrightarrow \psi$, alors $\vdash E^+_{agt}\phi \leftrightarrow E^+_{agt}\psi$.

($\neg$N+) $\nvdash E^+_{agt}\top$

($EE^+\wedge$) $E^+_{agt}\phi \wedge E^+_{agt}\psi \rightarrow E^+_{agt}(\phi \wedge \psi)$

($EE^+\vee$) $E^+_{agt}(\phi \wedge \psi) \rightarrow E^+_{agt}\phi \vee E^+_{agt}\psi$

(EQE) Si $\vdash \phi \leftrightarrow \psi$, alors $\vdash E_{agt}\phi \leftrightarrow E_{agt}\psi$.

($\neg$N) $\nvdash E_{agt}\top$

($EE\wedge$) $E_{agt}\phi \wedge E_{agt}\psi \rightarrow E_{agt}(\phi \wedge \psi)$

($EE\vee$) $E_{agt}(\phi \wedge \psi) \rightarrow E_{agt}\phi \vee E_{agt}\psi$

(ET) $E_{agt}\phi \rightarrow \phi$

La relation $\Rightarrow_S$ est axiomatisée comme une modalité classique qui satis-

fait les règles et schémas d'axiomes suivants.

(EQV) Si $\vdash \phi \leftrightarrow \phi'$ et $\vdash \psi \leftrightarrow \psi'$, alors $\vdash (\phi \Rightarrow_S \psi) \rightarrow (\phi' \Rightarrow_S \psi')$.

(CC) $((\phi \Rightarrow_S \psi) \wedge (\phi \Rightarrow_S \psi')) \rightarrow (\phi \Rightarrow_S (\psi \wedge \psi'))$

(CA) $((\phi \Rightarrow_S \psi) \wedge (\phi' \Rightarrow_S \psi)) \rightarrow ((\phi \vee \phi') \Rightarrow_S \psi)$

De plus nous avons accepté que cette relation soit transitive.

(S) $((\phi \Rightarrow_S \psi) \wedge (\psi \Rightarrow_S \theta)) \rightarrow (\phi \Rightarrow_S \theta)$

Les liens entre $\Rightarrow_S$ et D_S sont exprimés par les schémas suivants.

(D) $(\phi \Rightarrow_S \psi) \rightarrow D_S(\phi \rightarrow \psi)$

(C) $(\phi \Rightarrow_S \psi) \rightarrow (\phi \rightarrow D_S\phi)$

(DD) $D_S\phi \leftrightarrow D_S D_S\phi$

On peut montrer que dans cette axiomatique on a le théorème suivant.

(ConsC) $\vdash \phi \Rightarrow_S \psi \rightarrow (\neg D_S\phi \rightarrow \neg\phi)$

D'autre part on accepte les schémas suivants.

(DEB) $D_S ExpBel_I\phi \leftrightarrow \exists m\ bel(I, m, \phi, S)$

(CP) $counts(I, m, \phi, S) \leftrightarrow D_S Done_J^{-1} counts(I, m, \phi, S)$

La signification intuitive de (DEB) est que pour l'institution S l'agent institutionnel I croit ϕ ssi il y a un message dans la mémoire physique de cet agent qui pour l'institution S signifie ϕ.

La signification intuitive de (CP) est que pour l'institution S le fait que le message m qui est dans la mémoire de l'agent institutionnel I signifie ϕ ne change pas après qu'un autre agent J ait réalisé une certaine action.

4 Actions de communication entre agents institutionnels

Nous allons présenter dans la logique qui a été définie dans la section précédente les actions de communication sur lesquelles portent les normes de confidentialité et d'intégrité ainsi que les propriétés qui justifient leurs définitions.

Dans cette section on suppose que l'agent qui réalise l'action et l'agent destinataire sont des agents institutionnels. En fait, comme on l'a vu plus haut, les agents institutionnels n'agissent pas eux-même, mais dans cette section on ne précisera pas quels sont les agents qui agissent pour le compte

des agents institutionnels.

4.1 Informer

$Inf_{I,J,S}\phi$: l'agent institutionnel I a donné la possibilité à l'agent institutionnel J de savoir que I croit ϕ.

Définition formelle :

$Inf_{I,J,S}\phi \stackrel{\text{def}}{=} D_S(\exists m(Done_I^{-1}bel(I,m,\phi,S) \wedge E_I mailbox(J,m)))$

Dans cette définition formelle la modalité D_S indique qu'on définit une action qui compte pour S comme une action réalisée par I. La formule $E_I mailbox(J,m)$ signifie que l'agent I a fait en sorte qu'il y ait dans la mailbox de J le message m, et la formule $Done_I^{-1}bel(I,m,\phi,S)$ signifie qu'avant que I réalise cette action ce message était présent dans la mémoire de I et que la signification qui lui est attribuée par l'institution S est représentée par la formule ϕ. Sans cette condition la signification du message m ne serait pas définie.

Dans la sémantique on a :

$M,w \models Inf_{I,J,S}\phi$ ssi $\forall w'(wD_S w' \Rightarrow$
$\exists m \exists w'' \exists \alpha(w'' D_{I:\alpha} w'$ et $M,w'' \models bel(I,m,\phi,S) \wedge E_I^+ mailbox(J,m))$

On accepte le schéma d'axiome suivant :

(PERSB) $Done_I^{-1}bel(I,m,\phi,S) \wedge E_I mailbox(J,m) \rightarrow bel(I,m,\phi,S)$

Ce schéma signifie intuitivement que si l'agent I a rangé le message m dans la mailbox de J, alors cela ne change pas le fait que ce message était dans la mémoire de I.

Le Théorème 1 ci-dessous signifie que si l'agent I a réalisé l'action $Inf_{I,J,S}\phi$, alors pour S il y a dans la mailbox de J un message qui signifie ϕ pour I.

Theorem 1 *Si on a (PERSB), alors on a :*
$\vdash Inf_{I,J,S}\phi \rightarrow D_S(\exists m(bel(I,m,\phi,S) \wedge mailbox(J,m))$

Preuve.

(1) $Inf_{I,J,S}\phi$

(2) $D_S(\exists m(bel(I,m,\phi,S) \wedge E_I mailbox(J,m))$, d'après (1) et (PERSB)

(3) $D_S(\exists m(bel(I,m,\phi,S) \wedge mailbox(J,m))$, d'après (2), (ET) et le fait que D_S est un opérateur normal

(4) $\vdash Inf_{I,J,S}\phi \rightarrow D_S(\exists m(bel(I,m,\phi,S) \wedge mailbox(J,m))$, d'après (1) et (3)

Fin de la preuve.

On accepte les schémas d'axiomes suivants dont on donne la signification intuitive de chacun.

Si I a mis un message qu'il interprète comme ϕ dans la mailbox de J, alors J sait que I a fait cette action.

(TRANS) $D_S(\exists n(Done_I^{-1}bel(I,n,\phi,S) \wedge E_I mailbox(J,n)) \rightarrow Know_J \exists n(Done_I^{-1}bel(I,n,\phi,S) \wedge E_I mailbox(J,n)))$

Si J sait qu'il y a un message dans sa mailbox, alors J prends connaissance de ce message.

(ACS) $(Done_I^{-1}bel(I,m,\phi,S) \wedge E_I mailbox(J,m)) \wedge D_S(Know_J \exists n(Done_I^{-1}bel(I,n,\phi,S) \wedge E_I mailbox(J,n)) \rightarrow Know_J(Done_I^{-1}bel(I,m,\phi,S) \wedge E_I mailbox(J,m)))$

Si J sait qu'il y a dans sa mailbox un message que I interprète comme ϕ, alors J sait que I a l'intention que J sache que I croit ϕ.

(SEM) $D_S Know_J((Done_I^{-1}bel(I,m,\phi,S) \wedge E_I mailbox(J,m)) \rightarrow Int_I Know_J Bel_I \phi)$

J sait que I est sincère à propos de ce qu'il a l'intention de communiquer à I.

(SINC) $D_S Know_J(Int_I Know_J Bel_I \phi \rightarrow Bel_I \phi)$

Theorem 2 *Si on a (TRANS), (ACS), (SEM) et (SINC), alors on a* $\vdash Inf_{I,J,S}\phi \rightarrow D_S Know_J Bel_I \phi$

Preuve.

(1) $Inf_{I,J,S}\phi$

(2) $D_S(\exists n(Done_I^{-1}bel(I,n,\phi,S) \wedge E_I mailbox(J,n))$, d'après (1) et la Définition de $Inf_{I,J,S}\phi$

(3) $D_S Know_J(Done_I^{-1}bel(I,m,\phi,S) \wedge E_I mailbox(J,m))$, d'après (2), (TRANS) et (ACS)

(4) $D_S Know_J Bel_I \phi$, d'après (3), (SEM) et (TRANS)

(5) $\vdash Inf_{I,J,S}\phi \rightarrow D_S Know_J Bel_I \phi$, d'après (1) et (4)

Fin de la preuve.

Le Théorème 2 montre qu'il suffit que les conditions (TRANS), (ACS), (SEM) et (SINC) soient satisfaites pour que la possibilité qui est donné à J en rangeant le message m dans sa mailbox de savoir que I croit ϕ devienne effective.

4.2 Insérer

$Ins_{J,I,S}\phi$: l'agent institutionnel J a inséré dans la mémoire de l'agent institutionnel I l'information ϕ.

$Ins_{J,I,S}\phi \stackrel{\text{def}}{=} D_S(Done_J^{-1}(\neg\exists\, m\, bel(I,m,\phi,S))\wedge\exists m(counts(I,m,\phi,S)\wedge E_J memory(I,m)))$

L'opérateur D_S indique que l'action qui est définie compte pour S comme une action réalisée par J. La formule $E_J memory(I,m)$ signifie que J a fait en sorte que le message m soit rangé dans la mémoire de I et la formule $counts(I,m,\phi,S)$ indique ce message signifie ϕ pour S. D'autre part, la formule $Done_J^{-1}(\neg\exists\ m\ bel(I,m,\phi,S)$ signifie qu'avant que J réalise cette action il n'y avait aucun message dans la mémoire de I qui signifie ϕ pour S. Cette dernière condition signifie intuitivement qu'on n'appelle pas insérer une action qui consiste à insérer une information qui était déjà explicitement présente dans la mémoire de I.

Dans la sémantique on a :

$M,w \models Ins_{J,I,S}\phi$ ssi $\forall w'(wD_S w' \Rightarrow$
$\exists w''(w''D_J w'$ et $\neg\exists m(M,w'' \models bel(I,m,\phi,S)))$ et $\exists m(M,w' \models count(I,m,\phi,S)\wedge$
$M,w'' \models E_J^{+} memory(I,m)))$

Theorem 3 $\vdash Ins_{J,I,S}\phi \rightarrow D_S ExpBel_I\phi$

Preuve.

(1) $Ins_{J,I,S}\phi$

(2) $D_S(\exists m(counts(I,m,\phi,S)\wedge E_J memory(I,m)))$, d'après (1) et la définition de $Ins_{J,I,S}\phi$ et le fait que D_S est un opérateur normal

(3) $\vdash D_S E_J memory(I,m) \rightarrow D_S memory(I,m))$, d'après (ET) et le fait que D_S est un opérateur normal

(4) $D_S(\exists m(counts(I,m,\phi,S) \wedge memory(I,m)))$, d'après (2) et (3)

(5) $D_S(\exists m((memory(I,m) \Rightarrow_S ExpBel_I\phi) \wedge memory(I,m)))$, d'après (4) et la définition de *counts*

(6) $D_S D_S ExpBel_I \phi$, d'après (5) et (D)

(7) $D_S ExpBel_I \phi$, d'après (6) et (DD)

(8) $\vdash Ins_{J,I,S}\phi \rightarrow D_S ExpBel_I \phi$, d'après (1) et (7)

Fin de la preuve.

Le Théorème 3 signifie intuitivement que si J insère ϕ dans I, alors pour S I croit explicitement ϕ.

4.3 Supprimer

$Sup_{J,I,S}\phi$: l'agent institutionnel J a supprimé dans la mémoire de l'agent institutionnel I l'information ϕ.

Notation.

$M_{J,I,\phi,S} \ = \ \{ \ n \ : \ D_S Done_J^{-1}(bel(I,n,\phi,S))\}$

$M_{J,I,\phi,S}$ définit l'ensemble des messages qui signifient ϕ et qui sont dans I avant que J réalise l'action qui a pour effet de les supprimer.

$Sup_{J,I,S}\phi \stackrel{\text{def}}{=} D_S(Done_J^{-1}(\exists\, m\, bel(I,m,\phi,S)) \wedge E_J(\bigwedge_{n \in M_{J,I,\phi,S}} \neg memory(I,n)))$

L'opérateur D_S indique que l'action qui est définie compte pour S comme une action réalisée par J. La formule $E_J(\bigwedge_{n \in M_{J,I,\phi,S}} \neg memory(I,n))$ signifie que J a fait en sorte que tous les messages dans I qui signifient ϕ aient été supprimés[6]. D'autre part la formule $Done_J^{-1}(\exists\ m\ bel(I,m,\phi,S)$ signifie qu'avant que J réalise cette action il y avait au moins un message dans I qui signifie ϕ. Cette dernière condition signifie intuitivement qu'on n'appelle pas supprimer une action qui ne supprime aucun message dans I qui signifie ϕ parce qu'aucun message de cette nature n'est présent dans I.

Dans la sémantique on a :

$M, w \models Sup_{J,I,S}\phi$ ssi $\forall w'(wD_S w' \ \Rightarrow \ \exists w'' \exists \alpha(w'' D_{J:\alpha} w'$ et
$\exists m(M, w'' \models bel(I,m,\phi,S) \wedge E_J^+(\bigwedge_{n \in M_{J,I,\phi,S}} \neg memory(I,n))))$

On a le schéma :

(DD) $D_S \neg\phi \rightarrow \neg D_S \phi$

D'après (DEB) on a aussi :

[6] Il se peut cependant qu'après cette action I puisse déduire ϕ à partir d'autres croyances explicites. Nous n'aborderons pas ici le problème de la suppression des croyances implicites.

$\neg D_S ExpBel_I \phi \leftrightarrow \neg \exists m \; bel(I, m, \phi, S)$

On en déduit :
$\neg D_S ExpBel_I \phi \leftrightarrow \forall m(\neg memory(I, m, \phi, S) \vee \neg counts(I, m, \phi, S))$ et

(DE1) $\neg D_S ExpBel_I \phi \leftrightarrow \forall m(counts(I, m, \phi, S) \rightarrow \neg memory(I, m, \phi, S))$

On accepte les schémas d'axiomes suivants dont on donne la signification intuitive de chacun.

Le fait que J a supprimé des messages ne change pas le fait qu'on ait $counts(I, m, \phi, S)$.

(PERSC) $counts(I, m, \phi, S) \wedge E_J(\bigwedge_{n \in M_{J,I,\phi,S}} \neg memory(I, n)) \rightarrow Done_J^{-1} counts(I, m, \phi, S)$

Si l'action que vient de faire J est celle qui supprime tous les messages dans I qui signifient ϕ, alors si le message m est dans la mémoire de I quand cette action a été réalisée, il y était avant que J réalise cette action.

(PERSM) $E_J(\bigwedge_{n \in M_{J,I,\phi,S}} \neg memory(I, n)) \rightarrow (memory(I, m) \rightarrow D_S Done_J^{-1} memory(I, m))$

Theorem 4 $\vdash Sup_{J,I,S}\phi \rightarrow \neg D_S ExpBel_I \phi$

Preuve.

(1) $Sup_{J,I,S}\phi$

Soit m tel que : (2) $counts(I, m, \phi, S)$.

- Si on a : (3) $D_S Done_J^{-1} memory(I, m)$, on a :

(4) $D_S Done_J^{-1} counts(I, m, \phi, S)$, d'après (2) et (PERSC)

(5) $D_S Done_J^{-1} bel(I, m, \phi, S)$, d'après (3), (4) et le fait que D_S et $Done_J^{-1}$ sont des opérateurs normaux

(6) $m \in M_{J,I,\phi,S}$, d'après (5) et la définition de $M_{J,I,\phi,S}$

(7) $D_S E_J(\bigwedge_{n \in M_{J,I,\phi,S}} \neg memory(I, n))$, d'après (1) et la définition de $Sup_{J,I,S}\phi$ et le fait que D_S est un opérateur normal

(8) $D_S \bigwedge_{n \in M_{J,I,\phi,S}} \neg memory(I, n)$, d'après (7), (ET) et le fait que D_S est un opérateur normal

(9) $D_S \neg memory(I, m)$, d'après (6), (8) et le fait que D_S est un opérateur normal

(10) $\neg D_S memory(I, m)$, d'après (9) et (DD)

(11) $\neg memory(I, m)$, d'après (2), (10) et (ConsC)

- Si on a : (12) $\neg D_S Done_J^{-1} memory(I, m)$, on a :

(13) $\neg memory(I, m)$, d'après (1), (12) et (PERSM)

Donc dans le cas où on a (3) et où on a (12) on a : $\neg memory(I, m)$, et on en déduit :

(14) $counts(I, m, \phi, S) \rightarrow \neg memory(I, m)$, d'après (2), (11) et (13)

Donc on a :

(15) $\forall m(counts(I, m, \phi, S) \rightarrow \neg memory(I, m))$, car on a (14) pour un m quelconque

(16) $\neg D_S ExpBel_I \phi$, d'après (15) et (DE1)

Donc on a :

$\vdash Sup_{J,I,S}\phi \rightarrow \neg D_S ExpBel_I \phi$, d'après (1) et (16)

Fin de preuve.

Le Théorème 4 signifie intuitivement que si J a supprimé ϕ dans I, alors I ne croit pas explicitement ϕ.

5 Actions de communication entre agents institutionnels, humains et logiciels

5.1 Informer

$Inf_{(i:r,I),J,S}\phi$: l'agent humain i, agissant en tant que titulaire du rôle r dans l'agent institutionnel I, a communiqué à l'agent institutionnel J une information qui pour l'institution S signifie que I croit ϕ.

$Inf_{(i:r,I),J,S}\phi \stackrel{\text{def}}{=} \exists \alpha, \beta, p, m(Done^{-1}_{(i:r:\alpha);(p:\beta)} bel(I, m, \phi, S) \wedge$
$E_{i:r:\alpha} E_{p:\beta} mailbox(J, m))$

La signification de cette action est tout à fait similaire à celle de l'action $Inf_{I,J,S}\phi$, la seule différence étant que $E_{i:r:\alpha} E_{p:\beta} mailbox(J, m)$ signifie que c'est l'agent i qui en réalisant l'action α en tant que titulaire du rôle r qui a fait en sorte que l'agent logiciel p en réalisant l'action β a fait en sorte que l'on ait le message m dans la mailbox de J (voir la figure 1). Dans $Done^{-1}_{(i:r:\alpha);(p:\beta)}$ la formule $(i : r : \alpha); (p : \beta)$ dénote l'action $(i : r : \alpha)$ suivie de l'action $(p : \beta)$.

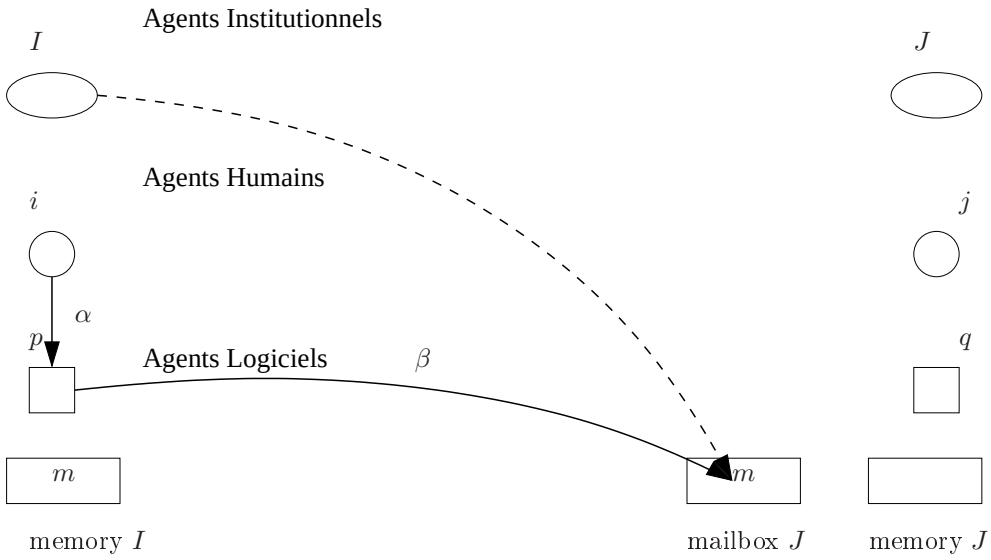

Figure 1: Le représentant i de l'agent institutionnel I informe J.

On accepte le schéma (PERSBL) ci-dessous dont la signification est similaire à celle du schéma (PERSB).

(PERSBL) $Done_I^{-1} bel(I, m, \phi, S) \wedge E_{i:r:\alpha} E_{p:\beta} mailbox(J, m) \rightarrow bel(I, m, \phi, S)$

Theorem 5 *Si on a (PERSBL), alors on a :*
$\vdash Inf_{(i:r,I),J,S}\phi \rightarrow \exists m(bel(I, m, \phi, S) \wedge mailbox(J, m))$

Preuve.

(1) $Inf_{(i:r,I),J,S}\phi$

(2) $\exists \alpha, \beta, p, m(Done^{-1}_{(i:r:\alpha);(p:\beta)} bel(I, m, \phi, S) \wedge E_{i:r:\alpha} E_{p:\beta} mailbox(J, m))$, d'après (1) et la définition de Inf

(3) $\exists \alpha, \beta, p, m(bel(I, m, \phi, S) \wedge E_{i:r:\alpha} E_{p:\beta} mailbox(J, m))$, d'après (2) et (PERSBL)

(4) $\exists m(bel(I, m, \phi, S) \wedge mailbox(J, m))$, d'après (3) et (ET) deux fois

(5) $Inf_{(i:r,I),J,S}\phi \rightarrow \exists m(bel(I, m, \phi, S) \wedge mailbox(J, m))$, d'après (1), (4)

Fin de preuve.

Le Théorème 5 montre que sous l'hypothèse (PERSBL) l'action que réalise l'agent humain i en utilisant l'agent logiciel p produit un effet brut qui peut compter comme l'effet institutionnel que produit l'action $Inf_{I,J,S}\phi$.

On notera que $Inf_{(i:r,I),J,S}\phi$ ne suffit pas à garantir que l'on ait le message m dans la mémoire de J.

On adopte les notations suivantes.

(COUNT1) $\forall m(E_{i:r:\alpha}E_{p:\beta}mailbox(J,m) \Rightarrow_S E_I mailbox(J,m))$

(COUNT2) $\forall m(Done^{-1}_{(i:r:\alpha);(p:\beta)}bel(I,m,\phi,S) \Rightarrow_S Done_I^{-1}bel(I,m,\phi,S))$

On notera que (COUNT1) et (COUNT2) ne sont pas des schémas d'axiomes et que les termes différents de m sont des constantes. On pourrait cependant accepter des hypothèses plus générales dans les théorèmes suivants.

Par exemple, il se pourrait que pour l'institution S on ait (COUNT1) quelque soit l'agent logiciel p déclenché par i et quelle que soit l'action β qu'il réalise, pourvu que l'effet soit que l'on ait $mailbox(J,m)$. De même on pourrait accepter que l'on ait (COUNT1) pour n'importe quel agent i, pourvu qu'il réalise α en tant que titulaire du rôle r.

Les mêmes remarques s'appliquent aux autres hypothèses de la forme (COUNT).

Theorem 6 *Si on a (COUNT1) et (COUNT2), alors on a :*
$\vdash Inf_{(i:r,I),J,S}\phi \rightarrow Inf_{I,J,S}\phi$.

Preuve.

(1) $Inf_{(i:r,I),J,S}\phi$

(2) $\exists\alpha,\beta,p,m(Done^{-1}_{(i:r:\alpha);(p:\beta)}bel(I,m,\phi,S) \wedge E_{i:r:\alpha}E_{p:\beta}mailbox(J,m))$, d'après (1) et la définition de Inf

(3) $\exists m(D_S Done_I^{-1}bel(I,m,\phi,S) \wedge D_S E_I mailbox(J,m))$, d'après (1), (COUNT1), (COUNT2) et (C)

(4) $D_S\exists m(Done_I^{-1}bel(I,m,\phi,S) \wedge E_I mailbox(J,m))$, d'après (3) et le fait que D_S est un opérateur normal

(5) $Inf_{I,J,S}\phi$, d'après (4) et la définition de Inf

(6) $Inf_{(i:r,I),J,S}\phi \rightarrow Inf_{I,J,S}\phi$, d'après (1) et (5)

Fin de preuve.

Le Théorème 6 montre que sous les hypothèses (COUNT1) et (COUNT2)

en réalisant l'action $Inf_{(i:r,I),J,S}\phi$ on réalise l'action $Inf_{I,J,S}\phi$.

Theorem 7 *Si on a (TRANS), (ACS), (SEM), (SINC), (COUNT1) et (COUNT2), alors on a :*
$\vdash Inf_{(i:r,I),J,S}\phi \rightarrow D_S Know_J Bel_I \phi$.

Preuve.

D'après les Théorèmes 6 et 2.

Fin de preuve.

Le Théorème 7 montre que si les hypothèses (TRANS), (ACS), (SEM), (SINC), (COUNT1) et (COUNT2) sont satisfaites, alors l'action $Inf_{(i:r,I),J,S}\phi$ a pour effet d'informer effectivement J du fait que I croit ϕ.

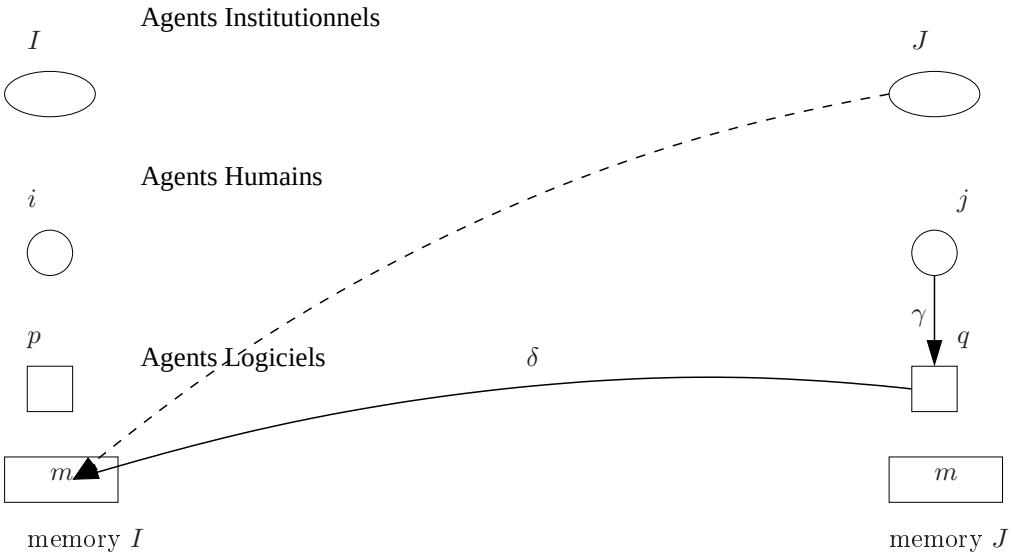

Figure 2: Le représentant j de l'agent institutionnel J insère m dans la mémoire de I.

5.2 Insérer

$Ins_{(j:s,J),I,S}\phi$: l'agent humain j, agissant en tant que titulaire du rôle s dans l'agent institutionnel J, a inséré une information dans la mémoire de I

qui pour l'institution S signifie que I croit ϕ.

$Ins_{(j:s,J),I,S}\phi \stackrel{\text{def}}{=} \exists \gamma, \delta, q, m(Done^{-1}_{(j:s:\gamma);(q:\delta)}(\neg\exists\ m\ bel(I,m,\phi,S))\wedge$
$\exists m(counts(I,m,\phi,S) \wedge E_{j:s:\gamma}E_{q:\delta}memory(I,m))$

La signification de cette action est tout à fait similaire à celle de l'action $Ins_{I,I,S}\phi$ la seule différence étant que $E_{j:s:\gamma}E_{q:\delta}memory(I,m)$ signifie que c'est l'agent j qui en réalisant l'action γ en tant que titulaire du rôle s a fait en sorte que l'agent logiciel q en réalisant δ a fait en sorte que l'on ait le message m dans la mémoire de I (voir la figure 2).

Theorem 8 $\vdash Ins_{(j:s,J),I,S}\phi \rightarrow D_S ExpBel_I\phi$

Preuve.

(1) $Ins_{(j:s,J),I,S}\phi$

(2) $\exists m(counts(I,m,\phi,S) \wedge E_{j:s:\gamma}E_{q:\delta}memory(I,m))$, d'après (1) et la définition de Ins

(3) $\exists m(counts(I,m,\phi,S)\wedge memory(I,m))$, d'après (2) et (ET) deux fois

(4) $D_S ExpBel_I\phi$, d'après (3) et (DEB)

(5) $Ins_{(j:s,J),I,S}\phi \rightarrow D_S ExpBel_I\phi$, d'après (1) et (4)

Fin de preuve.

Le Théorème 8 montre que l'action $Ins_{(j:s,J),I,S}\phi$ a pour effet que pour l'institution S I croit explicitement ϕ.

On adopte les notations suivantes.

(COUNT3) $\forall m(E_{j:s:\gamma}E_{q:\delta}memory(I,m) \Rightarrow_S E_J memory(I,m))$

(COUNT4) $\forall m(Done^{-1}_{(j:s:\gamma);(q:\delta)}(\neg bel(I,m,\phi,S)) \Rightarrow_S Done^{-1}_J(\neg bel(I,m,\phi,S)))$

Theorem 9 *Si on a (COUNT3) et (COUNT4), alors on a :*
alors $\vdash Ins_{(j:s,J),I,S}\phi \rightarrow Ins_{J,I,S}\phi$.

Preuve.

(1) $Ins_{(j:s,J),I,S}\phi$

(2) $\exists \gamma, \delta, q, m(Done^{-1}_{(j:s:\gamma);(q:\delta)}(\neg\exists\, m\, bel(I,m,\phi,S))\wedge\exists m(counts(I,m,\phi,S)\wedge$
$E_{j:s:\gamma}E_{q:\delta}memory(I,m))$, d'après (2) et la définition de Ins

(3) $\exists m(D_S Done_J^{-1}(\neg\exists\ m\ bel(I,m,\phi,S))\wedge$
$\exists m(counts(I,m,\phi,S)\wedge D_S E_J memory(I,m))$, d'après (2), (COUNT3), (COUNT4) et (C)

(4) $D_S(\exists m(Done_J^{-1}(\neg\exists\ m\ bel(I,m,\phi,S))\wedge$
$\exists m(counts(I,m,\phi,S)\wedge E_J memory(I,m))$, d'après (3)et le fait que D_S est un opérateur normal

(5) $Ins_{(j:s,J),I,S}\phi \rightarrow Ins_{J,I,S}\phi$, d'après (1) et (4)

Fin de preuve.

Le Théorème 9 montre que sous les hypothèses (COUNT3) et (COUNT4) en réalisant l'action $Ins_{(j:s,J),I,S}\phi$ on réalise l'action $Ins_{J,I,S}\phi$.

5.3 Supprimer

$Sup_{(j:s,J),I,S}\phi$: l'agent humain j, agissant en tant que titulaire du rôle s dans l'agent institutionnel J, a supprimé dans la mémoire de I les informations qui pour l'institution S signifie que I croit ϕ.

$M_{J,I,\phi,S} = \{\ n\ :\ D_S Done_J^{-1}(bel(I,n,\phi,S))\}$

$M_{(j:s:\gamma);(q:\delta),I,\phi,S} = \{\ n\ :\ Done^{-1}_{(j:s:\gamma);(q:\delta)}(bel(I,n,\phi,S))\}$

$Sup_{(j:s,J),I,S}\phi \stackrel{\text{def}}{=} \exists\gamma,\delta,q(Done^{-1}_{(j:s:\gamma);(q:\delta)}(\exists\ m\ bel(I,m,\phi,S))\wedge$
$E_{j:s:\gamma}E_{q:\delta}(\bigwedge_{n\in M_{(j:s:\gamma);(q:\delta),I,\phi,S}}\neg memory(I,n)))$

La signification de l'action $Sup_{(j:s,J),I,S}\phi$ est tout à fait similaire à celle de l'action $Sup_{J,I,S}\phi$.

La seule différence est que $E_{j:s:\gamma}E_{q:\delta}(\bigwedge_{n\in M_{(j:s:\gamma);(q:\delta),I,\phi,S}}\neg memory(I,n))$ signifie que c'est l'agent j qui en réalisant l'action γ en tant que titulaire du rôle s a fait en sorte que l'agent logiciel q a fait en sorte en réalisant l'action γ qu'il n'y ait aucun message dans I qui signifie ϕ.

On accepte le schéma d'axiome suivant.

(PERS) $(counts(I,m,\phi,S)\wedge Sup_{(j:s,J),I,S}\phi\wedge\neg D_S Done_J^{-1}memory(I,m)) \rightarrow$
$\neg memory(I,m)$

La signification intuitive de (PERS) est que si un message m signifie ϕ pour l'institution S et que l'agent j qui représente J vient de réaliser l'action qui supprime tous les messages dans I qui signifient ϕ et qu'avant de réaliser cette action ce message n'était pas dans I, alors après que cette action ait été réalisée m n'est toujours pas dans I. Plus simplement : un message qui

signifie ϕ mais qui n'est pas dans I ne peut avoir été rajouté dans I après que tous les messages dans I signifiant ϕ ait été supprimés.

Theorem 10 $\vdash Sup_{(j:s,J),I,S}\phi \rightarrow \neg D_S ExpBel_I \phi$

Preuve.

(1) $Sup_{(j:s,J),I,S}\phi$

Soit m tel que (2) $counts(I, m, \phi, S)$.

- Si on a (3) $Done^{-1}_{(j:s:\gamma);(q:\delta)} memory(I, m)$

(4) $m \in M_{(j:s:\gamma);(q:\delta),I,\phi,S}$, d'après (3) et la définition de $M_{(j:s:\gamma);(q:\delta),I,\phi,S}$

(5) $E_{j:s:\gamma}E_{q:\delta}(\bigwedge_{n\in M_{(j:s:\gamma);(q:\delta),I,\phi,S}} \neg memory(I, n))$, d'après (1) et la définition de $Sup_{(j:s,J),I,S}$

(6) $\bigwedge_{n\in M_{(j:s:\gamma);(q:\delta),I,\phi,S}} \neg memory(I, n)$, d'après (5) et (ET) deux fois

(7) $\neg memory(I, m)$, d'après (4) et (6)

- Si on a (8) $\neg Done^{-1}_{(j:s:\gamma);(q:\delta)} memory(I, m)$

(9) $\neg memory(I, m)$, d'après (1), (2), (8) et (PERS)

Donc on a (10) $counts(I, m, \phi, S) \rightarrow \neg memory(I, m)$, d'après (2), (7) et (9)

(11) $\forall m(counts(I, m, \phi, S) \rightarrow \neg memory(I, m))$, car on a (10) pour un m quelconque

(12) $\neg D_S ExpBel_I \phi$, d'après (11) et (DE1)

Donc on a (13) $\vdash Sup_{(j:s,J),I,S}\phi \rightarrow \neg D_S ExpBel_I \phi$, d'après (1) et (12)

Fin de preuve.

Le Théorème 10 signifie que l'action $Sup_{(j:s,J),I,S}\phi$ a pour effet que pour l'institution S l'agent I ne croit pas explicitement ϕ.

On adopte les notations suivantes.

(COUNT5) $E_{j:s:\gamma}E_{q:\delta}(\bigwedge_{n\in M_{(j:s:\gamma);(q:\delta),I,\phi,S}} \neg memory(I, n)) \Rightarrow_S E_J(\bigwedge_{n\in M_{J,I,\phi,S}} \neg memory(I, n))$

(COUNT6) $Done^{-1}_{(j:s:\gamma);(q:\delta)}(\exists\, m\, bel(I, m, \phi, S)) \Rightarrow_S Done^{-1}_J(\exists\, m\, bel(I, m, \phi, S))$

Theorem 11 *Si on a (COUNT5) et (COUNT6), alors on a :*
alors $\vdash Sup_{(j:s,J),I,S}\phi \rightarrow Sup_{J,I,S}\phi$.

Preuve.

La preuve est tout à fait similaire à la preuve du Théorème 9.

Fin de preuve.

Le Théorème 11 montre que sous les hypothèses (COUNT5) et (COUNT6)

5.4 Cas de deux agents humains

On considère que les deux agents n'agissent pas en tant que titulaire d'un rôle, car s'ils agissaient en tant que titulaire d'un rôle ils agiraient en tant que représentants d'agents institutionnels et on serait ramené au cas de la communication entre agents institutionnels.

Les actions d'insérer et de supprimer ne sont pas définies dans ce cas parce que l'agent qui agit ne peut pas faire plus que de transmettre un message à l'autre agent, il ne peut pas faire en sorte que les croyances de ce dernier soient modifiées, il ne peut que lui proposer de les modifier et cela est réalisé par une action de type "informer". D'autre part on suppose que pour communiquer les agents humains utilisent des agents logiciels.

$Inf_{i,j}\phi$: l'agent humain i a communiqué à l'agent humain j une information qui signifie ϕ pour i.

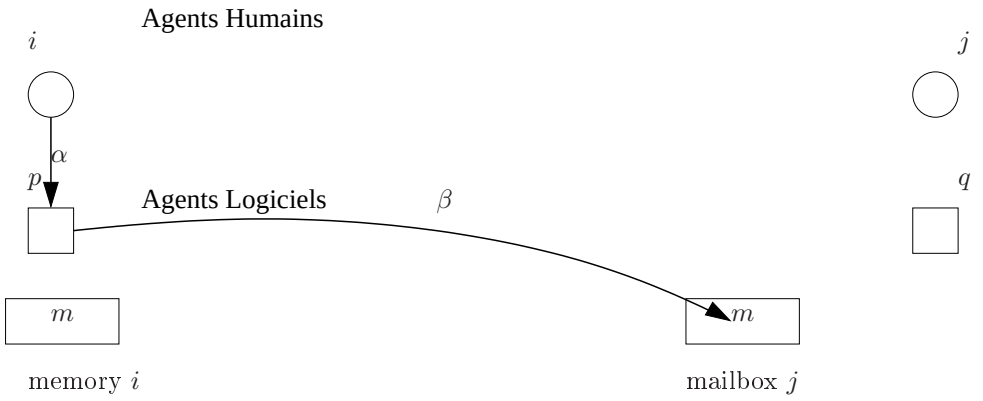

Figure 3: L'agent humain i informe j.

$$Inf_{i,j}\phi \stackrel{\text{def}}{=} \exists \alpha, \beta, p, m(Done^{-1}_{(i:\alpha);(p:\beta)} hbel(I, m, \phi) \wedge E_{i:\alpha} E_{p:\beta} mailbox(j, m))$$

La signification de la formule $E_{i:\alpha}E_{p:\beta}mailbox(j,m)$ est que l'agent i en réalisant l'action α a fait en sorte que l'agent logiciel p en réalisant l'action β a fait en sorte que le message m soit dans la mailbox de j. La formule $Done^{-1}_{(i:\alpha);(p:\beta)}hbel(I,m,\phi)$ signifie qu'avant que ces actions soient réalisées il y avait dans la mémoire de i un message qui signifie ϕ pour i. Par définition $hbel(I,m,\phi)$ dénote la formule $memory(i,m)\wedge sem(i,m,\phi)$ et $memory(i,m)$ exprime que i a mis sur le support physique qu'il utilise pour communiquer avec l'agent logiciel p le message m (voir la figure 3).

On accepte le schéma d'axiome suivant.

(PERSB') $(Done^{-1}_{(i:\alpha);(p:\beta)}hbel(I,m,\phi)\wedge E_{i:\alpha}E_{p:\beta}mailbox(j,m)) \rightarrow hbel(I,m,\phi)$

Theorem 12 *Si on a (PERSB'), alors on a :* $\vdash Inf_{i,j}\phi \rightarrow \exists m(hbel(I,m,\phi)\wedge mailbox(j,m))$

Preuve.
La preuve est tout à fait similaire à la preuve du Théorème 1.

Fin de la preuve.

On accepte les schémas d'axiomes suivants

(TRANS') $\exists\alpha,\beta,p,m(Done^{-1}_{(i:\alpha);(p:\beta)}hbel(I,m,\phi)\wedge E_{i:\alpha}E_{p:\beta}mailbox(j,m)) \rightarrow Know_j(\exists\alpha,\beta,p,m(Done^{-1}_{(i:\alpha);(p:\beta)}hbel(I,m,\phi)\wedge E_{i:\alpha}E_{p:\beta}mailbox(j,m)))$

(ACS') $Done^{-1}_{(i:\alpha);(p:\beta)}hbel(I,m,\phi)\wedge E_{i:\alpha}E_{p:\beta}mailbox(j,m)\wedge Know_j(\exists\alpha,\beta,p,m(Done^{-1}_{(i:\alpha);(p:\beta)}hbel(I,m,\phi)\wedge E_{i:\alpha}E_{p:\beta}mailbox(j,m))) \rightarrow Know_j(Done^{-1}_{(i:\alpha);(p:\beta)}hbel(I,m,\phi)\wedge E_{i:\alpha}E_{p:\beta}mailbox(j,m))$

(SEM') $Know_j(Done^{-1}_{(i:\alpha);(p:\beta)}hbel(I,m,\phi)\wedge E_{i:\alpha}E_{p:\beta}mailbox(j,m) \rightarrow Int_iKnow_jBel_i\phi)$

(SINC') $Know_j(Int_iKnow_jBel_i\phi \rightarrow Bel_i\phi)$

Theorem 13 *Si on a les (TRANS'), (ACS'), (SEM') et (SINC'), alors on a :*
$\vdash Inf_{i,j}\phi \rightarrow Know_jBel_i\phi$

Preuve.

(1) $Inf_{i,j}\phi$

(2) $\exists\alpha,\beta,p,m(Done^{-1}_{(i:\alpha);(p:\beta)}hbel(I,m,\phi)\wedge E_{i:\alpha}E_{p:\beta}mailbox(j,m))$, d'après (1) et la définition de Inf

(3) $Know_j(Done^{-1}_{(i:\alpha);(p:\beta)} hbel(I, m, \phi) \wedge E_{i:\alpha} E_{p:\beta} mailbox(j, m))$, d'après (2), (TRANS') et (ACS')

(4) $Know_j Bel_i \phi$, d'après (3), (SEM') et (SINC')

(5) $Inf_{i,j} \phi \rightarrow Know_j Bel_i \phi$, d'après (1) et (4)

Fin de la preuve.

6 Définition des normes

Nous allons voir dans cette section comment les normes portant sur la confidentialité et l'intégrité peuvent être exprimées en utilisant les définitions formelles des actes de communication que nous venons de voir dans la section précédente. Pour la confidentialité nous avons distingué le cas où on veut imposer des normes, soit à des agents institutionnels, soit à des agents humains. Pour l'intégrité nous n'avons considéré que des agents institutionnels car, comme on l'a déjà mentionné, on ne peut pas insérer ou supprimer des informations dans un agent humain. Cependant, les agents humains peuvent faire l'objet de normes relatives à l'intégrité mais en tant que représentants d'agents institutionnels.

6.1 Confidentialité

Le but des normes relatives à la confidentialité est de restreindre les connaissances d'un agent concernant les informations que détient (que connaît ou croit) un autre agent. Il y a alors deux questions qui se posent à ceux qui veulent définir les normes.

Première question : quelles sont les connaissances du destinataire d'un acte de communication que veut restreindre le législateur?

Deuxième question : quelles sont les actions qui peuvent avoir pour effets que le destinataire connaît ces informations, et sous quelles hypothèses ces effets peuvent-ils être obtenus?

6.1.1 Agents institutionnels

Réponse à la première question.

Ce sont des connaissances qui expriment que celui qui informe croit que

telle information est vraie, c'est-à-dire de la forme : $Know_J Bel_I \phi$, où I est l'agent qui informe et J l'agent qui est le destinataire.

Pourquoi pas : $Know_J Know_I \phi$? Parce qu'il peut se faire que le législateur veuille restreindre la diffusion de ce que I croit même si ce que I croit n'est pas justifié, ou même si ce que I croit est faux. Par exemple, si la signification de ϕ est : *le patient x a un cancer du type y*, il se peut que ce diagnostic soit faux, mais même si le diagnostic est faux tout le monde n'est pas nécessairement autorisé pour autant à savoir que I croit que ce diagnostic est vrai.

Pourquoi pas : $Bel_J Bel_I \phi$? Parce que le législateur n'a pas à restreindre les inférences fausses, ou sans justification, que peut faire le destinataire J à partir des actes de communication de I. Par exemple, si le législateur veut restreindre la transmission d'un diagnostic de cancer et que J croit, à tort, que la signification de m est : *le patient x a un cancer du type y*, tandis qu'en réalité la signification que I donne à m est : *le patient x a une infection du type z*, alors le législateur n'a pas de raison de restreindre la transmission du message m.

Pourquoi pas : $Bel_J Know_I \phi$? Pour les deux raisons précédentes.

Comme il s'agit ici d'agents institutionnels les connaissances et croyances ne sont pas des faits bruts mais des faits institutionnels qui sont formellement représentés par $D_S Know_J Bel_I \phi$ et non par $Know_J Bel_I \phi$.

Réponse à la deuxième question.

Ce sont les actions réalisées par I qui donnent la possibilité au destinataire J de savoir que I croit ϕ. Ce sont donc des actions de la forme $Inf_{I,J,S}\phi$, qui on pour effet, comme on l'a montré avec le Théorème 1 qu'il y a dans la mailbox de J un message qui pour l'institution S signifie ϕ. Si en plus les hypothèses (TRANS), (ACS), (SEM) et (SINC) sont satisfaites, alors, comme on l'a montré avec le Théorème 2, on a $D_S Know_J Bel_I \phi$. Comme le législateur doit envisager le "pire" des cas c'est bien les actions de la forme $Inf_{I,J,S}\phi$ qui doivent être interdites ou permises.

Interdiction

Les interdictions sont de la forme suivante.

(CI1) $Forb_S Inf_{I,J,S}\phi$

Si I a réalisé l'action $Inf_{I,J,S}\phi$, la norme (CI1) est violée par l'agent institutionnel I.

Si un agent humain i, en utilisant un agent logiciel p, réalise l'action $Inf_{(i:r,I),J,S}\phi$ et que l'on a les normes constitutives suivantes :

$\forall m(E_{i:r:\alpha}E_{p:\beta}mailbox(J,m) \Rightarrow_S E_I mailbox(J,m))$

$\forall m(Done^{-1}_{(i:r:\alpha);(p:\beta)}bel(I,m,\phi,S) \Rightarrow_S Done^{-1}_I bel(I,m,\phi,S))$

alors, d'après le Théorème 6, on a : $Inf_{I,J,S}\phi$, et donc il y a violation de (CI1) par I. Cependant, l'agent humain i n'a pas violé (CI1).

Pour avoir une meilleure garantie que la norme (CI1) soit respectée le législateur peut imposer à I l'obligation suivante.

(CI2) $Obg_S(prevent_{Inf}(I,J,\phi,S))$

où $prevent_{Inf}(I,J,\phi,S)$ est définie de la façon suivante :

$prevent_{Inf}(I,J,\phi,S)$: l'agent institutionnel I a pris les mesures de protection définies par les règles de l'art destinées à empêcher qu'aucun agent humain h ne puisse réaliser une action telle que $Inf_{(h:r,I),J,S}\phi$ (qui, d'après le Théorème 5, a pour effet $\exists m(bel(I,m,\phi,S) \wedge mailbox(J,m))$).

La norme (CI2) est violée par I si on a $\neg prevent_{Inf}(I,J,\phi,S)$, c'est-à-dire si I n'a pas mis en place toutes les mesures qui sont recommandées.

Cependant, même si ces mesures ont été mises en place, il se peut qu'un agent h réussisse à faire l'action $Inf_{(h:r,I),J,S}\phi$. Dans ce cas cet agent ne viole ni (CI1) ni (CI2). Si le législateur veux que h puisse être sanctionné il doit rajouter la norme suivante.

(CI3) $\forall h \quad Forb_S Inf_{(h:r,I),J,S}\phi$.

On notera que dans (CI3) l'agent h n'est pas nécessairement un représentant de I. Si, par exemple, I est un hôpital, h peut être un représentant de l'hôpital ou une personne employée par l'hôpital, mais qui ne le représente pas, ou toute autre personne qui n'est pas employée par l'hôpital.

La norme (CI3) est violée par h si on a $Inf_{(h:r,I),J,S}\phi$ et si h est un représentant de I il y a en plus violation de (CI1) par I.

Permission

Les permissions s'expriment de la façon suivante.

(CI4) $Perm_S Inf_{I,J,S}\phi$

Naturellement les permissions ne peuvent pas faire l'objet de violations.

6.1.2 Agent humain

On appelle i et j deux agents humains. Comme pour les agents institutionnels, si i est l'agent qui informe et j le destinataire, les situations que l'on veut réglementer avec les normes de confidentialité sont celles où l'on a : $Know_j Bel_i \phi$. On a vu que, d'après le Théorème 12, les actions qui sont susceptibles de conduire à une situation où on a $Know_j Bel_i \phi$ sont de la forme $Inf_{i,j}\phi$. On a vu aussi que, d'après le Théorème 13, ces situations sont effectivement atteintes si les hypothèses (TRANS'), (ACS'), (SEM') et (SINC') sont satisfaites.

Interdiction

Les interdictions sont de la forme suivante.

(CH1) $Forb_S Inf_{i,j}\phi$

Si i a réalisé l'action $Inf_{i,j}\phi$ la norme (CH1) est violée par l'agent humain i.

Permission

Les permissions s'expriment de la façon suivante.

(CH2) $Perm_S Inf_{i,j}\phi$

6.2 Intégrité

On appelle J l'agent institutionnel qui réalise une action ayant pour effet de modifier les connaissances ou croyances explicites de l'agent institutionnel I.

6.2.1 Agent institutionnel

Insertion.

On a choisi d'appeler "insérer" les actions qui, dans une situation où pour l'institution S l'agent I ne croit pas explicitement ϕ, ont pour effet que pour S l'agent I croit ϕ. Ces actions sont formellement représentées par $Ins_{J,I,S}\phi$ et on a vu, d'après le Théorème 3, que quand une action de ce type a été réalisée on a $D_S ExpBel_I \phi$ tandis qu'avant sa réalisation on avait $\neg D_S ExpBel_I \phi$.

Interdiction.

Les interdictions sont de la forme suivante.

(II1) $Forb_S Ins_{J,I,S}\phi$

Si J a réalisé l'action $Ins_{J,I,S}\phi$, la norme (II1) est violée par l'agent institutionnel J.

Si un agent humain j, en utilisant un agent logiciel q a réalisé l'action $Ins_{(j:s,J),I,S}\phi$ et que l'on a les normes constitutives suivantes :

$\forall m(E_{j:s:\gamma}E_{q:\delta}memory(I,m) \Rightarrow_S E_J memory(I,m))$

$\forall m(Done^{-1}_{(j:s:\gamma);(q:\delta)}(\neg bel(I,m,\phi,S)) \Rightarrow_S Done_J^{-1}(\neg bel(I,m,\phi,S)))$

alors, d'après le Théorème 9, on a $Ins_{J,I,S}\phi$ et donc on a violation de (II1) par J. Cependant l'agent humain j n'a pas violé (II1).

Pour avoir une meilleure garantie que la norme (CI1) soit respectée le législateur peut imposer à I l'obligation suivante.

(II2) $Obg_S prevent_{Ins}(I,J,\phi,S)$

$prevent_{Ins}(I,J,\phi,S)$: l'agent institutionnel I a pris les mesures de protection définies par les règles de l'art destinées à empêcher qu'aucun agent humain h ne puisse réaliser l'action $Ins_{(h:s,J),I,S}\phi$ (qui d'après le Théorème 8 a pour effet $D_S ExpBel_I\phi$).

La norme (II2) est violée par I si on a $\neg prevent_{Ins}(I,J,\phi,S)$.

Même si la norme (II2) est respectée il se peut qu'un agent h réussisse à faire l'action $Ins_{(h:s,J),I,S}\phi$. Cet agent ne viole ni (II1) ni (II2). Si le législateur veux que h puisse être sanctionné il doit rajouter la norme suivante.

(II3) $\forall h \;\; Forb_S Ins_{(h:s,J),I,S}\phi$.

On notera que dans (II3) h peut être n'importe quel agent, il n'est pas nécessairement un représentant de J.

La norme (II3) est violée par h si on a $Ins_{(h:s,J),I,S}\phi$ et si h est un représentant de J il y a également violation de (II1) par J.

Permission.

Les permissions s'expriment de la façon suivante.

(II4) $Perm_S Ins_{J,I,S}\phi$

Suppression.

On a choisi d'appeler "supprimer" les actions qui, dans une situation où pour l'institution S l'agent I croit explicitement ϕ, ont pour effet que pour S l'agent I ne croit pas explicitement ϕ. Ces actions sont formellement représentées par $Sup_{J,I,S}\phi$ et on a vu, d'après le Théorème 4, que quand

une action de ce type a été réalisée on a $\neg D_S ExpBel_I \phi$ tandis qu'avant sa réalisation on avait $D_S ExpBel_I \phi$.

Les interdictions sont de la forme suivante.

(SI1) $Forb_S Sup_{J,I,S} \phi$

Si J a réalisé l'action $Sup_{J,I,S}\phi$, la norme (SI1) est violée par l'agent institutionnel J.

Si un agent humain j, en utilisant un agent logiciel q, a réalisé l'action $Sup_{(j:s,J),I,S}\phi$ et que l'on a les normes constitutives suivantes :

$E_{j:s:\gamma} E_{q:\delta}(\bigwedge_{n \in M_{(j:s:\gamma);(q:\delta),I,\phi,S}} \neg memory(I,n)) \Rightarrow_S E_J(\bigwedge_{n \in M_{J,I,\phi,S}} \neg memory(I,n))$

$Done^{-1}_{(j:s:\gamma);(q:\delta)}(\exists\ m\ bel(I,m,\phi,S)) \Rightarrow_S Done^{-1}_J(\exists\ m\ bel(I,m,\phi,S))$

alors, d'après le Théorème 11, on a $Sup_{J,I,S}\phi$ et on a violation de (SI1) par J. Cependant, l'agent humain j n'a pas violé (SI1).

Pour avoir une meilleure garantie que la norme soit (SI1) soit respectée le législateur peut imposer à I l'obligation suivante.

(SI2) $Obg_S prevent_{Sup}(I,J,\phi,S)$

$prevent_{Sup}(I,J,\phi,S)$: l'agent institutionnel I a pris les mesures de protection définies par les règles de l'art destinées à empêcher qu'aucun agent humain h ne puisse réaliser l'action $Sup_{(h:s,J),I,S}\phi$ (qui d'après le Théorème 10 a pour effet $\neg D_S ExpBel_I \phi$).

La norme (SI2) est violée par I si on a $\neg prevent_{Ins}(I,J,\phi,S)$.

Même si la norme (SI2) est respectée il se peut qu'un agent h réussisse à faire l'action $Sup_{(h:s,J),I,S}\phi$. Cet agent ne viole ni (SI1) ni (SI2). Si le législateur veux que h puisse être sanctionné il doit rajouter la norme suivante.

(SI3) $\forall h \quad Forb_S Sup_{(h:s,J),I,S}\phi$.

On notera que dans (SI3) h peut être n'importe quel agent, il n'est pas nécessairement un représentant de J.

La norme (SI3) est violée par h si on a $Sup_{(h:s,J),I,S}\phi$ et si h est un représentant de J il y a également violation de (SI1) par J.

Permission.

Les permissions s'expriment de la façon suivante.

(SI4) $Perm_S Sup_{J,I,S}\phi$

7 Conclusion

Nous avons défini les actions de communication qui peuvent faire l'objet de normes de confidentialité ou d'intégrité. Ces actions sont réalisées par des agents humains qui peuvent éventuellement utiliser des agents logiciels et elles peuvent compter comme des actions réalisées par des agents institutionnels.

Nous avons défini un cadre formel pour exprimer les notions qui interviennent dans ces normes. Il s'agit d'une logique multi modale propositionnelle. Les particularités de cette logique sont qu'elle permet de distinguer les croyances explicites et implicites des agents institutionnels, les actions et les effets dont ces actions sont la cause, les obligations, interdictions et permissions, la notion de counts as et la modalité associée qui exprime qu'une proposition est vraie du point de vue d'une institution.

Nous avons donné la priorité à la définition de la sémantique du langage car c'est elle qui sert de référence pour comprendre la signification des modalités utilisées. Nous avons donné aussi une définition partielle de l'axiomatique associée mais la définition d'une axiomatique complète pour cette sémantique reste un problème ouvert.

La correspondance entre les actions réalisées par les agents logiciels, humains et institutionnels est exprimée avec l'opérateur de counts as. Ces correspondances sont justifiées soit par la relation de causalité pour la correspondance entre actions réalisées par les agents logiciels et celles réalisées par les agents humains, soit par la notion de rôle pour la correspondance entre celles réalisées par les agents humains et celles réalisées par les agents institutionnels, et ces justifications peuvent être exprimées dans cette logique.

Nous avons défini trois types d'actions de communication : informer, insérer et supprimer, et nous avons donné des théorèmes qui expriment les propriétés des effets de ces actions. Certains de ces effets ne sont obtenus que si certaines conditions sont satisfaites et nous avons explicité ces conditions.

Ce cadre formel a été utilisé ensuite pour exprimer les normes de confidentialité et d'intégrité. Pour cela nous avons distingué les situations qu'un législateur peut vouloir interdire ou permettre et les actions qui conduisent, ou qui sont susceptibles de conduire, à ces situations. Ces normes ont été définies tout d'abord pour les agents institutionnels et également pour les agents humains. L'une des idées directrices est que les agents institutionnels font l'objet d'obligations qui leur imposent de mettre en place les mesures

qui réduisent la possibilité de violer les normes.

En conclusion, nous avons proposé un cadre formel qui pourrait être utilisé par les législateurs pour définir les normes de confidentialité et d'intégrité qui leur paraissent appropriées et qui, d'autre part, pourrait être utilisé par les informaticiens pour vérifier que les agents logiciels qu'ils réalisent satisfont ou non ces normes.

Il reste cependant de nombreuses questions ouvertes qui pourraient faire l'objet de recherches ultérieures. Nous en mentionnons plusieurs dans ce qui suit.

Les normes qui ont été présentées s'appuient sur des choix qui peuvent être discutés et pourraient être remis en question par des législateurs.

Par exemple, les normes proposées visent à contrôler des actions qui sont la cause de certains effets qui, soit sont susceptibles de modifier les croyances des agents, soit modifient effectivement ces croyances. Les effets causés par ces actions sont rappelés brièvement ci-dessous.

Effet causé par l'action d'informer : il y a un message dans la mailbox de J qui signifie ϕ et qui donne la possibilité à J de savoir que I croit ϕ. Si les hypothèses que l'on a vu ne sont pas satisfaites, il se peut que l'action d'informer n'ait pas pour effet que J sait que I croit ϕ. Cependant, I aura violé la norme. Est-ce bien ce que veux le législateur?

Effet causé par l'action d'insérer : il y a un message dans la mémoire de I qui signifie ϕ alors qu'il n'y en avait aucun auparavant. Si l'agent n'avait rien fait il n'y aurait eu aucun message qui signifie ϕ et I n'aurait pas cru explicitement ϕ. Donc l'action de J est la cause du fait que I croit explicitement ϕ. Pour savoir qu'il a réalisé l'action d'insérer il est nécessaire avant qu'il réalise l'action que J sache qu'il n'y a aucun message dans la mémoire de I qui signifie ϕ. S'il insère un message qui signifie ϕ alors qu'il y en a déjà un, il n'a pas fait l'action insérer et donc il ne viole pas la norme. Est-ce bien ce que veux le législateur?

Effet causé par l'action de supprimer : il n'y a plus de message dans la mémoire de I qui signifie ϕ alors qu'il y en avait au moins un avant. Si l'agent n'avait rien fait il y aurait eu un message qui signifie ϕ et I aurait cru explicitement ϕ. Donc l'action de J est la cause du fait que I ne croit pas explicitement ϕ. Pour pouvoir réaliser l'action supprimer J doit savoir quels sont les messages qui sont dans la mémoire de I et qui signifient ϕ. S'il ne le sait pas et qu'il n'a supprimé que certains de ces messages, il n'a pas fait l'action de supprimer et il n'a pas violé la norme. Est-ce bien ce que veux le

législateur?

D'autre part, dans le cas des croyances des agents institutionnels, nous n'avons considéré que des normes qui s'appliquent aux actions qui modifient les croyances explicites. Faut-il imposer des normes qui s'appliquent aux croyances implicites et, dans ce cas, quelles sont les hypothèses qu'il faut faire sur les capacités de raisonnement de ces agents?

D'autre questions se posent concernant les relations entre les normes.

La première est : faut-il accepter des normes d'un niveau supérieur qui expriment comment les normes qui s'appliquent à des agents institutionnels se "propagent" en d'autres normes qui s'appliquent à des agents humains qui sont titulaires de tel ou tel rôle dans les agents institutionnels?

La seconde est liée aux risques de contradictions entre des normes qui donnent des permissions à certains agents (par exemple la permission qu'à l'agent J d'insérer un message dans la mémoire de I) et les obligations faites à d'autres agents de mettre en place des mesures qui empêchent certains agents de faire certaines actions (par exemple l'obligation qu'à l'agent I de mettre en place des mécanismes qui visent à empêcher les agents qui n'en ont pas la permission d'insérer ce message, mais qui pourraient aussi empêcher J d'insérer ce message alors qu'il en a la permission). Cette question rejoint la question plus générale de la compatibilité entre les mesures qui visent à garantir la sécurité et les normes qui définissent les libertés.

References

[1] L. Aqvist. Old foundations for the logic of agency and action. *Studia Logica*, 72, 2002.

[2] J. Carmo and O. Pacheco. Deontic and action logics for organized collective agency, modeled through institutionalized agents and roles. *Fundamenta Informaticae*, 48:129–163, 2001.

[3] B. F. Chellas. *Modal Logic: An introduction.* Cambridge University Press, 1988.

[4] F. Cuppens. Roles and Deontic Logic. In A. J. I. Jones and M. Sergot, editors, *Second International Workshop on Deontic Logic in Computer Science*, Oslo, Norway, 1994.

[5] R. Demolombe. Relationships between actions performed by institutional agents, human agents or software agents. In Guido Governatori and Giovanni Sartor, editor, *Proceedings of the 10th International Conference on Deontic Logic in Computer Science, LNAI 6181*. Springer Verlag, 2010.

[6] R. Demolombe and V. Louis. Norms, institutional power and roles: toward a logical framework. In F. Esposito, Z. W. Ras, D. Malerba, and G. Semeraro, editors, *Foundations of Intelligent Systems*. Springer, LNAI 4203, 2006.

[7] M. Esteva, J. Rodríguez-Aguilar, J. Lluis Arcos, C. Sierra, P. Noriega, and B. Rosell. Electronic Institutions Development Environment. In *Proceedings of the 7th international joint conference on Autonomous Agents and Multiagent Systems (AAMAS08)*. 2008.

[8] D. Grossi. *Desigining Invisible Hancuffs. Formal Investigations in Institutions and Organizations for Multi-Agent Systems*. PhD thesis, Utrecht University, 2007.

[9] D. Grossi, J-J. Ch. Meyer, and F. Dignum. The Many Faces of Counts-as: A Formal Analysis of Constitutive Rules. *Journal of Applied Logic*, 6, 2008.

[10] R. Hilpinen. On Action and Agency. In E. Ejerhed and S. Lindstrom, editors, *Logic, Action and Cognition: Essays in Philosophical Logic*. Kluwer, 1997.

[11] J.F. Horty and N. Belnap. The deliberative STIT: a study of action, omission, ability, and obligation. *Journal of Philosophical Logic*, 24:583–644, 1995.

[12] A. J. Jones and M. Sergot. A formal characterisation of institutionalised power. *Journal of the Interest Group in Pure and Applied Logics*, 4(3), 1996.

[13] I. Pörn. Action Theory and Social Science. Some Formal Models. *Synthese Library*, 120, 1977.

[14] A.S. Rao and M.P. Georgeff. Modeling Rational Agents within a BDI Architecture. In *Proc. 2nd Int. Conf. on Knowledge Representation and Reasoning*. Morgan Kaufmann, 1991.

[15] F. Santos and O. Pacheco. Specifying and reasoning with institutional agents. In *Proceedings of ICAIL*, 2003.

[16] G. Sartor. *Legal Reasoning: A Cognitive Approach to the Law*. Springer, Berlin, 2005.

[17] J. R. Searle. *Speech Acts: An essay in the philosophy of language*. Cambridge University Press, New-York, 1969.

[18] K. Segerberg. Bringing it about. *Journal of Philosophical Logic*, 18:327–347, 1989.

[19] K. Segerberg. Outline of a logic of action. In F. Wolter, H. Wansing, W. de Rijke, and M. Zakharyaschev, editors, *Advances in Modal Logic, Volume 3*. World Scientific Publishing Co., 2002.

[20] G. H. von Wright. *Norm and Action*. Routledge and Kegan, 1963.

[21] F. Lopez y Lopez, M. Luck, and M. d'Inverno. Normative agent reasoning in dynamic societies. In *Proceedings of the Third International Conference on Autonomous Agents and Multi-Agent Systems*. IEEE Computer Society, 2004.

Intégration du tatouage piloté par une politique de sécurité OrBAC dans la plateforme SELKIS

Contents

Abstract

Le projet SELKIS a pour objectif de développer des solutions d'analyse et de conception de Systèmes d'Information (SI) sécurisés qui abordent les aspects fonctionnels et sécuritaires dès les premières étapes de leur conception. L'approche SELKIS convient à une grande variété de systèmes d'information et dans ce projet, elle est appliquée à des SI de santé. Elle doit assurer des propriétés de sécurité à différentes étapes du cycle de vie du système et des logiciels : la disponibilité, la confidentialité, leur intégrité ainsi que la traçabilité qui permet d'établir les responsabilités notamment lors d'opérations de modification ou de mise à jour.

Les solutions proposées par le projet SELKIS font appel à des méthodes formelles, permettant ainsi l'utilisation de techniques de vérification et de preuve qui assurent d'une part la cohérence et l'adéquation des politiques de sécurité par rapport aux spécifications fonctionnelles et d'autre part la cohérence de l'implémentation par rapport à la spécification. Le projet repose sur des méthodes et techniques existantes qui ont fait leur preuve chacune dans son domaine respectif. Par exemple, les politiques de sécurité sont décrites avec le modèle OrBAC qui permet, comme expliqué et argumenté dans le chapitre 3, la prise en compte d'une grande variété d'exigences de sécurité en termes d'accès et d'usage.

En tant qu'un des partenaires de ce projet, l'une des tâches de notre groupe (Télécom Bretagne) est de valider cette approche formelle sur une étude de cas du domaine médical. Il s'agit du partage et de l'échange sécurisés d'informations médicales dans la plateforme du service d'Echo-Doppler du CHRU de Brest. L'objectif est de rendre des données des patients (image médicale, compte rendu, etc.) accessibles et modifiables à distance par les médecins autorisés. Notamment, afin de protéger les images médicales en cas de partage et d'échange entre les différentes organisations, nous utilisons les solutions de tatouage développées dans le cadre de nos travaux de thèse pour contrôler leur intégrité, leur authenticité et l'usage que les utilisateurs en font.

1. Méthodologie

La conception et la réalisation de systèmes sûrs nécessitent, tout d'abord, de définir précisément les politiques de sécurité correspondantes. Pour ce faire, nous nous appuyons sur la méthode EBIOS (Expression des Besoins et Identification des Objectifs de Sécurité) [EBI 10] qui permet d'évaluer et de traiter les risques relatifs à la sécurité des systèmes d'information. Cette méthode permet d'identifier des objectifs et exigences de sécurité en appliquant une analyse de risques et facilite ainsi la définition de la politique de sécurité. Notons que les bases de connaissances d'EBIOS décrivent des types d'entités, des méthodes d'attaques, des vulnérabilités, des objectifs de sécurité et des exigences de sécurité. Elles sont directement applicables à la plupart des secteurs, mais chacun peut aisément se les approprier et les adapter à son contexte particulier.

Ainsi, nous l'avons adaptée à notre contexte au travers des sept étapes suivantes : 1) description du système, 2) identification des fonctions et des ressources à protéger, 3) identification des besoins de sécurité, 4) analyse des menaces et des vulnérabilités, 5) identification des objectifs et des exigences de sécurité, 6) définition des règles de sécurité et 7) modélisation formelle.

Il est important de souligner qu'EBIOS n'est pas un catalogue de solutions ou de règles de sécurité prêtes à l'emploi. Son apport s'arrête à l'étape 5. Il ne permet donc pas de fournir de solutions immédiates aux problèmes de sécurité.

Pour aider à la définition des règles de sécurité, il est nécessaire dans un premier temps d'établir un scénario représentatif et de préciser les lois en vigueur. Le scénario décrit ne peut être pertinent vis-à-vis de la sécurité que s'il permet d'exprimer :

- les éléments de base de la politique de sécurité, notamment la classification des utilisateurs (rôles, groupes), les types d'accès possibles, les ressources, etc.
- les règles de description du fonctionnement du système, afin de pouvoir en déterminer l'impact sur les objectifs de sécurité. Il s'agit de spécifier les flux d'informations, les processus, la hiérarchie de rôles ainsi que les contraintes organisationnelles, etc.

Sur cette base, nous pouvons identifier clairement les informations à protéger, les menaces, les objectifs de sécurité, afin de définir les règles de sécurité qui permettent de satisfaire ces objectifs, de faire face aux menaces, et d'assurer la protection du système. Seuls les éléments essentiels sont présentés ici pour illustrer notre approche. Ainsi, certaines règles de sécurité, notamment celles concernant le tatouage sont exprimées formellement, avant implémentation dans le système.

2. Présentation des cas d'utilisation du service d'écho-doppler du CHRU de Brest

Dans le cadre du projet SELKIS, deux scénarios d'utilisation du SI du Service d'Echo-Doppler (SED) du CHRU de Brest ont été analysés. Le premier concerne la prise en charge d'un patient pour un examen d'échographie à la demande d'un service du CHRU ou d'un praticien de ville et le second est la demande d'expertise. Dans le cadre de ce manuscrit de thèse, nous nous limitons au premier scénario pour illustrer notre solution. Avant de le décrire, nous présentons le service d'écho-doppler et l'architecture du SI de l'hôpital.

2.1. Description du SED

Le SED est un plateau technique d'imagerie situé au CHRU de Brest. Il est constitué de trois salles d'examens d'échographie, d'un secrétariat, d'une salle d'attente, de deux salles de consultation et une salle d'interprétation. Le personnel est constitué de deux médecins permanents, une secrétaire, des médecins internes ou en formation. les tâches assurées par ce service sont : 1- diagnostiquer les pathologies des vaisseaux périphériques, 2- former des médecins et des internes et 3- maintenir des astreintes – médecine de garde. Le SED réalise 1500 consultations et 6500 examens par an. Le service répond aux demandes d'examen qui émanent d'autres services de l'hôpital et de praticiens de ville. Un examen consiste en une exploration à l'aide d'un échographe (*cf.* Figure 2.1). Au cours de l'examen, le médecin opérateur enregistre des images significatives qui sont ensuite transmises au système d'archivage (PACS). Les images sont imprimées sur une feuille de papier, à partir de la console de visualisation connectée à l'archivage, pour preuve légale de l'examen. Le médecin opérateur dicte sont compte rendu à la secrétaire en se basant sur les données collectées au cours de l'examen. Au final, les images d'intérêt et le compte rendu sont transmis au patient et au médecin prescripteur.

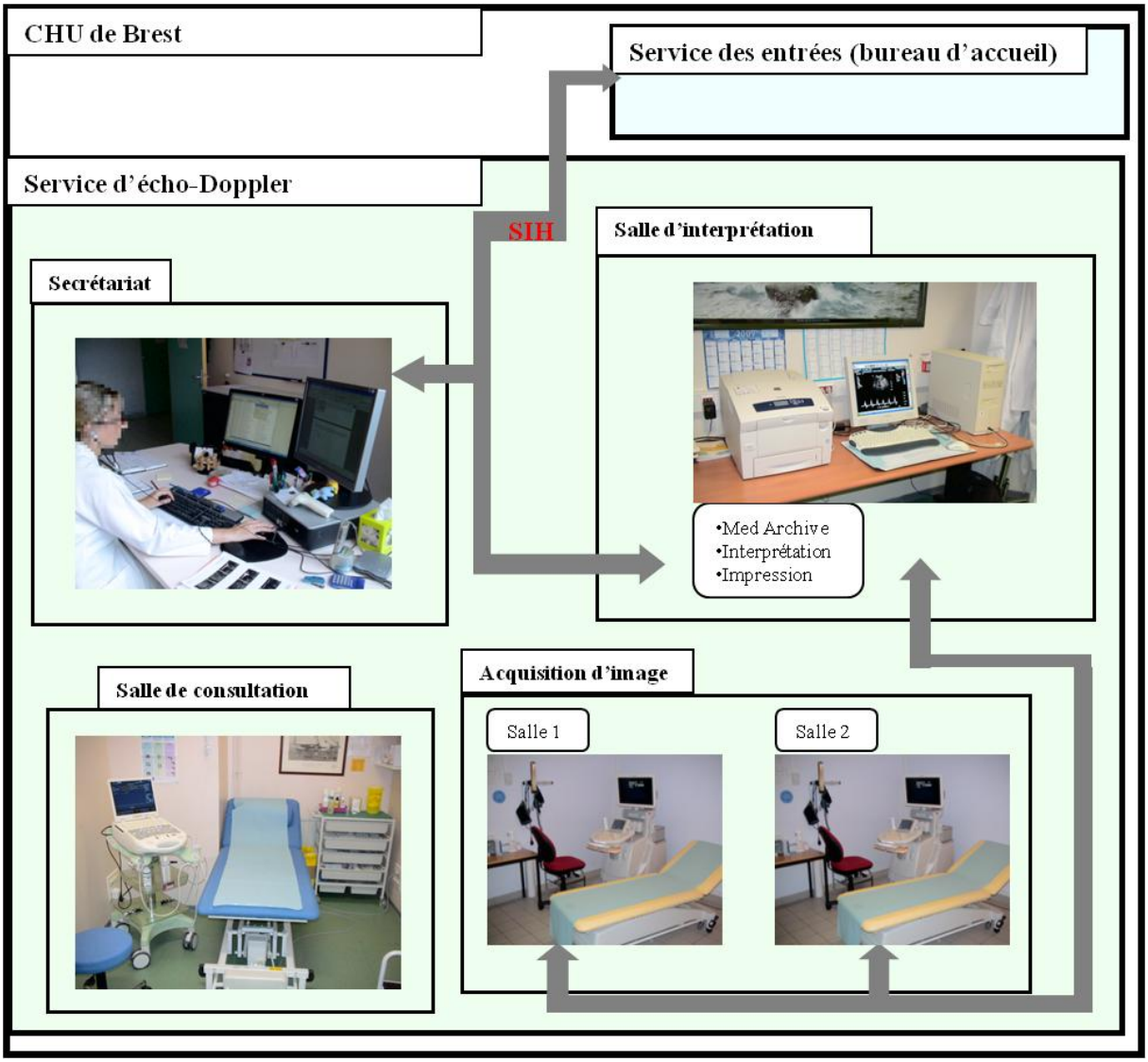

Figure 2.1 - *Présentation du SED*

2.2. Architecture du SI du SED

Le système d'information du SED est connecté au Système d'Information Hospitalier (SIH), SIH qui permet la gestion des rendez-vous, des comptes rendus via QPlanner / QDoc de AGFA[1] et la gestion du dossier patient via SUSIE, le portail IPS de McKESSON[2], etc. Ces logiciels ne communiquent pas tous ensembles. Le SI du SED inter-connectent également des échographes PHILIPS, l'archivage et la console d'interprétation MEDECOM[3]. Cette configuration impacte fortement le flux d'information et les méthodes de travail dans le service.

L'architecture du réseau actuel du service d'Echo-Doppler CHRU est décrite dans la Figure 2.2. On y retrouve connectés en LAN, un PACS qui permet l'acquisition et l'archivage des images, dans ce cas sous forme d'objets DICOM et le SIH.

[1] http://www.agfa.com/he/france/fr/internet/main/
[2] https://www.mckesson.fr/
[3] http://www.medecom.fr/

Actuellement, l'accès Internet par le médecin prescripteur au serveur de résultats n'est pas disponible. L'ouverture sécurisée à son réseau sera mise en place dans un futur proche. Quoiqu'il en soit, dans cet environnement, il nous faut suivre les procédures de sécurité et les politiques d'échange de données prescrites par la DSIS (le service en charge de l'informatique au CHRU).

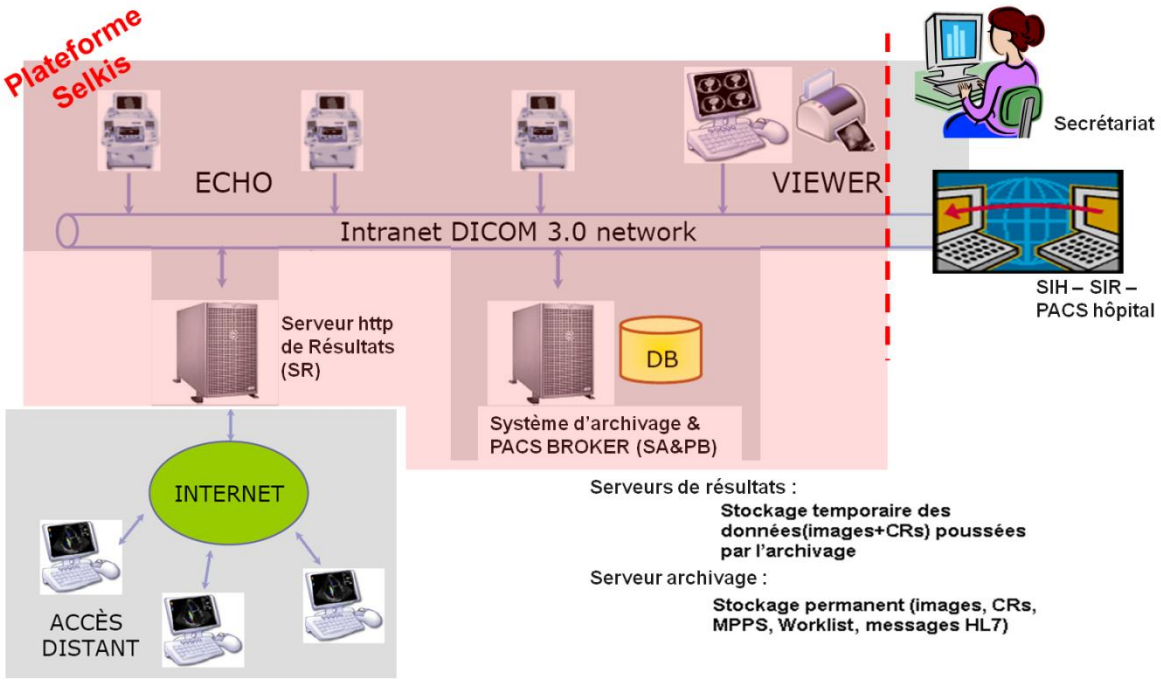

Figure 2.2 - *L'Architecture du SI du SED*

2.3. Cas d'un examen normal

Dans ce scénario, la demande d'examen peut émaner d'un service de l'hôpital ou d'un médecin de ville. Si pour la première situation, le patient est connu du CHRU, cela n'est pas forcément le cas pour la seconde. Pour homogénéiser ces deux procédures qui par la suite se déroulent de la même manière, nous considérons que le patient est passé par le service des admissions de l'hôpital et en conséquence qu'il est clairement identifiable par le SIH) de l'hôpital. Ainsi, la demande d'examen est faite directement au secrétariat du SED soit par FAX, soit par téléphone.

Les grandes étapes du scénario, qui va de la demande d'examen à la mise à disposition des résultats, sont les suivantes :

1. le secrétariat gère le planning et prend en compte de la demande de rendez-vous. L'hypothèse diagnostique est notée et l'examen à réaliser pour y répondre identifié. Le rendez-vous est pris.
2. le patient se présente à la date convenue et il est pris en charge par le personnel infirmier,
3. l'examen est réalisé par un médecin opérateur dans la salle d'examen à l'aide de l'échographe,
4. les images d'intérêt sont imprimées et le CR enregistré par le médecin opérateur,
5. le CR est saisi au secrétariat et signé par le médecin opérateur senior présent dans le service avant la transmission d'une copie au patient et au médecin prescripteur.

Un point important dans ce scénario concerne le médecin opérateur qui peut être un praticien hospitalier senior, un praticien hospitalier junior ou un interne. Dans ces deux derniers cas, il ne peut signer le CR et doit donc présenter le CR au praticien hospitalier senior présent dans le service.

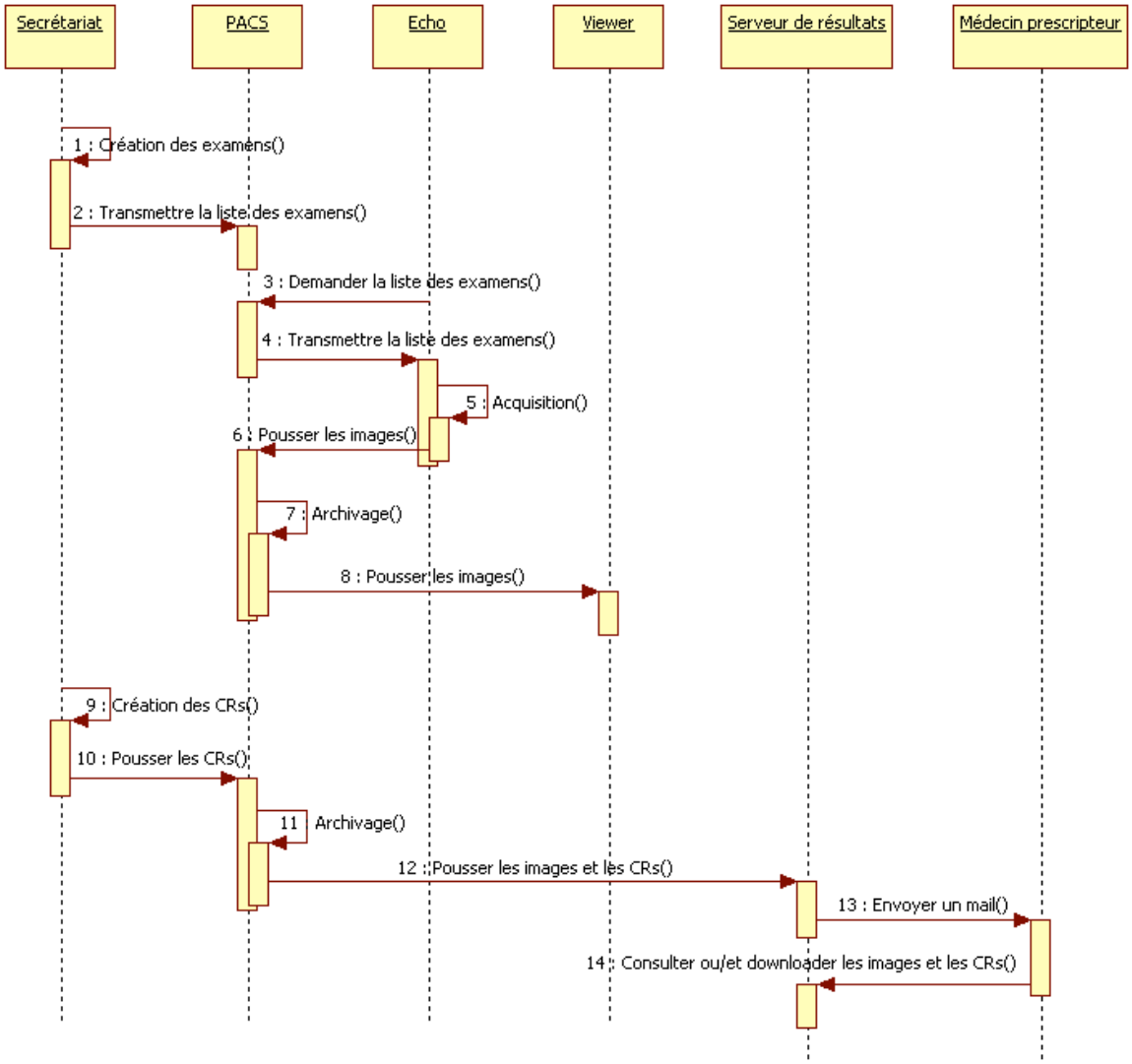

Figure 2.3 - *Diagramme de séquences pour un examen normal.*

A ce scénario, nous pouvons également associer le diagramme de séquences décrit dans la Figure 2.3. En parallèle, les flux de données suivants sont à considérer :

1. la saisie du rendez-vous au secrétariat alimente une liste d'examens (« worklist »). Cette liste indique entre autre le type d'examen, le nom du patient, le nom du médecin opérateur, le nom du médecin prescripteur, le nom de l'échographe – donc la salle d'examen -, le jour et l'heure, etc.,
2. les modalités interrogent la liste des examens pour obtenir leurs examens planifiés,
3. après acquisition sur l'échographe, les images sont poussées automatiquement vers le PACS (y compris le serveur d'archivage),
4. le PACS fait suivre automatiquement les images sur la station de diagnostic,
5. le CR est saisi au niveau du secrétariat sur un logiciel adapté comme QDoc (i.e. en lien avec le SIH),

6. le CR, lorsque validé par le praticien, est poussé automatiquement du SIH sur le PACS,
7. les images et le CR sont automatiquement mis à disposition via le serveur de résultats, accompagné de l'envoie d'un courriel au médecin prescripteur pour l'informer de la disponibilité des résultats,
8. le médecin prescripteur peut, à tout moment, se connecter sur le serveur de résultats et consulter ou/et télécharger les images et le CR de ses patients, sans attendre le dossier papier.

Notons que qu'ici, les images sont créées par les modalités d'acquisition sur l'échographie. Elles sont mises à disposition sur le serveur de résultats, afin de permettre le partage des données avec les médecins extérieurs du service (i.e. le médecin prescripteur). Une fois que les images ont été téléchargées par le médecin prescripteur, il pourra les envoyer aux autres médecins pour par exemple effectuer un travail en groupe. Dans ce cas il s'agit d'échange.

2.4. Identification des fonctions et des ressources à protéger

Cette étape consiste à identifier les fonctions (ou les activités) et les ressources entrant dans le champ de l'analyse des risques.

A. Les fonctions

Le recensement des activités médicales, administratives, logistiques,…, entrant dans le domaine de sécurité peut se faire sur la base d'une identification d'activités se rattachant à des grands processus métiers mais il peut être fait de manière plus spécifique. A partir d'un des scénarios décrits précédemment, nous avons identifié les fonctions suivantes :

- Gestion des rendez-vous (RdV)
- Transmission
    - de la liste des examens du poste de la secrétaire à la modalité d'acquisition (échographie).
    - des images de la modalité d'acquisition au serveur d'archivage (SA) et de l'archivage au serveur de résultats (SR).
    - du compte-rendu (CR) du SIH à l'archivage (SA) et de l'archivage au serveur de résultats.
    - des données de SR vers le SA.
- Acquisition d'image par la modalité d'acquisition
- Rédaction du CR au niveau SIH
- Consultation
    - des images
    - de l'historique
    - du CR
- Archivage
    - du CR
    - de l'image
- Envoi
    - des mails du SR vers le médecin extérieur
- Télécharger
    - des images par PS distant

    - du CR par PS distant
- Recueil et conservation du consentement du patient

B. Identification des ressources

Habituellement, les ressources d'infrastructure informatique regroupent les matériels et périphériques informatiques, les logiciels et de réseaux de télécommunication, comme les ressources humaines qui les organisent et les mettent en œuvre. La caractérisation du système comprend une identification des ressources utilisées par l'activité, à un niveau de maille cohérent avec celui des activités. Dans le cas du service SED, nous avons identifié les ressources suivantes :

- Modalité échographie
- Poste et applications autres personnels (poste secrétaire, bureau d'accueil (SIH))
- Réseaux de communication LAN
- Réseaux de communication Internet
- Frontal WEB (serveur de résultats)
- Station d'interprétation / de Post-traitement
- Archivage
- Base de données (SIR/SIH)
- agenda (RDV, Qplanning)
- worklist
- Identifiant du patient (IPP)
- Image
- Compte-rendu
- Support de sauvegarde (amovible) (DVD)
- Journaux de traces (sécurité)
- Annuaire utilisateurs en local (LDAP)
- annuaire utilisateurs distants
- consentement du patient

2.5. Identification des besoins de sécurité

Notons que chaque élément essentiel (fonction et ressource) a des besoins de sécurité qui s'expriment en termes de DICP : Disponibilité, Intégrité, Confidentialité et Preuve (ou Traçabilité). A l'aide de l'échelle de caractérisation des besoins de sécurité présentée dans le Tableau 2.1, nous pouvons évaluer les besoins de sécurité relatifs aux éléments essentiels. Notons que cette échelle de caractérisation a été établie par le GMSIH dans le contexte de sa politique de sécurité cadre.

NIVEAU de risque	DISPONIBILITE	INTEGRITE	CONFIDENTIALITE	PREUVE & CONTROLE
1 courant	**Interruption > à 1 journée** Une indisponibilité des informations est acceptée sous réserve qu'elle ne remette pas en cause le service fourni.	**Signalement** La perte d'intégrité momentanée des informations est acceptée, sous réserve qu'elle soit signalée et ne remette pas en cause le service fourni	**Public** Les informations peuvent être lues par tous	**Faible** Les éléments d'auditabilité sont faibles et non immédiatement disponibles Exemple : journalisation d'accès
2 Sensible	**Interruption < ou = 1 journée** Indisponibilité tolérée sous réserve qu'elle soit momentanée, signalée et sans conséquence pour le service fourni	**Signalement et correction** Perte tolérée si signalée et corrigée dans un délai suffisant pour ne pas avoir de conséquence grave sur len service fourni	**Restreint** Les informations sont diffusées ou accessibles par des populations identifiées et contrôlables	**Auditable** Les éléments de traçabilité des opérations existent et peuvent être rendus disponibles
3 Majeur (ou critique)	**Interruption < ou = 1 / 2 journée** Les informations doivent toujours être fournies pour remplir le service attendu	**Justification a posteriori** Les informations doivent rester intègres pendant la période d'utilisation ; toute perte en dehors de la période d'utilisation doit être signalée et corrigée ; si la perte d'intégrité est constatée pendant la période d'utilisation, le service ou traitement est arrêté jusqu'au rétablissement de l'intégrité	**Secret médical ou professionnel** Les informations sont protégées par le secret médical ou le secret professionnel et par la législation sur les données à caractère personnel et médical. **Données nominatives non médicales** Les informations sont protégées par la législation sur la protection des données nominatives	**Preuve interne** Fourniture d'une preuve opposable (mais contestable)
4 Stratégique (ou vital)	**Interruption comprise entre 15' et 1h** Les informations doivent en permanence être accessibles et utilisables par tous les services concernés	**Certification a priori** Les informations sont certifiées intègres pendant toute leur durée de vie ou leur période de validité	**Haute protection** Le secret médical est renforcé Exemple : informations dont la prise de connaissance non autorisée entraîne nécessairement un préjudice pour la personne concernée (ex : certaines pathologies)	**Preuve externe** Fourniture d'une preuve incontestable

Tableau 2.1 - *Échelle d'évaluation des besoins de sécurité.*

Les Tableaux 2.2 et 2.3 illustrent notre évaluation en DICP de quelque fonction et ressources. Il est important de souligner que l'image, la cible à laquelle nous nous intéressons, a des besoins qui sont tous de niveau « vital » (niveau 4). C'est-à-dire quelles images doivent en permanence être accessibles et utilisables par tous les services concernés (D), qu'elles sont certifiées intègres pendant toute leur durée de vie ou leur période de validité (I), que le secret médical est renforcé (C) ; qu'elles doivent être associées à des preuves incontestables en justice en cas de litige (ex : signature électronique).

Besoins de sécurité sur les fonctions	D	I	C	P
Gestion RDV	3/4	4	3/4	3
Transmission d'image	4	4	3/4	3
Consultation de l'image	3/4	4	3/4	3
Rédaction du compte-rendu (CR)	3/4	4	3/4	3
Transmission du CR	3/4	4	3/4	3

Tableau 2.2 - *Besoin de sécurité sur certaines fonctions.*

Besoins de sécurité sur les ressources	D	I	C	P
Réseau de communication (LAN, Internet)	4	3	4	3
Image	4	4	4	4
Compte-rendu (CR)	3	4	4	4
Identité du patient	4	4	4	3

Tableau 2.3 - *Besoin de sécurité sur certaines ressources.*

2.6. Analyse des menaces et des vulnérabilités

Une fois les éléments sensibles déterminés, les risques sur chacun de ces éléments peuvent être estimés en fonction de la combinaison de menaces qui pèsent sur eux et des vulnérabilités inhérentes à ces ressources. Parmi les 42 menaces génériques recensées par EBIOS, nous en avons retenu certaines comme : la divulgation d'information, une information sans garantie d'origine, l'altération des données, l'usurpation de droits, les erreurs de saisie ou d'utilisation. Ces menaces ont été choisies en fonction de leur pertinence pour les systèmes d'information de santé : par exemple, on craint plus dans les systèmes d'information hospitaliers le détournement d'informations médicales à caractère personnel (donc la divulgation) ou le vol de matériels « attractifs » (micro-ordinateurs,…) que l'espionnage à distance (secteurs militaire, recherche et développement, et commercial fortement concurrentiel).

Par la suite, nous estimons le risque à partir de l'évaluation des impacts des menaces sur les biens identifiés. A nouveau, le GMSIH a construit une échelle d'évaluation (*cf.* Tableau 2.4) qui établit une classification des impacts de sinistres informatiques en distinguant les risques dont l'impact est « acceptable » (seuils d'impact 1 et 2) des risques dont l'impact est « inacceptable » (seuils d'impact 3 et 4).

Axes d'impact	Seuils d'impact			
	1 **limité**	2 **important**	3 **grave**	4 **critique**
Atteinte à la qualité des soins	Perturbation momentanée et limitée de l'organisation des soins	Perturbations limitées dans la délivrance des soins	Désorganisation des soins, ou atteinte limitée à l'état de santé d'un patient	Atteinte grave à l'état de santé d'un patient
Pertes financières	Pertes non significatives sur le plan financier	Pertes < 0,5% du budget global ou du chiffre d'affaires	Pertes de l'ordre de 0,5% du budget global ou du chiffre d'affaires	Pertes > 0,5 %du budget global ou du chiffre d'affaires
Engagement de la responsabilité	Plainte(s) de patient(s) signalant un dysfonctionnement	Plainte(s) de patient(s) signalant un dysfonctionnement grave, ou débouchant sur un recours	Plainte de patients débouchant sur une sanction disciplinaire à l'encontre d'un responsable à l'encontre d'un responsable	Plainte de patients débouchant sur la condamnation civile ou pénale d'un responsable d'établissement
Atteinte à l'image de l'établissement (auprès des patients, partenaires, tutelles, médecins de ville,…)	Divulgation limitée d'incidents	Altération significative de l'image de l'établissement	Altération très importante de l'image de l'établissement	Altération définitive de l'image de l'établissement

Tableau 2.4 - *Échelle d'évaluation des impacts sur les QFRI, Q : Qualité des soins, F : Financier, R : Responsabilité (juridique), I : Image (de l'établissement).*

Afin de pouvoir déterminer l'impact des menaces sur les biens, nous devons aussi considérer l'axe DICP. Autrement dit, la survenance d'un risque sur une fonction ou une ressource peut avoir des conséquences variable (ex : la perte d'intégrité et/ou la perte de confidentialité). Ces conséquences nous permettent de donner les limites d'impact QFRI pour les menaces étudiées. Par exemple, la perte d'intégrité sur les ressources « image » est considérée comme pouvant être la cause directe ou indirecte d'une atteinte grave à la santé du patient. Cela peut aller jusqu'à causer le décès du patient en cas d'association de l'image d'un patient avec un autre patient, donc impact est Q4. En revanche, la perte de confidentialité d'une image peut donner lieu à la divulgation d'une information sensible (cas des VIP). Dans ce cas, le problème est limité à l'état de santé du patient mais ne peut aller jusqu'à causer le décès du patient, donc Q3. Nous donnons dans le Tableau 2.5 quelques exemples de mesures QFRI.

	Divulgation d'information	Information sans garantie d'origine	Altération des données	Abus ou usurpation de droits	Erreurs de saisie ou d'utilisation
Transmission d'une image (D4 I4 C3/4 P3)	-	-	Q3 F1 R2 Im2	Q3 F1 R2 Im2	Q3 F1 R2 Im2
Consultation de l'image (D3/4 I4 C3/4 P3)	-	-	Q3 F1 R2 Im2	Q3 F1 R2 Im2	Q3 F1 R2 Im2
Image (D4 I4 C4 P4)	R3 Im3	Q4 R3 Im3	Q4 R3 Im3	Q3 R3 Im3	Q3 R3 Im3
Compte-rendu (D3 I4 C4 P4)	R3 Im3	Q3 R3 Im3	Q3 R3 Im3	Q3 R3 Im3	Q2 R3 Im3

Tableau 2.5 - *Analyse des menaces.*

Dans le même temps, l'architecture et les composants du SI comportent un certain nombre de vulnérabilités; des faiblesses qui représentent autant d'opportunité pour que la menace se réalisent. Le fait, par exemple, d'utiliser Internet est un facteur de vulnérabilité du point de vue de la confidentialité des informations transportées. Pour déterminer les vulnérabilités exploitables par la menace, la méthode EBIOS dispose d'une base de connaissance de vulnérabilités génériques classées selon le type des biens à protéger. Nous pouvons notamment citer :

La divulgation d'information :
- Absence de vérification des accès partagés accordés.
- Présence d'un réseau de communication avec l'extérieur permettant l'échange d'information.
- Absence de filtrage et de journalisation sur les relais de communication inter-réseaux.
- Possibilité d'utiliser les ressources sans garder des traces.
- Absence de contrôle (voire de traces) des échanges avec l'extérieur.
- Absence de politique de protection de l'information.

L'Information sans garantie d'origine :
- Absence de moyen sûr d'identification.
- Absence de conservation des traces des activités.
- Absence de sensibilisation aux risques d'usurpation d'identité (mauvais usage des moyens garantissant l'authentification tels que les mots de passe).
- Absence d'organisation permettant de garantir l'identification d'une personne au sein de l'organisme ou d'un projet.

L'altération des données :
- Absence de moyens de protection et de contrôle de l'intégrité des données.
- Absence de procédure et de dispositif d'habilitation à la modification des données.
- Absence de mise en œuvre de règles de sécurité de base applicables au système d'exploitation et aux logiciels.

L'abus ou usurpation de droits :
- Absence de dispositif de contrôle d'accès robuste.
- Absence de politique d'audit.

- Le principe du moindre privilège n'est pas appliqué.
- Absence de procédures de contrôles des habilitations du personnel accédant au site ou aux locaux.
- Absence de contrôle sur les attributions des droits des utilisateurs.

Les erreurs de saisie ou d'utilisation :

- Absence de responsabilité.
- Absence de formation à l'utilisation et la maintenance des nouveaux logiciels.

2.7. Identification des objectifs et des exigences de sécurité

A. Les objectifs de sécurité

Après avoir analysé les menaces et les vulnérabilités, l'étape qui suit est d'identifier les objectifs de sécurité. Ceux-ci doivent couvrir la totalité des risques identifiés précédemment, tout en tenant compte des hypothèses, des règles de sécurité et des différents éléments du contexte (les contraintes et enjeux notamment). Un état du système qui satisfait l'ensemble des objectifs de sécurité est évidemment un état sûr, autrement dit, considéré comme acceptable du point de vue de la sécurité.

Voici quelques objectifs de sécurité que nous avons identifiés dans le contexte de notre cas d'étude (notons qu'un objectif peut couvrir plusieurs menaces à la fois) :

- Pour contrer les menaces de type **Divulgation d'information**, nous avons identifié :

  Obj.Accès : tout accès aux systèmes doit être protégé par un dispositif d'identification et d'authentification. Cet objectif couvre entre autre la vulnérabilité « absence de vérification des accès partagés accordés ».

  Obj.Licen : il doit exister une gestion des licences, de leur enregistrement et de leur conservation.

  Obj.Trace : les traces des opérations doivent être exploitables y compris lorsqu'elles sont générées par des systèmes différents (possibilité de reconstruire l'enchaînement des événements). Cet objectif répond, par exemple, à la vulnérabilité « absence de contrôle (voire de traces) des échanges avec l'extérieur ».

  Obj.Habil : il doit exister une gestion active des habilitations au sein des systèmes pour le traitement des informations en fonction des besoins d'en connaître et d'en modifier.

  Obj.Trans : les interfaces de communication doivent protéger les transmissions en confidentialité, intégrité et disponibilité.

  Obj.Auth.Com : l'authentification et la non-répudiation des communications doivent pouvoir en cas de besoin être établies.

- Pour contrer les menaces de type **Information sans garantie d'origine**, nous avons identifié :

  Obj.Auth.Doc : les documents réalisés par le service doivent pouvoir être authentifiés, supportés par les objectifs : Obj.Trace et Obj.Auth.Com.

- Pour contrer les menaces de type **Altération des données**, nous avons identifié :

  Obj.Integ : l'intégrité des logiciels et des données doit être garantie, supportés par les objectifs : Obj.Habil.

- Pour contrer les menaces de type **Abus ou usurpation de droits**, nous avons identifié :

Obj.Accès, Obj.Trace, Obj.Habil et Obj.Auth.Com.

- Pour contrer les menaces de type **Erreurs de saisie ou d'utilisation**, nous avons identifié : Obj.Trace.

B. Les exigences de sécurité

La détermination des exigences de sécurité s'effectue en fonction des objectifs de sécurité identifiés. Ces exigences peuvent être issues de l'ISO/IEC 15408 (critères communs) [CC 99] ou créées. Elles résultent du raffinement des objectifs de sécurité en un ensemble d'exigences de sécurité pour la cible de l'étude de sécurité et d'exigences de sécurité pour l'environnement qui, si elles sont satisfaites, garantiront que la cible de l'étude de sécurité peut satisfaire à ses objectifs de sécurité [EBI 04].

Voici quelques exigences nécessaires à la réalisation des objectifs de sécurité relatifs au SED :

Exi.Accès : des mécanismes de contrôle d'accès aux ressources du service sont à mettre en œuvre via les fonctionnalités du système d'exploitation.

Exi.Poli.Acces : pour un contrôle d'accès complet, toutes les opérations entre tout sujet et tout objet de la cible sont couverts par la politique de sécurité pour les contrôles d'accès.

Exi.Auth : le service doit mettre en œuvre un mécanisme d'authentification forte pour l'accès au système sur la base de l'utilisation de certificats.

Exi.Integ : les utilisateurs autorisés doivent avoir la capacité de contrôler l'intégrité des données de sécurité.

Exi.Trace.Géné : des enregistrements d'audit doivent pouvoir être générés pour des événements spécifiés.

Exi.Traces.Detec : des règles doivent permettre d'analyser les événements audités pour détecter des violations potentielles de la sécurité.

Dans le cadre nos travaux, nous proposons d'utiliser nos solutions de tatouage réversible et de contrôle d'accès et d'usage qui se fonde sur le modèle OrBAC pour assurer ces exigences de sécurité (*i.e.* Exi.Accès, Exi.Auth, Exi.Integ, Exi.Trace.Géné et Exi.Traces.Detec). Avant de les mettre en œuvre, nous devons établir la politique de sécurité (Exi.Poli.Acces) qui permet de formaliser ces besoins de sécurité afin de garantir un haut niveau de protection du système.

3. Définition de la politique de sécurité

Les résultats de notre analyse de risques permettent de définir la politique de sécurité correspondante, notamment celle relative au contrôle d'accès et d'usage. Deux approches sont possibles pour exprimer les règles de sécurité qui répondent aux problèmes soulevés par l'analyse de risque :

- Une génération automatique (dans la mesure du possible) à partir d'une modélisation orientée objectifs, au moyen de Kaos [Dar 93] [KAO 00], des aspects fonctionnels de l'étude de cas et des scénarios décrits précédemment. Ensuite, cette spécification Kaos est enrichie par l'analyse de risque EBIOS que nous avons menée. Enfin un module dérive de cette spécification Kaos enrichie, la politique OrBAC correspondante [Gra 11].
- Une identification et hiérarchisation des différentes entités du modèle OrBAC (organisations, activités, rôles, vues et contextes) relatives à notre étude de cas. Puis,

l'utilisation des objectifs et exigences de sécurité résultante de l'analyse de risques pour spécifier les permissions, interdictions et obligations conformément au modèle.

Nous avons adopté la seconde approche. Ainsi, nous associons aux objectifs de sécurité et aux exigences de sécurité définis dans la section précédente (tels que Obj.Auth.Com et Obj.Trace) des règles d'accès et d'usage relatives à la manipulation et la distribution des images médicales et qui sont des instanciations de règles génériques. Il s'agit de la **Règle d'accès,** la **Règle de distribution**, la **Règle d'affichage**, la **Règle d'intégrité,** la **Règle d'authenticité,** la **Règle de création**, la **Règle de traçabilité** et la **Règle d'administration**.

3.1. La modélisation OrBAC

3.1.1. Les organisations

L'entité centrale dans le cas de la modélisation de la politique de sécurité liée au service d'écho-doppler du CHRU de Brest est l'organisation *CHRU_Brest*. Sa structure est donnée par le schéma de hiérarchie d'organisations suivant :

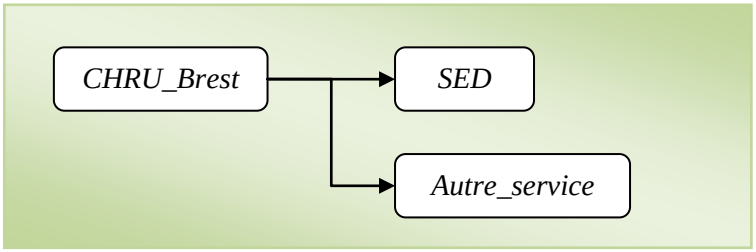

Figure 2.4 - *Hiérarchie d'organisations.*

En particulier, l'organisation *SED* représente l'entité du service d'écho-doppler et est constituée de l'ensemble du personnel de santé et administratif, ainsi que du réseau local correspondant. Les différentes règles de sécurité vont être identifiées relativement à cette organisation.

Pour simplifier, nous utilisons l'organisation *Autre_service* pour présenter un autre service de l'hôpital. Ce service peut avoir sa propre politique de sécurité mais toujours sous le contrôle de la politique de sécurité globale de CHRU de Brest.

3.1.2. Les rôles

Rappelons que les sujets dans le cadre du système correspondent à des professionnels de santé, des personnes administratives, des patients et des machines hôtes. L'attribution des droits d'accès à ces sujets se fait par le biais de la structuration en rôles. Les sujets se voient attribuer des droits d'accès au système. Un rôle correspond à un profil de règles de contrôle d'accès et n'a de sens que dans l'organisation où il a été défini.

Une hiérarchie des rôles permet l'héritage des différentes permissions, interdictions et obligations. Nous présentons la hiérarchie de rôles que nous avons identifiée en Figure 2.5. Nous supposons que l'API OrBAC et le module de tatouage sont intégrés dans *modalitéAcquis, SA, SR, Viewer*.

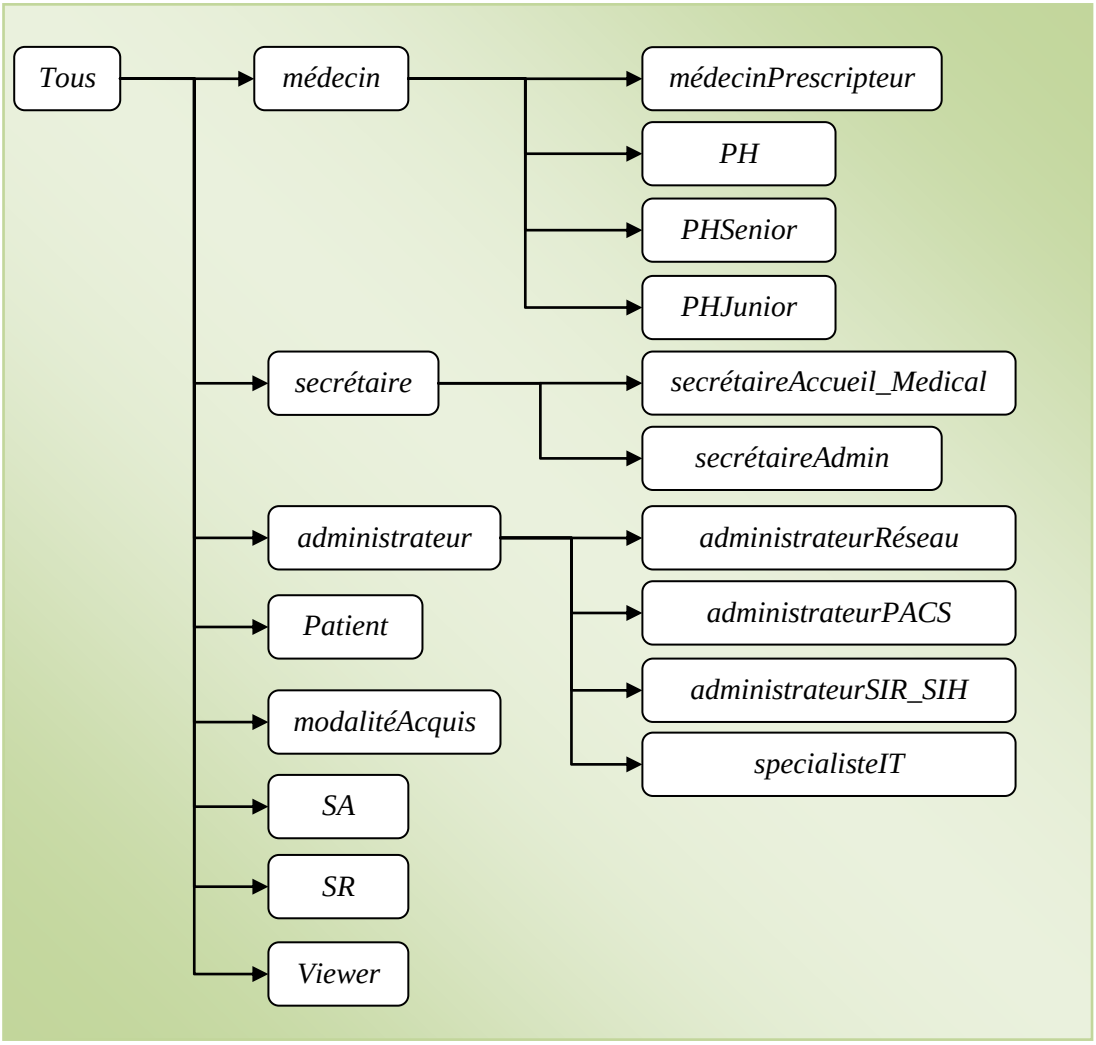

Figure 2.5 - *Hiérarchie de rôles.*

3.1.3. Les vues

Dans le cadre du SED, le dossier médical comporte les images et les comptes-rendus. Il y a aussi la vue Marque qui doit être considérée pour spécifier les règles de tatouage. Nous modélisons les vues comme décrit dans la Figure 2.6.

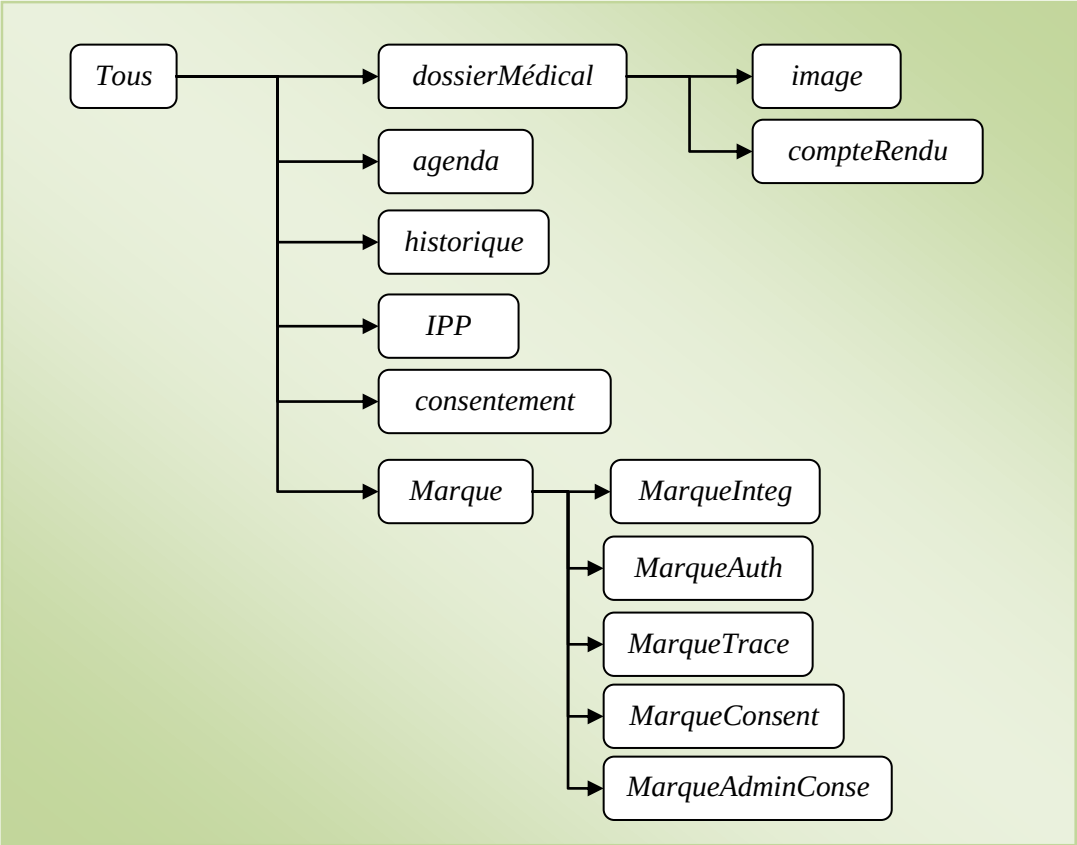

Figure 2.6 - *Hiérarchie de vues.*

3.1.4. Les activités

Les activités correspondent aux divers services offerts par le système à ses utilisateurs. Il doit leur permettre, entre autres, de : rechercher des documents liés à un patient (recherche) … Ces différentes activités sont implantées par des actions concrètes. Ainsi, l'activité gestion est une abstraction des actions : créer, supprimer, mise à jour (*MAJ*) et consulter. La structuration des actions est donnée par le schéma de hiérarchie d'activités dans la Figure 2.7.

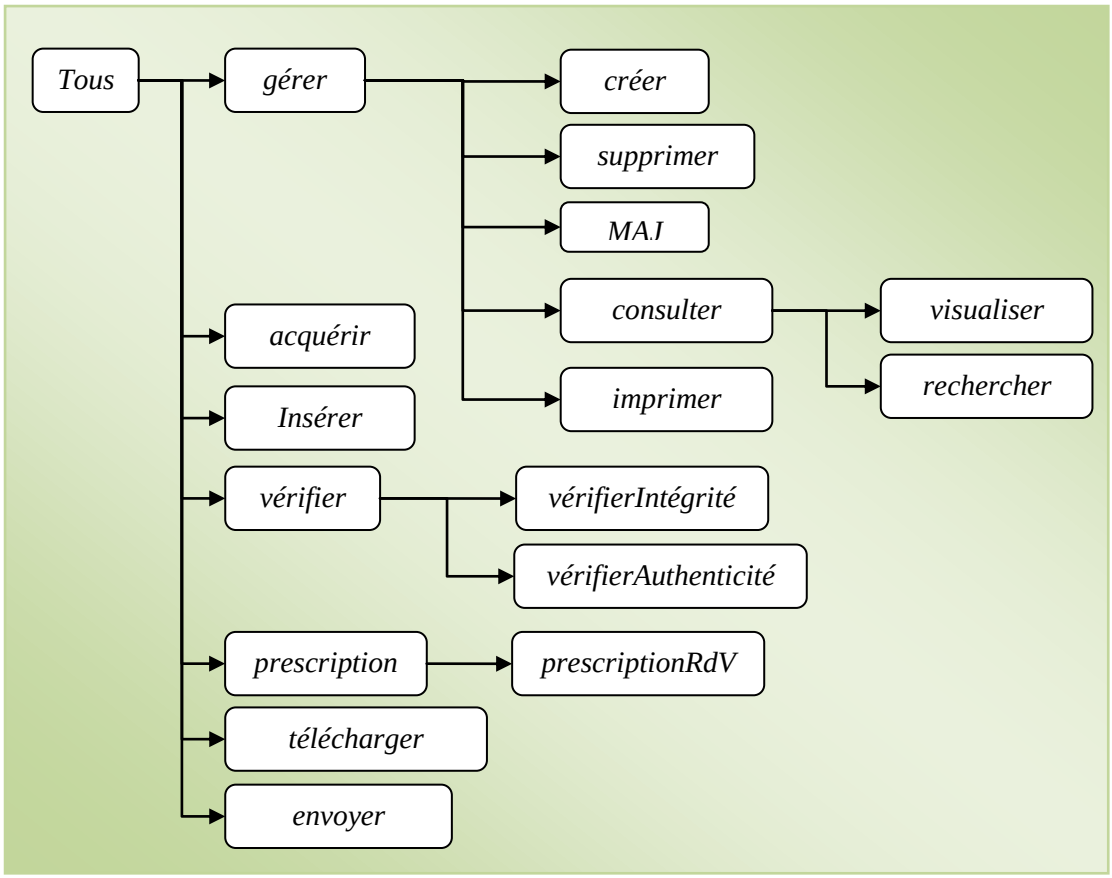

Figure 2.7 - *Hiérarchie d'activités.*

3.1.5. Les règles au sens OrBAC

Avec l'ensemble des concepts du modèle OrBAC, nous pouvons spécifier les règles de tatouage d'une manière formelle.

Règle d'accès :
obligation (SED, SR, MAJ, MarqueTrace, DemandeTéléchagerImage, delay(1s))
permission (SED, médecinPrescripteur, télécharger, Image, MarqueTraceMAJ)

Règle de distribution :
permission (Autre_service, médecinPrescripteur, envoyer, Image,
MarqueTraceMAJ & MarqueConsentValide)

Règle d'affichage :
prohibition(CHRU_Brest, médecin, consulter, Image, Lecture_Application &
MarqueConsentNonValide)

Règle d'intégrité :
obligation(CHRU_Brest, viewer, VériferIntegrité, image,
Lecture_Application, delay(1s))

Règle d'authenticité :
obligation(CHRU_Brest, viewer, vériferAuthenticité, Image,

Lecture_Application, delay(1s))

Règle de création :
obligation(SED, modalitéAcquis, insérer, Mark, ImageCrée, delay(1s))

Règle de traçabilité :
obligation(CHRU_Brest, viewer, MAJ, MarqueTrace, ImageConsulté, delay(1s))

Règle d'administration :
permission (CHRU_Brest, S, MAJ, MarqueConsent, DateLimitExpired) :-
use (CHRU_Brest, watermarkAdminConsent_xx.dcm, MarqueConsent),
authority (MarqueAdminConsent _xx.dcm, Autre_service),
grantee (MarqueAdminConsent _xx.dcm, S),
privilege (MarqueAdminConsent _xx.dcm, MAJ),
target (MarqueAdminConsent, MarqueConsent),
context (MarqueAdminConsent, DateLimitExpired).

Une fois que les règles ont été définies, nous les utilisons pour guider l'implémentation de tatouage. Autrement dit, les fonctions de tatouage doivent assurer la réalisation des règles de la politique de sécurité pouvant être assurées par du tatouage.

4. Implémentation

Afin de pouvoir appliquer l'ensemble de règles de tatouage, nous proposons d'intégrer le couple (API OrBAC - module de tatouage) dans les modalités d'acquisition (*modalitéAcquis*) et le logiciel Medview (*viewer*). Medview est une solution logicielle développée par la société Medecom qui permet de fournir les services aux utilisateurs tels que rechercher un examen, afficher/traiter/imprimer les images médicales, etc. Ce logiciel se trouve dans le serveur d'archivage (*SA*) et le serveur de résultat (*SR*). De cette manière, nous considérons les modalités d'acquisition et le logiciel Medview comme les moniteurs de référence qui permettent d'interroger à la fois l'API OrBAC et le module de tatouage pour appliquer les règles de sécurité dans le système.

4.1. L'API OrBAC

Rappelons que l'API OrBAC a été créée pour aider les développeurs de logiciel à inclure des mécanismes de contrôle d'accès et d'usage dans leurs applications. Elle implémente le modèle OrBAC pour spécifier des politiques de sécurité et le modèle AdOrBAC pour les administrer. Notons que l'outil MotOrBAC est interrogé au moyen de cette API et permet d'éditer et d'administrer des politiques OrBAC. Ainsi, nous utilisons cet outil pour éditer nos règles de tatouage puis les enregistrer sous la forme de fichiers XML RDF. L'API OrBAC peut ensuite lire ces fichiers et les interpréter afin d'appliquer les règles souhaitées.

Nous distinguons ici deux services qui sont fournis par l'API OrBAC : le contrôle d'accès et le contrôle d'usage.

Dans le cas du contrôle d'accès, si un sujet veut effectuer une action sur un objet, ce sujet envoie une demande au moniteur de référence (*e.g.* un *viewer*). Le monteur fait appel à l'API OrBAC qui, après avoir chargé la politique abstraite et évaluer les conditions contextuelles (i.e. les contextes), fournit la décision d'accès : acceptation, refus ou violation.

Dans le cas du contrôle d'usage, la tâche de l'API OrBAC est de superviser la réalisation de certaines actions obligatoires (*e.g.* mise à jour de la marque dans l'image). Ces actions sont

effectuées dans le système d'information et certaines d'entre-elles génèrent des évènements qui peuvent constituer des entrées pour l'API. Celui-ci, conformément à la politique d'usage et les différents contextes spécifiés dans les règles de sécurité, détecte les violations et les réalisations des obligations et génèrent de nouveaux évènements en sortie.

Dans les deux cas, l'API fait appel au module de tatouage pour évaluer les règles de tatouage. Nous verrons dans la section suivante comment les transitions se font entre les deux modules.

4.2. Intégration du tatouage

Le module de tatouage peut amener de nouvelles fonctionnalités, néanmoins nous distinguerons la fonction de tatouage indépendamment du service de sécurité auquel cette fonction contribue.

A. Fonction de tatouage

Fonctions élémentaires dans le module

Ce sont des fonctions qui peuvent être appelées par les fonctions principales « *Insere* » et « *Extraire* » (décrites dans la suite). Quatre fonctions à considérer dans ce cas : «*MsgToMarque* », « *Hash256* », « *Chiffre* » et « *Dechiffre* ».

La fonction «*MsgToMarque* » permet la conversion de l'information à insérer en une séquence de bits correspondant à une marque. Les entrées de cette fonction sont des informations liées au contrôle d'intégrité, d'authenticité et d'usage (l'information de trace et du consentement du patient) de l'image. La sortie de cette fonction est une marque binaire comportant 5 champs : champ empreinte de l'image, champ identifiant du patient, champ consentement, champ informations de trace et champ informations administratives. Elle est ensuite insérée dans l'image par notre algorithme de tatouage réversible.

Plus précisément, pour obtenir les bits dans le champ « empreinte de l'image », la fonction « *Hash256* » est appliquée à l'image pour obtenir les 256 bits correspondant. Pour le champ « identifiant du patient », il correspond aux 256 bits en sortie de la fonction « *Hash256* » qui prend en entrée le NIR, le prénom et la date de naissance du patient.

Le champ consentement, contient les attributs d'un « e-consent » du patient. En utilisant l'outil MotOrBAC, l'administrateur de sécurité formalise ce « e-consent » à partir du consentement du patient formulé oralement ou par écrit. Ce e-consent comporte les attributs **authority**, **grantee**, **privilege** et **dateLimit**. L'attribut **authority** identifie les organisations dans lesquelles le consentement est valide. Par exemple; si nb_o est le nombre total d'organisations connues de la politique, l'une d'elle peut être identifiée par un numéro num_o dans l'intervalle [*0, nb_o*-1]. La représentation de l'attribut **authority** dans le champ e-consent commence donc par un code C_o codé sur $\log_2(nb_o)$ bits qui précise combien d'organisations sont concernées par ce e-consent suivit de leurs identifiants num_o chacun codé chacun sur $\log_2(nb_o)$ bits. Il est important de signaler que ce codage est propre à une politique de sécurité. Pour généraliser, il conviendra de pouvoir identifier n'importe quelle organisation de manière unique à travers le monde. Le codage deviendra alors dynamique. De la même manière, nous pouvons associer le couple (C_a, num_a) à l'attribut **privilege** considérant nb_a activités possibles. Egalement, il est possible d'associer le couple (C_r, num_r) à l'attribut **grantee**. L'attribut **dateLimit**, indiquant la durée du consentement, est présenté par le jour, le mois et l'année. Dans le champ e-consent, nous utilisons 5 bits pour coder le jour, 4 bits pour

coder le mois, 12 bits pour coder l'année (jusqu'au 2050). Ainsi, 21 bits sont nécessaires pour coder cet attribut.

Pour le champ informations de trace, les entrées de «*MsgToMarque* » sont les identifiants des sujets et les dates d'accès correspondants. Un code C_t est utilisé pour préciser combien de sujets sont tracés dans ce champ. Lorsque le sujet est un médecin, l'identifiant correspond aux 256 bits la fonction « *Hash256* » appliquée aux entrées de la fonction «*MsgToMarque* » suivantes : le n°RPPS, le prénom et la date de naissance du médecin. Notons que le nombre maximal de médecins qui peut être tracé dans ce champ est de 2^k où k est le nombre de bits utilisé pour coder C_t. La date d'accès à l'image de chaque médecin est également codée en utilisant 21 bits.

Enfin, pour le champ informations administratives, c'est-à-dire qui peut modifier le consentement, il s'agit d'une liste de sujets codés chacun sur 256 bits car ils correspondent à des identifiants des sujets. Le codage est le même que précédemment : un code C_{adm} qui précise combien de sujets font parti de la liste, suivi des identifiants des sujets autorisés chacun associé à une date limite codée par 21 bits.

Pour avoir une idée sur la longueur finale (en bits) de la marque, faisons quelques hypothèses. Il est évident que les deux premiers champs de la marque (« empreinte de l'image » et « identifiant du patient ») représentent dont 512 bits.

Pour le champ consentement, dans notre cas d'étude, nous pouvons considérer dans la politique globale qu'il y a 16 organisations (nb_o = 16) qui comportent le CHRU de Brest, le service SED, les autres services du CHRU de Brest et certains cabinets de ville (ou cliniques privées), 1024 rôles (nb_r = 1024) et 1024 activités (nb_a = 1024) qui sont répartis dans les 16 organisations. Supposons maintenant toutes les 16 organisations (C_o = 16) sont concernées par un consentement du patient et dans chaque organisation, qu'il y a 4 rôles (C_r = 4×16 = 64) et 4 activité (C_a = 64) sont attribués par ce consentement. Par exemple, dans le service SED, les rôles *médecin*, *administrateur*, *modalitéAcquis* et *viewer* sont attribués dans le consentement et les activités *gérer*, *acquérir*, *télécharger* et *envoyer* sont permises par ce consentement. Ainsi, chaque organisation, rôle et activité est codée par 4 bits, 10 bits et 10 bits respectivement dans la marque de consentement. Avec 21 bits pour coder la date limite, nous obtenons 1389 bits au total pour présenter le champ consentement.

Dans le champ informations de trace, nous décidons de tracer les quatre derniers médecins qui ont accédé à l'image. Avec C_t = 4, nous pouvons insérer les 4 identifiants des médecins et les 4 dates d'accès correspondantes, soit 1110 bits au total.

Si maintenant nous considérons que le médecin prescripteur du patient et le patient lui-même ont le droit de modifier la marque de consentement de l'image, la marque d'administration du consentement est à 554 bits.

Enfin, la longueur de la marque est de 3565 bits. Pour les images échographies, notre méthode de marquage couvre facilement cette capacité d'insertion avec une valeur PSNR de 62.55 *dB* par image. Cette grande valeur de PSNR montre que la qualité de l'image a été parfaitement conservée.

Nous pouvons également faire varier les paramètres d'entrées pour augmenter la longueur de la marque à insérer. Par exemple, nous pouvons tracer à la fois 128 médecins dans la marque de trace. Si toutes les autres entrées restent inchangées, nous obtenons une marque de 37918 bits. Dans ce cas, notre méthode est toujours applicable avec une PSNR de 57.067 *dB* par image. Le résultat est acceptable en termes de distorsion. Il faut souligner que la capacité d'insertion dépend de la nature de l'image mais aussi de la taille de l'image. Par exemple, pour une image PET de la taille 144x144 pixels, au-delà de 6600 bits en capacité, la qualité de l'image devient difficile à conserver. Dans ce cas, au lieu de travailler sur une seule image, nous pouvons insérer les informations dans un volume de l'image.

Pour assurer la sécurité de la marque, nous utilisons les fonctions de chiffrement asymétrique (ou à clé publique) « *Chiffre* » et « *Dechiffre* ». Les entrées de ces fonctions sont les clés de chiffrement (privé ou publique) et la marque. La sortie est la marque chiffrée.

Fonction principale : Insertion

Cette fonction permet d'insérer un message (*Msg*) dans l'image. En pratique, le message tatoué comporte le message utile et d'autres données nécessaires à la réversibilité comme les seuils utilisés dans le processus de classification.

Nous pouvons distinguer deux cas pour l'insertion :

- La protection de l'image pour la 1ère fois,
- La mise à jour du contenu de la marque.

La mise à jour est une combinaison d'une opération d'extraction et de marquage avec, entre ces deux étapes, la mise à jour du contenu de la marque, i.e. le message.

L'écriture peut se faire en tâche de fond. Par ailleurs elle intervient le plus souvent lorsque l'utilisateur a fini d'exploiter l'image (e.g. la **règle de traçabilité**).

L'insertion peut être sécurisée suivant une clé de tatouage (*Cle*) qui, dans le cadre de cette application, paramètre une fonction de permutation des bits du message à tatouer.

La fonction « *Insere* » permet de satisfaire une partie des services que nous parcourons un à un dans la suite. Un service est référencé par son code <\service>, code à spécifier par l'utilisateur lors de l'appel de la fonction « *Insere* ». Les entrées et sorties de cette fonction dépendent donc du service appelé.

Pour résumer, l'appel à la fonction d'écriture ou de tatouage est de la forme :

<\flag><\Image_tatouee>=*Insere*(<\Image><\Cles><\Msg><\service>)

Où :

<\flag> <indique le succès ou l'échec du processus de tatouage>

<\Image_tatouee> <chemin de l'image tatouée ou les pixels de l'image tatouée>

<\Image> <chemin de l'image considérée par le tatouage >

<\Cles> <les clés de chiffrement et de tatouage>

<\Msg><contenu du message à insérer dans l'image>

<\service><n° du service spécifique à accomplir sur la base de la fonction de tatouage ex : insertion simple d'un message, mise à jour de contenu de la marque, demande de contrôle d'intégrité, etc.

Fonction principale : Extraction

L'extraction peut donner accès :

- Simplement au message,
- Au message et à l'image restaurée.

La lecture de la marque peut être assujettie à la connaissance d'une clé de tatouage (*Cle*).

La fonction « *Extraire* » permet de restaurer l'image initiale, de vérifier l'intégrité des données, de contrôler l'authentification des images …

Pour résumer, la fonction d'extraction est de la forme :

<\flags><\Msg_lu><\image_restauree>=*Extraire*(<\Image><\Cles><\Msg><\service>)

Où :

<\service><n° du service spécifique à accomplir ex : lecture simple du message, extraction de la marque, restauration de l'image initiale, contrôle d'intégrité,… >

<\Msg><Message fourni dans le cadre d'un service. Il s'agit d'une structure de données qui contient une signature ou un identifiant dont on veut vérifier la présence dans l'image ou la validité.>

<\Image><chemin de l'image tatouée>

<\image_restauree> < chemin vers ou pixels en flux de l'image restaurée >

<\flags>< un ensemble de flags liés à la bonne réalisation de l'appel et/ou de la réponse du service appelé>

<\Msg_lu> <message extrait>

B. Services de Sécurité

Service insertion d'une donnée (<\service=1>)

La mise en œuvre de ce service s'appuie directement sur la fonction d'insertion. Son objectif est de protéger l'image dès qu'elle a été crée par la modalité d'acquisition.

Dans le scénario du service SED, nous appliquons la **règle de création** au moment de l'acquisition de l'image sur l'échographe. Une fois une image créée, elle est prise en charge par le module de tatouage. Le service d'insertion doit être déclenché en insérant l'information de protection (*Msg*) dans l'image, afin de fournir en sortie une image tatouée et protégée par la marque. Ce *Msg* doit être préparé par l'administrateur du service et transmis au module de tatouage. Une solution possible est de fournir cette information par la "*worklist*". Le module de tatouage de la modalité interroge cette liste pour obtenir le *Msg*. Ensuite, l'image tatouée va être poussée dans le serveur d'archivage. Le rôle de l'API OrBAC est de prendre en charge la **règle de création** afin de superviser la réalisation des actions de création de l'image et d'insertion du message dans la modalité d'acquisition. Si l'insertion du message n'a pas été faite ou la sortie du module de tatouage flag indique l'échec du processus de tatouage, l'API OrBAC va considérer comme cet état de fait comme une violation du système et déclenche une alerte.

Service lecture de la marque tatouée

Ce service s'appuie directement sur la fonction d'extraction.

- *Vérification de l'intégrité et l'authenticité d'une image* (<\service=2>)

Ce service permet de réaliser les **règles d'intégrité** et **d'authenticité** au moment où le praticien accède et interprète les images sur la station de diagnostic. L'API OrBAC prend en charge ces deux règles et surveille l'action d'affichage de l'image dans le viewer. Si une image est en train d'être consulté par un praticien, L'API OrBAC va déclencher le contexte « *Lecture_Application* » dans les deux règles d'obligation. Ensuite, ce moniteur au travers l'API surveille si les vérifications d'intégrité et d'authenticité de l'image affichée ont été effectuées. Concernant les actions de tatouage, le module déclenche automatiquement la vérification lorsque l'image est affichée sur l'écran de diagnostic. Les entrées de la fonction « *Extraire* » sont l'image affichée, les clés et l'identifiant du patient à vérifier. En sortie, deux flags seront fournis, un flag indiquant la validité de l'intégrité et l'autre indiquant la validité de l'authenticité. Si la vérification n'a pas été faite ou si les flags indiquent la non validité de l'intégrité ou l'authenticité, l'API OrBAC va déclencher une alerte pour avertir le praticien que l'image est probablement corrompue.

Notons qu'en cas de besoin, le praticien a le droit de consulter l'image originale sur le viewer. Cette option est d'assurer par la sortie *image_restauree* de la fonction « *Extraire* ».

Dans ce cas la marque de protection est enlevée de l'image. Par mesure de sécurité, l'API OrBAC va surveiller cette opération dans le viewer. Si c'est le cas, il imposera au système de tracer cette opération dans le journal du service SED.

- *Vérification des droits d'accès d'une image* (<\service=3>)

Ce service permet de réaliser les **règles d'affichage et de distribution**. La **règle d'affichage** s'applique au moment où l'utilisateur accède aux images sur le viewer. La **règle de distribution** s'applique au moment où l'image arrive au poste de travail du médecin prescripteur. Nous supposons que du côté du médecin prescripteur, le système est compatible avec l'API OrBAC et le module de tatouage. Dans ces cas, c'est l'API OrBAC qui interroge le module de tatouage pour avoir les informations contenues dans la marque de consentement, afin de vérifier le profil d'utilisateur. Ces informations sont fournies par la sortie *Msg_lu* de la fonction « *Extraire* ».

Service de mise à jour de la marque tatouée

Ce service s'appuie sur la combinaison d'une opération d'extraction et d'insertion avec, entre ces deux étapes, la mise à jour du contenu de la marque.

- *Contrôle d'accès d'une image* (<\service=4>)

Ce service permet de réaliser la **règle d'accès**. Cette règle s'applique en cas de partage d'une image. Ainsi, le médecin prescripteur est autorisé à télécharger l'image sur le serveur résultat du SED dans le contexte où la marque de trace de cette image a été mise à jour par le module tatouage. Plus clairement, l'envoi est conditionné par l'ajout de l'identifiant du médecin dans la marque.

Concrètement, l'API OrBAC surveille les requêtes des médecins sur le serveur. Dès qu'une requête est émise (*i.e.* une demande de téléchargement de l'image sur le serveur), le contrôle d'accès et d'usage au travers l'API impose au module de tatouage d'insérer l'identifiant du requêteur et la date de demande dans l'image (dans la marque de trace). Si une telle opération n'a pas été effectuée, une violation de la politique est signalée. Dans le cas contraire, le contexte « *MarqueTraceMAJ* » est activé, le médecin prescripteur obtient la permission de télécharger l'image sur le serveur.

- *Contrôle de trace d'une image* (<\service=5>)

Ce service permet d'assurer la **règle de traçabilité**. Cette règle est activée lorsque l'utilisateur a terminé la consultation de l'image avec le viewer. L'API OrBAC surveille la fin de cette application dans le viewer. Si c'est le cas, le contexte « *ImageConsulte* » est activé. L'identifiant de l'utilisateur (i.e. le médecin) est automatiquement inséré dans l'image par la fonction « *Insere* ». Nous gardons ainsi la trace du médecin dans les images qu'il a consultées. Si ce n'est pas le cas, l'API OrBAC déclenchera une alerte dans le système.

- *Contrôle d'administration de la marque d'une image* (<\service=6>)

Ce service permet d'assurer la mise à jour de la marque de consentement dans l'image. Une fois la date limite du consentement expirée, l'API OrBAC va examiner le profil du médecin en appelant le module de tatouage pour extraire la marque d'administration du

consentement. Si le profil du médecin est compatible avec les attributs de la marque d'administration, ce médecin est alors autorisé à mettre à jour la marque de consentement.

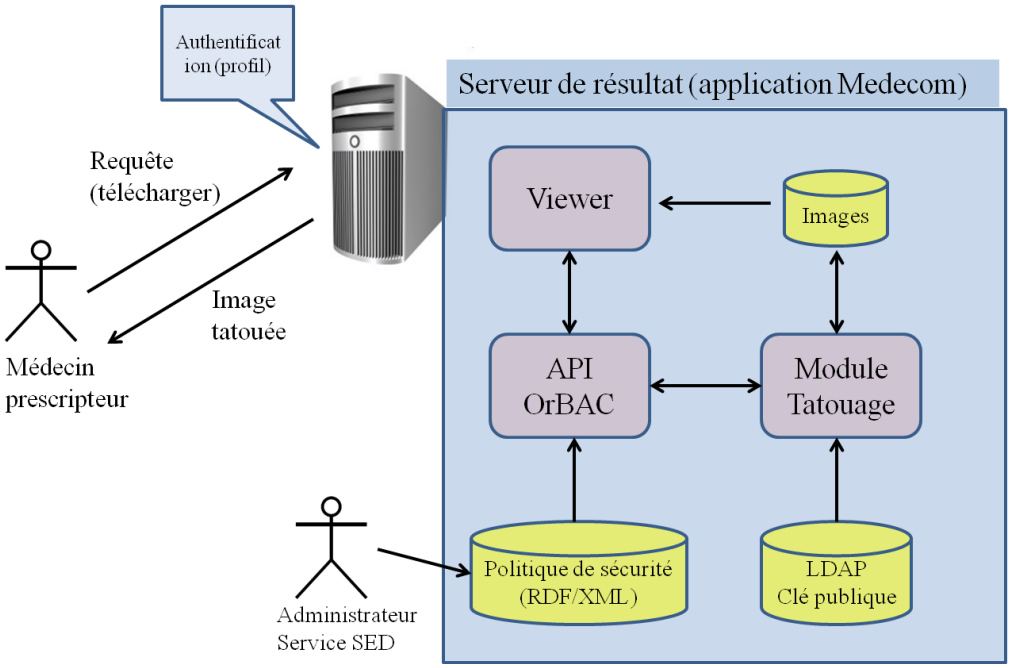

Figure 2.8 - *Architecture du serveur de résultat.*

5. Gestion des clés du module de tatouage

Rappelons que dans le module de tatouage, nous combinons des techniques cryptographiques à notre méthode de tatouage réversible. L'objectif étant de rendre difficile à une personne non autorisée de pouvoir : accéder au message tatoué, de l'interpréter le cas échéant ou, encore, de le modifier. Concrètement, nous utilisons une clé de tatouage et deux paires de clés (de l'émetteur et du destinataire respectivement) de chiffrement asymétrique pour assurer la sécurité de la marque.

Notons que notre méthode de tatouage est de type privé, c'est-à-dire que la clé de tatouage symétrique. Cette clé doit rester totalement confidentielle et transmise au correspondant de manière sûre. Pour assurer la confidentialité de la clé de tatouage, nous proposons de la chiffrer asymétriquement avant la transmission. En utilisant la clé publique du destinataire pour chiffrer la clé de tatouage, la confidentialité est assurée.

Au niveau du SED, l'authenticité des clés publiques est disponible sur l'annuaire LDAP ("*Light Directory Access Protocol*") ou le serveur de résultat (*cf.* Figure 2.8). Néanmoins, cette authenticité n'est pas garantie dans un environnement ouvert tel qu'Internet et il est possible qu'un pirate modifie l'annuaire ou le serveur web qui héberge les clés publiques et remplace ainsi la clé publique d'un médecin par la sienne. Un certificat électronique permet de résoudre ce type de problème. Rappelons qu'un certificat permet d'établir un environnement de confiance entre deux entités distantes ayant besoin de communiquer entre elles et de s'échanger des informations non-répudiables (nécessité de signature) ou confidentielles (application de chiffrement). En effet, un certificat est souvent destiné à remplir trois rôles : l'authentification de l'émetteur, garantir l'intégrité des documents, et

éventuellement un horodatage. Selon la norme X509, un certificat électronique doit contenir notamment : le nom de l'autorité de certification, le nom et le prénom de la personne, son entreprise, son adresse électronique, sa clé publique, les dates de validité du certificat ainsi qu'une signature électronique. Cette signature, calculée sur les informations contenues dans le certificat, est l'empreinte de ces informations chiffrées avec la clé privée de l'autorité de certification qui a délivré ce certificat. En cas de partage d'image au niveau du SED, le médecin prescripteur doit donc fournir son certificat. Ce certificat est signé avec une autorité AC. Le serveur de résultat peut vérifier la signature de ce certificat pour s'assurer que ce document a bien été crée par l'autorité AC et qu'il n'a pas été modifié. Avec cette assurance, le module de tatouage peut récupérer la clé publique de médecin contenue dans ce certificat.

Une Infrastructure de Gestion de Clés (IGC ou PKI pour Public Key Infrastructure) recouvre l'ensemble des services mis en œuvre pour assurer la gestion complète des clés publiques, c'est-à-dire l'enregistrement des utilisateurs et la vérification des attributs, la génération de certificats, la publication des certificats valides et révoqués, l'identification et l'authentification des utilisateurs, l'archivage des certificats, etc.

Afin de mettre en œuvre une IGC dans le SED, nous proposons deux solutions.

La première solution consiste à considérer la un tiers de confiance comme autorité de certification qui

- génère un couple de clés publique-privée pour elle-même,
- diffuse la valeur de sa clé publique auprès des structures qu'elle connaît et des annuaires (LDAP),
- crée, délivre et révoque les certificats des utilisateurs qu'elle gère.

Le mode de distribution des clés secrètes peut être de type :

- Exportation dans un fichier ;
- Envoi vers des serveurs de clés ;
- Envoi par messagerie ;
- Envoi par courrier (clé secrète incorporée à un support physique de type carte à puce, token, disquette…).

Notons que les clés (publiques et privées) sont périodiquement renouvelées afin de minimiser les risques d'attaque cryptographie.

L'inconvénient majeur de cette solution est le problème d'interopérabilité entre les différents établissements de santé.

La deuxième solution s'appuie sur la Carte Professionnel de Santé (CPS). Rappelons que le décret Confidentialité (2007) impose l'usage de la CPS qui permet l'authentification forte de son détenteur et à ce dernier de pouvoir signer électroniquement un document (Art. R.1110-3).

En fait, cette carte contient :

- Les informations concernant le titulaire de la carte.
- Les clés privées de signature et d'authentification du titulaire.
- Les certificats des clés publiques de signature et d'authentification du titulaire.

Dans l'IGC du « CPS », le GIP « CPS » se charge de la gestion de la vie des certificats qui est associée à celle de la carte CPS. Nous pouvons tous simplement utiliser les clés publique et privée de la carte CPS du médecin pour chiffrer et déchiffre la marque, afin d'assurer sa sécurité. Actuellement, les médecins dans le SED n'ont pas de CPS. Nous espérons que le système CPS arrivera bientôt dans ce service.

6. Conclusion

Nous avons présenté les travaux relatifs à la conception d'un modèle, une méthodologie et un mécanisme qui répondent au mieux à des exigences de traçabilité, d'intégrité des informations dans le cadre du projet. Nous avons montré que la technique de tatouage, permet d'insérer des informations utiles dans l'image médicale pour contrôler l'intégrité, l'authenticité et les droits d'accès et d'usage. Le modèle OrBAC a été utilisé pour spécifier explicitement la politique de sécurité et instrumenter le tatouage sans contraindre les utilisateurs. En particulier, dans ce cas concret, nous avons montré que le tatouage permet d'insérer une quantité d'information suffisante pour contribuer à la protection des images, sous des contraintes de distorsion forte. Pour atteindre un niveau de sécurité intéressant, nous avons montré que 3565 bits sont nécessaires à tatouer ce qui est plus faible que la capacité offerte par notre approche. Dans le même temps, nous avons montré comment l'API OrBAC joue le rôle de moniteur de référence en charge de l'application de la politique de sécurité. En combinant ces deux dimensions, modèle d'accès et d'usage et mécanismes de tatouage, notre système offre une protection continue de l'information.

Références

[CC 99] CC. Critère communs pour l'évaluation de la sécurité des technologies de l'information, partie 2 : Exigences fonctionnelles de sécurité. Version 2.1, août 1999, CCIMB-99-032, 394p. Disponible sur : www.ssi.gouv.fr.

[Dar 93] A. Dardenne, A. van Lamsweerde, S. Fickas, Goal-Directed Requirements Acquisition, in *Science of Computer Programming*, vol. 20, p. 3-50, 1993.

[EBI 04] EBIOS. Expression des Besoins et Identification des Objectifs de Sécurité, section 5 : Outillage pour le traitement des risques SSI. 2004, 284 p. Disponible sur : www.ssi.gouv.fr.

[Gra 11] M. Graa, N. Cuppens-Boulahia, F. Autrel, H. Azkia, F. Cuppens, G. Coatrieux, A. Cavalli, A. Mammar, Using Requirements Engineering in an Automatic Security Policy Derivation Process, SETOP'2011, K.U.Leuven Nieuwe Valk, Leuven (Belgium), September 2011.

[KAO 00] KAOS, Goal-Driven Requirements Engineering: the KAOS Approach, 2000, [On line], disponible sur : http://www.ingi.ucl.ac.be/research/projects/AVL/ReqEng.html.