Gestion de la sécurité dans une infrastructure de services dynamique : Une approche par gestion des risques

Pascal Bou Nassar

N° d'ordre 2012ISAL0102 Année 2012

Thèse

Gestion de la sécurité dans une infrastructure de services dynamique : Une approche par gestion des risques

Présentée devant
L'institut national des sciences appliquées de Lyon - France

Pour obtenir
Le grade de docteur

Formation doctorale : Informatique – Gestion de la sécurité
École doctorale : Informatique et Mathématiques

Par
Pascal Bou Nassar
Soutenue le 21 décembre 2012 devant la Commission d'examen

Jury MM.

Frédérique BIENNIER	Directeur de thèse - Professeur (INSA de Lyon)
Youakim BADR	Directeur de thèse - Maître de conférences (INSA de Lyon)
Kablan BARBAR	Directeur de thèse - Professeur (Université Libanaise)
Ernesto DAMIANI	Professeur (Università degli Studi di Milano)
Agnès FRONT	Maître de conférences (Université Pierre-Mendès-France)
Farouk TOUMANI	Professeur (Université Blaise Pascal-Clermont-Ferrand)
Olivier GARRO	Professeur (Université de Technologie de Belfort Montbéliard) - Directeur (Bureau Asie-Pacifique de l'Agence Universitaire de la Francophonie)

Laboratoire de recherche : Laboratoire d'InfoRmatique en Image et Systèmes d'information (LIRIS)

A Mon EPOUSE
Maria Dolorès

Remerciements

Je voudrais exprimer mes sentiments les plus sincères envers les personnes qui ont fait tout leur possible pour que ce travail de recherche puisse voir le jour.

Tout d'abord, je tiens à remercier, Mme Frédérique Biennier, Professeur à l'INSA de Lyon et directeur de cette thèse pour son aide et ses conseils qui ont été déterminants pour mon avenir professionnel. Je tiens à exprimer ma gratitude à M. Youakim Badr, maître de conférences à l'INSA de Lyon, qui m'a encadré efficacement tout au long de ces années. Cher Youakim, je te remercie pour ton soutien inestimable, pour le temps que tu m'as accordé et pour tous les conseils. Mes profonds remerciements à M. Kablan Barbar, professeur à l'université Libanaise, directeur du centre d'études et de recherches en informatique juridique et co-directeur de cette thèse pour sa confiance et son encouragement.

Mes sincères remerciements à mon employeur, l'Agence Universitaire de la Francophonie (AUF) pour m'avoir donné la chance de poursuivre ce projet avec toutes les responsabilités que j'assume en tant que responsable technique du Moyen-Orient. Je tiens tout particulièrement à remercier mon ex-Directeur Professeur Olivier Garro pour son appui. Cher Olivier, j'espère que tu trouveras dans cet aboutissement le fruit de la confiance que tu m'as accordée. Je tiens également à remercier Mme Salwa Nacouzi, directrice du bureau Moyen-Orient et Victor Bruneau directeur des ressources informatiques pour leur encouragement continu. Je n'oublie pas de remercier mes collègues, l'équipe du Moyen-Orient et en particulier, mon ami Toufic Wehbe pour sa lecture avisée et commentée de mon rapport.

Je remercie tous les membres du jury pour la grande attention qu'ils ont bien voulu porter à mon travail. Je remercie M. Ernesto Damiani, Professeur à l'Université de Milan et Mme Agnès Front, Maître de conférences à l'Université Pierre-Mendès-France pour avoir bien accepté d'être rapporteurs de cette thèse.

Pour conclure, je garde une place toute particulière à Vanessa El-Khoury pour son amitié, à ma famille, et surtout, mon épouse Maria Dolorès pour son amour sans limites, son encouragement et les sacrifices qu'elle a fait pour moi. Rien n'aurait été possible sans sa présence et son soutien.

Résumé

Les changements de contexte économiques imposent de nouvelles stratégies organisationnelles aux entreprises : recentrages métier et développement de stratégies de collaboration interentreprises. Ces tendances du marché laissent prévoir une croissance exponentielle d'écosystèmes de service accessibles à la fois aux clients finaux et aux partenaires. Tout laisse prévoir que ces écosystèmes s'appuieront largement sur les architectures orientées services permettant de construire des systèmes d'information capable d'avoir l'agilité requise et de supporter l'interconnexion des processus métier collaboratifs en composant dynamiquement les processus à partir de services distribués. Ce type d'architecture qui permet d'assurer l'alignement du système d'information sur les besoins métier de l'entreprise, rend indispensable la prise en compte des contraintes de sécurité tant au niveau individuel des services qu'au niveau de la composition.

Dans un environnement de services distribués et dynamiques, la sécurité ne doit pas se limiter à fournir des solutions technologiques mais à trouver une stratégie de sécurité prenant en compte les dimensions métier, organisationnelle et technologique. En outre, la sécurité doit être appréhendée comme un processus continu qui vise l'optimisation des investissements de sécurité et assure la pérennité des mesures de sécurité mises en œuvre. Or les modèles et architectures de référence du domaine des services ont sous-estimé la définition des besoins en termes de sécurité, les biens à protéger et l'identification des risques pesant sur ces biens. Pour cela, nous proposons d'aborder la problématique de la sécurité par une approche de gestion des risques permettant d'identifier les différents types de risques et de proposer les mesures de sécurité les plus adéquates au contexte. Toutefois, la gestion des risques s'avère un vrai défi dans un environnement ouvert de services collaboratifs. En effet, les méthodes de gestion des risques développées dans le cadre des systèmes d'information ne répondent pas aux exigences de sécurité dans un environnement ouvert et ne sont pas adaptées aux environnements dynamiques.

Pour pallier ces limites, nous proposons un cadre méthodologique de gestion de la sécurité portant sur les phases préparation, conception, exécution et supervision du cycle de vie des services. Nous proposons un modèle de services sécurisés permettant de définir des patrons de sécurité, un modèle de classification des biens à protéger et une ontologie pour définir les concepts associés à ces biens. En outre, nous développons une méthodologie de conception d'une architecture orientée services sécurisée puis abordons la construction de processus métier sécurisés avant de proposer un service de gestion des vulnérabilités de l'infrastructure.
Dans son ensemble, notre contribution apporte une vue "mixte" sur la sécurité dans un écosystème de service, c'est-à-dire intègre à la fois une vision technologique et une vision organisationnelle.
Mots clés : Service, Architectures Orientées Services, Sécurité, Gestion de la sécurité, Gestion des risques.

Abstract

Changes in economic environment impose new organizational strategies to companies: refocusing business and creating collaboration strategies. These trends point to an exponential growth of service ecosystems accessible to both end users and partners. All foreshadows that these ecosystems rely heavily on service-oriented architectures that can build information systems having the required agility and supporting the interconnection of collaborative business processes by composing processes dynamically from distributed services. This type of architecture that ensures business and information systems alignment, makes it essential to take into account security constraints at the services' and the composition's levels.

In a distributed and dynamic services' environment, security should not be limited to providing technological solutions but to find a security strategy taking into account the business, organizational and technological dimensions. Besides, the security must be considered as an ongoing process that aims to optimize security investments and ensures the sustainability of implemented security measures. However, the models and reference architectures in the services' domain have underestimated the definition of security requirements, assets to protect and the identification of risks to those assets. Therefore, we propose to address the security management issues by a risk management approach to identify the different types of risks and propose the most appropriate security measures to the context. Nevertheless, risk management is a real challenge in an open collaborative services' environment. The methods of risk management developed in the context of information systems do not meet the security requirements in an open environment and are not suitable for dynamic environments.

To overcome these limitations, we propose a methodological framework for security management covering the phases: preparation, design, execution and supervision of the services' lifecycle. We propose a model of secure services to identify security patterns, an assets' classification model and an ontology defining the concepts associated with those assets. Moreover, we develop a methodology for designing secure service oriented architectures, we address the development of secure business processes then we propose a security service for managing and supervising the infrastructure components' vulnerabilities.

As a whole, our contribution provides a "mixed" vision on security in a service ecosystem, that is to say, incorporating both a technological and an organizational vision.

Key words: Service, Service Oriented Architecture, Security, Security Management, Risk management.

Table des matières

Liste de figures

Liste des tableaux

Chapitre 1. Introduction Générale

1.1 Les services et leurs enjeux

Dans un contexte de mondialisation des marchés, de concurrence et de recentrage métier, de nouvelles stratégies organisationnelles sont imposées aux entreprises. Le besoin d'agilité des processus métier accentué par de nouveaux besoins et demandes des clients s'ajoute au besoin de développer des stratégies de collaboration interentreprises. Cette dynamique nécessite une importante capacité d'adaptation et de réactivité. Par conséquent, il est indispensable que les entreprises fassent tomber les barrières culturelles, fonctionnelles, organisationnelles et technologiques pour que l'ensemble des entreprises impliquées dans une collaboration soit perçu comme un tout homogène et cohérent [1]. Afin de combler ces besoins, il est important d'articuler l'action autour du système d'information de l'entreprise et de l'interconnexion des processus métier.

L'une des solutions technologiques permettant de répondre à ces attentes est l'architecture orientée services. Elle permet de construire un système d'information capable d'avoir l'agilité requise et de supporter l'interconnexion des processus collaboratifs, en composant dynamiquement les processus à partir de services distribués, accessibles à la fois aux clients finaux et aux partenaires. Ces services sont autonomes et permettent d'assurer une interopérabilité technologique voire sémantique. L'interopérabilité technologique est établie en se basant sur des standards. L'interopérabilité sémantique nécessite la description sémantique détaillée des fonctionnalités offertes par les services. L'approche orientée service permet ainsi de garantir l'agilité de l'entreprise [2], grâce à la composition et substitution de services, et facilite le changement de fournisseurs pour répondre à des nouveaux besoins et construire des nouveaux processus métier.

En plus des avantages mentionnés ci-dessus, l'approche service présente d'autres sources de valeur ajoutée. Premièrement, elle permet d'assurer l'alignement du système d'information sur les besoins métier de l'entreprise. En effet, les services sont issus d'un besoin métier et non pas d'un besoin technologique. Deuxièmement, elle permet la réduction des coûts de développement de nouveaux processus et de nouvelles applications grâce à la réutilisation des services. Enfin, l'approche orientée service permet l'interconnexion des processus métier collaboratifs d'une façon plus simple grâce à la publication des services et aux mécanismes d'échange de messages.

Toutefois, ce type d'architecture rend indispensable la prise en compte des contraintes de sécurité tant au niveau individuel des services qu'au niveau de la composition selon les dimensions métier et technologique. Par conséquent, il est judicieux d'aborder la problématique de la sécurité par une approche globale.

1.2 La gestion de la sécurité

La gestion de la sécurité impose d'abord de définir les besoins en termes de sécurité, et de les prendre en compte dans la définition d'une stratégie globale au niveau de l'organisation : c'est la politique de sécurité de l'entreprise. En effet, les processus métier reposent sur l'organisation, les procédures et la technologie. Sécuriser les aspects technologiques de ces processus est donc insuffisant, car il est nécessaire de considérer également les aspects non techniques.

Depuis les années 80, de nombreux standards ont été définis pour certifier les composants « technologiques » du système d'information avec une reconnaissance internationale. Ce n'est que récemment que la gestion globale de la sécurité (technologique et organisationnelle) a été prise en compte pour définir une politique de sécurité globale. La gestion de la sécurité est un processus qui se base à la fois sur les aspects organisationnels et technologiques de l'entreprise pour garantir la sécurité en prenant en compte l'information, le traitement (processus, fonctions ou applications, etc.), la technologie (matériel et systèmes d'exploitation, etc.), et l'environnement (acteurs, partenaires, locaux, etc.). D'après [3], une politique de sécurité qui ne prendrait pas en compte tous ces éléments et domaines serait instable et incomplète. Elle produirait une solution dangereuse reposant sur un faux sentiment de sécurité plus dommageable encore que de ne rien faire.

Un des défis majeurs de la gestion de la sécurité est d'adapter la sécurité selon les enjeux du système étudié. Les questions de savoir 'quels sont les biens à protéger' et 'quelles sont les mesures les plus adéquates pour le système' sont d'une importance majeure. En fait, les biens à protéger sont à la base de la prise de décision et l'attribution de priorités dans la sécurisation du système. Pour cela, il faut d'une part, se mettre d'accord sur la définition de ces biens, et d'autre part, identifier les risques pesant sur eux. Pour cela, la gestion des risques permet d'identifier l'ensemble des risques pesant sur la sécurité du système, de fixer les objectifs de sécurité et d'en déduire les mesures de sécurité permettant d'atteindre ces objectifs [4].

Avec l'évolution des exigences de la définition d'une politique de sécurité globale, différentes méthodes de gestion des risques ont été élaborées comme par exemple la méthode EBIOS, développée par l'Agence Nationale de la Sécurité des Systèmes d'Information (ANSSI), qui se focalise sur la sécurisation du système d'information de l'entreprise en définissant un cadre de gestion des risques pesant sur ce système.

En mettant l'accent sur différentes méthodes de gestion des risques, nous nous sommes aperçus qu'elles s'orientent vers la gestion de la sécurité dans des environnements statiques de systèmes d'information. Toutefois, dans le cadre de notre travail, notre attention s'oriente autour de la gestion de la sécurité dans un environnement dynamique de services dans un cadre collaboratif ouvert.

1.3 Problématique

La gestion des risques indispensable dans la gestion de la sécurité est donc un vrai défi dans un environnement ouvert de services collaboratifs. Partant du constat que les méthodes de gestion des risques ne sont pas adaptées aux environnements dynamiques, nous avons soulevé plusieurs questions :

- ✓ Comment adapter les méthodes de gestion des risques actuelles pour répondre aux exigences de sécurité dans un environnement ouvert basé sur des services ? Il faut en particulier étudier la prise en compte des aspects métier, organisationnels et technologiques de la sécurité et le périmètre étendu de l'entreprise, voire les nouveaux partenariats et collaborations.
- ✓ Comment gérer la sécurité dans les différentes phases du cycle de vie des services ? En effet, la sécurité ne se limite pas à la conception de services sécurisés, mais à l'application de la sécurité dans les phases d'exécution et de supervision.
- ✓ Comment prendre en compte la dynamique de l'environnement et l'adaptation de la sécurité en fonction des changements éventuels du contexte, voire la re-conception des processus métier en fonction des contraintes de sécurité ?

C'est autour de ces questions cruciales, correspondant à la gestion de la sécurité que s'articuleront nos travaux. À cette fin, nous proposons un cadre structuré permettant l'intégration des modèles, des méthodologies de conception et d'architecture d'exécution et de supervision dans le monde des services et de gestion du risque.

1.4 Notre approche

Comme le montre la Figure 1-1, la structure de ce rapport reflète la démarche permettant la prise en compte de la sécurité dans les différentes phases du cycle de vie des services. Nous développerons le mémoire en deux parties:

La première partie présente l'état de l'art. Elle se compose de trois chapitres portant sur :

- ✓ Les architectures orientées services et leur valeur ajoutée dans le développement des entreprises.
- ✓ La sécurité et les différents processus dans la gestion de la sécurité.
- ✓ La gestion des risques et les différentes méthodologies existantes.

Pour conclure cette partie, nous introduisons un cadre méthodologique de gestion de la sécurité pour répondre aux différentes exigences à partir des conclusions tirées dans les deux premiers chapitres.

La deuxième partie présente notre contribution. Elle se compose de trois chapitres :

- ✓ Le chapitre 5 couvre les modèles et les concepts d'une SOA sécurisée, ceci pour aborder la phase de préparation du cycle de vie des services.

- ✓ Le chapitre 6 présente notre méthodologie de conception intégrant un cycle de gestion de risques, une architecture de sélection de services sécurisés et un service de gestion des vulnérabilités de l'infrastructure.
  Cette méthode permet de couvrir les phases de conception, d'exécution et de supervision du cycle de vie des services.

Enfin, dans le chapitre 7, nous présentons l'outil de conception d'une SOA sécurisée et nous appliquons la méthodologie de conception sur un cas d'illustration : une plateforme de e-learning orientée services.

Ce chapitre est suivi par une conclusion générale et une présentation des différentes perspectives.

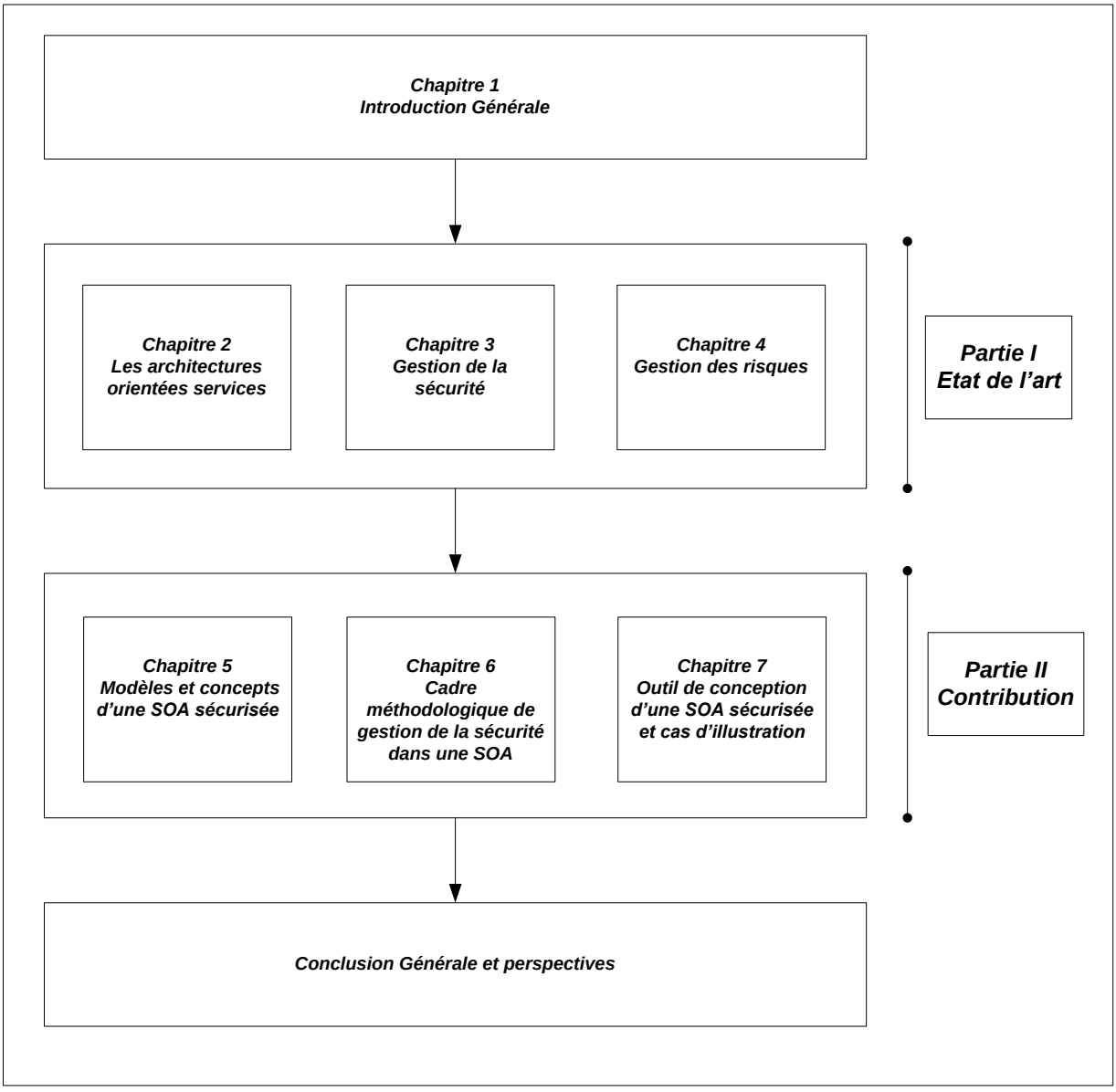


Figure 1-1 : Structure du mémoire de thèse

PARTIE I : ETAT DE L'ART

Chapitre 2. Les Architectures Orientées Services

Résumé

Dans ce chapitre, nous abordons les concepts autour des services et des architectures orientées services. Nous mettons en évidence la valeur ajoutée de l'adoption de ces architectures du point de vue métier et technologique et présentons les méthodologies de développement des SOA. Enfin, nous abordons les standards permettant la mise en œuvre des SOA.

Sommaire

2.1 Introduction

Ce premier chapitre d'état de l'art porte sur les architectures orientées services. Dans un premier temps, notre abordons les caractéristiques des services qui sont à la base du paradigme de l'orientation service. Ensuite, nous définissons les architectures orientées services et identifions la valeur ajoutée du développement de telles architectures.

Dans un deuxième temps, nous abordons les différents concepts à la base des architectures orientées services et qui sont pertinents pour la prise en compte de la sécurité. Nous présentons le modèle opérationnel de l'architecture, les standards existants, le cycle de vie des services, la composition et le bus des services. Enfin, nous abordons la gouvernance SOA qui représente un processus intéressant pouvant être à la base de la gestion d'un environnement de services.

Nous aborderons enfin les méthodologies de développement des architectures SOA pour identifier une méthodologie permettant de développer des SOA sécurisées puis la mise en œuvre des SOA en utilisant la technologie des services web. En effet, ces derniers représentent l'instance la plus répandue des architectures orientées services.

2.2 Les services

2.2.1 Définition

La notion de service est assez générale et existe dans plusieurs domaines métier et technologique. Par conséquent, la signification du terme varie selon le domaine et le contexte étudiés. Nous citons à titre d'exemple les définitions suivantes :

1. Dans le domaine de l'économie, l'INSEE définit le service comme étant une prestation qui consiste en « *la mise à disposition d'une capacité technique ou intellectuelle* » [5].
2. Dans le domaine du marketing et management, C. Gronroos définit le service comme «*Any activity or benefit that one party can offer to another which is essentially intangible and does not result in the ownership of anything*» [6].
3. Dans le domaine de l'informatique et des solutions applicatives, l'Open Group définit les services comme étant « *une représentation logique d'une activité métier répétitive* » [7]. Dans ce même domaine, O. Perrin décrit les services comme étant « *les fonctions d'une application offertes sur le réseau. Ces services sont, de manière intrinsèque, distribués, hétérogènes, dynamiques et surtout faiblement couplés* » [8].

Dans notre travail, nous nous intéressons aux services dans le domaine de l'informatique et des solutions applicatives. Par conséquent, nous retenons la définition de l'Open Group (qui met en évidence l'alignement métier dans le développement des applications) et nous l'enrichissons par la définition de O. Perrin d'où notre proposition de définition des services:

« Les services sont des représentations logiques d'activités métier répétitives. Ils sont distribués, hétérogènes, dynamiques et surtout faiblement couplés »

Pourquoi développer des services?

Les applications traditionnelles ont été principalement conçues pour répondre à des exigences métier identifiées. La conception de ces applications ne prenait pas en compte l'utilisation des fonctionnalités par d'autres applications pour répondre à de nouveaux besoins. Pour combiner les fonctionnalités de deux ou plusieurs applications, les approches traditionnelles ne suivaient aucun standard ce qui rendait cette opération extrêmement compliquée. Ce problème était même le plus souvent impossible à résoudre dans un environnement distribué entre différents partenaires.

Les services sont apparus comme une solution très intéressante pour ce problème. En effet, dans un environnement de services distribué, le concept d'application, tel que nous le connaissons depuis longtemps, change radicalement pour laisser la place au concept de composition de services et aux applications composées [9]. Les services contiennent un ensemble de fonctionnalités bien définies, accompagnées d'un ensemble d'informations décrivant les capacités du service ainsi qu'un ensemble de règles pour gérer cette information [10]. Les interfaces sont définies de manière standardisée ce qui facilite la réutilisation. Les limites du système ne sont donc plus celles de l'entreprise ce qui rend le partenariat entre plusieurs entreprises facilement envisageable.

2.2.2 Taxonomie de services

Dans [11], M. Rosen identifie sept types de services selon leur granularité:

1. Les *processus métier* couvrent l'ensemble de l'entreprise et font usage des services sous-jacents.
2. Les *services métier*, de forte granularité, exposent les fonctions métier au sein de l'entreprise.
3. Les *services de domaine*, de granularité moyenne, représentent les services métier spécifiques à un domaine particulier de l'entreprise. Ces services sont exposés uniquement au sein de leur domaine.
4. Les *services utilitaires* (utility services), de faible granularité, fournissent des fonctionnalités communes à travers l'entreprise.
5. Les *services d'intégration* exposent les applications existantes en tant que services à l'usage du reste de l'entreprise et fournissent un accès cohérent aux données.
6. Les *services externes* fournissent un accès aux systèmes et aux applications des fournisseurs ou des partenaires externes.
7. Les *services de l'infrastructure* sont utilisés dans la construction des services, indépendamment du domaine d'activité (par exemple, la sécurité, l'audit et l'orchestration).

Comme le montre la Figure 2-1, les processus métier et les services métier sont au sommet de cette hiérarchie.

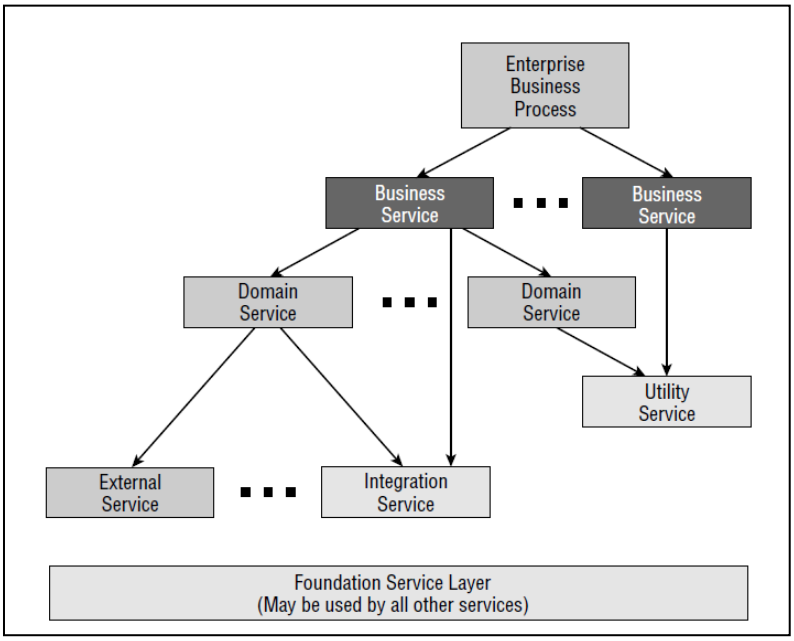


Figure 2-1 : Hiérarchie des services [11] p.58

Dans [9], T. ERL classifie les services suivant trois niveaux d'abstraction en se basant sur la réutilisation et la portée des services:

1. Les *services d'entités* (ou de données) correspondent à des services exposés par les objets métier de l'entreprise et permettent de gérer ces objets métier. Les objets métier représentent les concepts métier et correspondent à un regroupement de classes d'un point de vue informatique. Par exemple les services de type CRUD (create, read, update, delete) appartiennent à ce niveau.
2. Les *services centrés sur les tâches* (task services) permettent d'implémenter les activités d'un processus métier. Ces services sont directement liés à la logique métier des processus.
3. Les *services utilitaires* permettent de fournir des fonctionnalités communes à travers l'entreprise.

Dans [12], P. Bonnet élabore un méta modèle de service (présenté en Annexe 1) séparant les services métier et les opérations associées qui sont découverts lors de la modélisation des processus, des services identifiés au niveau du travail de conception de l'architecture orientée services, services qui sont attachés aux catégories (objets métier). Dans ce modèle, le service est le résultat d'un travail d'urbanisation sur le système d'information d'une entreprise. L'auteur met en relief trois types de service:

1. Le *processus métier* représente l'exécution d'un traitement métier de bout en bout pour une même préoccupation de gestion. Il se décompose en opérations et phases :

   a. L'opération représente le traitement offert par un service métier. En fait, un service métier peut contenir plusieurs opérations indépendantes. Notons que l'opération est identifiée dès la phase de spécification des processus et représente un ensemble d'activités contigües.
   b. La phase est le regroupement d'activités d'un processus dont l'implémentation n'est pas réalisée sous la forme d'une opération d'un service métier. À l'inverse, les activités qui concourent à la réalisation d'une opération d'un service métier forment une opération dans le processus
2. Le service métier, formé d'une ou de plusieurs opérations est découvert au niveau de la modélisation des processus métier. Les opérations sont des façades qui assemblent des services. En d'autres termes, les opérations d'un service métier sont réalisées par un ou plusieurs services. Ceci crée le lien entre les services métier et les services.
3. Le service 'technique' est identifié à l'interface de l'architecture applicative résultant du pattern d'architecture applicative SOA. Ce travail consiste à créer une architecture applicative qui décompose les traitements (opération, phase) sous la forme de services rattachés à des paquets de classes. Les paquets forment des catégories (objet métier) dotées d'une façade d'accès qui contient l'ensemble des services que cette catégorie expose.

Jusqu'à présent aucune taxonomie de services n'a été normalisée. Toutefois, nous notons que les classifications citées ci-dessus ne sont pas contradictoires. En effet tous ces travaux évoquent les concepts essentiels suivants :

- La séparation entre les services de niveau métier et les services de niveau informatique.
- Le partage des caractéristiques essentielles des services informatiques, permettant de valider leur conformité au paradigme « orienté-services »

Nous retenons alors la taxonomie des services suivante :

✓ *Services de niveau métier (services abstraits):*

1- Les *processus métier* sont un ensemble d'activités structurées formant le cœur du métier. Ils sont formés d'activités manuelles, d'activités semi-automatiques ou d'activités automatiques. Les deux derniers types d'activité seront réalisés par les opérations des services métier. Nous identifions deux types de processus métier, les *processus métier intra-entreprises* qui sont privés à l'entreprise et les *processus métier interentreprises* qui sont des processus métier collaboratifs entre partenaires.

2- Les *services métier* sont issus de la modélisation des processus métier et encapsulent une logique métier bien précise. Un service métier expose une ou plusieurs opérations qui seront implémentées par un ou plusieurs services informatiques. À noter qu'un service métier doit décrire les aspects métier et opérationnels liés à sa mise en œuvre. Il est totalement indépendant des implémentations techniques.

✓ *Services de niveau informatique (services concrets):*

1- Les services mis en œuvre dans une SOA, que nous appelons services dans le reste de ce travail, implémentent les opérations des services métier. Nous identifions deux types de services différents :
   - les *services atomiques* qui peuvent fournir des fonctionnalités par eux-mêmes et ne dépendent pas des autres services.
   - les *services composites* qui orchestrent des services atomiques pour offrir des fonctionnalités plus avancées.
   Nous retenons également la classification des services atomiques proposée par [9]: les services d'entités (de données), les services utilitaires (utility services) et les services centrés sur les tâches (task services).

2- Les *services de support* ou *d'infrastructure* sont des services qui facilitent la construction d'autres services, tels que les services d'orchestration, de routage, de sécurité, de supervision, etc. Ces services ne fournissent pas de fonctionnalité métier mais des capacités techniques essentielles pour supporter l'exécution de chaînes de services.

Dans ce qui suit, nous détaillerons les caractéristiques des services permettant la mise en œuvre du 'paradigme orienté service' en permettant la composition d'applications ou de processus métier à partir de services distribués.

2.2.3 Caractéristiques des services

Afin de répondre au 'paradigme orienté services', un service doit présenter les caractéristiques détaillées dans les paragraphes suivants.

2.2.3.1 Couplage faible

Le couplage faible assure la flexibilité et réduit les dépendances entre les différents services. Cette propriété permet d'avoir des services faiblement couplés pour simplifier la création et l'évolution des applications. Les aspects à considérer pour assurer le couplage faible sont les suivants :

a- Un service ne peut pas appeler un autre service : Il délègue cette fonction à un traitement spécialisé dans l'enchaînement (fonction d'orchestration) [12]. Cette propriété signifie aussi que le service doit intégrer les fonctionnalités dont il est lui-même responsable.

b- Une meilleure prise en compte de l'interopérabilité : Un service doit être indépendant des technologies d'implémentation. Il doit délivrer les fonctionnalités à travers son interface. En effet, le service doit être accompagné d'une description qui explique aux consommateurs la manière de l'invoquer et d'utiliser ses fonctionnalités. Il doit être activable indépendamment de sa technologie grâce à l'utilisation de standards. Par exemple, dans le monde des services web, l'envoi et de la réception des messages d'invocation se font en XML. SOAP est le format des messages utilisés. WSDL décrit l'interface du service. etc.

2.2.3.2 Découverte, sélection et consommation

Les services devront être publiés de manière à garantir leur découverte, leur sélection et leur consommation sans l'intervention du fournisseur. Les consommateurs utilisent les mécanismes de découverte pour localiser les services d'une manière transparente et utilisent la description des services pour les sélectionner et les 'consommer'. La description fonctionnelle et non fonctionnelle des services est à la base de leur découverte. Elle devra contenir des métadonnées concernant la localisation, les capacités et les exigences du service, et devra être réalisé avec précision à la conception des services. La sélection peut se faire sur la base des propriétés fonctionnelles et non fonctionnelles. À titre d'exemple, un consommateur peut rechercher la liste des services qui répondent au besoin fonctionnel et décider d'en sélectionner un particulier selon ses paramètres de qualité de service ou de sécurité.

2.2.3.3 Réutilisation

Cette propriété devra être prise en compte dès les premières phases de conception des services. En effet, les services doivent encapsuler une logique de traitement suffisamment générique pour être utilisés dans des contextes d'utilisation différents [13]. Cependant, le degré de réutilisation varie selon la portée du service et de sa granularité. Par exemple, un service de donnée a un potentiel fort de réutilisation. En revanche, les services propres à des processus métier particuliers ont une réutilisation limitée au processus en question.

2.2.3.4 Le contrat des services

Le contrat des services permet de décrire les services d'une façon normalisée. Il permet aux consommateurs de juger de l'utilisabilité du service pour le sélectionner. De plus, il permet aux consommateurs de connaître les capacités du service, ses exigences et la manière dont il fonctionne.
Dans [9], T. Erl définit le contrat des services comment un ensemble de documents contenant:

- ✓ La description de chaque opération du service.
- ✓ Les types de messages tels que les messages en entrée, les messages générés comme réponse et les messages d'exception ou d'erreur.
- ✓ La description de chaque type de données contenu dans les messages.
- ✓ La localisation physique du service et les protocoles de communication.
- ✓ Les informations et règles sur l'exploitation du service comme le temps de réponse, le temps pendant lequel le service doit être disponible, etc.

Dans le cas des services web, le contrat de service (intitulé 'interface de service') est décrit en deux parties. La partie abstraite déclare les messages en entrée et en réponse au traitement offert. La partie concrète décrit les standards et protocoles techniques utilisés pour l'activation du service.

2.2.3.5 Asynchrone / Sans état

Un service fonctionne de manière asynchrone. C'est-à-dire qu'il ne bloque pas le consommateur pendant qu'il s'exécute. Ce principe est intéressant pour réduire les goulets d'étranglement (performance, robustesse) [12]. En effet, les opérations d'un service s'exécutent sans maintenir un état pendant une longue période [9]. Une fois les opérations réalisées, le service ne retient aucune information sur l'état des messages ou du consommateur. Toutefois, l'information du contexte (comme les données de session, les droits d'accès et les règles de validation) peut s'avérer très importante dans le cas de la composition de services. C'est la raison pour laquelle la tâche de conservation et de passage de cette information est déléguée au service d'orchestration qui se charge de maintenir le bon déroulement des activités et de l'exécution de la composition.

2.2.3.6 Autonomie

Lors d'un travail d'architecture applicative, les services sont identifiés en décomposant les traitements métier (décomposition des activités métier). Ces services qui forment l'inventaire des services, devront être autonomes. L'autonomie hérite de la propriété 'couplage faible' et signifie que le service peut évoluer indépendamment des autres services de l'inventaire. Nous distinguons l'autonomie dans la phase de la conception et celle de la phase d'exécution. Dans le premier cas, l'autonomie permet d'assurer des changements au niveau du service sans affecter les autres services de l'inventaire. Dans la phase de l'exécution, cette propriété signifie que le service peut contrôler le traitement métier qu'il expose et les ressources qu'il utilise d'une façon autonome.

Après avoir présenté les définitions et les propriétés des services, nous abordons dans la section suivante les architectures orientées services.

2.3 Les architectures orientées services

2.3.1 Définition

Les architectures orientées services (Service-Oriented Architecture SOA – SOA) présentent un style d'architecture permettant d'assurer l'alignement métier et l'agilité dans le développement des applications. Nous listons plusieurs définitions qui illustrent différents points de vue d'une SOA:

Définition 1: "A Service Oriented Architecture is a form of distributed systems architecture that is typically characterized by the following properties: logical view, message orientation, description orientation, granularity, network orientation, platform neutral" [14] p.61
Cette définition du W3C se focalise sur les caractéristiques des services sans mettre en relief l'alignement métier dans une SOA.

Définition 2: "SOA is an architectural style for building enterprise solutions based on services. More specifically, SOA is concerned with the independent construction of business-aligned services that can be combined into meaningful, higher-level business processes and solutions within the context of the enterprise" [11] p.33
Dans cette définition, M. Rosen met en évidence l'alignement métier dans la vocation de la SOA.

Définition 3: "A Service Oriented Architecture is a software architecture that is based on the key concepts of service, service repository, and service bus. A service consists of a contract, one or more interfaces, and an implementation" [15] p.67
Dans cette définition, D. Krafzig donne un caractère technique à la SOA en se focalisant sur les composants techniques de l'architecture.

Dans notre travail, nous nous basons sur la définition de S. Chaari [1] dans laquelle, il intègre les aspects métier et technique de la SOA:
La SOA est un style d'architecture qui permet la réorganisation du système d'information. Elle permet l'encapsulation des fonctionnalités d'un système d'information en un ensemble de services faiblement couplés appartenant à la fois au niveau métier et au niveau technique. Les services, munis d'un contrat d'utilisation et d'une interface de description, seront publiés dans des registres de services afin qu'ils puissent être invoqués par d'autres services. [1] p.45

2.3.2 Valeur ajoutée des architectures orientées service

L'objectif principal des architectures orientées services est d'assurer un alignement entre les activités métier et le système d'information d'une manière qui optimise l'efficacité et l'agilité de l'entreprise. En effet, dans une SOA, l'entreprise se base sur le modèle du domaine métier qui décrit le fonctionnement des activités pour concevoir des services réutilisables en tirant parti des propriétés des services. Ceci permettra d'adapter rapidement le système d'information suite aux changements métier.

Dans les architectures orientées services, les nouvelles activités métier pourront être perçues comme étant des processus métier réalisés par des services. En effet, l'objectif principal dans une SOA est de découper les activités métier en des séries d'activités réutilisables qui pourront être complètement ou partiellement automatisées par des services métier. Les services reflètent à la fois l'image métier des activités et l'image technique des fonctions implémentées au niveau du système d'information.

Les responsables métier percevront les services comme des fonctions métier réutilisables par différents clients ou partenaires. Les responsables techniques percevront les services comme étant des fonctionnalités applicatives, disponibles pour être réutilisées pour construire facilement et rapidement de nouvelles applications suite à des changements ou de nouveaux besoins métier.

La Figure 2-2 illustre les concepts explicités ci-dessus. Dans cette figure, nous trouvons les processus métier qui représentent le cœur des activités métier de l'entreprise. La couche service représente l'interface entre les processus métier et les composants de l'architecture applicative. En particulier, nous avons représenté dans la couche infrastructure trois applications pour mettre en évidence la possibilité d'exposer les fonctionnalités des applications existantes dans l'entreprise en tant que services. Ceci permet d'optimiser la réutilisabilité des applications existantes dans la construction de nouvelles applications.

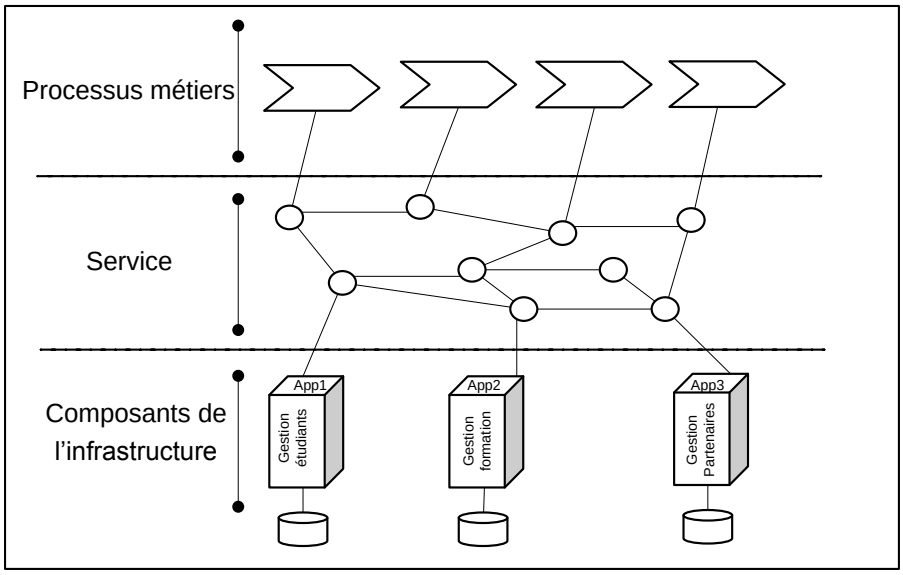


Figure 2-2 : Perspective d'une Architecture orientée services

2.4 Les concepts autour des services et des architectures orientées services

Dans cette partie, nous abordons les concepts qui sont à la base des services et des architectures orientées services. Ces concepts nous permettent d'identifier les différentes problématiques liées à la sécurité dans un environnement de services.

2.4.1 Modèle opérationnel de l'architecture orientée services

Une architecture orientée services définit un modèle opérationnel identifiant trois rôles : le fournisseur de service, le client de service et l'annuaire des services. Ce modèle fournit également un moyen générique de recherche et d'invocation des services.
Comme le montre la Figure 2-3, le fournisseur de services crée le service et publie sa description dans un annuaire de services. L'annuaire enregistre la description des services et fournit des capacités de recherche. Il intègre deux interfaces : une pour la publication, dédiée aux fournisseurs, et une pour la recherche dédiée aux clients.

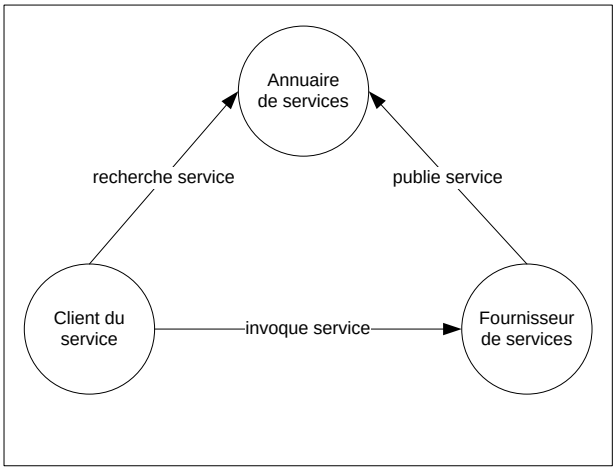


Figure 2-3 : Modèle opérationnel de l'architecture orientée services

Un client recherche auprès de l'annuaire les services qui répondent à ses besoins. Il obtient l'information de localisation du fournisseur et peut alors invoquer le service correspondant à son besoin. Si le client possède les informations nécessaires pour invoquer le service, il peut contacter directement le fournisseur.

Outre l'importance de ce modèle du point de vue opérationnel, les concepts de publication, recherche, sélection et invocation doivent être maîtrisés pour assurer la sécurité des services. En effet, compte tenu de la prolifération des services publics, plusieurs questions se posent comme par exemple :
Qui gère les annuaires publics ?
Est-ce que nous pouvons faire confiance à ces annuaires et aux informations publiées ?
Est-ce qu'il existe un mécanisme permettant de choisir des services sécurisés parmi d'autres ?

2.4.2 Modèles conceptuels de services, standards et leur complémentarité

Un modèle conceptuel d'un domaine est un modèle abstrait où les concepts de base sont identifiés grâce à leurs principales caractéristiques et liés les uns aux autres. Nous nous intéressons à ce concept afin de pouvoir présenter la structure et les relations entre les différents éléments des services. Dans ce qui suit, nous identifions les différents modèles et étudions leur complémentarité.

Dans le cadre du projet SeCSE, un modèle conceptuel a été proposé [16], permettant de fournir une définition claire de la notion de service et des concepts associés tels que la publication, la découverte, la composition, l'exécution et la supervision. Ce modèle a été conçu pour être une référence commune pour les partenaires impliqués dans le projet en décrivant les acteurs, les entités et les activités pertinentes ainsi que les relations entre eux. Les concepts sont modélisés par des diagrammes de classes UML complétés par un dictionnaire de données.

Le modèle présenté dans [17] vise à établir le lien entre les concepts des processus métier et les services SOA. Ce modèle permet de modéliser la dépendance entre les phases de conception et de déploiement. Cela permet, dans une approche MDA, de transformer les processus métier en services déployés dans un environnement de production.

Les organismes de standardisation, tels que l'OASIS, l'Open Group et l'OMG ont développé des modèles et des architectures de référence complémentaires, faisant preuve d'un large consensus sur les concepts de base. La Figure 2-4 reprise du modèle de référence de l'OASIS permet de faire le lien entre les différents types de modèles.

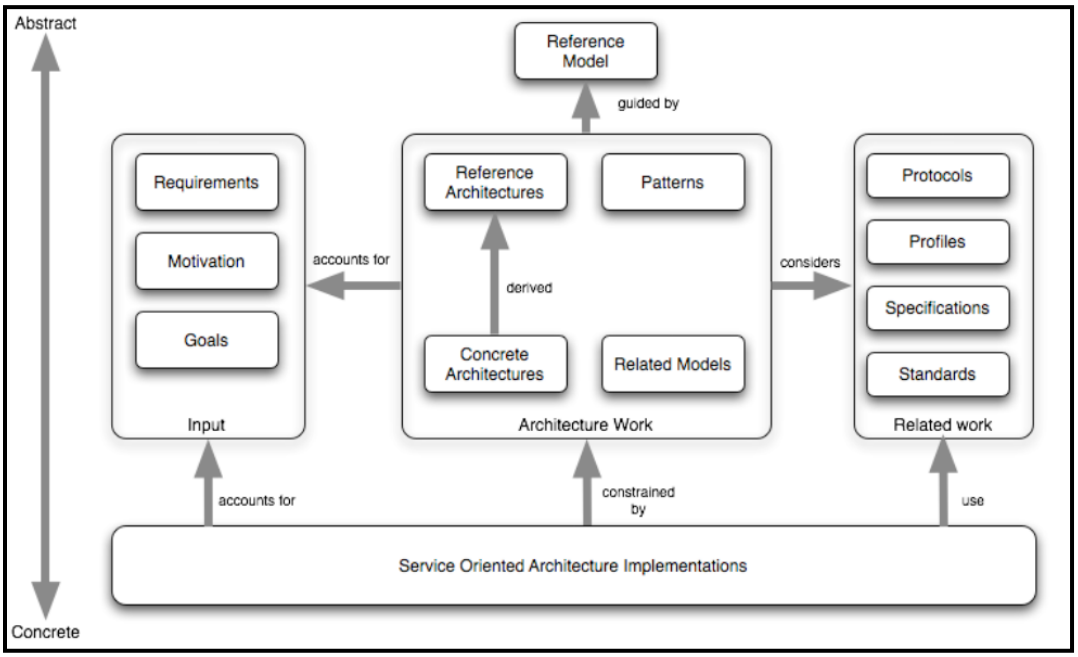


Figure 2-4 : Lien entre les différents modèles [18] p.5

Nous y notons que :

1. Le modèle de référence définit les éléments de base et sera à la base des architectures de référence
2. Les architectures concrètes sont la combinaison des architectures de référence, des patterns de conception et des architectures technologiques.

Une étude de ces différents standards, menée dans [19], montre comment ces derniers se complètent, en particulier :

- ✓ La conception d'un projet SOA se fait en utilisant le langage de modélisation SOAML [20] et les architectures de référence [21] [22].
- ✓ L'évaluation de la maturité des services pour entamer un projet SOA peut se faire en utilisant le modèle OSIMM [23]. La portée du projet peut être un projet de l'entreprise ou bien un projet inter entreprises.

- ✓ La définition d'un modèle de gouvernance de l'architecture orientée services par l'entreprise peut se faire en utilisant la plateforme « SOA Governance » [24]

Le modèle de référence proposé par l'OASIS [18] met l'accent sur les concepts fondamentaux de base d'une SOA : le service, sa description, sa visibilité, l'interaction et le contexte d'exécution. Ce modèle abstrait ne prend pas en considération la partie déploiement SOA. Le document « SOA Ontology » [7] développé par l'Open Group élargit la description des services à l'aide de nouveaux concepts et crée un langage commun pour la description des concepts SOA. L'architecture de référence proposée par l'OASIS [22] décrit la manière de réaliser une SOA. Elle fournit des directives et permet de gérer une SOA à plusieurs partenaires. De plus, cette architecture permet de créer un modèle de confiance et de politique de sécurité.

La Figure 2-5 montre les perspectives de réalisation d'une SOA où le modèle de description informe les participants sur l'existence des services et des conditions d'utilisation. Certaines de ces conditions découlent du modèle des politiques et des contrats. Les informations contenues dans la description des services complétées par les détails de la politique constituent la base du modèle de visibilité. Le processus dans lequel les services sont utilisés est décrit dans le modèle d'interaction avec les services.

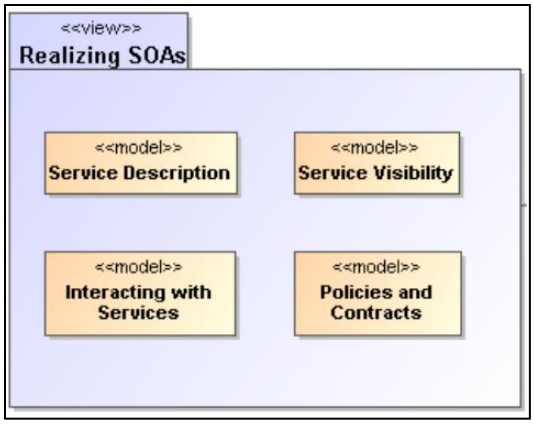


Figure 2-5 : Réaliser une SOA [5] p.48

Cet état de l'art permet de constater que les modèles conceptuels abordés ont atteint un bon niveau de maturité dans la conception, le déploiement et la gouvernance d'un projet SOA. Toutefois, nous notons que :

- ✓ Aucun des modèles conceptuels cités ci-dessus n'a été développé dans l'objectif de concevoir des SOA sécurisées où la construction d'une SOA sécurisée est la finalité attendue.
- ✓ Dans les modèles qui évoquent les notions de sécurité et de réseaux de confiance (comme l'architecture de référence de l'OASIS), la sécurité liée aux aspects métier et organisationnels n'est pas prise en considération.
- ✓ Aucun de ces modèles conceptuels ne se base sur la notion de risques pour identifier les menaces et en déduire les politiques de sécurité correspondantes.

2.4.3 Le cycle de vie des services

T. ERL définit le cycle de vie des services comme étant la séquence des phases suivantes : l'analyse, la conception, le développement, le test, le déploiement et l'administration [13]. Des conseils sont également donnés sur la possibilité de superviser le service pendant la phase d'administration, mais ce sujet n'a pas été approfondi. Oracle propose un modèle de cycle de vie qui sépare clairement la conception (identification des processus métier, modélisation de services, construction et composition) de l'exécution (déploiement, sécurisation, évaluation) [25]. M. Papazoglou propose un modèle du cycle de vie [26] qui commence par une phase initiale de planification, suivie par une série de phases itérativement répétées: l'analyse et la conception, la construction et les tests, l'approvisionnement, le déploiement, l'exécution et la supervision. La supervision est effectuée avant la mise en œuvre des services au moyen de tests de performance, et lorsque les services deviennent opérationnels, au moyen de techniques de surveillance de qualité de service.

Parmi les définitions mentionnées ci-dessus, nous trouvons que:

- ✓ La phase ‘planification’ qui est prise en compte uniquement dans la définition de M. Papazoglou est très importante. En effet, l’étude préalable des caractéristiques des services est essentielle avant d’entamer l’analyse et la conception. La formation des responsables métier et techniques aux concepts d’une SOA est primordiale pour une bonne préparation du projet SOA. Aussi, par la suite nous désignerons cette phase sous le terme de préparation.
- ✓ Les phases ’analyse’ et ‘conception’ sont fortement couplées et peuvent être intégrées dans une même phase.

Suite à ces constats, nous proposons :

- ✓ de définir le cycle de vie des services selon la séquence des phases suivantes : la préparation, la conception, la construction, le déploiement, le test, l’exécution et la supervision.
- ✓ d’intégrer la sécurité dans ces différentes phases et non pas seulement après la phase de déploiement comme c’est le cas de la définition proposée par ‘Oracle’. En effet, la définition des éléments relatifs à la sécurité (objectifs, besoins, mesures) dans la phase de préparation permettra de les intégrer de manière cohérente dans le reste du cycle de vie des services et par conséquent optimiser la gestion de la sécurité dans une SOA.

2.4.4 La composition des services

La composition des services englobe les fonctionnalités nécessaires pour l'agrégation de différents services en un seul service composite pouvant être utilisé soit directement par les

clients soit dans la composition de nouveaux services. Une fois composée, l'exécution de la chaîne de services peut être faite grâce à:

1- L'*orchestration* qui permet de spécifier l'ordre d'exécution des services et de décrire le flot de contrôle du processus qui sera géré par un moteur d'exécution. M. Papazouglou définit l'orchestration comme: *'Orchestration describes how services interact at the message level, including the business logic and execution order of interactions under control of a single end point. It is an executable business process that can result in a long-lived, transactional, multistep process model. With orchestration, one of the business parties involved in the process always controls the business-process interactions'* [27] p.67. L'orchestration suppose un contrôleur principal pour la composition des services et que les messages envoyés entre les activités ne sont pas visibles de l'extérieur. En d'autres termes, l'orchestration repose sur l'interaction entre les services qui composent un processus.

2- La chorégraphie se place dans une vision plus collaborative dans laquelle plusieurs processus échangent des messages en ayant une vue commune des différentes interactions, des différents messages et des activités de chaque partenaire. Cette vue commune a été préalablement définie par les partenaires, selon ce que chaque partenaire a déclaré être capable de faire. C'est donc une vue externe ou publique [8]. M. Papazouglou définit la chorégraphie comme: '*Choreography is typically associated with the public (globally visible) message exchanges, rules of interaction, and agreements that occur between multiple business process end points rather than a specific business process executed by a single party*' [27] p.68

La *composition* est un concept très important dans notre travail puisqu'il nous oblige à repenser à la cohérence globale de la sécurité. En effet, la composition des services doit prendre en compte à la fois la qualité de service (QoS) et la qualité de protection (QoP) des services qui sont composés. Les exigences et les politiques de sécurité des services doivent donc être respectées au moment de la composition ce qui implique également :

- ✓ La propagation des contextes de sécurité au niveau de la chaîne des services composant un processus métier intra ou inter entreprise.
- ✓ L'adaptation dynamique de la sécurité lors d'une composition dynamique des services, en mettant en place des solutions d'optimisation et d'adaptation.

2.4.5 Le bus de services

Un bus de services (ESB-Enterprise Service Bus) [28] permet l'interconnexion et le couplage faible des applications et des services à intégrer et facilite la gestion des applications et services. Comme le montre la Figure 2-6, le bus de services permet l'intégration de différents types de services et facilite la communication entre eux permettant l'interconnexion de services

développés dans différentes technologies (applications Java ou .Net, services web) en fournissant une infrastructure logicielle qui facilite l'intégration des services, le routage des messages, la médiation, le contrôle des services et parfois la gestion de la sécurité.

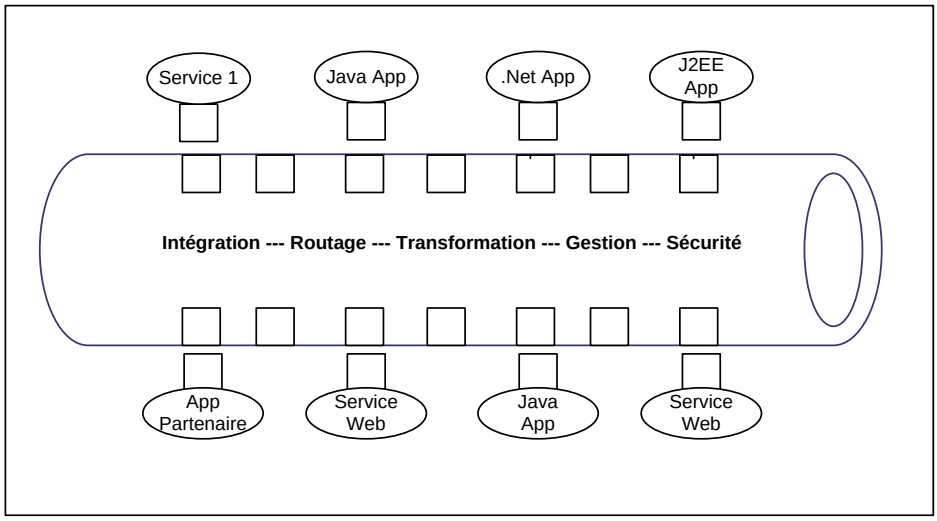


Figure 2-6: Bus de service - ESB

Un ESB est parfois considéré comme une nouvelle génération d'EAI (Intégration d'applications d'entreprise) construite sur des standards. La différence majeure avec l'EAI réside dans le fait que l'ESB propose une intégration complètement distribuée grâce à l'utilisation des conteneurs de services [29]. Tout service pourra être branché sur le bus de service pour faire partie d'un écosystème de services.

Dans notre travail, nous nous intéressons aux ESB puisqu'ils nous permettent de mettre en place des solutions de sécurité efficaces. Ci-dessous quelques utilisations sont données à titre d'exemple :

- ✓ Brancher des services de sécurité à l'ESB. Par exemple, nous pouvons déléguer la tâche d'authentification sur les services à un service d'authentification connecté à l'ESB.
- ✓ Inspecter les messages en fonction des politiques préétablies. Ceci permettra de renforcer et de vérifier la bonne application des politiques de sécurité.
- ✓ Permettre la propagation des informations de sécurité (jetons d'authentification, autorisations…) dans la composition de services interconnectés via le bus. Ceci permet notamment de vérifier les jetons de sécurité (ex : jetons SAML) encapsulés dans les messages et en transmettant les messages aux services concernés [30].
- ✓ Intégrer au sein de l'ESB des modules permettant d'assurer les services d'authentification, d'autorisation, d'audit, de gestion des clefs, de chiffrement, de signature et de supervision du routage.

2.4.6 La gouvernance SOA

Afin d'expliciter ce concept, nous commençons par rappeler les concepts liés à la gouvernance de l'entreprise et des systèmes d'informations. La gouvernance de l'entreprise définit les règles et la manière dont l'entreprise mène ses activités en se basant sur la stratégie métier et le marché. La gouvernance des systèmes d'information définit le processus et les règles permettant de diriger, contrôler et optimiser la gestion du système d'information pour atteindre les objectifs métier de l'entreprise.

Directement décliné de ces principes, la gouvernance SOA permet de s'assurer que les concepts et les principes de l'architecture orientée services est gérée de façon appropriée et que les objectifs métier sont respectés. E. Marks définit la gouvernance SOA comme suit :

'SOA governance refers to the organization, processes, policies, and metrics required to manage an SOA successfully. A successful SOA is one that meets defined business objectives over time. In addition, an SOA governance model establishes the behavioral rules and guidelines of the organization and participants in the SOA, from architects and developers to service consumers, service providers, and even applications and the services themselves' [31] *p.248*

Dans un environnement SOA, les responsables métier et techniques devront assurer la qualité des services en prenant en compte la totalité des phases du cycle de vie de ces services. À titre d'exemple, les responsables métier devront veiller à la bonne application des contrats de qualité de service. Les responsables techniques devront veiller à la mise en place, la maintenance et la supervision des services.
Après le déploiement d'un service, le suivi doit être organisé pour contrôler et superviser l'architecture. La gouvernance SOA définit les politiques et les processus nécessaires pour contrôler le développement, le déploiement et la gestion des services en intégrant entre autres la sécurité. Ces politiques et processus de gouvernance sont d'une importance cruciale dans la définition du contexte de la SOA à sécuriser et pour permettre la gestion de la sécurité.

Toutefois, la performance globale est largement dépendante des phases de conception et de déploiement de l'architecture. Nous introduisons quelques démarches méthodologiques liées à ces phases dans la section suivante.

2.5 Méthodologies de développement des architectures orientées services

Dans cette partie, nous nous intéressons aux méthodologies de conception des SOA pour trouver celles qui permettent de construire des SOA en assurant d'une part l'alignement métier et d'autre part la sécurité des services développés. Nous présentons différentes approches et méthodologies de conception qui font référence dans le domaine avant de définir des critères de comparaison pour établir un tableau comparatif de ces différentes méthodologies.

2.5.1 Stratégies utilisables dans la conception des architectures orientées services.

Dans le Tableau 2-1, nous décrivons les approches utilisées dans la conception des SOA. Celles-ci spécifient la découverte des services et la façon d'aborder les aspects métier et technologiques dans une SOA

Type	Description
Approche Ascendante (AS) (Botton – UP)	Cette approche consiste à intégrer les solutions logicielles qui ont été conçues et créées pour des projets antérieurs. La découverte des services commence par la collecte des ressources logicielles existantes, par exemple les composants d'application, les middlewares et le code source.
Approche Descendante (DS) (Top – Down)	Cette approche consiste à découvrir les processus métier de l'entreprise et identifier les activités correspondantes qui seront mises en œuvre par des services.
Approche (OI) (Outside – In)	C'est une combinaison des deux approches précédentes dont les activités se déroulent en parallèle. Un référentiel de services est établi afin d'utiliser les services existants lors de la conception ou de créer de nouveaux services si nécessaire.
Approche (MO) (Middle – Out)	Cette approche consiste à former les personnes qui s'engagent dans la conception d'un projet SOA afin de considérer d'une part, les besoins métier et l'organisation du projet et d'une autre part, les livrables immédiats. Cela est réalisable si nous nous basons sur une architecture de référence de sorte que le contexte soit présent dans toutes les activités de conception [11].

Tableau 2-1 : Type des méthodologies de conception des SOA

D'après la description des différentes approches, nous avons retenu les avantages suivants des approches Middle-Out et Outside-In :

- ✓ L'approche Middle-Out est intéressante puisqu'elle permet de se baser sur un modèle ou un cadre de référence simplifiant le dialogue entre les responsables métier et technique pour introduire une SOA. De plus, elle permet d'assurer une cohérence globale dans la conception.
- ✓ L'approche Outside-In est intéressante puisqu'elle permet d'assurer un alignement métier et d'optimiser la réutilisation des services. Ceci suppose dans un premier temps, d'identifier les services existants qui répondent aux besoins et dans un deuxième temps, de créer les services qui manquent.

Suite à ce constat, nous proposons de combiner les deux approches « Outside-In » et « Middle-Out » afin de profiter des caractéristiques de ces deux stratégies dans la conception des architectures orientées services.

2.5.2 Panorama des méthodologies de développement des SOA

Plusieurs méthodologies ont été proposées pour concevoir et mettre en œuvre des SOA :

- ✓ IBM Service Oriented Modeling and Architecture (SOMA) [32] est une méthodologie de modélisation composée de trois étapes: l'identification, la spécification et la réalisation des processus métier, des services et des composants. Le processus de SOMA est itératif et incrémental et couvre les phases d'analyse et de conception. Cependant, la méthode est propriétaire ce qui limite l'accès aux spécifications détaillées.

- ✓ A partir de son cadre de référence (SOA Reference Framework), CBDI-SAE a développé une méthodologie SOA de conception et un modèle de service [33]. La méthodologie vise à optimiser l'alignement métier en suivant une approche descendante. Le cycle de vie de services est couvert dans sa totalité, y compris les activités de déploiement, de surveillance et de gouvernance. Toutefois, l'accès à cette méthode est restreint : si la version 2 est en diffusion publique, la version la plus récente (version 3) est encore à accès réservé.

- ✓ Service Oriented Architecture Framework (SOAF) [34] est une plateforme composée de cinq phases principales: l'élicitation des informations, l'identification des services, la définition du service, la réalisation du service et la planification. SOAF se base sur la modélisation des processus métier et des services selon une approche descendante à la base des besoins métier et une approche ascendante à partir des processus métier existants. SOAF couvre les phases d'analyse et de conception du cycle de vie des services.

- ✓ Dans [26], Papazoglou propose une méthodologie de développement de service (SODM) qui couvre la totalité du cycle de vie des services, tout en tenant compte des points de vue du fournisseur et des consommateurs des services. Cette méthodologie se base sur des modèles de développement comme le RUP (Rational Unified Process), la CBD (Component Based Developpment) et le BPM (Business Process Modelling). La méthodologie utilise un processus itératif et incrémental incluant une phase préparatoire et huit phases principales (l'analyse, la conception, la construction, les tests, l'approvisionnement, le déploiement, l'exécution et la supervision des services)

- ✓ Dans [13], T. Erl présente une méthodologie qui s'oriente vers l'implémentation d'une SOA par des services web. Dans ce travail, l'auteur se focalise sur deux phases principales du cycle de vie des services : l'analyse et la conception. Dans la première, il présente la façon d'identifier les services en suivant une approche descendante. Dans la deuxième, il détaille la spécification des services identifiés et présente la façon de les implémenter par des services web.

- ✓ Dans [35], S. Jones présente une méthodologie de conception qui met en évidence les propriétés métier d'une architecture orientée services. Il souligne l'importance des concepts métier et se focalise sur les éléments {Qui, Quoi, Comment et Pourquoi} dans l'identification des services métier selon une approche descendante. Toutefois, l'auteur n'aborde pas les aspects technologiques des architectures orientées services.

2.5.2 Comparaison des méthodologies de développement des SOA

Pour comparer les différentes méthodologies de conception, nous avons retenu différents critères:

Critère 1 : Flexibilité
Ce critère décrit la capacité de la méthode à adapter l'architecture orientée services aux nouveaux besoins métier. Ce critère nous permet de différencier les méthodologies qui utilisent des approches agiles dans le développement pour plus de flexibilité des autres qui adoptent des approches plus rigides limitant leur flexibilité.
Critère 2 : Accessibilité
L'accessibilité dénote l'accès aux détails de la spécification de la méthode. Ce critère nous permet de différencier les méthodologies propriétaires dont les spécifications ne sont pas accessibles et celles non propriétaire qui sont accessibles en détail.
Critère 3 : Approche d'identification des services
Ce critère est en rapport avec les approches utilisées dans la conception des architectures orientées services: Approche Ascendante (AS), Approche Descendante (AD), Approche 'Outside In' (OI) et Approche 'Middle Out' (MO). Ce critère nous permet de déterminer comment la méthodologie aborde l'identification des services et la réutilisation des ressources.
Critère 4 : Portée
Ce critère décrit les phases du cycle de vie des services couvertes par la méthodologie. Dans notre travail, nous proposons d'intégrer la sécurité dans la totalité des phases du cycle de vie des services. Par conséquent, il est important de déterminer si les méthodologies couvrent la totalité de ce cycle de vie, ou bien se limitent à des phases particulières.
Critère 5 : Richesse de la méthodologie
Ce critère dénote la richesse de la méthodologie en termes de guides d'utilisation, de directives et de bonnes pratiques pour évaluer l'autonomie de la méthodologie et déterminer si elle présente un support facilitant la conception des services.
Critère 6 : Prise en compte de la sécurité
La prise en compte de la sécurité dès les premières phases de la conception est un aspect crucial pour assurer la sécurité des architectures orientées services. Nous cherchons une méthodologie qui met en évidence les différents aspects de la sécurité tels que la prise en compte des risques, des objectifs de sécurité et des mesures de sécurité.

La comparaison des méthodologies selon ces critères est résumée dans le Tableau 2-2

Critère	Méthodologies de conception					
	SOMA	CBDI-SAE	SOAF	SODM	T. ERL	S. Johns
Flexibilité	++	+++	+	++	+	SO
Accessibilité	Non accessible	Non accessible	Version 2 oui	Accessible	Accessible	Accessible
Approche d'identification des services	Approche Outside-In	Approche Outside-In	Approche Outside-In	Approche Outside-In	Approche Descendante	Approche Descendante
Portée	Analyse et conception	Tout le cycle de vie	Analyse et conception	Tout le cycle de vie	Analyse, conception et implémentation	Analyse
Richesse de la méthodologie	+++	+++	++	++	+++	+
Prise en compte de la sécurité	Non	Non	Non	Non	Non	Non

+ : niveau mineur de flexibilité (capacité de la méthode à adapter l'architecture orientée services aux nouveaux besoins métier) / richesse de la méthode en termes de guides d'utilisation, de directives et de bonnes pratiques.
++ : niveau moyen de flexibilité / richesse
+++ : niveau majeur de flexibilité / richesse
SO : Sans Objet (Le critère ne s'applique pas)

Tableau 2-2 : Comparaison des méthodologies

On remarque que CBDI-SAE et SODM couvrent la totalité des phases du cycle de vie des services.

- ✓ CBDI-SOA est une méthodologie assez riche. Toutefois, elle n'est pas accessible et ne prend pas en compte la sécurité.
- ✓ La méthodologie SODM est accessible, mais ne prend pas les aspects de la sécurité.

Les méthodologies restantes se limitent à l'étude d'une phase particulière du cycle de vie des services et ne prennent pas en compte les aspects de la sécurité.

Cette étude montre qu'aucune des méthodologies étudiées n'est complète et ne répond à nos besoins pour développer des architectures orientées services sécurisées. Pour cela, nous proposons de développer une méthodologie permettant d'assurer la conception et le déploiement des architectures orientées services sécurisées en intégrant une approche de gestion des risques.

2.6 Mise en œuvre d'une SOA : Les services web

L'instance la plus connue des architectures orientées services est basée sur les services web bien que d'autres technologies (CORBA [36] ou Enterprise Java Beans [37]) permettent aussi cette mise en œuvre.

2.6.1 Définition

Le point fort de la technologie des services web est qu'elle est indépendante des plateformes d'implémentation et des langages de programmation. Le groupe de travail « Web services Architecture » de la W3C (World Wide Consortium) a défini les services web comme : *'A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP-messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards'* [38] section 2.

Cette définition montre qu'un service Web peut être vu comme étant une application accessible par une interface spécifiée selon le standard Web Services Description Langage WSDL [39]. De plus, l'interaction entre les services web se fait via le standard SOAP [40]. Ainsi, les services web utilisent des standards pour décrire leur fonctionnement et la façon d'interagir avec eux.

2.6.2 La description, la découverte et l'invocation des services web

Plusieurs standards garantissent l'interopérabilité des services web pour permettre :

- ✓ *La description :* Le standard WSDL décrit l'interface du service web. Cette description est fondée sur le langage XML et spécifie la façon de communiquer avec le service. WSDL permet de définir, d'une manière abstraite, l'ensemble des opérations et des messages qui peuvent être transmis vers et depuis un service Web donné. Le document WSDL décrit l'information concernant (1) les opérations offertes par le service, (2) le type de donnée des messages en entrée et en sortie, (3) le protocole de transport utilisé, (4) l'adresse de localisation du service.

- ✓ *La découverte : Afin d'invoquer un service web, il 'suffit' au client de consulter son interface et de comprendre sa description. Ceci est réalisable en publiant la description du service dans un annuaire. Le standard 'Universal Description and* Discovery Interface' *(UDDI)* [41] *représente la spécification d'un annuaire conçu particulièrement pour les services web. UDDI permet la recherche, la découverte et la publication de services Web. Une requête de recherche renvoie le lien vers la description WSDL du service qui répond aux besoins.*

- ✓ *L'invocation : Le protocole SOAP définit l'ensemble de règles pour structurer et représenter les messages à échanger entre les services. Ces messages sont acheminés en utilisant un protocole de transport tel que http ou ftp (en dessus de TCP/IP). Un message SOAP est l'équivalent d'une enveloppe, formée de :*
  1) Une entête encapsulant des informations concernant le transport, l'expéditeur et le destinataire.
  2) Un corps du message encapsulant les données des applications telles que les appels de méthode, les paramètres et / ou les réponses aux requêtes correspondantes.

2.6.3 Enrichir la description des services web

Afin d'enrichir la description des services, nous pouvons inclure la description des propriétés non fonctionnelles tels que la qualité de services et la sécurité. Ceci est faisable en se basant sur le standard WS-Policy [42] qui décrit les exigences et les capacités du service web. Ce standard fournit un cadre pour étendre la description faite par le fichier de description WSDL. WS-Policy définit :

- ✓ Une syntaxe pour exprimer les politiques représentant les capacités et les exigences du service web.
- ✓ Un algorithme permettant de comparer et de trouver les similarités de deux politiques.

WS-Policy est complétée par trois autres standards.

1- WS-PolicyAssertions [43], spécifie la structure de quelques assertions génériques.
2- WS-Policy Attachment [44], définit comment associer les politiques au service web et ceci en l'intégrant directement au WSDL ou indirectement en l'associant à l'annuaire UDDI.
3- WS-SecurityPolicy [45], spécifie un ensemble d'assertions de sécurité correspondant à la sécurisation des messages SOAP.

Enfin, en enrichissant la description des services web, on enrichit implicitement la sélection de ces services. En effet, Les politiques ajoutées à la description permettent de sélectionner le service qui répond au mieux aux aspects fonctionnels et non fonctionnels.

2.6.4 La composition des services web

Comme nous l'avons vu dans la section 2.4.4, l'orchestration et la chorégraphie régissent la logique d'exécution après la composition des services. L'orchestration mise en œuvre par le composant d'orchestration de l'ESB, permet l'invocation de services web dans un ordre bien précis. Le 'Web Service Business Process Execution Langage' (WS-BPEL) [46] est le standard utilisé dans l'orchestration des services web. C'est un langage basé sur XML pour composer des chaines de services en se basant sur des services distribués. BPEL permet la modélisation des l'orchestration des services BPEL définit des processus qui seront exécutés dans un WfMS (workflow management System) qui est le moteur BPEL. De plus, le standard spécifie les activités à accomplir dans l'orchestration des services (par exemple, *invoke, receive, reply*).

Le 'Web Services Choreography Description Language' WS-CDL [47] est un exemple de langage pour la description des chorégraphies des services web. C'est un langage basé sur XML permettant à l'utilisateur de décrire des collaborations pair à pair (P2P) entre services en définissant leur comportement commun et observable. Un document de chorégraphie décrit les interactions entre participants. L'accomplissement de leur but commun se fait alors par des échanges ordonnés de messages suivant le plan déterminé dans ce document [48].

2.7 Conclusion

Les architectures orientées services représentent un style d'architecture permettant d'améliorer l'agilité du système d'information. Cependant la mise en place d'une telle architecture nécessite une préparation approfondie dans l'objectif d'atteindre la finalité d'une SOA qui est l'alignement métier et applicatif.
Dans ce premier chapitre de l'état de l'art, nous avons abordé les concepts, les modèles et les standards dans le domaine des architectures orientées services. Nous avons d'abord recherché des modèles de référence permettant de définir les concepts d'une SOA, puis des architectures de référence, des standards et des technologies permettant de réaliser une SOA.
En abordant, les concepts autour de la SOA, nous avons pu identifier des solutions intéressantes permettant d'améliorer la sécurité. Les ESB permettent de centraliser la sécurité en connectant des services de sécurité, par exemple des services d'authentification et d'autorisation. L'implémentation de la gouvernance SOA pourra améliorer la sécurité en intégrant un processus de gestion et de supervision. Le standard WS-Policy peut être utilisé pour enrichir la description des services par des paramètres non fonctionnels tels que les paramètres de la sécurité et de la qualité de service. Toutefois, nous n'avons trouvé dans la littérature abordée :

1. ni un modèle permettant de faire le lien entre les caractéristiques des services, de la sécurité et des risques et pouvant être un modèle de référence dans la gestion de la sécurité dans une infrastructure de services
2. ni des méthodologies permettant le développement d'une SOA sécurisée.

Dans le chapitre suivant, nous complétons notre étude en abordant les concepts de la sécurité pour mettre en relief les défis de la sécurité dans les architectures orientées services afin de maîtriser la gestion de la sécurité dans ces architectures.

Chapitre 3. Gestion de la Sécurité

Résumé

Dans ce chapitre, nous nous intéressons aux concepts liés à la sécurité, aux défis de la sécurisation dans les environnements de services, ainsi qu'aux différentes solutions proposées dans la littérature. Nous abordons également la gestion de la sécurité et le besoin d'avoir un cadre de gestion pour la définition d'une politique de sécurité globale.

Sommaire

3.1 Introduction

L'adoption croissante des technologies de l'information et de la communication (TIC) améliore la productivité des entreprises et leur permet d'offrir de meilleurs services, d'améliorer leur efficacité et leur agilité. Toutefois, l'évolution rapide de ces technologies a pour contrepartie l'augmentation des risques liés aux systèmes d'information (risques liés à la manipulation, au stockage et à la transmission des données, etc.)

Ce deuxième chapitre de l'état de l'art porte sur la sécurité, ses concepts et ses implications dans les architectures orientées services. Nous étudions d'abord les concepts liés à la sécurité en mettant en évidence l'implication des aspects métier et technologique dans la sécurité d'un système.

Ensuite, nous traitons les défis de la sécurité dans les architectures orientées services et présentons différentes contributions permettant d'améliorer la sécurité dans les SOA en général et dans les services web en particulier. Ces contributions nous permettent de mettre en évidence l'importance d'un cadre de gestion dans la sécurisation des architectures orientées services.

Enfin, nous abordons les concepts liés à la gestion de la sécurité en abordant les cadres de gestion dont l'objectif est l'élaboration d'une stratégie globale de sécurité d'une part et les standards existants dans le domaine de la gestion de la sécurité des systèmes d'information d'autre part. Ces standards définissent les processus, les directives à suivre et les bonnes pratiques dans la gestion de la sécurité et placent la gestion des risques au cœur de leur démarche de gestion.

3.2 Concepts liés à la sécurité

Dans [49] p.7, S. Zevin définit la sécurité comme étant '*la protection de l'information et des systèmes d'information contre tout accès et utilisation non autorisés, divulgation, perturbation, modification ou destruction'*. Selon C. Alberts, '*la sécurité revient à déterminer ce qui doit être protégé et pourquoi, ce qui a besoin d'être protégé et comment le protéger tant qu'il existe*' [50] p.5. Ces définitions nous montrent que la sécurité d'un système revient à la définition de ce système et à l'identification de la portée de la sécurité sur la totalité des composants formant le système.

La sécurité représente la '*satisfaction des besoins de sécurité des biens essentiels*' selon [3]. Les besoins de sécurité créent des objectifs de sécurité à atteindre et conduisent à mettre en place des mesures pour améliorer la sécurité d'un système. Dans ce qui suit, nous traitons les objectifs de sécurité, les mesures de sécurité et les stratégies de développement de systèmes sécurisés.

3.2.1 Les objectifs de sécurité

Usuellement trois objectifs de base de sécurité sont définis : la confidentialité, la disponibilité et l'intégrité. À ces objectifs s'ajoutent des objectifs de support de sécurité. Dans la littérature de la sécurité informatique, les objectifs de sécurité sont nommés les services de sécurité. Dans notre travail, et pour qu'il n'y ait pas de confusion, nous faisons la distinction entre les objectifs de sécurité et les services de sécurité. Nous désignons les services de sécurité comme étant des services offrant des fonctionnalités assurant les objectifs de sécurité. Les services de sécurité seront explicités dans la partie 'la sécurité en tant que service' de ce chapitre (p. 58). Les objectifs de sécurité conditionnent toute décision de sécurité prise par l'entreprise. Ces objectifs seront satisfaits par la mise en place de mesures technologique de sécurité ou bien par la mise en place de procédures de sécurité métier.

3.2.1.1 Les objectifs de base de sécurité

La confidentialité

La confidentialité est un objectif de sécurité permettant de protéger l'information au repos et lors de son échange contre toute divulgation et accès non autorisés. La confidentialité doit être assurée techniquement (mécanisme de chiffrement et de contrôle d'accès) et non techniquement (classification des informations et mise en place de politiques de contrôle d'accès) afin de ne donner l'accès qu'à ceux qui sont autorisés. Cet objectif peut porter sur la protection d'un message élémentaire ou d'un champ spécifique à l'intérieur d'un message en recourant aux objectifs support d'authentification et de contrôle d'accès.

Les informations peuvent avoir différents niveaux de confidentialité. Si certaines informations n'ont aucune exigence de confidentialité (information publique qui peut être accessible par tout le monde) d'autres informations doivent être plus étroitement contrôlées et partagées uniquement par les partenaires métier, voire pour les plus sensibles n'être accessible que par certaines personnes.

Un autre aspect de la confidentialité est la protection du flot de trafic contre l'analyse. Cela requiert qu'un attaquant ne puisse observer les sources et destinations, les fréquences ou autres caractéristiques du trafic sur un équipement de communication [51].

La disponibilité

L'objectif de la disponibilité est de garantir l'accès à un service, à une information ou à une ressource à tout moment pour les personnes autorisées. La disponibilité est assurée techniquement en assurant la protection des ressources et des biens (par exemple, les applications, les systèmes d'hébergement et les équipements réseau) et de s'assurer que ces biens fonctionnent correctement. Toutefois, il ne faut pas oublier que la disponibilité est aussi assurée en mettant en place des procédures et des politiques de sécurité. En effet, les dénis de service (indisponibilité du service) sont causés par les attaques malveillantes qui peuvent résulter de la non-application des politiques et des procédures de sécurité.

L'intégrité

D'après l'agence nationale de la sécurité des systèmes d'information (ANSSI) [52], l'intégrité est la propriété assurant qu'une information ou un traitement n'a pas été modifié ou détruit de façon non autorisée. Pour assurer l'intégrité de l'information en transit, on peut utiliser des mécanismes de signature électronique. L'intégrité de l'information au repos est assurée en utilisant des mécanismes de signature de cette information d'une part et en vérifiant par des mécanismes de détection d'intrusion que les systèmes hébergeant l'information fonctionnent d'une façon fiable d'autre part.

3.2.1.2 Les objectifs de support de sécurité

Dans le Tableau 3-1, nous décrivons brièvement les objectifs de sécurité de support nécessaires à la mise des objectifs de sécurité de base. La confidentialité est assurée en mettant en place des mécanismes d'identification, d'authentification, d'autorisation et de chiffrement. L'intégrité est assurée par les mécanismes d'authentification, d'audit, de non-répudiation et de signature. La disponibilité est assurée en mettant en place des mécanismes de contrôle d'accès, d'audit, de redondance, de filtrage (pare-feu), etc.

Objectifs de sécurité	Description
Identification	Cet objectif permet d'attribuer des identifiants aux utilisateurs ou aux services. En particulier, la fédération d'identité permet à un utilisateur d'utiliser le même identifiant pour divers domaines de confiances. L'identification sert à l'audit et la traçabilité des activités d'un utilisateur ou d'un service.
Authentification	Cet objectif permet de valider l'identifiant d'un utilisateur ou d'un service. Ceci est réalisé en présentant la preuve de possession de l'identité (soumission d'un mot de passe, d'une clé secrète, d'une signature numérique, etc.)
Contrôle d'accès (Autorisation)	Cet objectif permet de contrôler l'accès à l'information et aux systèmes. Pour réaliser ce contrôle, chaque entité essayant d'obtenir un accès doit d'abord être authentifiée avant que les droits d'accès puissent être calculés [51].
Non-répudiation	Cet objectif empêche aussi bien l'expéditeur que le receveur de nier le fait d'avoir transmis ou reçu une information. Lorsqu'un message est envoyé, le récepteur peut prouver que le message a été bien envoyé par l'expéditeur. De même, lorsqu'un message est reçu, l'expéditeur peut prouver que le message a bien été reçu par le receveur.
Audit	Cet objectif permet de contrôler le fonctionnement d'un système et de contrôler les mécanismes de sécurité et de conformité afin de détecter leurs défaillances et les corriger.

Tableau 3-1 : Les objectifs de sécurité de support

3.2.2 Les mesures de sécurité

Les mesures de sécurité représentent les différentes solutions de sécurité qui pourront être mises en place pour atteindre les objectifs de sécurité. L'ontologie de sécurité NRL-SO [53] classifie les mesures de sécurité en trois types: les protocoles, les mécanismes et les politiques de sécurité.

1. Les protocoles de sécurité sont définis comme une série d'étapes permettant de réaliser une tâche bien définie [54]. Ces protocoles peuvent être associés aux protocoles fonctionnels qu'ils supportent comme des protocoles de sécurité associés aux protocoles de routage (IPsec est associé à IP), de transport (SSL/TLS est associé à TCP), d'application (DNSsec associé à DNS).
2. Les mécanismes de sécurité représentent la mise en œuvre des protocoles. Nous pouvons trouver des mécanismes de sécurité réseaux (VPN), des mécanismes systèmes (Safehost), des mécanismes de services (Parefeu SOAP) [53].
3. Les politiques gouvernent les mécanismes et les protocoles en spécifiant les règles de sécurité à appliquer. Différents types de politiques de sécurité peuvent être définis:
   - ✓ Les politiques de sécurité métier sont définies par les responsables métier. Dans cette catégorie, on trouve par exemple les politiques spécifiant les droits d'accès à l'information.
   - ✓ Les politiques de sécurité applicative et architecturale sont définies par les architectes logiciels. Ces politiques sont utilisées dans la conception des applications. Par exemple, ces politiques intègrent la définition des rôles qui donnent des droits à l'invocation d'opérations dans une application.
   - ✓ Les politiques de sécurité opérationnelles sont définies par les administrateurs réseaux. Ces politiques sont utilisées dans la gestion de l'infrastructure technique comme par exemple la définition du nombre de tentatives de connexion sur un système informatique [55].

Afin de choisir les meilleures mesures de sécurité à mettre en place, il faut faire une veille sur les mesures de sécurité existantes et choisir parmi ces mesures celles qui sont les plus adéquates au contexte métier et technologique de l'entreprise. Par exemple, les mesures de sécurité suivantes pourront être mises en place pour assurer la confidentialité des données :

- ✓ Des politiques spécifiant les droits d'accès aux données.
- ✓ Des mécanismes de chiffrement ou des mécanismes d'authentification et d'autorisation (Mandatory Access Control 'MAC' [56], Role Based Access Control 'RBAC' [57])
- ✓ Des protocoles de chiffrement tels que les protocoles SSL/TLS pour les données en transit.

Assurer la confidentialité des données revient à choisir une ou plusieurs de ces instances, à les combiner en fonction des contextes métier et technologique de l'entreprise.

Dans ce qui suit, nous abordons les stratégies de développement de systèmes sécurisés. En particulier, nous mettons en évidence la notion de patron de sécurité. Un patron de sécurité représente des mesures de sécurité types à déployer pour répondre à un problème connu.

3.2.3 Les stratégies dans le développement de systèmes sécurisés

Sécuriser un système revient à prendre en compte la sécurité dès les premières phases de la conception. Le développement de systèmes sécurisés, qu'il s'agisse de systèmes d'informations, d'architectures à base de services ou d'applications monolithiques, peut être effectué en utilisant des méthodologies d'analyse des risques, des modèles de conception et des patrons de sécurité. Dans ce qui suit, nous présentons le concept des patrons de sécurité. Les méthodologies d'analyse des risques et les modèles de conception seront détaillés dans les chapitres 4 et 5.

De manière générale, un patron constitue une base de savoir et de savoir-faire pour résoudre un problème récurrent dans un domaine particulier. La spécification de ces connaissances réutilisables

1) permet d'identifier le problème à résoudre par capitalisation et organisation de connaissances d'expériences
2) propose une solution possible, correcte, générale et consensuelle pour y répondre
3) offre les moyens d'adapter cette solution au contexte spécifique [58].

A partir des spécifications d'un problème récurrent, les patrons permettent d'obtenir les informations et connaissances organisationnelles pour y faire face. Les patrons de sécurité aident les architectes et les développeurs à partager des connaissances sur la sécurité, à définir un nouveau paradigme de conception ou un style d'architecture, à identifier les risques qui ont été traditionnellement identifiés par prototypage ou par expérience intégrant les visions métier et technologiques de la sécurité [59]. Dans le Tableau 3-2, nous proposons des exemples de patrons de sécurité.

Nom du patron	Standards and Technologies	Description
Communication sécurisée	HTTPS; SSL (TLS), IPsec	Ce patron de sécurité décrit l'utilisation d'une couche de transport sécurisée dans le cadre de la communication client-serveur ou serveur-serveur.
Log d'évènements de sécurité	JMX; Java API for Logging	Ce patron de sécurité décrit la traçabilité des évènements de sécurité pour des raisons d'audit.
Passerelle de sécurité SOA	Intégration de services de sécurité au sein d'un ESB	Pour centraliser la sécurité, des services de sécurité peuvent être connectés à un ESB. En utilisant ce patron, nous pouvons simplifier la propagation des identités, renforcer les politiques, mettre en place des mécanismes d'audit, etc.

Tableau 3-2 : Exemples patterns de sécurité

Après avoir abordé les concepts génériques liés à la sécurité (objectifs, mesures et stratégies de développement des systèmes sécurisés), nous abordons dans ce qui suit la sécurité dans les SOA, les défis et les solutions proposées dans la littérature.

3.3 La sécurité dans les SOA

Les architectures orientées services permettent la composition de processus métier et d'applications à partir de services distribués sur le réseau. Dans cette partie, nous abordons les défis liés à la sécurité, les solutions proposées et les modèles de sécurité de référence.

3.3.1 Les défis de la sécurité dans une architecture orientée services

La réutilisation des fonctionnalités des applications n'est pas une tâche simple. D'une part, ces fonctionnalités n'ont pas été conçues pour être réutilisables et d'autre part, les technologies utilisées par les différentes applications ne sont pas obligatoirement compatibles. L'exploitation des fonctionnalités d'une application en tant que service favorise la réutilisation, améliore l'agilité et apporte un meilleur alignement métier et applicatif. Toutefois, cet apport s'accompagne de nouveaux défis au niveau de la sécurité puisque l'environnement est distribué et dynamique.

3.3.1.1 Des nouvelles perspectives de la sécurité

Pour prendre en compte la sécurité dans une architecture orientée services, nous distinguons deux perspectives :

A- La perspective verticale :

La perspective verticale de la sécurité prend en compte les éléments métier, organisationnels et technologiques dans un périmètre connu et identifie les mesures de sécurité à mettre en œuvre en fonction des besoins. Par exemple :

1. Au niveau *métier*, la sécurité implique la mise en place de :
   - ✓ Un cadre de gestion des politiques permettant de repenser la stratégie globale de la sécurité et d'attribuer des priorités dans la mise en œuvre des solutions de sécurité.
   - ✓ Une stratégie de veille sur les mesures de sécurité dans la mise en œuvre de l'architecture orientée services, cette stratégie recouvre la formation du personnel gérant les systèmes.
   - ✓ Une politique de classification et de contrôle d'accès aux données confidentielles.
2. Au niveau *technologique*, la sécurité implique la mise en place :
   - ✓ Des serveurs et des équipements d'hébergement redondants ainsi que des solutions de sauvegarde.
   - ✓ Des architectures réseau sécurisées, des systèmes de filtrages et de détection d'intrusion, etc.

B- La perspective horizontale

La perspective horizontale de la sécurité revient à prendre en compte le périmètre (éventuellement étendu) de l'entreprise afin d'assurer une sécurité globale et cohérente. L'ouverture du système d'information de l'entreprise et le développement des partenariats devront être pris en compte pour garantir la cohérence globale. En effet, la définition d'une

vision commune de la sécurité entre ces différents partenaires est le premier pas pour assurer une protection globale et une sécurité de bout en bout entre les consommateurs et les fournisseurs de services.

3.3.1.2 Des approches de sécurité inadéquates

Nous utilisons l'étude effectuée dans [60] pour synthétiser dans le Tableau 3-3 les défis dans l'application des approches de sécurité traditionnelles dans la sécurisation des SOA.

Objectif de sécurité	**Approche traditionnelle de sécurité**	**Défi dans le contexte de SOA**
Authentification	Le mécanisme de l'authentification fait partie de la responsabilité de l'application	✓ L'invocation du service peut se faire à partir de sources différentes et de milieux hétérogènes. ✓ Dans le cas d'une composition, peut-on faire confiance à l'authentification réalisée par un autre service ? Comment propager l'authentification au niveau de la composition ?
Contrôle d'accès	Le mécanisme du contrôle d'accès fait partie de la responsabilité de l'application (Modèles RBAC, ACL, etc.)	✓ Comment faire une composition de services quand les services impliqués dans la composition utilisent des mécanismes de contrôle d'accès différents ? ✓ L'intégration du mécanisme de contrôle d'accès dans le service lui-même peut nuire à sa réutilisation : le service est alors dépendant d'un contexte et ne peut plus être réutilisé dans un contexte où un autre mécanisme est requis.
Confidentialité / Intégrité / Non répudiation	Les applications se basent sur le chiffrement SSL/TLS pour assurer ces objectifs de sécurité sur la transmission des messages.	SSL/TLS assure une confidentialité point à point dans le contexte SOA, il existe plusieurs intermédiaires. Par conséquent, le risque d'atteinte à la confidentialité, à l'intégrité et à la non-répudiation s'accentue au niveau des services intermédiaires
Protection des données personnelles des utilisateurs	✓ Mécanismes de contrôle d'accès sur l'application ✓ Sécurisation de l'infrastructure technique pour des accès non autorisés sur les systèmes.	Les informations privées sont partagées et gérées par plusieurs acteurs et partenaires. Par conséquent, il existe un grand risque sur la divulgation des données personnelles.

Tableau 3-3 : Des approches de sécurité inadéquates

3.3.1.3 Un contexte et des services dynamiques

Pour maintenir une architecture orientée services sécurisée, les mesures de sécurité devront être adaptées aux changements du contexte métier et technologique. A la différence des architectures traditionnelles statiques où le déploiement des mesures de sécurité pouvait être réalisé et maintenu facilement, les services peuvent être déployés de manière dynamique et utilisés dans différents contextes. En effet, de nouveaux partenaires peuvent entrer en jeu, de nouveaux services pourront être dynamiquement ajoutés ou remplacés, des changements dans les infrastructures techniques des fournisseurs de services peuvent avoir lieu, etc.

3.3.1.4 La gestion des préférences utilisateurs

Les préférences de l'utilisateur correspondent à l'attente de l'utilisateur en matière de gestion de la sécurité pour ses données personnelles, pour l'invocation et l'utilisation du service. À titre d'exemple, l'utilisateur pourra spécifier ses exigences de sécurité pour la sauvegarde et la manipulation de ses données personnelles (nom, adresse, numéro de carte de crédit, etc.) au niveau du service. Il pourra demander une garantie sur l'application des politiques de sécurité associés au service (niveau de chiffrement, utilisation des clés, etc.).

La gestion des préférences utilisateurs est un vrai défi au niveau de la sécurité. En effet, il va falloir transmettre les préférences de l'utilisateur, confronter ces préférences aux politiques du service, valider les préférences de l'utilisateur et les capacités de protection du service, vérifier la bonne application des préférences lors de l'utilisation du service. Enfin, la prise en compte des préférences utilisateurs devra se baser sur des standards pour des raisons d'interopérabilité.

3.3.2 Les contributions dans la sécurisation des SOA

Dans ce qui suit, nous aborderons plusieurs travaux qui répondent aux défis que nous avons identifiés. Les modèles de références permettent de cadrer la portée de la sécurité dans une SOA et de prendre en compte les aspects métier et technologiques. Les nouvelles approches de la sécurité permettent de trouver une solution aux limites des mesures de sécurité existantes. Enfin, nous présenterons les standards pour la gestion des préférences des utilisateurs.

3.3.2.1 Les modèles de référence de la sécurité

L'architecture de référence de l'OASIS

Dans son architecture de référence des architectures orientées services [22], l'OASIS (Organization for the Advancement of Structured Information Standards) a élaboré un modèle de sécurité mettant en évidence une dimension métier dans la sécurisation des architectures orientées services. La sécurité est analysée en termes de permissions, d'obligations et de rôles associés aux acteurs pour l'utilisation des services et de mesures à mettre en place pour assurer la

sécurité. Ce modèle intègre également les éléments nécessaires à l'établissement d'un réseau de confiance entre les différents partenaires.

Telle qu'elle est modélisée (voir la Figure 3-1) la confiance est la relation entre un participant et un ensemble d'actions et d'événements. Ce modèle de confiance permet de désigner parmi les différents acteurs, ceux qui pourront mener des actions ou déclencher un événement. De plus, l'OASIS définit différents niveaux de confiance en distinguant les rôles et les attributs des participants leur permettant de réaliser différentes actions.

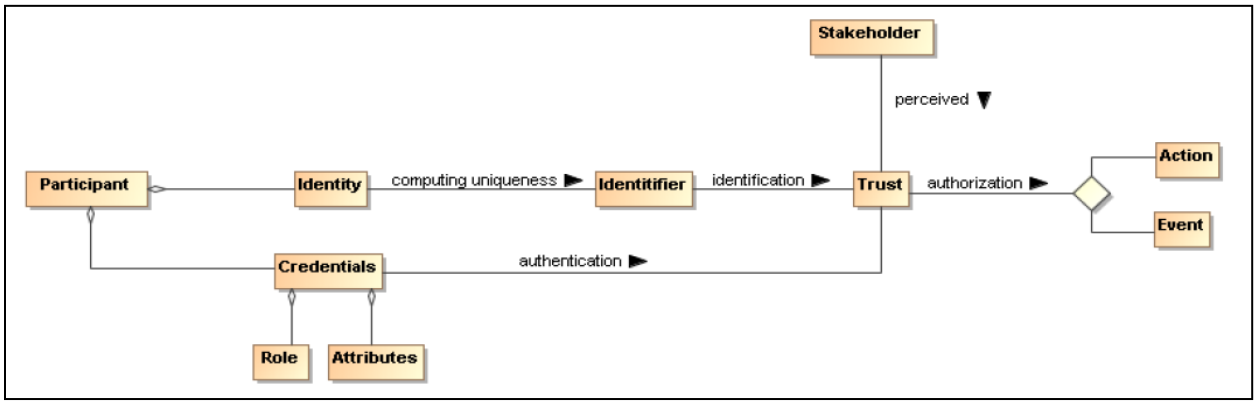


Figure 3-1 : Modèle de confiance - Architecture de référence OASIS [22] p.92

L'OASIS définit également la notion de domaine de confiance. Cette notion représente un espace d'action où les participants partagent la même relation de confiance. Un participant souhaitant effectuer une action dans un domaine de confiance devra se conformer aux politiques de sécurité mises en place. Il doit s'authentifier auprès du domaine afin d'exécuter les actions. La Figure 3-2 montre comment les politiques de sécurité sont intégrées pour assurer une sécurité cohérente.

Les politiques de sécurité définissent les consignes concernant la mise en œuvre des mécanismes de sécurité. Une entité de décision (Decision Point) permet de récupérer la politique à mettre en œuvre. L'entité d'application (Enforcement Point) permet de mettre en œuvre la politique afin d'autoriser les participants à exécuter les actions demandées. Enfin, une entité d'audit (Audit Point) a été ajoutée pour collecter les logs des interactions à des fins d'audit.

En mettant en évidence la confiance et l'application des politiques, ce document donne une nouvelle perspective métier de la portée de la sécurité. Cette perspective peut être complétée par le modèle de référence d'IBM.

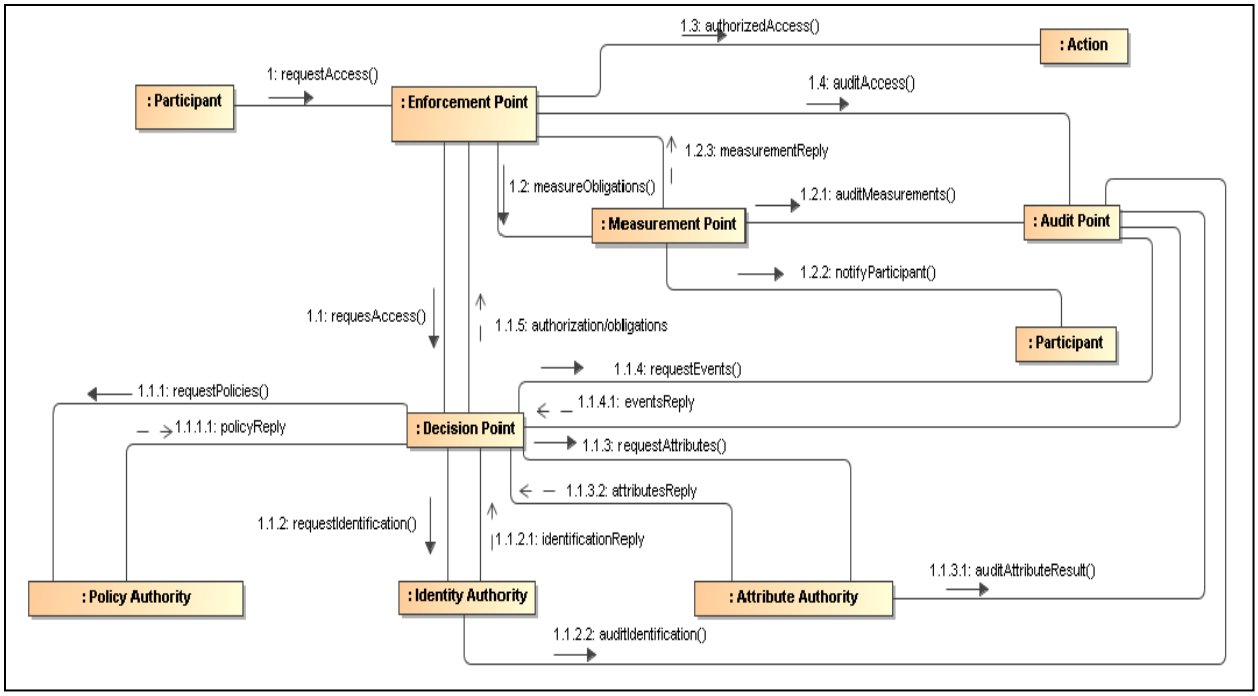


Figure 3-2 : Sécurité à la base des politiques - Architecture de référence OASIS [22] p.92

Le modèle de référence d'IBM

Le modèle de référence d'IBM [55] fournit un aperçu des composantes de sécurité d'une architecture SOA. Comme le montre la Figure 3-3, IBM différencie les services de sécurité métier et technologique.

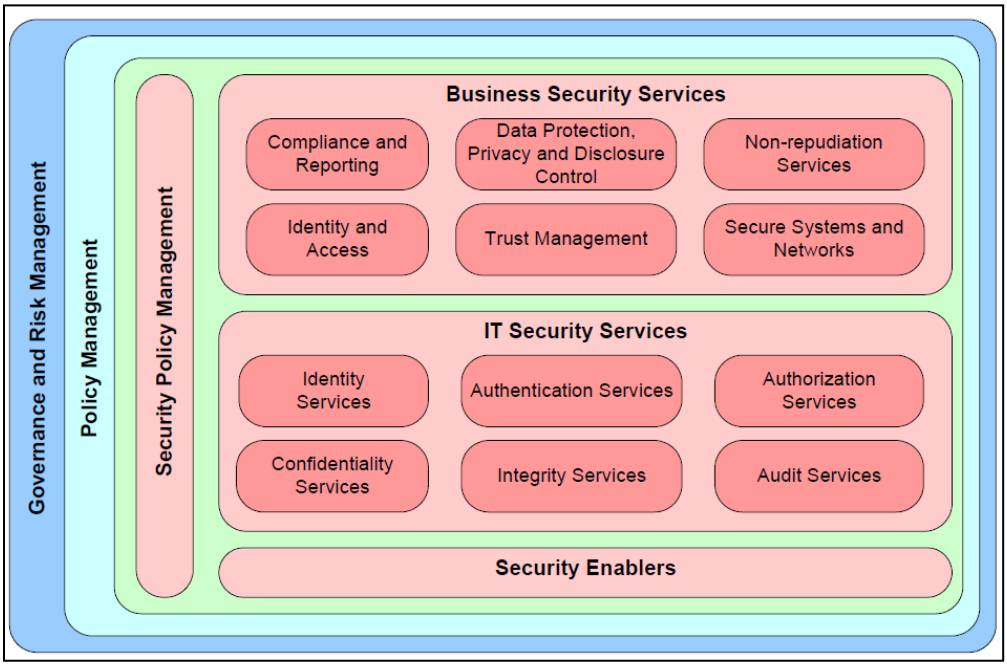


Figure 3-3 : IBM - Modèle de référence de sécurité SOA [55] p.57

Les services de sécurité métier sont utilisés dans la définition des politiques de sécurité et permettent de :

1. vérifier les lois et les obligations légales dans la mise en œuvre de la sécurité
2. identifier les ressources et les données nécessitant une protection ainsi que les mesures de sécurité nécessaires sur ces ressources
3. assurer la gestion de la non-répudiation des communications en offrant la preuve de l'origine, de la soumission, du transport et de la livraison des données
4. assurer la gestion de l'identité et du contrôle d'accès
5. assurer la gestion de la confiance entre les entités (entreprises, personnes, services)
6. assurer la définition des politiques de sécurité dans la gestion des systèmes et des réseaux (par exemple : mise en place de politiques pour la détection d'intrusion).

Les services de sécurité liés au système d'information couvrent à la fois les services de sécurité de base et les services de sécurité support (identification, authentification, autorisation, confidentialité, intégrité et audit). À titre d'exemple, le service d'authentification met en œuvre les mécanismes d'authentification par mot de passe. Le protocole Kerberos, le standard SAML peuvent être utilisés pour mettre en œuvre ce service.

Ce modèle de référence sépare clairement les niveaux métier et technologique dans une SOA :

- ✓ A niveau métier : il convient de prendre en compte la conformité des lois et des obligations légales, la classification des données et la gestion du contrôle d'accès aux ressources.
- ✓ Au niveau technologique : Il convient de prendre en compte la cohérence dans l'application de la sécurité dans une composition de services (au niveau individuel de chaque service et au niveau de la composition)

Nous remarquons la complémentarité de ces documents de référence pour couvrir les différentes facettes de la sécurité dans une SOA, ce qui oblige à repenser le cadre de la sécurité et les aspects à prendre en compte dans la sécurisation des services. Dans ce qui suit, nous aborderons de nouvelles approches permettant de combler les manques des solutions de sécurité traditionnelles lors de la mise en place d'une SOA.

3.3.2.3 Des nouvelles approches dans la sécurisation des SOA

Afin d'améliorer la sécurité dans un environnement de services, différentes solutions de sécurité ont été proposées :

1- La sécurité en tant que service

Cette approche consiste à intégrer les fonctions de la sécurité dans un service dédié. Il s'agit par exemple de services d'authentification, d'autorisation, de chiffrement/déchiffrement des messages, de signatures /vérification de signatures, de log des messages [60]. Ces services peuvent être invoqués par les applications pour assurer les fonctions de sécurité nécessaires. De

plus, ils peuvent être intégrés dans les bus de services (ESB) afin que leur invocation soit transparente pour les applications. Cette approche permet d'alléger les applications en déléguant les fonctions de sécurité au middleware et donc de simplifier l'accès à la sécurité dans une composition de service.

2- Intégration de la sécurité dans les messages échangés

Cette approche consiste à intégrer des paramètres de sécurité dans l'entête des messages pour désigner le chiffrement ou la signature de parties spécifiques des messages. Ceci leur permet d'être utilisables uniquement par les utilisateurs ou les services concernés même en passant par plusieurs intermédiaires [60]. Ceci permet d'avoir une sécurisation 'de bout en bout', c'est-à-dire que les nœuds intermédiaires traversés lors des échanges entre le consommateur de service et le fournisseur de service ne pourront pas accéder au contenu de ces échanges. Dans le monde des services web, les standards WS-Security présentés dans la 'sécurité des services web', permettent d'apporter l'interopérabilité nécessaire à cette fonctionnalité.

3- *La sécurité axée sur les politiques*

Cette approche consiste à formaliser les exigences et les mécanismes de sécurité sous forme de politiques de sécurité pour assurer une cohérence globale sur l'application des décisions de sécurité au sein de l'entreprise et entre les différents partenaires. Les politiques de sécurité permettent de gouverner les mécanismes et les exigences de sécurité. Dans le monde des services web, le standard WS-SecurityPolicy répond à ce besoin.

3.3.2.4 La gestion des préférences des utilisateurs

Pour permettre la gestion des préférences des utilisateurs, de nombreux travaux ont été proposés. Le W3C a élaboré le langage P3P [61] pour permettre à un service de décrire la manière dont il traite les données personnelles d'un utilisateur. L'utilisateur peux ensuite confronter ses préférences à la politique du service avant de l'utiliser. P3P gère l'information à travers des politiques spécifiant les informations que le service stocke, le temps de stockage, l'utilisation de l'information et le contrôle d'accès à cette information.

En se basant sur P3P, d'autres spécifications ont été créées pour la définition des préférences utilisateurs. APPEL [62] (A P3P Preference Exchange Language) est une spécification du W3C permettant à un utilisateur de spécifier la manière dont ses données personnelles doivent être traitées. Cette spécification définit des algorithmes à utiliser pour confronter les préférences des utilisateurs et les politiques des services. Cependant, APPEL permet de spécifier uniquement les éléments valides dans une politique P3P et ne permet pas de spécifier les éléments non valides [63]. Pour combler ce manque, R. Agrawal a proposé le langage Xpref [64] qui permet de spécifier les préférences des utilisateurs selon deux catégories : les éléments valides et les éléments non valides dans une politique P3P. Enfin, pour automatiser la phase de confrontation des préférences utilisateurs et des politiques des services, le W3C a élaboré le langage

SWAPPEL (Semantic Web APPEL) permettant d'intégrer une gestion sémantique des règles [65] et d'améliorer le matching des préférences de l'utilisateur et des politiques des services.

Une autre problématique est la prise en compte des conflits entre les préférences utilisateurs et les politiques des services. En effet, dans le cas d'un conflit une phase de négociation devra être établie pour trouver un compromis. Pour répondre à cette problématique, S. Trabelsi a proposé d'enrichir la description des politiques du service et des requêtes des utilisateurs. La tâche de négociation sémantique est déléguée à un service qui établit une phase de négociation en cas de conflit [65]. De même, Preibusch a proposé une plateforme de négociation entre les politiques du service P3P et les préférences de l'utilisateur APPEL et Xpref [67].

Comme nous l'avons indiqué dans le chapitre 2, l'instance la plus connue des architectures orientées services est basée sur les services web. Nous abordons dans ce qui suit les travaux portant sur la sécurité des services web et les standards qui ont été élaborés.

3.3.3 La sécurité dans les services Web

3.3.3.1 Les standards de la sécurité des services Web

De nombreux standards et recommandations ont été élaborés dans le domaine de la sécurité des services web. IBM et Microsoft ont établi les documents décrivant les stratégies techniques et la feuille de route de l'intégration de la sécurité dans une architecture à base de services web [68]. Les standards développés portent sur l'établissement d'un réseau de confiance, la définition des politiques de sécurité, l'application du contrôle d'accès, etc. La Figure 3-4 illustre ces standards que nous présentons brièvement dans ce qui suit.

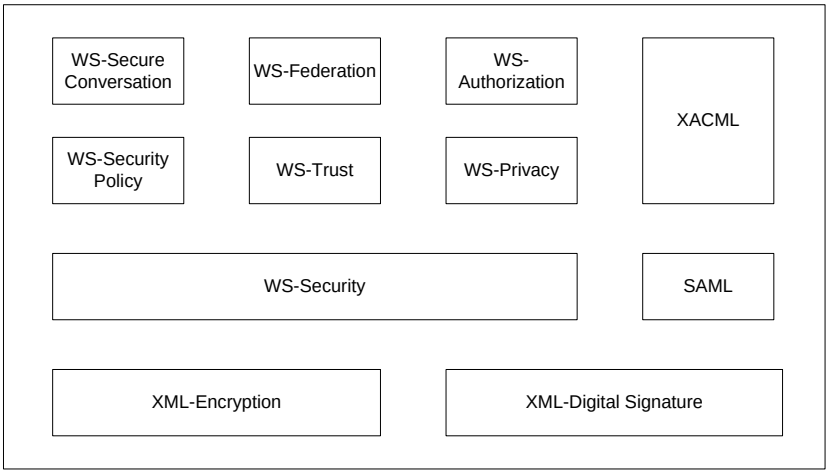


Figure 3-4 : Les standards de sécurité des services Web

L'OASIS propose une extension de la structure des messages SOAP pour améliorer la sécurité des messages transmis entre les services web. Le standard WS-Security est à la base de la sécurisation de ces interactions [69]. Ce standard définit la façon de chiffrer ou de signer les

messages ou une partie des messages en intégrant (dans l'entête du message) les informations nécessaires au chiffrement et à la signature. Il s'agit en particulier des clés et algorithmes de chiffrement. Pour cela, WS-Security s'appuie sur d'autres standards: XML-Digital Signature [70] et XML-Encryption [70] pour la signature et le chiffrement des documents XML. Ces deux standards spécifient le processus de chiffrement et de signature des données et la représentation de ces données en XML. Outre la confidentialité, WS-Security peut assurer l'authentification en intégrant dans l'entête des messages des mécanismes d'authentification (des certificats X509, des tickets Kerbero, etc.). Toutefois, WS-Security se limite à l'intégration de jetons de sécurité et ne permet pas d'offrir des fonctionnalités d'authentification avancées (validation de l'authentification, authentification dans des milieux hétérogènes) [71].

Le standard SAML (Security Assertion Markup Language) V2.0 [72] a été conçu par l'OASIS comme un cadre pour l'échange et la propagation des informations de sécurité entre partenaires de confiance. Les informations de sécurité sont exprimées comme des assertions sur des entités (personne ou service) ayant une identité dans le domaine de sécurité. Les assertions peuvent transmettre des informations sur les attributs des entités, sur les authentifications déjà effectuées ou sur les décisions d'autorisation en rapport avec des ressources spécifiques.

XACML [73] définit un langage pour la formulation des politiques de contrôle d'accès. Il spécifie les fonctionnalités nécessaires pour le traitement de ces politiques ainsi qu'un modèle de flux de données entre les composants fonctionnels de l'infrastructure. XACML fournit les mécanismes d'autorisation qui sont transférés par SAML. En d'autres termes XACML complémente SAML pour offrir le service d'autorisation.

Établissement d'un réseau de confiance

WS-Trust [74] se base sur les mécanismes de sécurité de WS-Security et définit un modèle pour l'établissement et le maintien des relations de confiance au travers des domaines de sécurité. Dans les architectures orientées services, la confiance suppose généralement l'émission, l'échange et la validation des jetons de sécurité pour contrôler l'accès à des services spécifiques [71]. Le standard WS-Federation étend le standard WS-trust pour la fédération des identités à travers les frontières organisationnelles. Ce standard permet la création d'un domaine virtuel de sécurité et le partage d'identité à travers le domaine créé où les systèmes d'authentification et d'autorisation sont distribués [75]. WS-Trust et WS-Federation fournissent ainsi un modèle pour la création d'un réseau de confiance entre différents partenaires.

Confidentialité des communications

Le standard WS-SecureConversation [76] permet d'établir des communications sécurisées. Il utilise des clés publiques pour l'échange des clés de session et spécifie les mécanismes pour l'établissement et le partage des contextes de sécurité. Ce protocole associé au niveau applicatif est l'équivalent du protocole SSL au niveau du transport.

Les politiques et les politiques de sécurité

Le standard WS-Policy [42] a été conçu comme un modèle générique permettant d'exprimer différents types de politiques. Citons à titre d'exemple, les politiques d'utilisation des ressources, de qualité de service, etc. WS-Policy utilise la notion d'assertion qui spécifie les besoins ou les capacités d'une entité (policy subject). La sémantique des assertions est spécifique à chaque domaine (sécurité, transaction, etc.). L'approche retenue par WS-Policy est de définir les assertions qui correspondent à des domaines particuliers dans des standards séparés tels que WS-SecurityPolicy [45] définissant les assertions de sécurité ou WS-PolicyAssertion [43]. Dans la Figure 3-5, nous illustrons une assertion d'intégrité de WS-SecurityPolicy spécifiant la partie du message à signer. Cette assertion peut être satisfaite en utilisant les mécanismes de sécurité des messages SOAP (WS-Security) ou en utilisant d'autres mécanismes hors de la portée des messages SOAP (SSL).

```
1  <wsp:Policy>
2   ...
3   <sp:SignedParts>
4    <sp:Body/>
5    <sp:Header Namespace="http://schemas.xmlsoap.org/ws/2004/08/addressing"/>
6   </sp:SignedParts>
7   ...
8  </wsp:Policy>
```

Ligne 3 : corps du message à signer
Ligne 4 : entête du message à signer

Figure 3-5 : Exemple d'une assertion d'intégrité

Le standard WS-PolicyAttachment [44] spécifie les entités du WSDL ou de l'annuaire UDDI auxquelles les politiques s'appliquent. Enfin, les assertions peuvent être combinées pour former des politiques de plus haut niveau.

3.3.3.2 Bilan sur la sécurité des services web

D'après nos recherches bibliographiques, nous avons pu constater que les aspects technologiques de la sécurité dans les services ont atteint un bon niveau de maturité. De nombreux besoins ont été abordés par des standards ou par des projets sur la sécurisation des services web. Toutefois, un défi majeur est la gestion de la sécurité dans un environnement distribué et dynamique. Comme nous l'avons évoqué dans la partie 'Portée de la sécurité', la sécurité revient à prendre en compte les niveaux métier, organisationnel et technologique non seulement au sein de l'entreprise mais aussi entre l'entreprise et ses partenaires.

Pour cela, nous présentons une liste non exhaustive des besoins de sécurité à prendre en compte :

- ✓ Se baser sur des méthodologies de conception de politique de sécurité et des techniques de modélisation des menaces pour la protection contre les attaques. Les techniques de

modélisation de menaces permettent d'identifier les failles de sécurité pour y remédier. Nous aborderons ces techniques en détail dans le chapitre suivant.

- ✓ Optimiser la disponibilité des services web, en empêchant les attaques par déni de services (DoS). Il faut pouvoir détecter ces attaques, continuer à fonctionner en présence de ces attaques et pouvoir reprendre les opérations après les attaques. Des mesures de redondance de données ou de services peuvent être mises en place.
- ✓ Ne pas négliger la gestion des traces (logs) dans les transactions des services web pour assurer la non-répudiation des services.
- ✓ Sécuriser les annuaires de services pour pouvoir vérifier l'authenticité des entrées de l'annuaire et la pertinence des informations concernant les services publiés. Des mécanismes de signatures peuvent être mis en place pour assurer cet objectif.
- ✓ Sécuriser l'environnement d'hébergement des services web. En effet, les services web ne sont pas isolés et la sécurité des éléments de l'infrastructure (serveurs d'hébergement, systèmes d'exploitation, équipements réseau, etc.) doit être assurée.

Par conséquent, une approche de 'gestion de la sécurité' s'avère primordiale pour optimiser la sécurité. Dans ce qui suit, nous aborderons les concepts et les standards liés à la gestion de la sécurité.

3.4 La gestion de la sécurité

La gestion de la sécurité est un processus permettant d'assurer la sécurité en intégrant les aspects organisationnels et technologiques. Ce processus permet d'identifier les biens à protéger et de développer des stratégies de protection contre les menaces éventuelles. L'objectif principal de la gestion de la sécurité est de cadrer les besoins de sécurité et de définir une stratégie globale afin d'assurer le niveau de sécurité requis sur l'information et les systèmes d'informations de l'entreprise. Dans le cadre de la gestion de la sécurité d'un système d'information, l'agence nationale de la sécurité des systèmes d'information de France (ANSSI) a défini dans son référentiel général de la sécurité [4], six principes à la base de la gestion de la sécurité :

1. *Adapter une démarche globale*

L'objectif est la cohérence d'ensemble de la démarche de sécurisation des systèmes d'information. Il convient à ce titre de n'oublier aucun élément pertinent, pour éviter toute faille qui réduirait la sécurité globale du système d'information [4].

2. *Adapter la sécurité du système d'information selon les enjeux*

Il est recommandé que la sécurité du système d'information soit adaptée aux enjeux du système et aux besoins de sécurité, afin d'y consacrer les moyens financiers et humains juste nécessaires mais suffisants.

3. Gérer les risques

Il est obligatoire de suivre une démarche qui consiste à :

1) Identifier l'ensemble des risques pesant sur le système
2) Fixer les objectifs de sécurité, pour répondre de manière proportionnée aux besoins de protection du système et des informations face aux risques identifiés
3) En déduire les fonctions de sécurité et leur niveau de mise en œuvre pour atteindre ces objectifs.

4. Élaborer une politique de Sécurité du Système d'Information (SSI)

Élaborer une stratégie globale de sécurité permet de définir le cadre d'utilisation du système d'information. Les politiques définissent entre autres les rôles et les responsabilités des différents acteurs, les règles d'utilisation des systèmes et de l'information, les règles permettant de contrôler l'accès sur l'information, les règles d'utilisation des données privées, les règles d'audit, de sauvegarde, etc.

5. Utiliser les produits et prestataires labellisés pour leur sécurité

La certification de produits ou prestataires permet d'attester de la confiance que l'on peut accorder à des produits de sécurité et à la compétence des professionnels en matière de SSI.

6. Viser une amélioration continue

Il est recommandé de chercher une amélioration constante de la SSI, par exemple en mettant en place un « système de management de la sécurité de l'information » (SMSI) pour planifier les actions de sécurisation et les mettre en œuvre puis les vérifier et améliorer la SSI.

3.4.1 Les standards dans la gestion de la sécurité

Le standard ITSEC de l'Union Européenne

Proposé en 1991, le standard Information Technology Security Evaluation Criteria de l'Union Européenne a été développé pour réaliser une synthèse entre les travaux de certification sécurité des différents états partenaires [77]. Les besoins en termes de « cible de sécurité » (i.e. le niveau de certification visé) sont décrits selon 8 groupes de critères : identification et authentification, contrôle d'accès, imputabilité, réutilisation d'objets, fidélité, continuité de service, échange de données (incluant l'authentification, le contrôle d'accès, la confidentialité des données, l'intégrité des données et la non-répudiation). Ces critères sont également regroupés en neuf familles selon le cycle de vie du projet permettant d'aboutir à un système d'information sécurisé : étude des besoins, conception de l'architecture, conception détaillée, mise en œuvre, configuration et contrôle, langages de programmation et compilateurs, sécurité pour les développeurs, documentation « opérationnelle », environnement opérationnel. Toutefois, si l'ITSEC vise une analyse globale du système support du système d'information, ce standard ne prend pas réellement en compte les aspects organisationnels. L'accent est placé sur les aspects

conception, développement et contraintes de mise en œuvre de ressources informatique et non sur l'adaptation de l'organisation pour répondre aux contraintes de la politique de sécurité. De même, ce standard « oublie » les composants « sécurité » d'un réseau ou d'un système informatique [78].

Les 'Common Criteria'

Afin de certifier de manière unifiée dans un cadre international le niveau de sécurité atteint par les systèmes des partenaires dans un environnement distribué, le standard 'Common Criteria' (CC) [79] définit à la fois des critères et une méthode d'évaluation.

Ce standard repose sur deux concepts principaux :

- Le profil de protection (PP) représente l'ensemble des besoins et d'objectifs de sécurité pour une catégorie de produits ou systèmes.
- La cible de sécurité (Security Target : ST) décrit les objectifs de sécurité et les besoins associés à une 'cible d'évaluation'.

L'originalité de ce standard est qu'il repose sur un modèle de gestion des risques intégrant différents concepts (Figure 3-6). Le risque porte sur les biens (assets) et est identifié en croisant les menaces et les vulnérabilités. Un risque est réduit en mettant en place de contremesures permettant de réduire les vulnérabilités et donc leur possible exploitation par des menaces. Ce modèle met également en relief la responsabilité des propriétaires dans la définition de la valeur des biens et des contremesures, ce qui permet d'intégrer les contraintes organisationnelles dans la définition des objectifs de sécurité. La méthode développée pour l'évaluation permet de contrôler la conformité de la cible (ST) vis-à-vis d'un ou de plusieurs profils de sécurité (PP).

Toutefois, ce standard vise majoritairement la certification des composants et n'intègre pas la dimension organisationnelle.

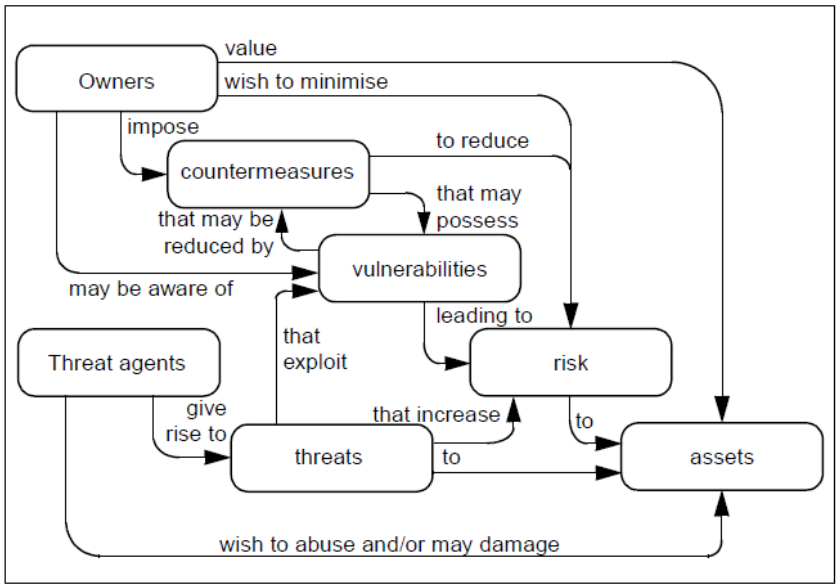


Figure 3-6 : Modèle de concepts de sécurité [79] p. 13

Les standards de l'ISO

L'ISO et l'IEC (International Electrotechnical Commission) ont publié les standards ISO27001/ISO27002 (anciennement ISO 17799) [80]. Ces standards établissent les lignes directrices et les principes pour préparer, implémenter, maintenir et améliorer la gestion de la sécurité. A la différence des standards précédents, la sécurité est prise en compte aux niveaux organisationnel et technologique. Le Tableau 3-4 liste les fonctions offertes par ces standards:

ISO 27001	Gestion de la responsabilité Audit Interne Amélioration de l'ISMS (Information Security Management System)
ISO 27002	Elaboration d'une politique de sécurité Organisation de la sécurité des informations Gestion des biens et des actifs Sécurité physique et environnementale Communications et la gestion des opérations Contrôle d'accès Systèmes d'acquisition de l'information Développement et maintenance Gestion des incidents de sécurité des informations Gestion de la continuité Conformité

Tableau 3-4 : Fonctions ISO 27001 / ISO 27002

Les standards ISO27001/ISO27002 font partie d'une série de standards publiés par l'ISO sur la gestion de sécurité:

- ✓ Le standard ISO27003 fournit une aide et des conseils dans la mise en œuvre d'un système de gestion de la sécurité. Il s'agit notamment de mettre l'accent sur la méthode PDCA (Plan, Do, Check, ACT) pour l'établissement, la mise en œuvre, le contrôle et l'amélioration du système de gestion.
- ✓ Le standard ISO27004 fournit les directives pour l'évaluation d'un système de gestion de sécurité.
- ✓ Le standard ISO27005 fournit les directives pour la gestion des risques dans une entreprise.
- ✓ Le standard ISO27006 fournit les directives pour l'accréditation des organismes qui offrent la certification ISO.
- ✓

Le cadre FISMA du NIST

Dans une même perspective, le NIST (National Institute of Standards and Technology) a développé des standards pour l'implémentation de la gestion de la sécurité (FISMA : Federal Information Security Management Act) [81].

FISMA est un cadre de gestion qui fait référence à de nombreux documents élaborés par le NIST dans la sécurisation des systèmes d'information. L'objectif du cadre FISMA est la mise en œuvre d'un plan de gestion de la sécurité. Le Tableau 3-5 résume la portée de ce cadre :

Standards pour la catégorisation de l'information et des systèmes d'information
Standards des exigences de sécurité minimales pour l'information et les systèmes d'information
Directives pour la sélection des contrôles de sécurité appropriés aux systèmes d'information
Guide pour l'évaluation des contrôles de sécurité dans les systèmes d'information et de la détermination de l'efficacité du contrôle de sécurité.
Directives pour la certification et l'accréditation des systèmes d'information

Tableau 3-5 : Portée du cadre FISMA

Parmi les documents auxquels FISMA fait référence, nous listons ci-dessous ceux qui sont au cœur de ce cadre :

- ✓ NIST 800-30 : Le guide de gestion des risques pour les systèmes d'informations.
- ✓ NIST 800-53 : Le guide des contrôles de sécurité dans les systèmes d'information.
- ✓ FIPS 199 (Federal Information Processing Standard 199): Standards pour la catégorisation de l'information et des systèmes d'information.

Bien que le cadre FISMA s'oriente vers les systèmes en développement et que les standards de l'ISO sont plus destinés aux systèmes en production, nous avons remarqué que ces deux cadres sont assez similaires dans la gestion de la sécurité :

- ✓ Les deux cadres se basent sur un processus de développement : ISO recommande le processus PDCA (Plan – Do – Check – Act), FISMA propose un cycle de développement plus classique (Initialisation, développement, acquisition/implémentation, opération et maintenance)
- ✓ Les deux cadres mettent la gestion de risques au cœur de leur démarche.
- ✓ Les deux cadres soulignent l'importance du support des responsables métier dans l'étude, l'implémentation et le suivi de la gestion de la sécurité.

Les cadres (ITSec/ISO/FISMA) font référence dans le domaine de la gestion de la sécurité et permettent de sensibiliser les responsables métier et technique à la gestion de la sécurité des systèmes d'information. Toutefois, aucun de ces cadres n'a été conçu pour la gestion de la sécurité dans un environnement de services distribués et dynamiques où la collaboration entre les différents partenaires est l'un des principaux objectifs. En effet, ces cadres ne correspondent pas au contexte de collaboration inter-entreprises et ne définissent pas une plateforme d'intégration des exigences de sécurité dans les modèles des processus métier [82].

3.4.2 Les processus d'implémentation de la gestion de la sécurité

La gestion de la sécurité suppose de mettre en œuvre plusieurs études et processus de gestion des risques (qui seront présentés dans le chapitre 4), de gouvernance de sécurité et d'organisation

d'un plan de poursuite d'activité [83] pour prendre en compte les besoins de sécurité, sur l'ensemble du cycle de vie du système d'information et des projets associés.

La gouvernance de la sécurité

Le processus de gouvernance simplifie la gestion d'une stratégie de sécurité globale. Parmi les fonctions principales de la gouvernance, le processus de planification et la détermination des priorités dans l'utilisation des ressources de l'entreprise tient une place importante. Ce processus comprend l'établissement du budget, l'allocation des ressources, ainsi que le support des décisions prises dans le processus de gestion des risques. D'après la spécification 800-39 du NIST, le processus de gestion des risques inclut :

- ✓ L'alignement stratégique des décisions de gestion des risques avec la mission de l'entreprise et les objectifs organisationnels;
- ✓ La vérification de l'application du processus de gestion des risques et de l'attribution des ressources nécessaires à ce processus
- ✓ La vérification que l'exécution du processus de gestion des risques garantit les objectifs métier et organisationnels.

Le plan de poursuite d'activité

Le plan de poursuite d'activité a pour but de garantir la survie de l'entreprise, en préparant à l'avance la continuité des activités stratégiques. Plus précisément, le plan de poursuite d'activité intègre :

- ✓ Un plan de secours informatique qui garantit la reprise des systèmes désignés comme critique dans le temps minimum fixé.
- ✓ La reprise des données avec le minimum de perte [84].

Le processus de définition du plan de secours informatique suppose l'engagement de la direction de l'entreprise et utilise l'analyse de risques pour réaliser les activités suivantes :

- ✓ Analyse de l'impact de l'indisponibilité des activités sur les objectifs métier de l'entreprise.
- ✓ Choix des stratégies de recouvrement en fonction des contextes métier et technologiques de l'entreprise
- ✓ Mise en place d'un plan de rétablissement des activités et des stratégies de recouvrement suite à des scénarios de risques.
- ✓ Sensibilisations des acteurs afin qu'ils puissent agir dans des scénarios de risques.
- ✓ Vérification du plan de continuité d'activité.

3.4.3 La gestion de la sécurité dans une architecture orientée services

Dans la littérature, nous n'avons pas trouvé un cadre définissant spécifiquement les principes de la gestion de la sécurité d'une SOA. En effet, une architecture orientée services fait partie du système d'information d'une entreprise et est donc sécurisée en appliquant les principes

généraux de la sécurité des Systèmes d'informations. Toutefois, les architectures à base de services ont des caractéristiques différentes (ouverture, portée, réutilisation). Nous pensons qu'il serait judicieux d'utiliser les principes de la gestion de la sécurité et de les compléter pour les adapter aux SOA selon le cadre suivant :

1. Adaptation d'une démarche globale dans la sécurisation d'une SOA en ciblant les éléments essentiels d'une SOA. Nous écartons les éléments du système d'information qui ne correspondent pas au contexte spécifique d'une SOA (influence sur le personnel, risques d'origines naturelles ou produits et prestataires labellisés) et étendons la portée du système d'information de l'entreprise pour couvrir la collaboration entre différents acteurs.
2. Adaptation de la sécurité selon les enjeux en suivant une démarche de préparation et de sensibilisation à un projet SOA.
3. Élaboration d'une politique de sécurité globale, amélioration continue et supervision de la sécurité d'une SOA. Ceci implique la mise en place d'un processus de gestion des risques, d'une stratégie de poursuite d'activités et d'un plan de gouvernance de la sécurité.

3.5 Conclusion

Dans ce deuxième chapitre de l'état de l'art, nous avons présenté les concepts liés à la sécurité. Nous avons noté que la sécurité est fortement dépendante des besoins et objectifs métier et qu'il est crucial de bien identifier ces objectifs pour pouvoir mettre en place les mesures de sécurité les plus adaptées au contexte de l'entreprise.

Ensuite, nous avons abordé les défis de la sécurité dans les SOA ainsi que les différentes contributions au niveau des standards et des modèles et de la sécurisation des services web. Nous avons noté la particularité de cette architecture concernant l'étendue du périmètre et la nécessité d'aborder les problématiques de la sécurité en suivant une approche de gestion de la sécurité. Cette dernière consiste à définir des processus en tenant compte des aspects métier, organisationnels et technologique de la sécurité.
En traitant les concepts de la gestion de la sécurité, nous avons trouvé qu'il est crucial d'élaborer une stratégie de sécurité globale du système. Par conséquent, nous nous sommes référés aux différents standards permettant la définition de cette stratégie (ITSec, standards de l'ISO et standards du NIST). Toutefois, aucun de ces standards ne répond aux exigences liées à la sécurité des environnements distribués et dynamiques tels que les environnements de services. Enfin, la gestion des risques est au cœur de l'implémentation du processus de gestion de la sécurité.

Dans le chapitre suivant de l'état de l'art, nous complétons notre étude en abordant les concepts de la gestion des risques dans l'objectif de trouver des travaux pertinents pour les environnements de services.

Chapitre 4. Gestion des Risques

Résumé

Dans ce chapitre, nous étudions les concepts liés à la gestion des risques. En effet, le processus de gestion des risques se place au cœur de la gestion de la sécurité. Nous nous focalisons sur quatre méthodes qui font référence dans le domaine pour étudier leur pertinence dans la gestion des risques d'une SOA.

Sommaire

4.1 Introduction

Dans les chapitres précédents, nous avons présenté les architectures orientées services ainsi que les problématiques de sécurité qui leur sont associées en montrant pourquoi les approches traditionnelles de la sécurité ne sont pas adaptées aux SOA.

Dans ce chapitre, nous présentons la gestion des risques. En effet, le processus de gestion des risques nous permet d'aborder à la fois la sécurité des éléments métier et technologiques en établissant un dialogue entre les équipes métier et technique et en facilitant la tâche de prise de décision dans le traitement des risques. Dans ce qui suit, nous commençons par définir les risques dans différents domaines d'applications et présentons quatre méthodes de gestion des risques. Nous étudions ensuite leur portée et leurs applications dans le domaine des services.

4.2 Le risque : définition et domaine d'application

De manière générique, le guide ISO 73:2009 [85] propose un vocabulaire sur les risques et leur gestion, ce guide définit le risque comme étant « l'effet d'une incertitude sur des objectifs ». Le terme «risque» est utilisé dans différents contextes et domaines comme par exemple, dans les domaines de la finance, du management, de la gestion des projets, des systèmes d'informations, etc. Dans le Tableau 4-1, nous récapitulons quelques définitions.

Domaine d'application	**Définitions**
Finance	'Risk refers to the variance of return' [86] p.77
Management	'Risk refers to the uncertainty that surrounds future events and outcomes. It is the expression of the likelihood and impact of an event with the potential to influence the achievement of an organization's objectives' [87] p.3
Gestion des projets	Le risque est la possibilité que survienne un événement dont l'occurrence entrainerait des conséquences (positives ou négatives) sur le déroulement de l'activité du projet [88] p.30
Systèmes d'information	Le risque est un scénario qui combine un événement redouté (sources de menaces, bien essentiel, critère de sécurité, besoin de sécurité, impacts) et un ou plusieurs scénarios de menaces (sources de menaces, bien support, critère de sécurité, menaces, vulnérabilités) [3] p.92

Tableau 4-1: Les définitions du risque

Dans notre travail, nous utilisons la définition du risque d'après [3] puisque nous nous intéressons aux risques *affectant les SOA et représentant des évènements redoutés portant atteinte aux éléments essentiels de l'architecture* du système d'information à base de services.

Définition 1 : Un risque est défini comme la probabilité d'occurrence d'un évènement redouté et les conséquences de cet évènement sur les éléments essentiels d'une SOA.
A noter que :

- Le terme risque dans ce travail est associé à la sécurité du système d'information
- Les éléments essentiels représentent les biens métier et technologiques à protéger dans un environnement de services. Ces éléments seront détaillés dans le chapitre suivant.

Nous définissons plusieurs concepts liés au risque :

Définition 2 : Un événement redouté - Unwanted Incident – est un scénario générique représentant une situation crainte par l'organisme. Cet événement correspond à la combinaison d'un bien essentiel, d'un critère de sécurité, du besoin de sécurité associé, des sources de menaces susceptibles d'en être à l'origine et des impacts potentiels [3]. Dans le contexte de notre travail, un évènement redouté est un évènement qui nuit ou réduit la valeur des éléments essentiels d'une SOA.
Définition 3 : Une menace est une cause potentielle d'un évènement redouté. Les menaces peuvent être classées en différentes catégories: (a) Des menaces d'origine humaine délibérée (b) Des menaces d'origine humaine accidentelle (c) Des menaces d'origine non humaine.
Définition 4: Les scénarii de menace sont les scénarios qui conduisent à un ou plusieurs événements redoutés. Ces scénarii sont déclenchés par les menaces et permettent d'expliquer et de décrire comment les menaces peuvent être les causes initiales des événements redoutés.
Définition 5 : Une vulnérabilité est une propriété essentielle d'une ressource entraînant une sensibilité à une source de risque pouvant induire une conséquence [85]. Dans le contexte de notre travail, nous définissons les vulnérabilités comme étant des faiblesses ou des failles pouvant être exploitées et pouvant nuire aux valeurs des éléments essentiels d'une SOA.
Définition 6: Une mesure de sécurité est le moyen de traitement des risques. Dans notre travail, une mesure de sécurité consiste à mettre en place une ou plusieurs des solutions de sécurité suivantes: (a) un protocole de sécurité (b) un mécanisme de sécurité (c) une politique de sécurité (d) un service de sécurité.
Définition 7: Un besoin de sécurité est la définition précise et non ambiguë du niveau d'exigence opérationnelle relative à un bien essentiel pour un critère de sécurité donné (disponibilité, confidentialité, intégrité…) [3]. Dans notre travail, nous retenons cette définition en remplaçant le terme « besoin » par « objectif » dans le sens où la sécurisation d'une SOA est un objectif à atteindre en dérivant les exigences de sécurité sur les différents éléments essentiels de la SOA.
Dans ce qui suit, nous introduisons la gestion des risques (Risk Management) et les processus associés : l'analyse du risque (Risk Analysis) et l'évaluation du risque (Risk Assessment).

4.3 La gestion des risques

Une définition abstraite et indépendante du domaine d'application de la gestion des risques est : un processus constitué d'une série d'activités, dans lesquelles plusieurs acteurs définissent le système d'étude et collaborent entre eux afin de transformer la connaissance du système en des objectifs et des décisions permettant de mieux gérer le système [89] p.37. Pour mener à bien une étude de gestion des risques, les acteurs devront avoir des origines et formations multidisciplinaires, tels que des experts métier, des experts techniques, des experts des lois et des réglementations, des auditeurs, etc. Ainsi, la gestion des risques permet la capitalisation des connaissances, l'amélioration de la prise des décisions et la création de nouvelles valeurs tout en justifiant l'investissement ainsi que les budgets alloués à la sécurisation des systèmes.

La norme (ISO/DIS 31000:2009) [85] définit la gestion des risques comme : '*Coordinated activities to direct and control an organisation with regard to risk*'. Cette norme fournit un guide générique ainsi que des conseils sur la mise en œuvre et la maintenance du processus de gestion des risques. Le processus de gestion des risques est défini en plusieurs étapes (Figure 4-1) :

- ✓ L'établissement du contexte : Dans cette activité, l'objectif est de définir les éléments du contexte et de délimiter le périmètre de l'étude. Il s'agit de prendre en compte les éléments internes à l'organisation tels que les parties prenantes, les capacités, la mission et les objectifs ainsi que les éléments externes tels que l'environnement économique, les lois et les règlementations.
- ✓ L'identification des risques : ici, l'objectif est d'identifier les risques à gérer. Ceci implique l'identification des sources de risques, des évènements redoutés et de leurs conséquences potentielles. Il s'agit de catégoriser les biens, les processus et les activités de l'organisation, d'identifier les périmètres des risques, de définir les risques et d'établir la typologie de ces derniers.
- ✓ L'analyse du risque : Dans cette activité, l'objectif est de déterminer les conséquences et la probabilité d'occurrence du risque, en prenant en considération les mesures du traitement des risques existants ainsi que leurs efficacités.
- ✓ L'évaluation du risque : elle consiste à comparer les niveaux des risques identifiés dans l'étape précédente par rapport à des seuils de risques bien définis permettant la prise de décision afin d'accepter, de réduire, de transférer et de supprimer le risque.
- ✓ Le traitement des risques : Cette activité consiste à identifier l'ensemble des mesures existantes pour le traitement des risques, l'évaluation et la mise en place de ces mesures de traitement des risques.
- ✓ La communication et le conseil : Cette activité implique de privilégier le dialogue entre les personnes qui mènent l'étude de gestion des risques au sein de toutes les activités citées ci-dessus. Le dialogue doit se baser sur un échange de compétences et d'expériences en vue d'atteindre une maturité maximale dans la prise des décisions.

✓ La supervision et la révision des risques : Cette activité consiste à superviser et réviser d'une façon périodique les traitements des risques et leurs résultats.

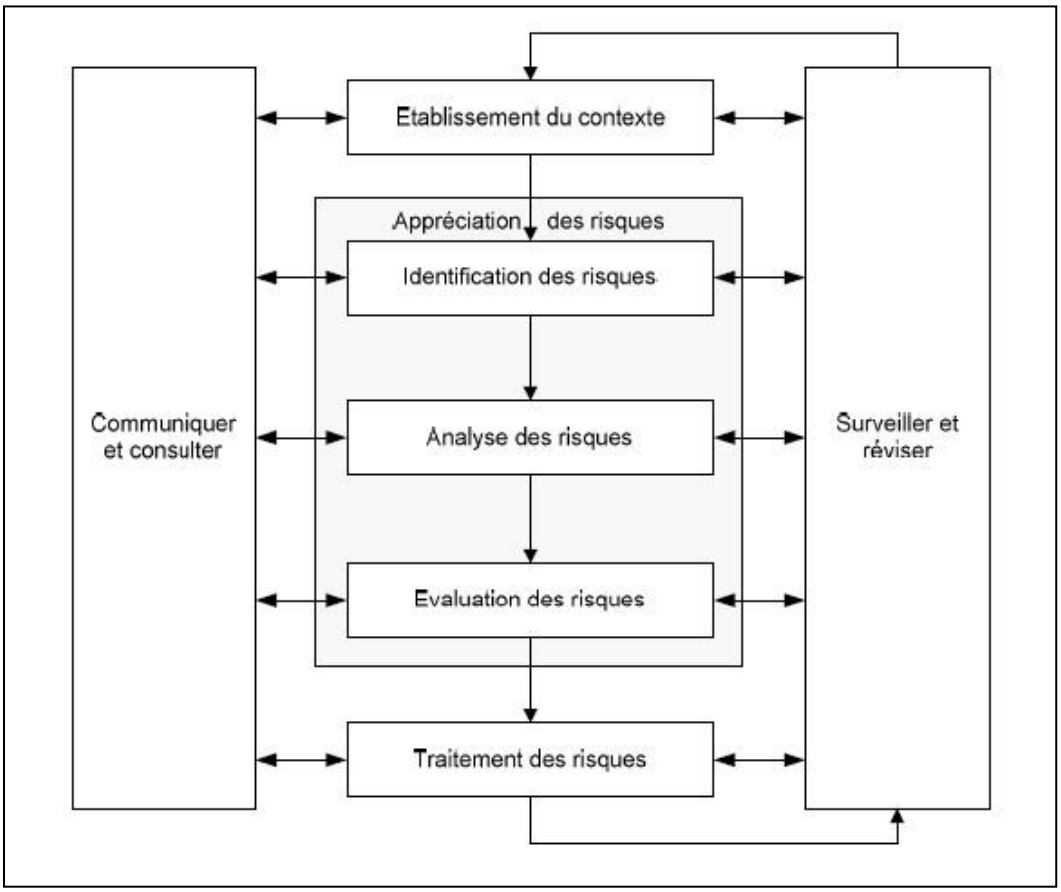


Figure 4-1 : Cycle de gestion des risques ISO 31000

Au sein du cycle de gestion des risques, les activités d'établissement du contexte, d'identification, d'analyse, d'évaluation et de traitement des risques représentent le processus d'analyse des risques (Risk Analysis). Au sein du cycle d'analyse des risques, les activités d'identification, d'analyse et d'évaluation des risques représentent le processus d'appréciation des risques (Risk Assessment) (Figure 4-1)

Dans son travail de thèse [90], Amadou SIENOU redéfinit le modèle générique de gestion des risques en intégrant l'activité « communiquer et consulter » dans le reste des activités, étant donné que la gestion des risques est un travail d'équipe impliquant plusieurs responsabilités. En outre, cette activité doit faire partie de toutes les étapes de l'analyse du risque. Ensuite, l'auteur considère que le processus défini dans le standard de l'ISO 31000 intègre plusieurs cycles imbriqués impliquant une difficulté au niveau du pilotage et par la suite, il propose de réorganiser les phases sous forme d'étapes (Figure 4-2)

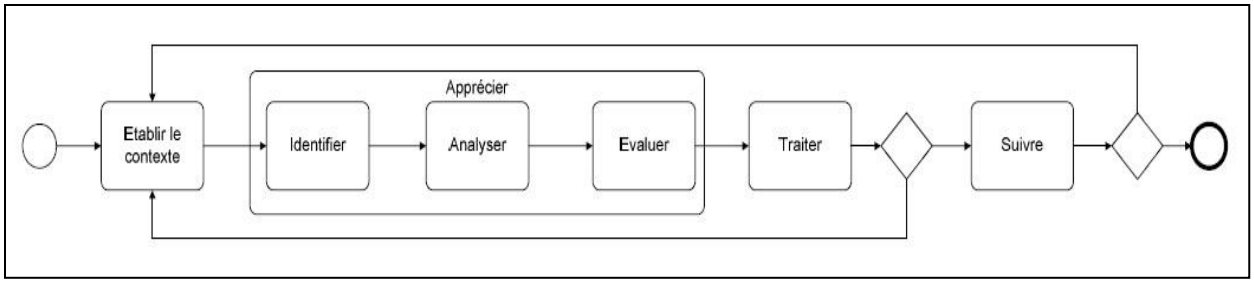


Figure 4-2 : Cycle de gestion des risques générique [90] p.74

Nous trouvons qu'il serait judicieux d'adapter ces cycles de gestion des risques génériques à un environnement de service. Cette adaptation (Figure 4-3) pourra être réalisée en :

- ✓ Ajoutant l'activité « identification des exigences de sécurité » en parallèle à l'activité « identification des risques ». En effet, dans le contexte des SOA, il est impératif de montrer l'importance des exigences de sécurité sur les éléments métier et technologiques et de se baser sur ces exigences pour l'évaluation des risques. La création de cette activité permettra de se focaliser sur l'identification des exigences de sécurité afin de mieux cadrer le périmètre de gestion des risques.

- ✓ Concaténant les tâches des deux activités « analyse » et « évaluation » des risques dans une seule activité « évaluation des risques » car ces deux activités sont fortement couplées. En effet, le regroupement des tâches de ces deux activités est effectué dans le cadre de la gestion des risques des systèmes d'information (voir le module 4 de la méthode EBIOS et la phase 3 de la méthode OCTAVE dans la section 4.4). Par conséquent, ce regroupement est possible dans le cadre de la gestion des risques des SOA qui permettent de construire des SI.

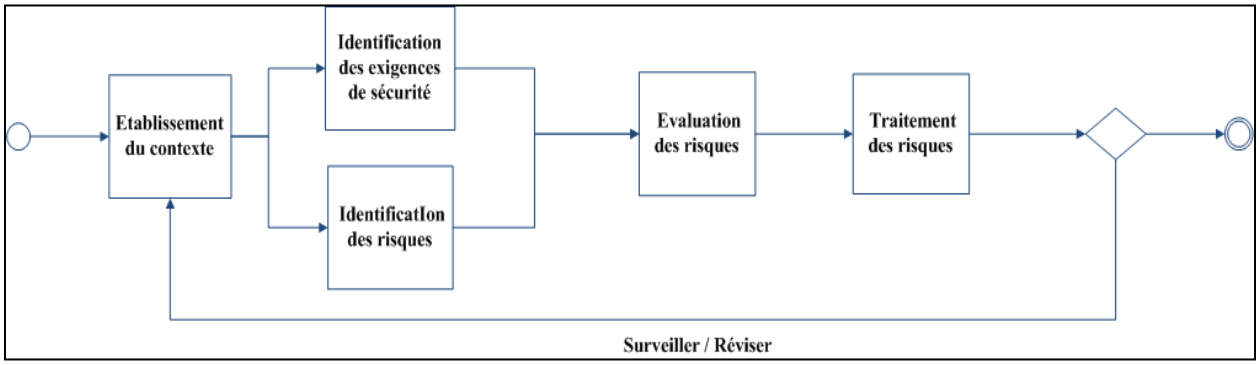


Figure 4-3 : Cycle de gestion des risques adapté aux SOA

La gestion des risques et les architectures orientées services

Nombreux sont les travaux de recherche disponibles dans la littérature visant à sécuriser les architectures orientées services par une approche de gestion des risques. Parmi ces travaux, certains s'orientent vers les aspects technologiques et d'autres couvrent à la fois les visions métier et technologiques.

Dans [91], les auteurs définissent une approche pour la sécurisation des applications dans un environnement de services dynamiques. L'approche se base sur l'identification des composants de l'application qui peuvent être cibles d'attaques ainsi que sur l'identification des vulnérabilités associées. Dans [92], l'identification des risques repose sur des bases de vulnérabilités internes ou externes (NVDB, OSVDB). L'approche utilisée est ascendante. L'étude des vulnérabilités se fait en passant par les composants de l'infrastructure, les applications, les services et enfin les processus. Ces deux contributions [91] et [92]) ne répondent pas à nos besoins puisqu'elles se focalisent sur les aspects technologiques des SOA et ne prennent en considération ni les aspects métier, ni les exigences de sécurité et les préférences des utilisateurs.

Dans le cadre de la sécurisation des services web, un travail sur l'identification des risques portant sur les données personnelles est proposé par [93]. L'analyse des risques utilise des diagrammes dénotant les flux, le stockage et l'utilisation des données personnelles en intégrant les préférences des utilisateurs. Ce travail est complémentaire aux approches technologiques précédentes. Toutefois, il se limite aux données personnelles et ne couvre pas tous les aspects métier et organisationnels.
Dans [94], l'auteur propose une plateforme de gestion des risques pour les architectures orientées services qui prend en compte à la fois les aspects métier et technologiques (Figure 4-4). La plateforme proposée permet la réduction des risques en améliorant le choix des partenaires, la découverte des services, leur composition et leur QoS. Toutefois, ce travail ne fait aucune référence aux méthodes de gestion des risques permettant d'améliorer, voire même d'optimiser le travail de la gestion des risques puisqu'il manque un cadre méthodologique explicitant les détails de la démarche à suivre, les outils et la documentation à utiliser.

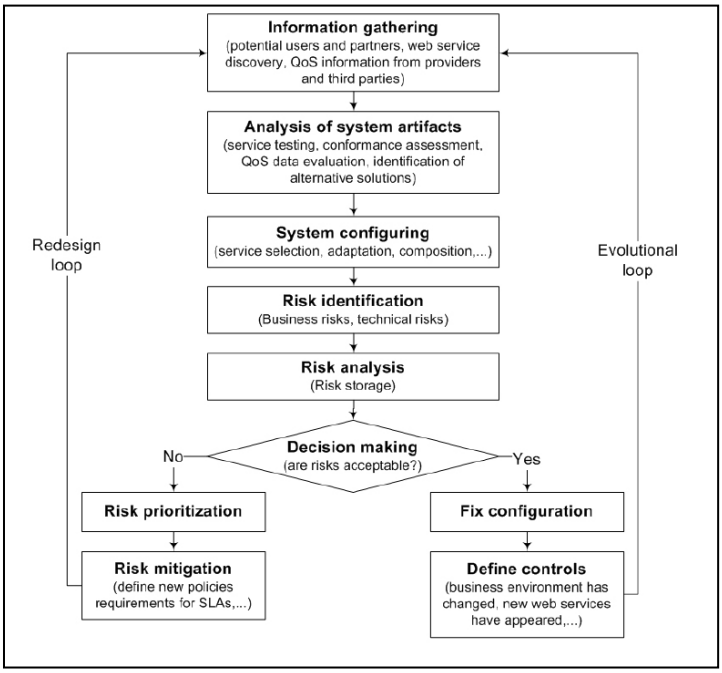


Figure 4-4 : Plateforme de gestion des risques pour les SOA [94]

Suite à ce constat, nous abordons dans ce qui suit quatre méthodes qui font référence dans le domaine de la gestion des risques pour identifier une méthode ou tout au moins des éléments adaptés au contexte SOA.

4.4 Méthodes de gestion des risques

Pour ce qui concerne la sécurisation des systèmes d'information, une méthode de gestion des risques est un outil d'analyse permettant d'identifier les risques pesant sur ces systèmes, puis y remédier en proposant des solutions. Les méthodes de gestion des risques s'appuient sur différentes stratégies d'analyse. Dans ce travail de recherche, nous avons choisi quatre méthodes qui couvrent différentes perspectives dans la gestion des risques: EBIOS [3] , OCTAVE [95], SNA [96] et CORAS [97]. En effet, EBIOS est une méthode assez reconnue et fait référence dans le domaine de la gestion des risques des systèmes d'information. La méthode OCTAVE s'oriente vers la gestion des risques portant sur les systèmes opérationnels (information, systèmes, logiciel, matériel, personnes) qui ont un effet immédiat sur l'organisme. La méthode SNA couvre une autre perspective qui est celle de la « survie » des systèmes, en présence d'attaques, des défaillances et d'accident. Enfin, la méthode CORAS est une méthode générique et se base sur la modélisation des risques pour conduire une analyse centrée sur les ressources et les biens de l'entreprise.

4.4.1 Expression des Besoins et Identification des Objectifs de Sécurité (EBIOS)

EBIOS [3] est une méthode développée et maintenue par l'ANSSI (Agence Nationale de la Sécurité des Systèmes d'Information) du secrétariat général de la défense nationale française (SGDN). La méthode consiste à formaliser les objectifs et les exigences de sécurité adaptés au contexte du système étudié. Cette méthode vise la sécurisation des 'gros' systèmes d'information et s'appuie sur le standard ITsec. La démarche méthodologique permet d'impliquer l'ensemble des acteurs du système d'information dans la problématique de la sécurité.
La démarche de la méthode comprend cinq modules (Figure 4-5)

- Étude du contexte
- Étude des événements redoutés
- Étude des scénarios de menace
- Étude des risques
- Détermination des mesures de sécurité

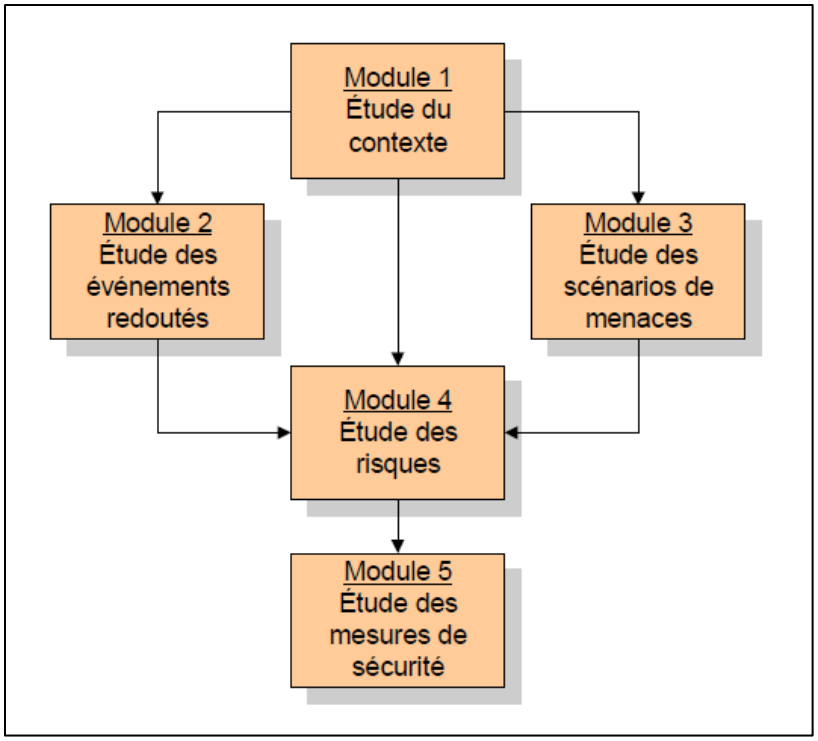


Figure 4-5 : Démarche EBIOS [3] p.13

Le premier module est l'étude du contexte. Il a pour objectif de formaliser le cadre de la gestion de risques c'est-à-dire identifier, délimiter et décrire le périmètre de l'étude en identifiant les biens essentiels (patrimoine informationnel), leurs relations et leurs propriétaires ainsi que les biens supports (techniques ou non techniques).

Le deuxième module inclut l'étude des événements redoutés, ces derniers représentant les scénarios génériques qu'il faut éviter sur le périmètre de l'étude. Dans cette méthode, les réflexions sont menées sur les biens essentiels (définis comme l'information ou les processus jugés important pour l'organisme) et non pas sur les biens de support (définis comme des biens sur lesquels reposent les biens essentiels comme le système d'exploitation). Les événements redoutés sont identifiés, analysés et évalués en termes de gravité et de vraisemblance.

Parallèlement à l'appréciation des événements redoutés, EBIOS propose un troisième module : l'étude des scénarios de menaces. Ces derniers représentent les modes opératoires génériques pouvant porter atteinte à la sécurité des informations appartenant au périmètre de l'étude. Dans ce module, les scénarios de menaces sont identifiés, analysés et évalués. Les données nécessaires sont obtenues en récapitulant :

- ✓ Les menaces qui pourraient se réaliser ;
- ✓ Les vulnérabilités exploitables sur les biens supports ;
- ✓ Les sources de menaces susceptibles à l'origine d'une attaque.

Le quatrième module permet de réaliser l'identification des risques pesant sur le périmètre de l'étude. L'objectif est de créer le lien entre les événements redoutés et les scénarios de menaces, c'est-à-dire entre ce que l'organisme craint et ce à quoi il est exposé. Dans ce même module, on identifie également les objectifs de sécurité afin de choisir la manière dont chaque risque devra être traité en fonction de l'évaluation de son impact et de la potentialité de son occurrence. Il convient de choisir parmi les options suivantes :

- L'évitement d'un risque consiste à changer le contexte de conception de telle sorte qu'on ne soit plus exposé à ce risque.
- La réduction d'un risque consiste à prendre des mesures de sécurité pour diminuer l'impact et/ou la probabilité d'occurrence.
- Le transfert consiste à transférer le risque, ou une partie du risque, à un tiers.
- La prise du risque est la décision d'accepter un risque lorsque son évitement, sa réduction ou son transfert s'avère impossible ou trop coûteux. Ceci permet de mettre en évidence les risques résiduels qui subsisteront.

Le dernier module de la méthode est l'étude des mesures de sécurité. Ce module consiste à formaliser les mesures de sécurité à mettre en œuvre, afin de déterminer les actions à entreprendre ainsi que les mesures de sécurité adéquates. Des activités d'élaboration et de suivi de la réalisation du plan de traitement sont mises en œuvre à la fin de ce module.

Les avantages de la méthode sont multiples :

- ✓ EBIOS supporte efficacement l'ensemble des actions, notamment la définition du périmètre, l'appréciation des risques et la spécification du traitement des risques.
- ✓ EBIOS contribue à l'élaboration du plan de traitement des risques d'un schéma directeur, des politiques de sécurité et des tableaux de bord.
- ✓ EBIOS présente des bases de connaissance intégrées ainsi qu'un logiciel libre (Figure 4-6) qui fournissent un support rapide et efficace.

Une version mise à jour en 2010 propose les améliorations suivantes :

- ✓ La convergence des concepts vers les normes internationales relatives au système de management de la sécurité de l'information et à la gestion des risques (ISO 31000).
- ✓ La mise en évidence des actions de communication et concertation, et des actions de surveillance et revue dans les descriptions des activités.

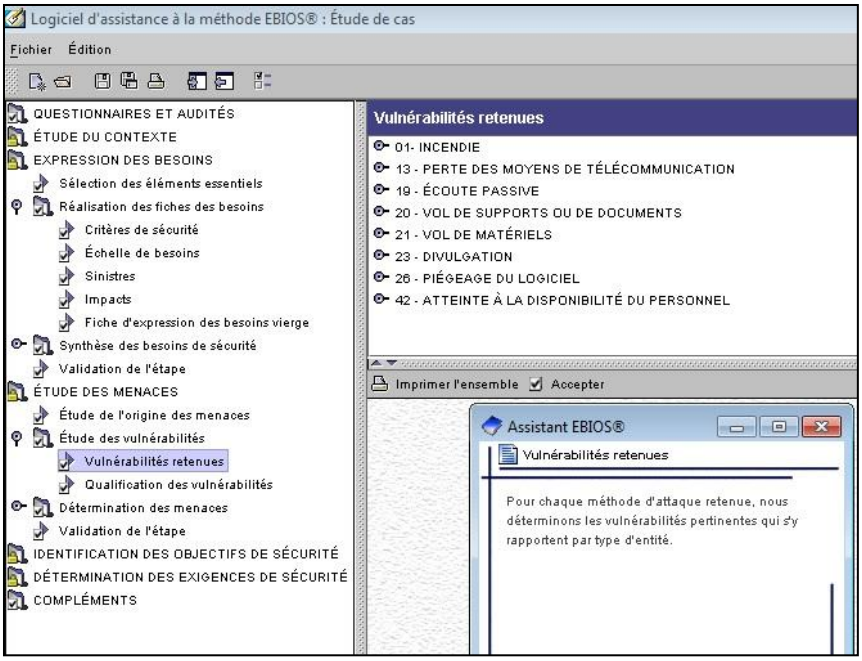


Figure 4-6 : Logiciel d'assistance à la méthode EBIOS

- ✓ L'ajout de la définition du cadre de la gestion des risques : vision projet, étude des sources de menaces.
- ✓ La prise en compte des aspects relatifs aux processus métier de l'entreprise.

Toutefois, la méthode reste assez cloisonnée. La mise en œuvre des étapes telles qu'elles sont conçues réduit la flexibilité de cette méthode et complique son application dans le domaine des architectures orientées services.

4.4.2 Operationally Critical Threat, Asset, and Vulnerability Evaluation (OCTAVE)

La méthode OCTAVE [95], développée au SEI (Software Engineering Institute) à l'université Carnegie Mellon fournit un moyen de traitement des risques en couvrant à la fois les dimensions organisationnels et techniques. La méthode est conçue pour être menée par le personnel de l'organisation sans faire appel à des consultants externes : elle est auto-dirigée. Cette méthode présente trois variantes : OCTAVE qui vise les grandes organisations, OCTAVE-S qui est la version réduite visant les PME/PMI et OCTAVE Allegro qui se focalise principalement sur le stockage, le transport, la gestion de l'information et les risques associés.

OCTAVE se concentre sur l'évaluation des vulnérabilités et des menaces qui touchent les biens critiques ayant un effet immédiat sur l'organisation. Trois phases sont au cœur de la méthode (Figure 4-7).

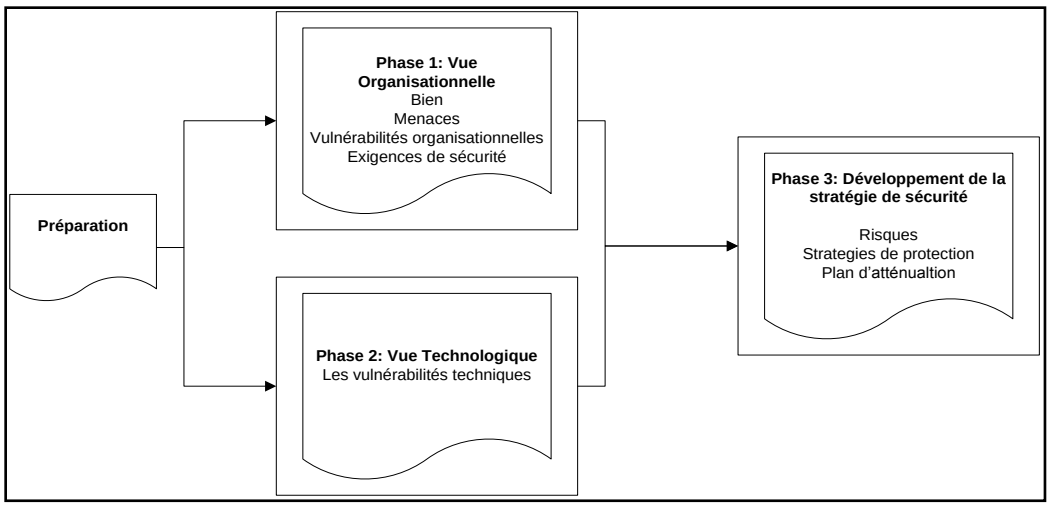


Figure 4-7 : Démarche OCTAVE

Phase 1 : Vue organisationnelle

Cette phase couvre les aspects organisationnels et se focalise sur la constitution des profils de menaces basés sur les actifs de l'entreprise. Il s'agit de l'identification des ressources importantes, des menaces et des exigences de sécurité associées. Les mesures de sécurité existantes sont identifiées pour produire des profils de menace.

Phase 2 : Vue technologique

Dans cette phase se fait l'identification des vulnérabilités de l'infrastructure à partir de composants clefs. Les analystes identifient les moyens d'accès aux actifs opérationnels afin d'identifier les composants à sécuriser.

Phase 3, Développement de la stratégie de sécurité

Les analystes se focalisent sur l'atténuation des menaces suite à l'évaluation des risques pour définir des stratégies de protection et créer un plan d'atténuation.

La méthode OCTAVE représente une approche structurée et systématique pour l'identification et la réduction des risques. Elle repose sur des séances de brainstorming et sur le travail d'équipe. En plus de la documentation complète qu'elle fournit pour l'identification des biens à protéger, des menaces et des risques, la méthode OCTAVE est accompagnée d'un catalogue de bonne pratique de sécurité. Ce catalogue peut être utilisé à deux stades de la démarche : lors de l'évaluation des mesures de sécurité mises en place et au moment de la mise en œuvre de la stratégie de protection. De plus, ce catalogue inclut des pratiques stratégiques et opérationnelles. L'inconvénient de la méthode est qu'elle se concentre sur la sécurisation des actifs opérationnels. En effet, les biens à protéger sont classés en cinq catégories : l'information, les systèmes, les applications logicielles, les équipements matériels, le personnel de l'organisation. Toutefois, dans le cadre de notre travail, nous nous intéressons à fournir une sécurité plus globale, qui va au-delà de la sécurisation individuelle de ces actifs. En particulier, nous nous intéressons davantage à la sécurisation des éléments essentiels métier que nous détaillerons dans le chapitre suivant.

4.4.3 Survivable Network Architecture (SNA)

La méthode SNA [96], développée au CERT et à l'université Carnegie Mellon propose de mettre en place des moyens de sécurisation rendant le système informatique capable de survivre en cas d'attaque. Elle se focalise sur les moyens de communication et les réseaux.
La méthode permet d'assurer la continuité des services selon trois stratégies ('les trois R') :

- Résistance : Identification des mesures permettant d'empêcher les attaques.
- Reconnaissance : Identification des attaques et évaluation des dommages résultants.
- Recouvrement : Prestation des services pendant et après l'attaque.

La démarche de la méthode est organisée selon un cycle en spirale (Figure 4-8) pour assurer la survie du système à tout moment. Ceci présente un atout primordial dans des contextes dynamiques tels que ceux des services. En effet, SNA définit le système selon un axe technologique comprenant les composants informatiques et réseaux et un axe organisationnel comprenant les processus métier associés aux missions de l'organisation.

L'étape 1 de la méthode permet aux analystes d'identifier les besoins et la mission d'un système. Elle permet d'identifier la structure et les propriétés de son architecture. Dans l'étape 2, les analystes identifient les services essentiels et les biens à protéger en se basant sur la mission du système et les impacts des défaillances. Pour cela, SNA identifie les scénarios d'utilisation caractérisant les services essentiels et les biens à protéger.

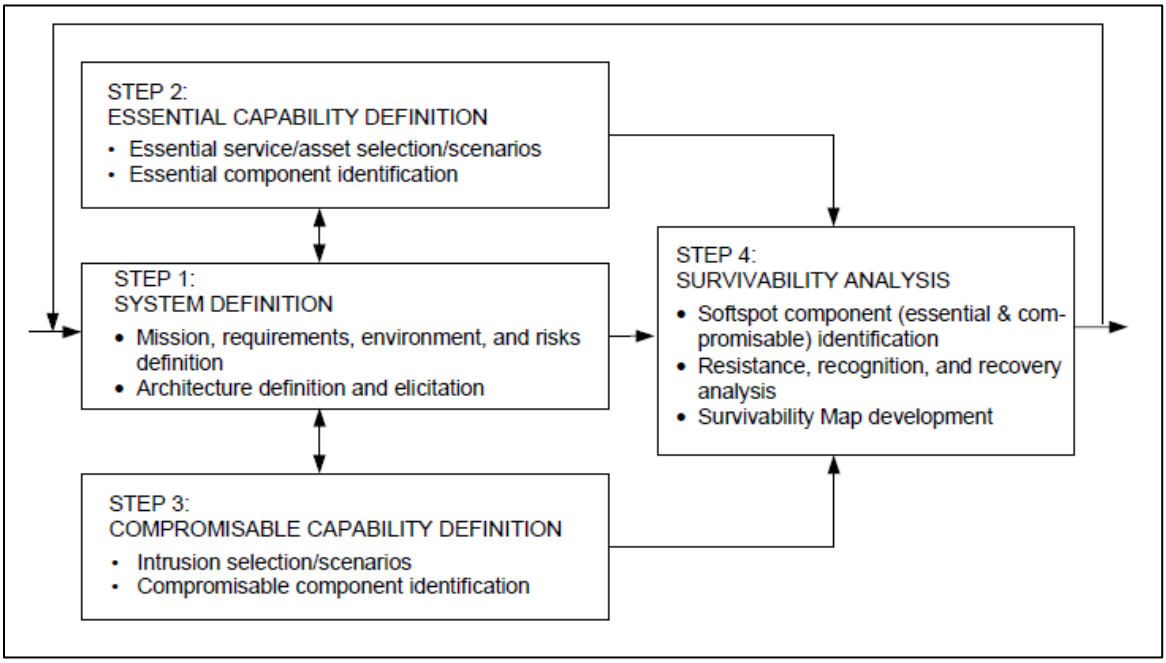


Figure 4-8 : Démarche SNA [96] p.18

Dans la troisième étape, l'équipe sélectionne les scénarios d'intrusion en utilisant sur le contexte de l'étude, l'évaluation des risques et des capacités des attaquants. Cette phase s'appuie sur la consultation des bases d'attaques du CERT pour la détermination de ces scénarios. Ces derniers sont ensuite 'mappés' sur l'architecture du système afin de déterminer ceux qui doivent être retenus vis-à-vis du système étudié.

Dans l'étape 4, l'équipe identifie les composants de l'architecture essentiels et vulnérables. L'équipe analyse alors ces composantes en termes de résistance, reconnaissance et récupération. L'analyse des trois «R» est résumée dans un tableau de survie (Tableau 4-2) :

Scénario d'intrusion	Effets	Architecture nécessaire	Résistance	Reconnaissance	Recouvrement
Scénario 1		Actuelle :			
		Recommandée :			
. . .					
Scénario n		Actuelle :			
		Recommandée :			

Tableau 4-2 : Tableau synoptique proposé par SNA

Le tableau de survie proposé dans la méthode, énumère pour chaque scénario d'intrusion ses effets, l'architecture actuelle et les stratégies recommandées pour améliorer la résistance, la reconnaissance et le recouvrement. Compte tenu de ses caractéristiques, la méthode SNA permet l'audit d'une SOA opérationnelle et donne de nouvelles perspectives sur l'étude de la continuité des services et le recentrage sur les missions essentielles. Toutefois, elle n'est que partiellement accessible et nécessite l'intervention d'experts en audit de sécurité.

4.4.4 CORAS

CORAS [94] est une approche d'analyse des risques qui peut être adaptée et appliquée à différents domaines. En particulier, elle a été appliquée dans le domaine de la sécurité informatique, la protection civile, la défense et la santé. Utilisant les concepts liés au risque (définis dans la partie '4.2 le risque : définition et domaine d'application'), CORAS propose une méthode d'analyse, un langage de modélisation des risques et un outil support.

La méthode se concentre sur l'analyse des risques autour des biens directs et les biens indirects à protéger. Un bien direct est une ressource de valeur nécessitant d'être protégée, tandis qu'un bien indirect est un bien menacé uniquement par des risques si d'autres biens direct le sont.
La démarche se compose de huit étapes (Figure 4-9)

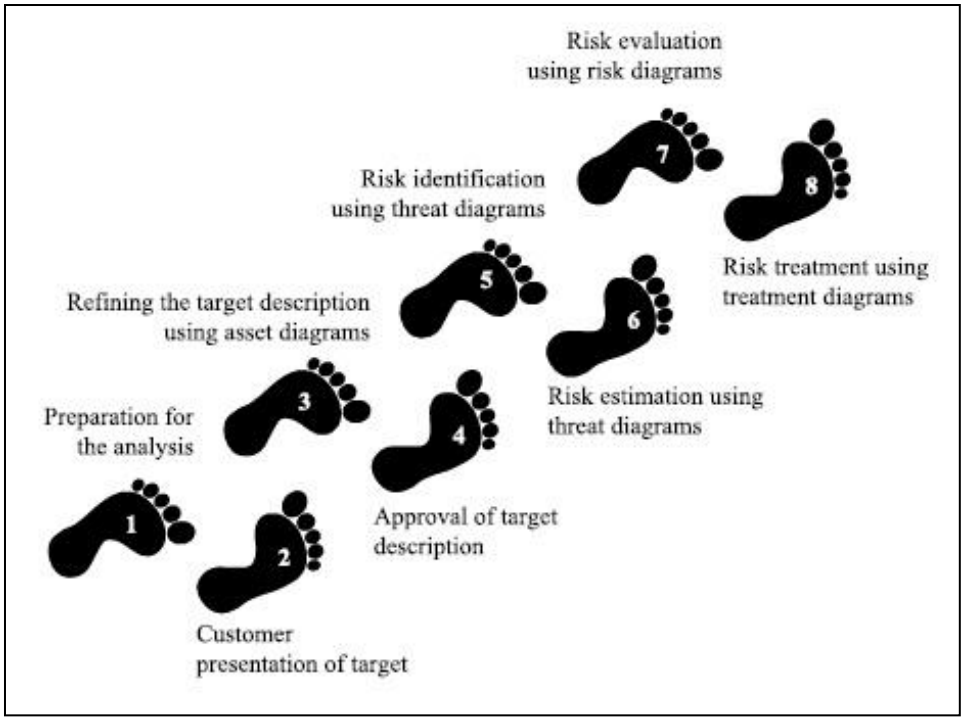


Figure 4-9: Démarche CORAS [97] p.24

L'objectif de la première étape est de préparer l'analyse des risques en définissant le périmètre de l'étude. Dans la deuxième étape, une réunion avec les représentants du client permet de recueillir l'information formant le contexte de l'étude (cible de l'étude, biens à protéger et portée de l'étude). Dans la troisième étape les analystes récapitulent les informations du contexte. Ce contexte sera approuvé par le client dans l'étape quatre. Dans cette quatrième étape, les échelles de probabilité d'occurrence (données en intervalles de temps: années, mois semaine, etc.) et les échelles de conséquences (ou effets) d'un évènement redouté (négligeable, importante, critique, etc.) sont définies.

Dans la cinquième étape, les risques sont identifiés en se basant sur les menaces, les vulnérabilités, les scénarios de menaces et les évènements redoutés. Dans l'étape six, l'évaluation des évènements redoutés permet d'estimer les risques vis à vis des biens à protéger en utilisant les diagrammes de menaces. L'étape sept est destinée à faire une deuxième évaluation des risques pourtant sur les biens indirects (ex : la réputation de l'entreprise). A la différence des autres méthodes (comme OCTAVE) qui considère la réputation comme une mesure d'impact, CORAS considère la réputation de l'entreprise comme un bien indirect.
Le processus d'évaluation permet enfin d'attribuer des priorités aux traitements des risques qui sont traités dans l'étape huit.

Le langage de modélisation proposé par CORAS utilise des symboles graphiques et des diagrammes pour la modélisation des risques :

- ✓ La Figure 4-10 illustre les symboles du langage de modélisation : menaces d'origine humaine accidentelle, menaces d'origine humaine délibérée, menaces d'origine non humaine, biens directs, biens indirects, acteur (entreprise ou personne qui mène l'étude de gestion des risques : 'party'), vulnérabilités, scénarios de menaces, scénarios de traitement (implémentation des mesures de sécurité), événements redoutés et le risque.
- ✓ Les diagrammes sont destinés à être utilisés lors des séances de brainstorming. Ils facilitent la communication entre les personnes menant l'étude, indépendamment de leur formation. La Figure 4-11 illustre une menace d'origine humaine accidentelle (en fait une erreur due à un manque de compétence) conduisant à l'atteinte à l'intégrité des données confidentielles suite à une corruption de la base de données.

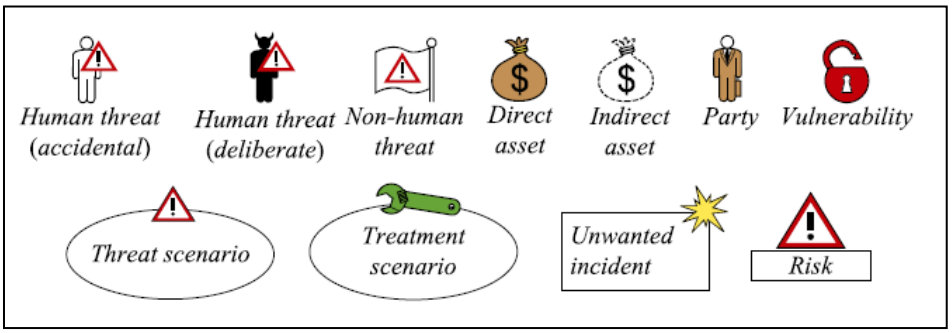


Figure 4-10 : Symboles du langage de modélisation des risques CORAS [97] p. 27

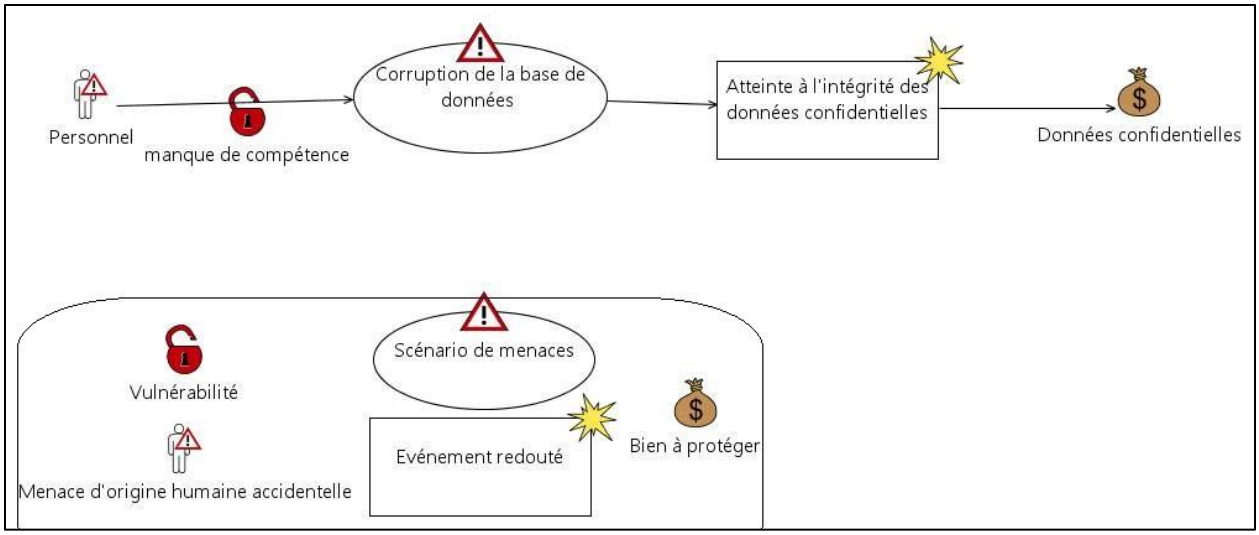


Figure 4-11 : Exemple de diagramme de menaces

Enfin, l'outil support permet de créer rapidement les diagrammes pour modéliser des risques, faciliter la documentation et présenter les résultats de l'analyse des risques.

Autres méthodes de gestion des risques

D'autres méthodes supportent également la gestion des risques :

1- MEHARI (Méthode Harmonisée d'Analyse de Risques) [98] est une méthode d'analyse et de gestion des risques développée par le CLUSIF (Club de la Sécurité de l'Information Français). La méthode fournit des outils et des bases de connaissance pour analyser les enjeux majeurs, étudier les vulnérabilités, réduire la gravité des risques et piloter la sécurité de l'information. MEHARI propose une démarche de trois phases :
   - ✓ La phase préparatoire permettant d'établir le contexte de l'étude.
   - ✓ La phase opérationnelle d'analyse des risques.
   - ✓ La phase de planification du traitement des risques.
2- CRAMM (CCTA Risk Analysis and Management Method) [99] est la méthode de gestion des risques élaborée par le gouvernement britannique et maintenue par Siemens/Insight. La méthode inclut des outils d'aide et de documentation pour supporter le travail des analystes de sécurité afin de déterminer les politiques de sécurité à appliquer sur le système.
3- Microsoft Application Threat Modeling [100] est une méthode qui propose des diagrammes de flux de données pour représenter graphiquement la cible de l'analyse. La cible est décomposée en éléments qui sont analysés vis-à-vis de menaces identifiées. La méthode utilise des arbres de menaces pour l'analyse d'attaques et s'appuie sur la liste STRIDE (Spoofing, Tampering, Repudiation, Information disclosure, Denial of Service, and Elevation of privilege) pour l'identification des risques.

Dans la suite de notre travail, et de manière à simplifier les comparaisons, nous avons retenu 4 méthodes abordant la gestion des risques selon des visions complémentaires :

- Analyse systématique des risques selon les dimensions métier et technologiques : plusieurs méthodes comme MEHARI, CRAMM et EBIOS couvrent ce point de vue. Dans la suite de notre travail, nous retenons la méthode EBIOS qui fournit une documentation et des catalogues de menaces et de vulnérabilités permettant d'identifier systématiquement les risques et de concevoir des SOA sécurisées.
- Analyse des scénarii d'accès conduisant à un risque : nous retenons la méthode OCTAVE qui repose sur ce principe et qui permet d'identifier les risques dans des ateliers de brainstorming en créant un dialogue commun entre les responsables métier et techniques.
- Intégration de la reprise d'activité : nous retenons la méthode SNA qui permet de doter le système de capacité de survie et d'enrichir la perspective de réduction des risques par celle de la continuité et de la reprise des activités.
- Intégration d'une vision regroupant sécurité et sureté de fonctionnement : nous retenons la méthode CORAS qui s'adapte à plusieurs secteurs d'activité et qui permet de modéliser les atteintes au bon fonctionnement du système.

Toutefois, les critères de choix que nous définissons plus tard dans ce chapitre peuvent être utilisés pour déterminer la pertinence de toute autre méthode dans le domaine des services à

condition qu'elle intègre une dimension « métier » (cette dimension étant absente de la méthode 'Microsoft Application Threat Modeling', nous ne l'avons pas retenu dans ce qui suit).

4.4.5 Stratégies d'analyse

A partir des méthodes vues précédemment, on constate qu'elles peuvent utiliser plusieurs modèles d'analyse :

1. Les modèles basés sur une structure arborescente dont le sommet peut être un évènement, une défaillance ou une attaque. Les nœuds restants représentent la façon d'atteindre le sommet. Prenons comme exemple :
   - ✓ La technique « arbre d'attaque » [101] qui décrit la sécurité d'un système en se basant sur les attaques auxquelles le système peut être exposé ;
   - ✓ Les profils de menace utilisés dans OCTAVE [95]. La Figure 4-12 illustre les menaces pouvant être causées par une personne en se connectant via le réseau interne ou externe de l'entreprise.
2. Les modèles basés sur des tableaux : Les techniques de ce type réalisent de manière systématique l'analyse de l'information, sauvegardée dans des tableaux suite à des séances de brainstorming entre les personnes comme la technique Failure Mode Effect Analysis / Failure Mode Effect and Criticality Analysis (FMEA/FMECA) [102] qui permet de déterminer les modes de défaillance possibles d'un système ainsi que leurs conséquences. Des séances de brainstorming sont organisées pour déterminer la description fonctionnelle du système. Les modes de défaillances ainsi que leurs conséquences sont identifiées et documentées dans des tableaux FMEA/FMECA. En outre, cette stratégie d'analyse est utilisée dans la méthode EBIOS [3].

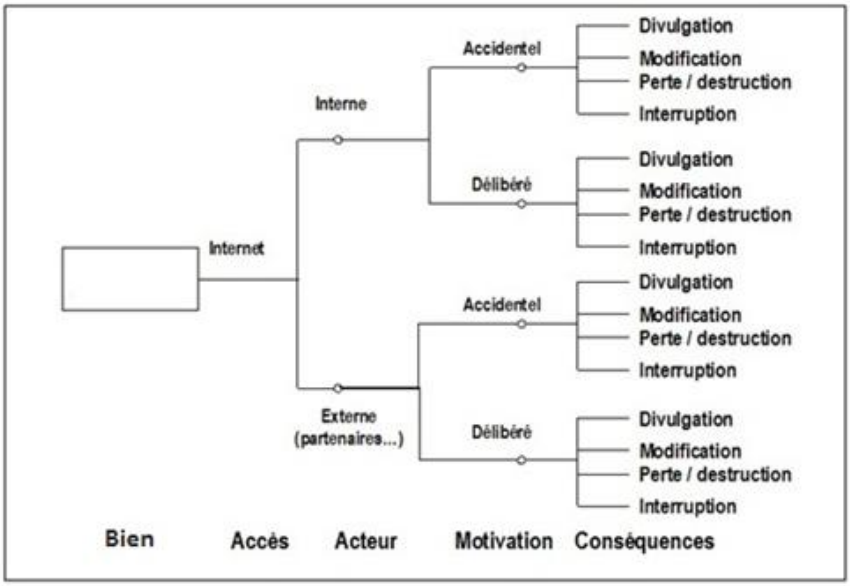


Figure 4-12 : Octave profils de menace

3. Les modèles basés sur des graphes pour déterminer les événements et les défaillances d'un système. Par exemple, les réseaux bayésiens [103] permettent d'identifier le nombre de défaillances dans les composants logiciels. Les nœuds du graphe représentent les facteurs contributifs et les arêtes représentent les dépendances entre ces facteurs. De plus, les nœuds sont caractérisés par des valeurs de probabilité qui dépendent des valeurs des nœuds parents. Tout changement de valeur au niveau des nœuds parents se répercute aux nœuds fils.

Les personnes qui mènent une étude de gestion des risques devront maîtriser la méthode et la technique utilisée afin de pouvoir réussir le projet de gestion des risques. Il n'existe pas une meilleure technique dans l'absolu mais certaines techniques sont mieux adaptées à des contextes particuliers. Les méthodes basées sur la notation d'arbre permettent de spécifier les détails par raffinements successifs. Par exemple, Octave utilise cette stratégie pour analyser les conséquences des évènements redoutés en utilisant la décomposition des évènements et des incidents. OCTAVE permet aussi de représenter les causes des évènements ainsi que les dépendances entre elles. Le recueil d'information est réalisé dans des séances de brainstorming dans lesquels les probabilités des évènements sont déduites. Cette méthode offre une excellente flexibilité dans la modélisation des menaces étant donné que les notations peuvent être alimentées et mises à jour à tout moment.

Les méthodes basées sur des tableaux sont très structurées et permettent de rendre l'étude de gestion des risques assez systématique en organisant toute l'information. Comme pour les méthodes basées sur la notation d'arbre, les tableaux peuvent être alimentés dans des séances de brainstorming pour lister les biens à protéger, les menaces et les risques portant sur le système étudié. Cette technique est facilement utilisable pour gérer les risques au niveau des systèmes d'information. Toutefois, elle manque de flexibilité pour qu'elle soit adaptée à des environnements dynamiques tels qu'un environnement de services.

Enfin, les approches basées sur des graphes s'avèrent très efficaces si des valeurs quantitatives précises peuvent être attribuées à la probabilité d'occurrence des évènements. La manipulation de la probabilité des nœuds parents se propage sur celle des nœuds fils. Toutefois, l'évaluation pertinente de cette probabilité reste un défi majeur.

Dans le cadre de notre travail, nous privilégions la flexibilité afin d'adapter l'étude de gestion des risques au changement du contexte. Nous objectif est de modéliser les risques en partant des éléments essentiels ou des menaces et de pouvoir alimenter les modèles à tout moment. Par conséquent, nous trouvons que les notations d'arbre sont les plus adaptées à notre environnement de services dynamiques.

4.4.6 Comparaison des méthodes

Afin de conduire efficacement une étude de gestion des risques, le choix de la méthode à utiliser est primordial. En effet, il faut se baser sur une méthode claire, bien documentée et qui répond aux besoins métier et techniques. Dans les sections précédentes, nous avons présenté plusieurs méthodes. L'objectif de cette section est de définir des critères de comparaison pour déterminer la/les méthode(s) la/les plus adaptée(s) pour la gestion des risques dans un environnement de services dynamiques.

4.4.6.1 Définition des critères d'analyse

Critère 1 : Objectif de la méthode

Ce critère définit l'objectif pour lequel la méthode a été conçue, et donc sa portée en fonction du périmètre de l'étude. Dans notre travail, nous cherchons une méthode qui permet de sécuriser les architectures orientées services en définissant les biens et les processus définis sous forme de workflows à protéger. La méthode recherchée doit donc prendre en considération les dimensions métier et techniques et placer les processus métier (et les éléments qui en dépendent) au cœur de l'étude. En outre, les SOA vont au-delà des frontières de l'entreprise. La méthode recherchée doit donc pouvoir couvrir les aspects liés aux partenariats, aux contrats de protection et aux obligations légales.

Les quatre méthodes que nous avons étudiées prennent en considération des aspects organisationnels et technologiques. Dans le cas de la méthode EBIOS, l'objectif est d'apprécier et de traiter les risques portant sur le système d'information d'un organisme. Cette méthode permet aussi de positionner la cible de l'étude vis-à-vis des partenaires de l'organisme et permet d'aboutir à la définition d'un plan de traitement des risques. OCTAVE, propose un cadre systématique pour évaluer les risques. Cette méthode permet d'avoir une vision globale des problèmes de sécurité tout en se focalisant sur les actifs opérationnels (qui ont un effet immédiat sur l'organisme). A la différence des autres méthodes, SNA a comme objectif de définir les conditions de survie des systèmes informatiques en se focalisant autour du réseau et des moyens de communication. Enfin, CORAS est une approche globale et systématique d'identification des biens essentiels et des risques correspondants. Cette méthode définit les concepts du risque d'une manière globale et abstraite, ce qui rend son utilisation possible dans tous les domaines.

En conclusion, les quatre méthodes permettent la gestion des risques dans une architecture orientée service.

Critère 2 : Les phases de gestion du projet sécurité

Ce critère définit la portée des méthodes et l'intégration de la gestion des risques dans les phases de gestion du système d'information. Nous définissons la phase d'organisation comme étant la sensibilisation des responsables métier et techniques aux concepts de la gestion des risques

durant la phase d'analyse du système d'information, et les phases de conception et d'exécution comme étant l'intégration de la gestion des risques dans les phases de conception et d'exécution du système d'information. OCTAVE se focalise sur la phase d'exécution du système d'information. SNA porte sur les phases de conception et d'exécution. Enfin, CORAS porte sur les trois phases d'organisation, de conception et d'exécution (Tableau 4-3).

Phases de gestion du projet sécurité / Méthode de gestion des risques	Organisation	Conception	Exécution
EBIOS	+	+	-
OCTAVE	-	-	+
SNA	-	+	+
CORAS	+	+	+
+ : la méthode couvre la phase de gestion du projet sécurité. - : la méthode ne couvre pas la phase de gestion du projet sécurité.			

Tableau 4-3 : Gestion des phases du projet SOA

Si CORAS assure la couverture la plus large, les autres méthodes font référence et devront être utilisées pour les phases du projet sécurité qu'elles couvrent.

Critère 3 : Concepts autour du risque

Ce critère regroupe trois sous-critères : la définition des biens à protéger, la définition des risques et la définition des besoins de sécurité.

3.1 Définition des biens à protéger

Dans notre travail, nous nous intéressons à une méthode qui prend en considération les biens essentiels ainsi que la dépendance entre eux. En d'autres termes, dans les architectures orientées services, nous ne pouvons pas isoler un processus métier ou un service des composants de l'infrastructure. Une sécurité globale doit porter sur l'ensemble des éléments (biens) interdépendants.

Les quatre méthodes étudiées permettent d'identifier les biens métier et technologiques à protéger ainsi que les dépendances entre eux. Les quatre méthodes sont adaptées à l'identification des biens à protéger dans une SOA mais à des phases différentes du projet. Toutefois, CORAS définit les biens d'une façon plus globale comme étant une ressource de valeur nécessitant une protection.

Cette analyse nous conduit à placer les quatre méthodes au même niveau de pertinence pour ce sous-critère.

3.2 Définition du risque

Ce sous-critère spécifie comment les méthodes définissent et identifient le risque. Dans notre travail, nous recherchons une méthode qui permet de définir les risques en rapport avec les biens à protéger et d'identifier ces risques en fonction des menaces et des vulnérabilités

EBIOS et CORAS définissent le risque comme étant un scénario combinant un événement redouté et un ou plusieurs scénarios de menaces. Dans OCTAVE, le risque est défini comme étant la possibilité de subir une perte ou un dégât, c'est-à-dire une situation dans laquelle une personne ou un phénomène naturel pourraient entraîner un impact négatif. Le risque est définit par une menace, une incertitude, et une conséquence telle que la divulgation, la modification, la perte ou la destruction ou bien l'interruption. Dans SNA, les risques sont définis en termes de conditions défavorables qui pourront nuire au système. Ces conditions défavorables sont le résultat d'attaques et d'intrusion sur le système.

En dépit de ces différences de définition, les quatre méthodes utilisent une même équation du risque : Risque = Menace x Vulnérabilité.

Il y a un donc consensus global sur la définition et l'identification du risque.

3.3 Définition des besoins de sécurité

Ce sous-critère définit les différents critères de sécurité qui caractérisent les besoins de sécurité pertinents par rapport au bien à protéger. Dans les méthodes de gestion de risque classiques, cette définition se base sur trois besoins de base : la confidentialité, l'intégrité et la disponibilité. Dans le cadre de notre travail, nous nous intéressons à une méthode qui permet de traiter :

- ✓ La classification de l'information qui représente l'attribution d'un niveau de sensibilité globale à l'information au niveau métier.
- ✓ La gestion des contrats de protection (création et administration) permettant de prendre en compte les contraintes liées aux partenariats entre entreprises.

Dans OCTAVE, les besoins de sécurité sont définis à partir des trois services de base de sécurité. Dans ses anciennes versions, EBIOS définissait les besoins de sécurité autour des trois services de base classiques. Dans sa version de 2010, une étape de définition des critères de sécurité a été ajoutée dans le module de l'établissement du contexte qui permet d'ajouter d'autres "critères de sécurité", tels que la preuve, l'imputabilité, l'auditabilité, la fiabilité, la traçabilité…

Dans SNA, les besoins de sécurité sont déterminés différemment en termes de survie du système en tenant compte des besoins de résistance, de reconnaissance et de recouvrement. Enfin, dans CORAS, outre les critères classiques, la définition des besoins de sécurité est étendue pour intégrer la confiance, la réputation, la conformité aux lois, etc. (besoins 'cachés' ou pris comme mesure d'impact dans les autres méthodes)

La définition des besoins de sécurité a donc été enrichie dans les dernières versions des méthodes. Les méthodes CORAS et EBIOS semblent mieux cadrer avec notre cahier des charges.

Critère 4 : Flexibilité de la méthode

Ce critère met en évidence la flexibilité de la méthode. En effet, dans un environnement de services, nous recherchons une méthode qui nous permettra de mettre à jour les informations du contexte et de maximiser la sécurité dans une démarche continue d'amélioration.
Selon leur démarche, OCTAVE, SNA et CORAS nous permettent de revenir sur les activités des différents modules de façon à optimiser la sécurité. Même si EBIOS a été remarquablement amélioré dans sa dernière version, sa flexibilité reste limitée par rapport aux autres méthodes.

Critère 5 : Analystes

Ce critère identifie les personnes qui appliquent la méthode. Certaines méthodes nécessitent de faire appel à des experts de sécurité externes tandis que d'autres peuvent être menées par le personnel de l'entreprise en leur fournissant des directives, des bonnes pratiques et des outils pour bien mener l'étude.

La méthode OCTAVE est 'auto-dirigée' et fournit donc les guides nécessaires pour que les acteurs de l'entreprise puissent mener l'étude de gestion des risques. La méthode CORAS est une méthode simple à appliquer. Toutefois, c'est une méthode généraliste et manque de support dans la sécurisation des systèmes d'information. Par conséquent, elle ne peut être mise en œuvre que par un expert sécurité (interne ou externe). EBIOS est une méthode assez complexe, nécessite beaucoup de compétences et devra être mise en œuvre par un expert sécurité (interne ou externe). Enfin, SNA devra être appliquée avec l'assistance d'une équipe externe.

OCTAVE occupe donc le premier rang pour ce critère suivi par CORAS et EBIOS et enfin SNA.

Critère 6 : Aide à la réalisation

Ce critère regroupe quatre sous-critères :

- La documentation recommandée et la méthode de collecte d'information
- Le support pour l'identification des menaces
- Le support pour l'identification des vulnérabilités
- Les outils de support.

6.1 Documentation recommandée et méthode de collecte d'information

Ce sous-critère consiste à identifier les documents nécessaires pour entamer l'étude de gestion des risques puisque cette documentation contribue à la création des profils des biens à protéger au cours des premières étapes. L'exhaustivité de ces listes constitue un critère important pour faciliter le travail des analystes.

EBIOS recommande de collecter les données sur le contexte de l'étude (documents stratégiques, documents relatifs aux missions, attributions et organisation, etc.) ainsi que les données concernant le contexte de la gestion des risques et la structure de travail.

OCTAVE encourage à collecter le plus possible d'informations sur les actifs (les biens) afin de faciliter le processus : organigramme, inventaire des systèmes informatiques, inventaire des logiciels, architecture réseau, documentation sur la politique de sécurité existante, documentation sur les formations, configuration des systèmes et des routeurs, données d'audit, outils de configuration, liste des évaluations de sécurité disponibles.
SNA recommande la collecte d'une liste exhaustive de documents classifiés en quatre catégories : missions métier, exigences fonctionnelles, environnement d'utilisation et utilisateurs et enfin architecture.
CORAS (plus généraliste) ne précise pas les documents à collecter et laisse ce choix aux analystes qui mènent l'étude.
Dans toutes ces méthodes, les informations sont collectées au cours de réunions (workshops et brainstorming) entre les différents acteurs.
SNA et OCTAVE sont les méthodes les plus riches en termes de précision sur la documentation requise. Ensuite viennent les méthodes EBIOS et CORAS.

6.2 Support à l'identification des menaces

Ce sous-critère précise comment les méthodes permettent d'identifier les menaces. Dans le cadre de notre travail, nous nous intéressons à une méthode qui donne des directives et des techniques claires pour assister l'analyste et optimiser le temps accordé à cette tâche.
Afin d'identifier les menaces, EBIOS procure des bases de connaissances qui fournissent une liste générique de menaces et des sources de menaces. Ces bases peuvent être utilisées et ajustées selon le contexte. La méthode OCTAVE repose sur des séances de « brainstorming » permettant d'identifier les menaces. La méthode fournit des arbres de menaces pour chaque catégorie de source de menace. Cette technique permet à l'analyste de définir le bien à protéger, l'accès utilisé, les acteurs, la motivation et les conséquences. Pour chaque bien, les menaces sont regroupées en catégories selon leur source (acteur humain, naturel, environnemental) et leur cause (accidentelle, délibérée). Pour identifier les menaces, SNA définit plusieurs profils d'attaquant et / ou d'attaque : attaque ludique, fait d'un employé mécontent, fait d'activiste (raisons éthiques ou politiques), espionnage industriel, fait d'états. En outre, l'attaque peut également être caractérisée en termes de ressources, de temps, d'objectif de l'attaquant et de la source d'attaque (réseau local ou réseau Internet). De plus, SNA utilise la base de connaissances du CERT pour la reconnaissance des attaques grâce aux signatures d'intrusions. Enfin, CORAS intègre des séances de brainstorming pour déterminer les menaces à retenir. La méthode fournit des diagrammes de menaces et un langage de modélisation. Toutefois, étant donné que la méthode ne procure pas de base de connaissances sur les menaces, elle repose sur l'expertise des analystes pour la détermination de ces menaces.
Les méthodes OCTAVE et SNA ont une documentation sur les attaques plus riche que celles apportées par les autres méthodes. Ensuite, la méthode EBIOS fournit une documentation générique adaptée pour la conception des systèmes. Par contre, la méthode CORAS est assez générique et manque de support sur ce sujet.

6.3 Support à l'identification des vulnérabilités

Ce sous-critère précise comment les méthodes permettent d'identifier les vulnérabilités c'est-à-dire quels sont les éléments destinés à assister l'analyste et optimiser le temps accordé à cette tâche.

EBIOS fournit des bases de connaissances génériques pour identifier les vulnérabilités et les menaces associées. Les vulnérabilités exploitables sont des caractéristiques génériques des biens supports. Les vulnérabilités représentent les failles de sécurité tant au niveau organisationnel que technique. Toutefois, ces bases de connaissances ne sont pas exhaustives et devraient systématiquement être adaptées pour que leur niveau de détail soit approprié au sujet étudié et à l'objectif poursuivi.

Dans la méthode OCTAVE, l'identification des vulnérabilités est réalisée dans des séances de « brainstorming » principalement pour ce qui concerne les vulnérabilités organisationnelles (mauvaise gestion des droits d'accès ou pratiques du personnel). Concernant les vulnérabilités techniques (i.e. liées à l'infrastructure), les experts techniques disposent de conseils, indications et d'un ensemble d'outils de test, d'audit et de scanneurs de vulnérabilités.

Dans la méthode SNA, les experts de sécurité externes établissent des séances de collecte d'information pour identifier les vulnérabilités organisationnelles et utilisent les bases de connaissances de CERT pour identifier les vulnérabilités au niveau de l'infrastructure.

Si les trois premières méthodes fournissent des bases de connaissances, en revanche CORAS n'utilise que les diagrammes de menaces réalisés par les analystes (leur niveau d'expertise est donc essentiel) pour déterminer les vulnérabilités dans des séances de brainstorming.

Comme dans le cadre de l'identification des menaces, l'identification des vulnérabilités est liée à la phase du projet que la méthode couvre. OCTAVE et SNA permettent de mieux gérer cette tâche durant la phase d'exécution et font référence à des bases de connaissance exhaustives. EBIOS fournit une base de vulnérabilité générique à utiliser dans la conception des systèmes. En revanche, CORAS manque totalement de support à ce sujet ce qui est une réelle faiblesse.

6.4 Outil support

Ce sous-critère met en évidence l'outil ou le langage de modélisation, procuré par la méthode pour faciliter l'étude de gestion des risques et fournir le plus de supports possibles à l'analyste.

EBIOS fournit un logiciel libre qui facilite l'application de la méthode. OCTAVE fournit des fiches de collecte d'information sur les biens, les secteurs de préoccupation, les exigences de sécurité, les enquêtes sur les pratiques existantes, les profils de sécurité et les stratégies de protection. SNA fournit des fiches de profils et de patterns d'attaques. CORAS fournit un langage de modélisation permettant de structurer l'information recueillie et d'aider les participants à définir un niveau de description approprié (compromis entre abstraction et détail). Ce langage facilite la communication entre les participants puisqu'il crée un vocabulaire commun, fournit un format standard pour documenter l'analyse des risques et contribue à mettre en évidence les résultats les plus importants.

4.4.6.2 Synthèse

Dans l'objectif de gérer la sécurité dans un contexte SOA, nous recherchons une méthode de gestion des risques qui :

- ✓ permet de gérer les différentes phases du cycle de vie des services,
- ✓ prend à la fois les dimensions métier et technologiques dans la gestion des risques,
- ✓ présente un niveau de flexibilité élevé,
- ✓ pourra être menée par les responsables techniques de l'entreprise,
- ✓ fournit un support exhaustif dans l'aide à la réalisation.

D'après notre étude des différentes méthodes, nous constatons qu'il n'existe aucune méthode parfaitement adaptée au contexte SOA. En effet, nous avons trouvé qu'elles sont complémentaires et peuvent être utilisées à différents phases dans la gestion des risques dans ce contexte.

Dans le Tableau 4-4, nous synthétisons l'analyse effectuée. Nous définissons une échelle pour évaluer l'intérêt de la méthode par rapport à la gestion des risques dans un contexte de services :

+ : Intérêt mineur

++ : Intérêt moyen

+++ : Intérêt majeur

Critère de sélection	Les méthodes de gestion des risques / Sous-critères	EBIOS	OCTAVE	SNA	CORAS
1. *Objectif de la méthode*	*X*	+++	+++	+++	+++
2. *Phases de gestion du projet sécurité*	*X*	++	+	++	+++
3. *Concepts liés au risque*	*Définition du bien de l'organisme*	+++	+++	+++	+++
	Définition du risque	+++	+++	+++	+++
	Définition des besoins de sécurité	+++	++	++	+++
4. *Flexibilité de la méthode*	*X*	+	+++	+++	+++
5. *Analystes*	*X*	++	+++	+	++
6. *Aide à la réalisation*	*Documentation recommandée et méthode de collecte d'information*	++	+++	+++	++
	Support à l'identification des menaces	++	+++	+++	+
	Support à l'identification des vulnérabilités	++	+++	+++	+
	Outil support	++	++	+	+++

Tableau 4-4 : Synthèse – Intérêt par rapport à la gestion des risques dans un contexte SOA

La Figure 4-13 illustre graphiquement les résultats de l'analyse et nous permet de revenir sur les points forts de chaque méthode au besoin. En particulier, nous utiliserons :

- ✓ Le langage de modélisation de la méthode CORAS pour assister les analystes dans l'identification des risques.
- ✓ La documentation et les catalogues des méthodes EBIOS et OCTAVE pour identifier les éléments liés au risques (menaces et vulnérabilités métier et technologique)

En outre, nous aborderons l'identification des risques par une approche de modélisation des menaces (approche OCTAVE, CORAS, SNA) qui assurera plus de flexibilité, nous mènerons notre étude dans des ateliers de 'brainstorming' en présence des responsables métier et technique (approche OCTAVE et CORAS) ce qui permettra de prendre en compte les dimensions métier et technique de la sécurité et enfin, nous ferons référence à des ressources complémentaires sur l'identification des menaces et vulnérabilités portant sur les services.

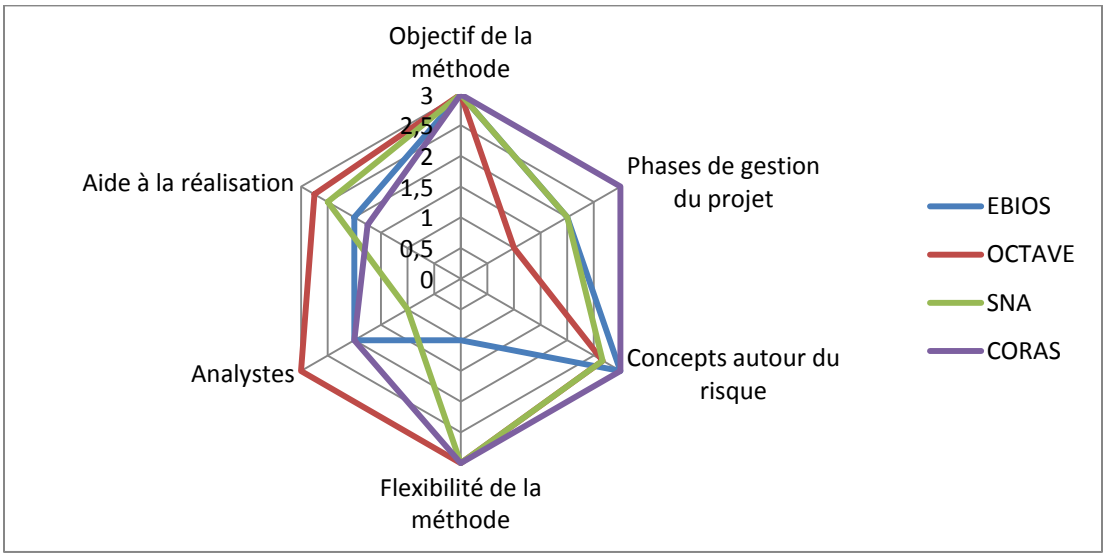


Figure 4-13 : Graphe de couverture

4.5 Conclusion

Comme nous l'avons vu dans le deuxième chapitre de l'état de l'art, nous considérons la SOA comme un style d'architecture et non pas une technologie. La gestion de la sécurité dans un environnement de services est relative aux besoins et aux objectifs de sécurité. Cette gestion ne doit pas se limiter aux aspects technologiques mais doit forcément prendre en compte les aspects métier et organisationnels. La gestion des risques permet de répondre à cet objectif en définissant les mesures de sécurité selon les besoins et les objectifs métier.

Suite à nos recherches bibliographiques, nous avons constaté que les trois points suivants ne sont pas évoqués dans la littérature abordée :

- Un modèle conceptuel de service sécurisé permettant de développer des patrons de sécurité et de prendre en compte les dimensions métier et technologique de la sécurité.
- Un consensus global sur la définition des biens à protéger dans un environnement de services. En effet, la définition des biens à protéger est parmi les aspects les plus importants dans une étude de gestion des risques. Cette définition permettra de répondre à une question primordiale: Que faut-il protéger dans une SOA ?
- Une méthodologie de conception d'une SOA sécurisée, qui détaille la démarche à suivre pour concevoir une SOA sécurisée. Cette méthodologie permettra de répondre aux questions :
  Comment faire pour concevoir une SOA sécurisée ?
  Quels sont les outils à utiliser et les bonnes pratiques à suivre ?

Partant de ces constats, nous établissons la feuille de route de notre travail. Notre contribution axée sur la sécurisation des services dans les différentes phases de leur cycle de vie, sera exposée dans les chapitres suivants.

PARTIE II : CONTRIBUTION

Chapitre 5. Modèles et concepts d'une SOA sécurisée

Résumé

Ce chapitre porte sur la phase de préparation du cycle de vie des services. Nous développons un modèle de services sécurisés permettant de définir des patrons de sécurité et un modèle de classification des éléments essentiels (biens à protéger). Enfin, nous développons une Ontologie de Conception de SOA sécurisée (OCSS) *pour définir les concepts associés à ces éléments.*

Sommaire

5.1 Introduction

Dans le monde des technologies de l'information et de la communication, la sécurité a souvent été considérée sous un angle purement technique. En fait, les mesures de sécurité sont souvent mises en place pour répondre à des problèmes spécifiques après la mise en œuvre des applications. Cette démarche qui néglige la dimension organisationnelle (les objectifs de sécurité (ex : gestion des droits d'accès) et risques associés à ce niveau (ex : indisponibilité d'un partenaire)) s'avère inadéquate dans un écosystème de services où les frontières du système d'information sont étendues, dépassant celles de l'entreprise et où l'environnement est dynamique (de nouveaux partenaires ayant potentiellement une vue et des besoins différents de la sécurité peuvent entrer en jeu)

Dans les chapitres précédents, nous avons présenté les technologies liées aux services, les méthodes de gestion des risques et montré les limites dans la gestion de la sécurité des SOA. Les modèles et architectures de référence du domaine des services ont selon nous sous-estimé l'identification des risques métier et organisationnels et la définition des biens à protéger. De même, les méthodologies de développement des SOA se focalisent sur les dimensions fonctionnelles et prêtent moins d'attention à la sécurité.
Pour pallier ces limites, nous proposons d'aborder la gestion de la sécurité par une approche de gestion des risques adaptée au cadre SOA. Nous proposons un cadre méthodologique de gestion de la sécurité portant sur les phases préparation, conception, exécution et supervision du cycle de vie des services (Figure 5-1)

Pour permettre aux responsables métier et techniques d'aborder efficacement la conception du système d'information sécurisé dans une stratégie SOA, il importe qu'ils partagent un même corpus de concept (métier, technique et sécurité) : c'est ce que nous désignons par le terme de 'phase de préparation'. Pour répondre à ce besoin, nous proposons un *modèle conceptuel de service sécurisé*, *une classification des biens* à protéger et *une ontologie de ces biens.*

La phase de conception est dévolue à la modélisation des processus métier ainsi qu'à l'identification et la spécification des services. Il faut donc se focaliser sur deux axes complémentaires : la conception de services réutilisables et la spécification des besoins de sécurité (prenant en compte l'évolution de l'architecture et les changements organisationnels). Nous proposons alors, *une méthodologie de conception d'une SOA sécurisée* permettant de remplir cet objectif. En outre, nous proposons d'annoter les services par des paramètres de sécurité qui pourront être intégré lors de la sélection des services dans la phase d'exécution.

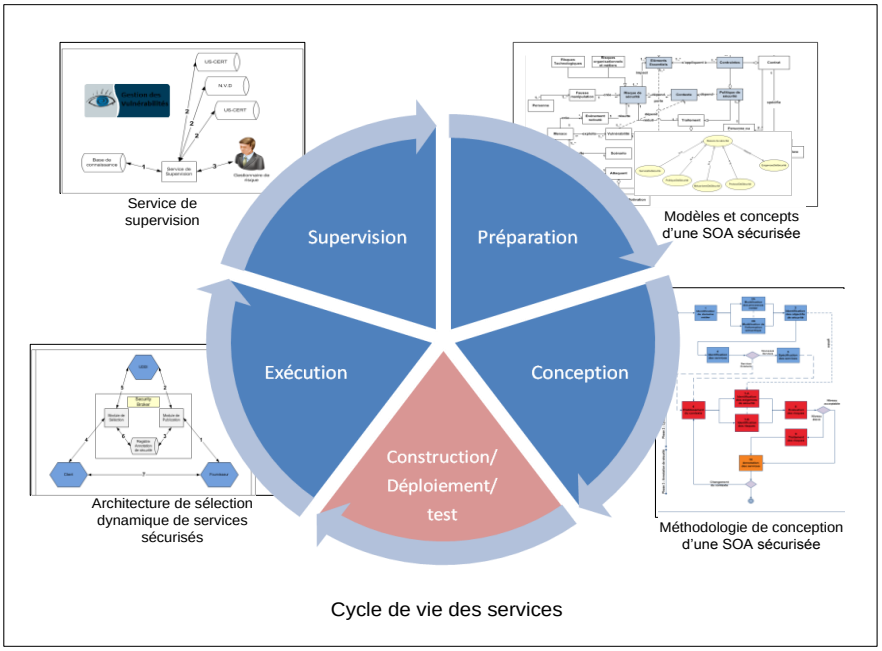


Figure 5-1: Portée de notre contribution

Les phases de construction, de déploiement et de test du cycle de vie des services permettent de mettre en œuvre et déployer les services. La sécurité est donc directement liée aux technologies utilisées. La sélection de ces composants technologique est réalisée en fonction des exigences issues de la conception. Nous ne détaillerons pas ces étapes dans notre recherche car elles sont traitées par une autre thèse développée dans l'équipe (Francis Ouedraogo) [104] : une stratégie de transformation de modèles permet de générer les politiques correspondant au contexte de déploiement.

Une fois déployés, les services se trouvent dans un environnement opérationnel et vont être exploités par d'autres applications et services, c'est la phase d'exécution. Outre l'exécution des processus définis lors de l'étape de conception, les services peuvent aussi être invoqués pour mettre en œuvre des processus ad-hoc en sélectionnant les services au fil de l'exécution des différentes tâches. Pour répondre à ce besoin, nous proposons *une architecture de sélection dynamique de services sécurisés* permettant de composer des processus métier sécurisés utilisant l'annotation des services.

Enfin, la phase de supervision doit permettre de gérer les changements du contexte. Dans cette phase, nous nous focalisons sur la gestion des vulnérabilités des composants de l'infrastructure et nous proposons un service de sécurité permettant de gérer et de superviser ces vulnérabilités.

Pour supporter ces différentes phases, nous proposons un cadre méthodologique intégrant les besoins et la gestion de sécurité dans ces différentes phases. Dans ce chapitre portant sur la phase de préparation du cycle de vie des services, nous développons :

- ✓ Un modèle conceptuel de service intégrant service, politiques de sécurité et risque. Ce modèle sera utilisé pour définir des patterns de sécurité.
- ✓ Une classification *des éléments essentiels* à protéger selon les plans d'abstraction.
- ✓ Une Ontologie de Conception de SOA Sécurisée (OCSS) permettant de définir les concepts liés aux biens essentiels à protéger puisqu'une étude de gestion des risques ne peut se faire qu'après la définition du bien à protéger.

5.2 Modèle conceptuel d'un service sécurisé

Plusieurs modèles conceptuels de service ont été définis (voir chapitre 2 section 2.4.2) pour améliorer l'alignement métier et applicatif dans la conception des architectures orientées services. Toutefois, ces modèles n'ont pas été développés pour supporter la conception de SOA sécurisée et prêtent donc moins d'attention à l'identification des risques métier et organisationnels pesant sur ces architectures. Afin de combler ce manque, nous proposons un modèle conceptuel. Ce modèle est construit à partir de trois modèles distincts :

- ✓ un modèle de service qui met en évidence les éléments essentiels associés au service,
- ✓ un modèle de risque qui montre la prise en compte des différents types de risques dans la sécurité,
- ✓ un modèle de politiques de sécurité faisant le lien entre les objectifs et les mesures de sécurité à mettre en place,

et d'un ensemble de relations entre ces modèles.

En dehors de son application dans le développement de patrons de sécurité, ce modèle doit aussi permettre de :

- ✓ Fournir des informations permettant de prendre en compte, dès la première phase du cycle de vie des services, la sécurité et les risques.
- ✓ Faciliter la communication entre les responsables du domaine métier et ceux du domaine technique, en tenant compte de la sécurité des processus métier, des services et des infrastructures.
- ✓ Montrer l'importance des éléments communs aux trois modèles.

5.2.1 Modèle conceptuel de service

Dans le contexte des architectures orientées services, la composition des services offre une nouvelle perspective de développement d'applications à partir de services existants, afin de satisfaire les besoins des utilisateurs. Nous rappelons que le service est un module logiciel affichant ses fonctionnalités par l'intermédiaire d'une interface. Plusieurs types de services peuvent être identifiés tels que les services centrés sur les tâches, les services de données (services permettant de gérer les objets métier ex : devis) et les services utilitaires (service de support ex : copie, soumission, transfert, etc.)

Pour élaborer le modèle conceptuel de service, nous introduisons le concept d'***élément essentiel*** c'est-à-dire un élément duquel le service dépend pour sa conception et son déploiement. La sécurité de ces éléments devra donc être maîtrisée pour assurer la sécurisation du service. Ces éléments essentiels peuvent être :

1. des *éléments métier* qui décrivent le cadre métier, organisationnel et légal. Ce cadre est constitué de l'ensemble des partenaires, des acteurs et leurs rôles, des accords du niveau de protection, des préférences de sécurité, des lois et des obligations légales,
   1.1. des processus métier ainsi que des différentes activités qui les composent : activités automatiques (pouvant être informatisées et gérées par un système de gestion de workflow), semi-automatiques (ou interactives : ces activités automatisées nécessitent une intervention humaine) et manuelles. Une activité décrit un fragment de travail, qui constitue une étape logique à l'intérieur d'un processus. Les activités automatiques ou semi-automatiques sont mises en œuvre grâce à des services métier,
   1.2. des documents métier (synonyme d'objets métier) associé à la représentation sémantique des données à échanger au sein d'un processus métier. Un document métier spécifie l'information qui 'traverse' les activités du processus métier et est utilisé par les activités du processus métier implémentées par des services. Ce document change d'état selon l'avancement du processus.

   Nous faisons la distinction entre documents métier et données pour y inclure la sémantique associée au domaine métier en définissant un modèle de données structuré pour définir des documents métier et optimiser l'interopérabilité. Prenons par exemple un processus métier : « Demande de devis », le processus manipule le document métier « Devis » qui doit être défini à partir d'un modèle de données structuré afin qu'il soit compréhensible par l'ensemble des acteurs concernés.
2. des services (atomiques ou composites) qui implémentent les opérations des services métier.
3. des données pouvant être stockées ou échangées par les services de données.
4. des messages c'est-à-dire les unités de communication entre les services contenant les documents métier ou données nécessaires pour exécuter une tâche particulière. Les documents métier/données sont encapsulés dans les 'corps des messages'.
5. des assertions de sécurité et la qualité de protection (QoP) représentant les exigences et les mesures de protection des services. Nous désignons ces deux éléments (assertions et QoP) comme étant des éléments essentiels, notre approche de gestion étant récursive.
6. des éléments de l'infrastructure associés aux composants de l'infrastructure hébergeant les services comme les logiciels d'hébergement, les systèmes d'exploitation, les équipements utilisés pour stocker l'information (serveur de fichier, serveur de base de données) ainsi que les éléments de support qui sont liés directement à l'utilisation des services tel que la connexion au réseau Internet.

Ces éléments essentiels seront définis dans une ontologie (Voir section 5.4.1)

Dans notre modèle, le processus métier est formé de services métier (activités automatiques et semi-automatiques) et d'activités manuelles. Chaque service métier offre des opérations qui sont réalisées par un ou plusieurs services. Un service métier échange des messages encapsulant des documents métier ou des données et est mise en œuvre par des services utilisant des composants de l'infrastructure.

Pour gérer les relations entre fournisseur et consommateur (trait pointillé), nous ajoutons la classe contrat. Le contrat spécifie l'interface cachant l'implémentation du service et spécifiant les opérations, les assertions de sécurité et la qualité de protection du service. Le contrat inclut donc la description fonctionnelle et non fonctionnelle du service.

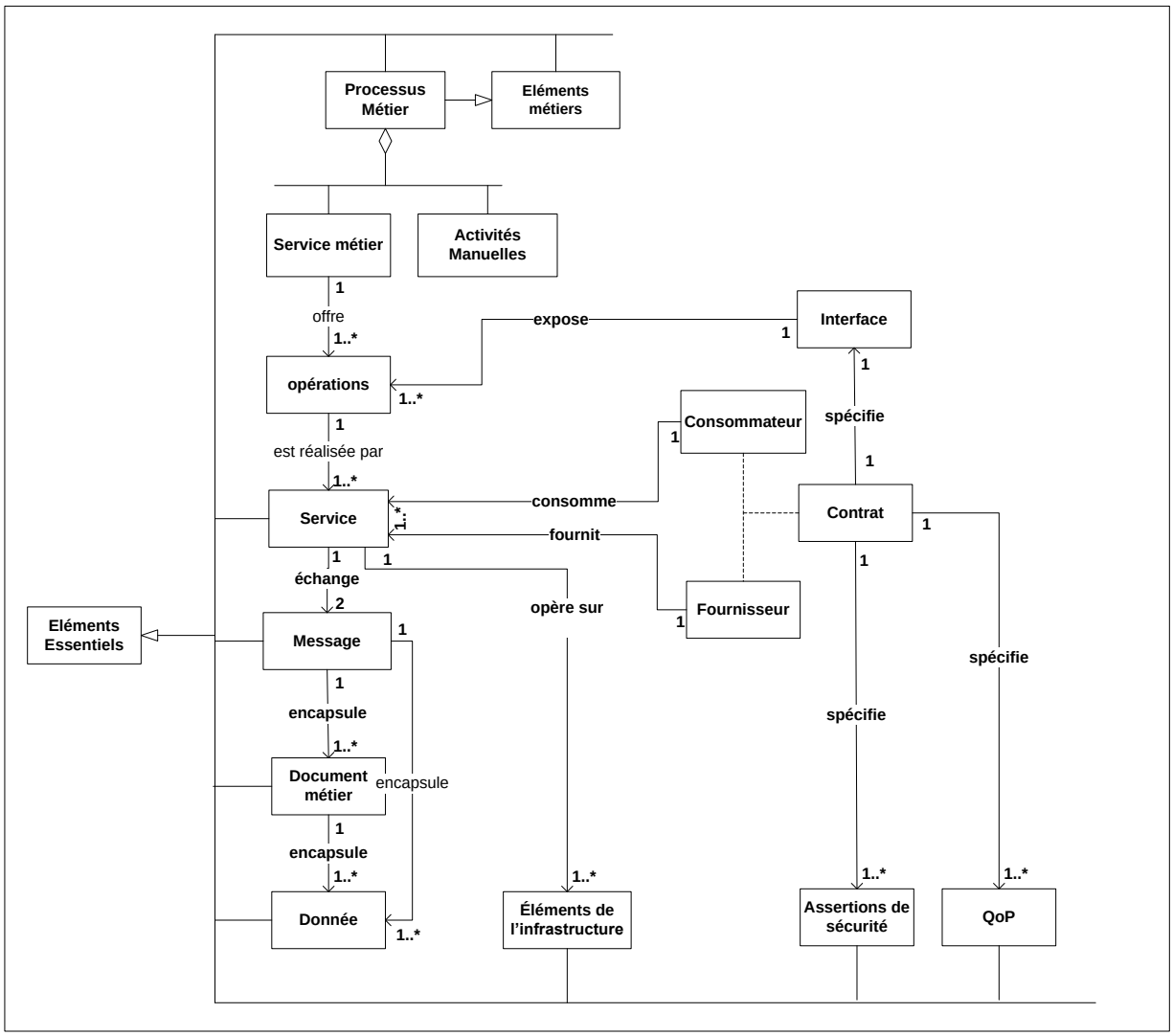


Figure 5-2 : Modèle conceptuel de service

5.2.2 Modèle conceptuel de politique de sécurité

Un service est membre d'un écosystème. Il présente des capacités et a un rôle bien défini, des responsabilités et des droits. L'évolution continue dans les environnements de service exige de nouvelles stratégies pour sécuriser les ressources. De même, les solutions de sécurité doivent être facilement déployables, utilisables et adaptables aux changements. Une politique permet de définir les conditions d'utilisation d'un service ainsi que ses exigences.

La Figure 5-3 présente un modèle de politique issu de l'architecture de référence de l'OASIS. Dans ce modèle, la politique est définie par des contraintes sur l'utilisation, le déploiement ou la description d'une ressource incluant les droits (le droit d'un participant d'effectuer une action), les permissions et les obligations spécifiant l'exigence pesant sur un participant d'effectuer une action ou de maintenir un état bien défini [22].

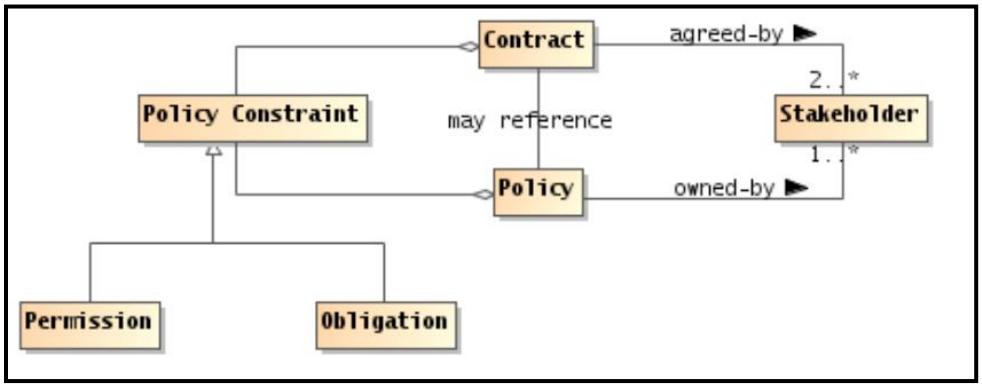


Figure 5-3 : Modèle de Politique [5] p.56

Nous développons un modèle de politique de sécurité en créant un lien entre les objectifs de sécurité à atteindre, les mesures de sécurité et les éléments essentiels en couvrant à la fois les risques métier, organisationnel et technologique. En d'autres termes, la sécurité dans notre travail ne se limite pas à la sécurisation des données transmises ou à la sécurisation des données stockées et donc à une vision technologique de la sécurité puisque nous intégrons la classification du patrimoine informationnel, la classification des partenaires et des acteurs et la gestion des droits d'accès. La politique de sécurité répond à la fois aux objectifs de sécurité métier (classification de l'information et utilisation des ressources) et technologique (assertions en référence aux services de sécurité à mettre en œuvre)

Pour élaborer notre modèle, nous avons défini trois objectifs de sécurité génériques permettant de couvrir les aspects métier de la sécurité dans un écosystème de service.

1. La gestion des éléments essentiels métier représente la création et l'administration des contrats, des droits et des obligations comme la gestion des obligations des consommateurs, la gestion des droits d'accès aux documents métier, la gestion des accords sur la qualité de la

protection offerte lors de la prestation de service, la gestion des contrats de partenariat (Ces concepts sont définis dans la partie OCSS p. 113).

2. La classification des processus métier et des documents métier représente l'attribution d'un niveau de sensibilité aux processus métier et aux documents métier, en adoptant l'échelle discrétisée proposée par [105] : noir pour les données privées, gris pour les données qui nécessitent un niveau de protection standard et blanc pour les données publiques.
3. La confiance est associée à la classification des acteurs et à l'établissement d'un réseau de partage se basant sur des politiques de sécurité spécifiques. Le réseau de partage permet la propagation des identités entre différents domaines et la gestion des échanges entre les acteurs.

La Figure 5-4 présente notre modèle conceptuel de politique de sécurité et dans lequel nous regroupons les objectifs de sécurité métier, technologique et de support.

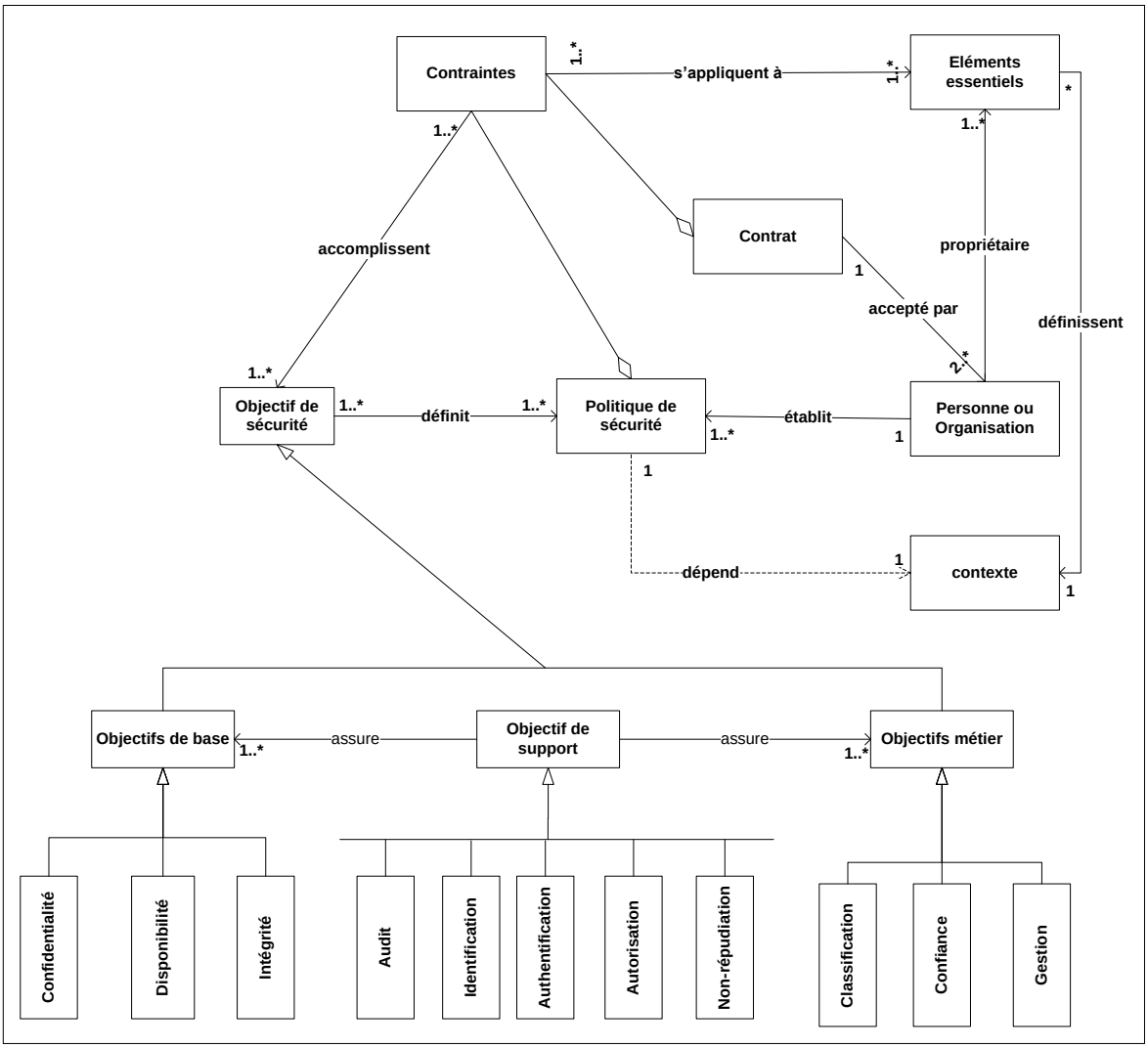


Figure 5-4 : Modèle conceptuel de politique de sécurité

- ✓ Les objectifs de sécurité seront atteints en respectant les contraintes imposées par le contrat et par les politiques de sécurité.
- ✓ Les contraintes s'appliquent aux éléments essentiels et permettent d'atteindre les objectifs de sécurité.
- ✓ La politique de sécurité dépend du contexte. Ce dernier représente l'ensemble des éléments essentiels identifiés au moment de la conception.

Le contexte est d'une extrême importance pour la définition de politiques de sécurité adaptées (pour éviter une sur ou sous protection). A ce titre, il fera l'objet d'une étude approfondie dans le chapitre suivant.

Nous présentons dans la partie suivante le modèle de risques relatifs aux différents niveaux d'abstraction, ce qui nous conduit à identifier des risques métier et organisationnels et des risques technologiques relatifs aux services et à l'infrastructure. Ce modèle prend également en compte la dynamique des services, étant donné que les risques varient selon le contexte.

5.2.3 Modèle conceptuel de risque de sécurité

Comme nous l'avons vu dans le chapitre 3, la gestion des risques est un processus qui se place au cœur de la gestion de la sécurité. La gestion des risques s'appuie sur l'analyse des menaces et des vulnérabilités pour déterminer les contre-mesures appropriées. Ce processus est un travail complexe vu la diversité des méthodologies et des domaines d'application. De plus, la gestion des risques est plus complexe dans les environnements distribués et dynamiques tels que les écosystèmes de services où les échanges se font entre différents partenaires.

La Figure 5-5 illustre un modèle de risque élaboré par le standard Common Criteria [106]. Dans ce standard, le risque résulte des menaces et des vulnérabilités associées aux biens à protéger dans une vision 'mono-contexte'. Ce modèle, qui porte sur la certification d'éléments technologiques, a été adopté dans le travail de P. Herrmann [107] pour élaborer la plateforme MoSSBP (Modeling Security Semantics of Business Processes) pour la modélisation et l'analyse des exigences de sécurité des processus métier. Toutefois, comme dans les Common Criteria, MoSSBP ne distingue pas les risques organisationnels des risques technologiques. De même, le modèle « Information System Security Risk Management (ISSRM) » [108] est aussi adapté à des environnements statiques dans lesquels les contextes de conception ou d'exécution ne changent pas.

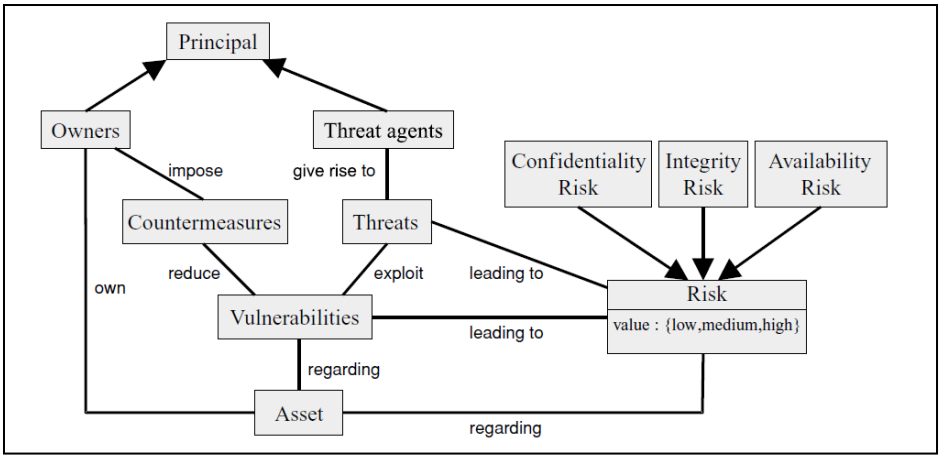


Figure 5-5 : Modèle de risques - Common Criteria

Pour fournir un modèle des risques de sécurité adapté à l'environnement de services (Figure 5-6), nous avons intégré les concepts suivants:

- ✓ Identification des différents *types de risque* portant sur les éléments essentiels métier, organisationnels et technologiques du système d'information :
  - Un processus métier est un élément essentiel métier, son indisponibilité est un risque métier.
  - Les droits d'accès représentent un élément essentiel organisationnel, l'altération de ces droits est un risque organisationnel.
  - Un routeur est un composant de l'infrastructure, son indisponibilité est un risque technologique.

  Cette séparation en différents types de risques permet de mieux les identifier dans des séances de brainstorming entre les responsables métier et technique. L'intégration de relations entre risques permet d'inclure des chaînes de causalité dans le modèle et donc faciliter le travail d'inventaire. Par exemple, un risque métier peut résulter d'un problème technique (l'indisponibilité d'un routeur peut entrainer l'indisponibilité d'un processus métier) ou l'inverse (une attaque 'métier' de déni de service sur un processus métier peut entrainer de nombreux échanges conduisant à la congestion des éléments de l'infrastructure et donc rendre indisponible un routeur)
- ✓ Association des *risques* au contexte en précisant les éléments essentiels formant le contexte de conception ou d'exécution. Par exemple, le niveau de risque diffère pour un service hébergé au sein de l'entreprise ou externalisé.
- ✓ Définition d'une catégorie de *traitement des risques* permettant soit l'évitement, le transfert, la prise ou la réduction de risque. Ces concepts seront détaillés dans la section 6.2.2, étape 9.
- ✓ Création d'une relation entre le risque et les *mesures de sécurité* c'est-à-dire les traitements qui réduisent les risques portant sur les éléments essentiels. Une mesure peut être une *politique de sécurité*, un *protocole de sécurité*, un *mécanisme de sécurité* ou un *service de sécurité*. (Voir section 5.4.2)

- ✓ Définition des *patrons de sécurité* à partir des patrons de menaces et de vulnérabilités génériques pour réduire les risques correspondant au contexte.
- ✓ Identification des *sources de risques* (attaques délibérées ou fausses manipulations). Ces sources créent des menaces, exploitant les vulnérabilités et menant à des évènements redoutés.

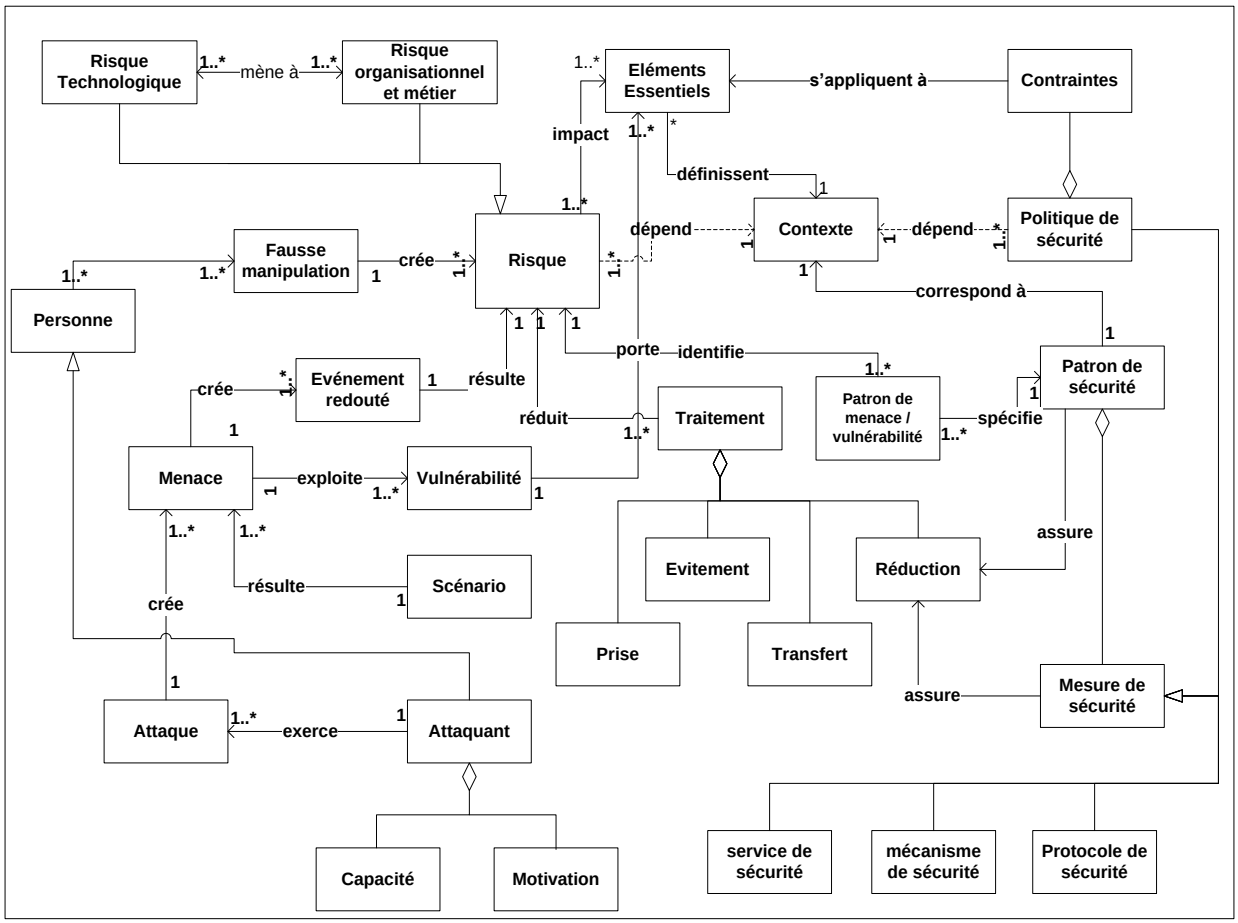


Figure 5-6 : Modèle conceptuel de risque de sécurité

5.2.4 Modèle conceptuel de service sécurisé

Dans les trois modèles élaborés précédemment, nous avons identifié les éléments communs suivants : élément essentiel, risque, contexte, contrainte et mesure de sécurité. Nous nous basons sur ces éléments pour créer les associations entre les différents modèles pour construire le modèle conceptuel de service sécurisé (Figure 5-7) :

- ✓ Le service n'est pas isolé : sa sécurité dépend des éléments essentiels métier et technologiques qui forment le contexte.
- ✓ L'identification des risques se fait en identifiant les événements redoutés portant sur les éléments essentiels en se référant à des patrons de menaces et de vulnérabilités.

- ✓ Le traitement des risques est défini en termes de prise, évitement, transfert ou réduction du risque considéré. Ceci conduit à définir des mesures déclinées en politique, protocole, mécanisme ou service aux niveaux métier et technologique.
- ✓ Des patrons de sécurité spécifiés à partir de patrons de menaces et de vulnérabilités peuvent faciliter le traitement des risques et permettre de répondre à des problèmes de sécurité spécifiques selon le contexte.

La création des instances de ce modèle se fait en plusieurs étapes :

1- Identification des éléments essentiels formant le contexte (concepts encadrés en bleu). Tout d'abord, nous identifions les éléments métier qui décrivent le cadre métier, organisationnel et légal en rapport avec les processus métier. L'identification des éléments essentiels se fait en utilisant les processus métier pour identifier les services qui les composent. A leur tour, les services sont utilisés pour identifier les composants de l'infrastructure qui les hébergent.
2- Identification des risques qui portent sur les éléments essentiels (concepts encadrés en rouge)
3- Traitement des risques pour répondre au mieux aux objectifs de sécurité (concepts encadrés en vert)

Pour illustrer l'utilisation de notre modèle, nous proposons l'exemple suivant : une agence d'organisation de voyage offre en collaboration avec ses partenaires (hôtels et compagnies aériennes) des services de réservation en termes de choix de destination et d'hôtel. Si nous nous focalisons sur le processus métier 'réservation de voyage en ligne', ce processus se compose des services de réservations d'hôtels, réservation de vols et de paiement en ligne. Ces services, hébergés au niveau de l'infrastructure technique de l'agence sont accessibles via un portail web et forment le contexte avec les éléments essentiels métier (partenaires, processus métier, etc.) et les composants de l'infrastructure (serveurs d'applications, serveurs web, etc.)

Dans notre approche de gestion de la sécurité, nous identifions les risques métier et technologiques portant sur les éléments essentiels du contexte. L'indisponibilité du processus métier 'réservation de voyage' peut résulter de différents événements redoutés causé par l'indisponibilité :

- ✓ des partenaires (les partenaires privilégient d'autres agences de voyage ou arrêtent de fournir des services spécifiques)
- ✓ des services (suite à des attaques de déni de service XML-DOS)
- ✓ des composants de l'infrastructure (suite à des attaques de déni de service).

Le risque 'inaccessibilité du processus métier réservation de voyage' devra être traité en mettant en place des mesures de sécurité permettant de réduire l'impact et/ou la probabilité d'occurrence des événements redoutés. Par exemple, nous mettons en place :

- ✓ des contrats spécifiant la disponibilité des services entre les partenaires (niveau métier)
- ✓ le patron de sécurité 'Message inspector gateway' [59] permettant d'intercepter le trafic et de filtrer les requêtes (niveau service)
- ✓ des systèmes de filtrage (Pare-feu, routeurs, etc.,) au niveau des composants de l'infrastructure.

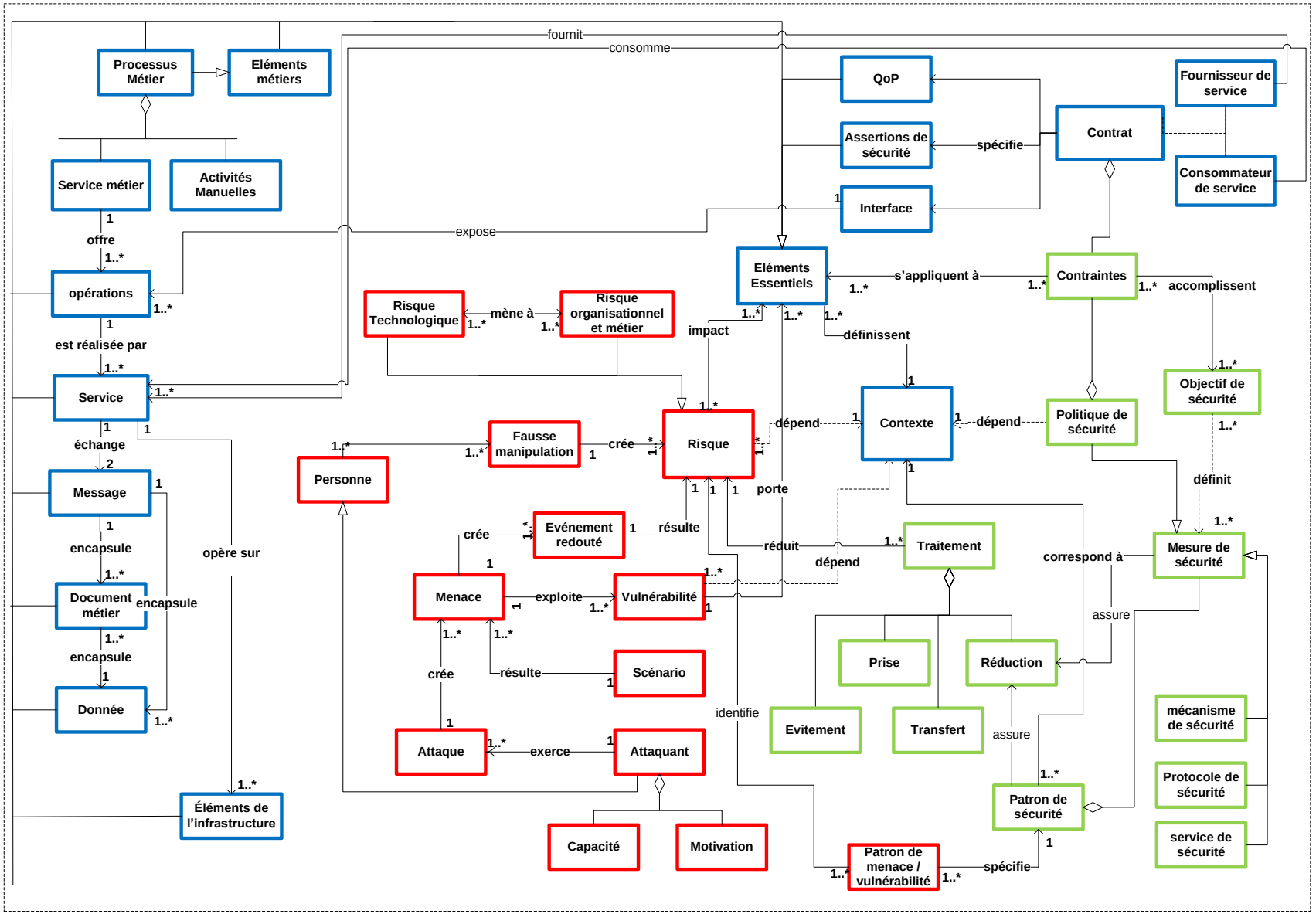


Figure 5-7 Modèle conceptuel de service sécurisé

5.3 Classification des éléments essentiels

Afin de pouvoir définir les éléments essentiels à protéger dans un environnement de services, nous proposons une classification de ces éléments pour mettre en évidence les aspects métier et technologique à couvrir. Pour conserver le couplage lâche, caractéristique de la SOA, nous proposons une organisation en trois plans (Figure 5-8) :

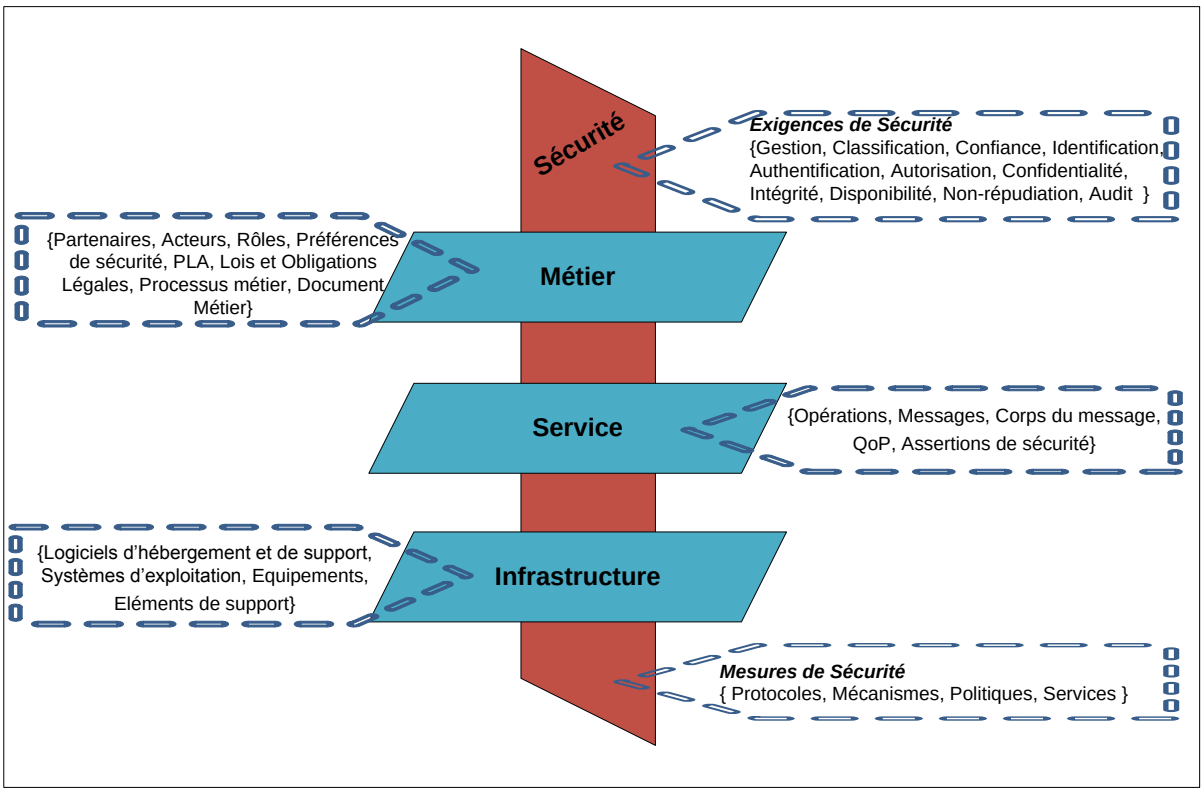


Figure 5-8 : Classification des éléments essentiels

1. Le plan métier décrit le cadre métier, organisationnel et légal. Ce cadre est constitué de l'ensemble des éléments suivants:
   - Les processus métier et des documents métier. Ces éléments métier définissent ce que l'entreprise sait faire.
   - Les partenaires, les acteurs et leurs rôles, les accords du niveau de protection (PLA global) et les préférences de sécurité qui représentent les éléments organisationnels.
   - Les lois et des obligations légales.

2. Le plan service regroupe les concepts associés aux services : opérations, messages, corps des messages, assertions de sécurité et qualité de protection.

3. Le plan de l'infrastructure décrit le cadre technologique et l'environnement d'hébergement des services : logiciels d'hébergement et de support, systèmes d'exploitation, équipements et éléments de support.

Nous définissons le plan sécurité comme un plan orthogonal aux autres plans. Ce plan décrit les exigences, ainsi que les mesures de sécurité qui sont associés à chaque élément essentiel. Ces exigences seront satisfaites en mettant en place une ou plusieurs mesures de sécurité sur les plans service et infrastructure. Par exemple, les exigences de sécurité métier sont définies en termes de gestion, classification et confiance et permettent de définir les politiques de sécurité qui seront déclinées en termes d'identification, authentification, autorisation, confidentialité, intégrité, disponibilité, non-répudiation ou audit.

Pour construire un référentiel commun, nous présentons dans ce qui suit l'Ontologie de Conception d'une SOA Sécurisée (OCSS) comme l'union de deux ontologies :

- ✓ L'ontologie « éléments essentiels d'une SOA» dans laquelle nous définissons les éléments des trois plans métier, service et infrastructure.
- ✓ L'ontologie « Profil de sécurité » définit les exigences et les mesures de sécurité.

5.4 Ontologie de Conception d'une SOA Sécurisée

Comme nous l'avons déjà dit, la sécurisation d'une SOA ne peut se faire qu'en identifiant les biens (métier et technologiques) à protéger : ce sont les éléments essentiels. Définir une ontologie de conception d'une SOA sécurisée permet de :

- ✓ Favoriser une compréhension mutuelle des éléments essentiels par les responsables métier et technique et d'optimiser l'alignement métier et technologique.
- ✓ Rendre la conception d'une SOA sécurisée indépendante de toute technologie d'implémentation en se basant sur les concepts des éléments à sécuriser et non pas les technologies de sécurisation.
- ✓ Formaliser la description des éléments essentiels en définissant clairement et sans ambigüité les différents concepts.

En outre, la flexibilité offerte par les systèmes et outils de définition et gestion des ontologies permettra de faire évoluer cette ontologie simplement en cas de besoin.

Pourquoi les ontologies ?

Une ontologie est une spécification explicite d'une conceptualisation d'un domaine. La conceptualisation permet d'identifier (par un processus d'abstraction) les concepts essentiels référencés par les termes du domaine. La spécification rend explicite le sens associé à ces concepts en leur associant une définition [109]. Dans notre travail, nous avons choisi les ontologies pour les raisons suivantes :

1- Elles pourront être facilement étendues et permettent d'envisager plusieurs travaux futurs. En effet, les éléments essentiels sont les éléments que nous avons jugés pertinents vis-à-vis de la sécurité et l'ajout d'autres éléments est envisageable.
2- Elles améliorent la communication entre les responsables techniques et métier en fixant un vocabulaire commun et en facilitant la réutilisation des sources de connaissances.

La Figure 5-9 illustre l'ontologie de conception d'une SOA sécurisée OCSS qui est l'union des ontologies éléments essentiels et profil de sécurité. Un élément essentiel des plans métier, service ou infrastructure a un profil de sécurité qui regroupe les mesures de sécurité mises en place. Ces mesures de sécurité répondent aux exigences de sécurité.

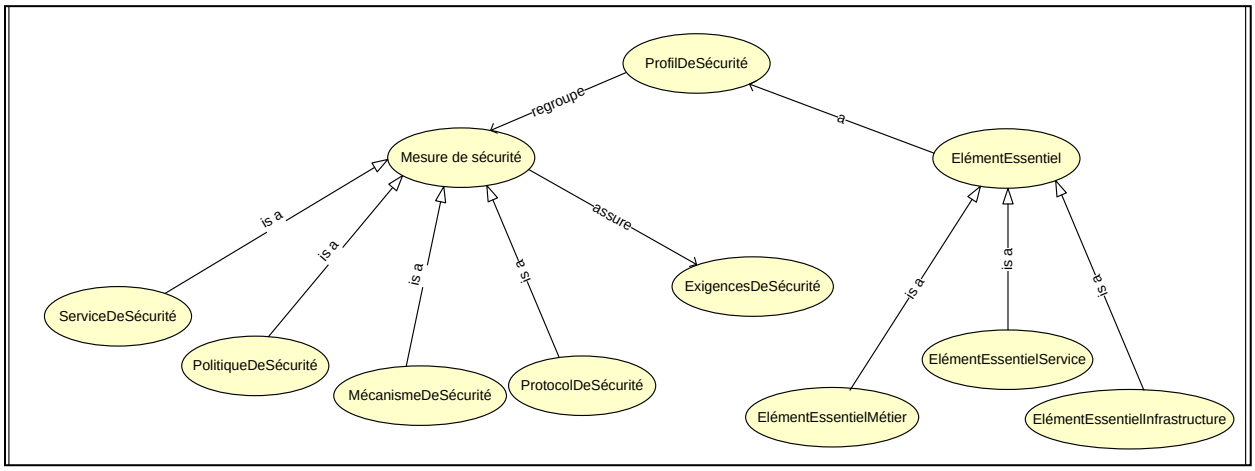


Figure 5-9 : Ontologie de conception d'une SOA sécurisée

Les éléments essentiels du plan métier sont sécurisés par des politiques de sécurité. Ces dernières incluent les règles de gestion de la sécurité (ex : gestion des droits d'accès, classification des documents métier, etc) formant le profil de sécurité de ces éléments. Les éléments essentiels des plans service et infrastructure quant à eux pourront être sécurisés par des mesures de sécurité de quatre types différents (les politiques de sécurité, les mécanismes de sécurité, les protocoles de sécurité et les services de sécurité) que nous détaillerons dans l'ontologie 'profil de sécurité'

5.4.1 L'ontologie « Éléments essentiels d'une SOA»

L'ontologie 'éléments essentiels d'une SOA' (Figure 5-10) est construite sur trois concepts : éléments essentiels métier, éléments essentiels service et éléments essentiels de l'infrastructure associés aux trois plans définis dans la section 5.3.

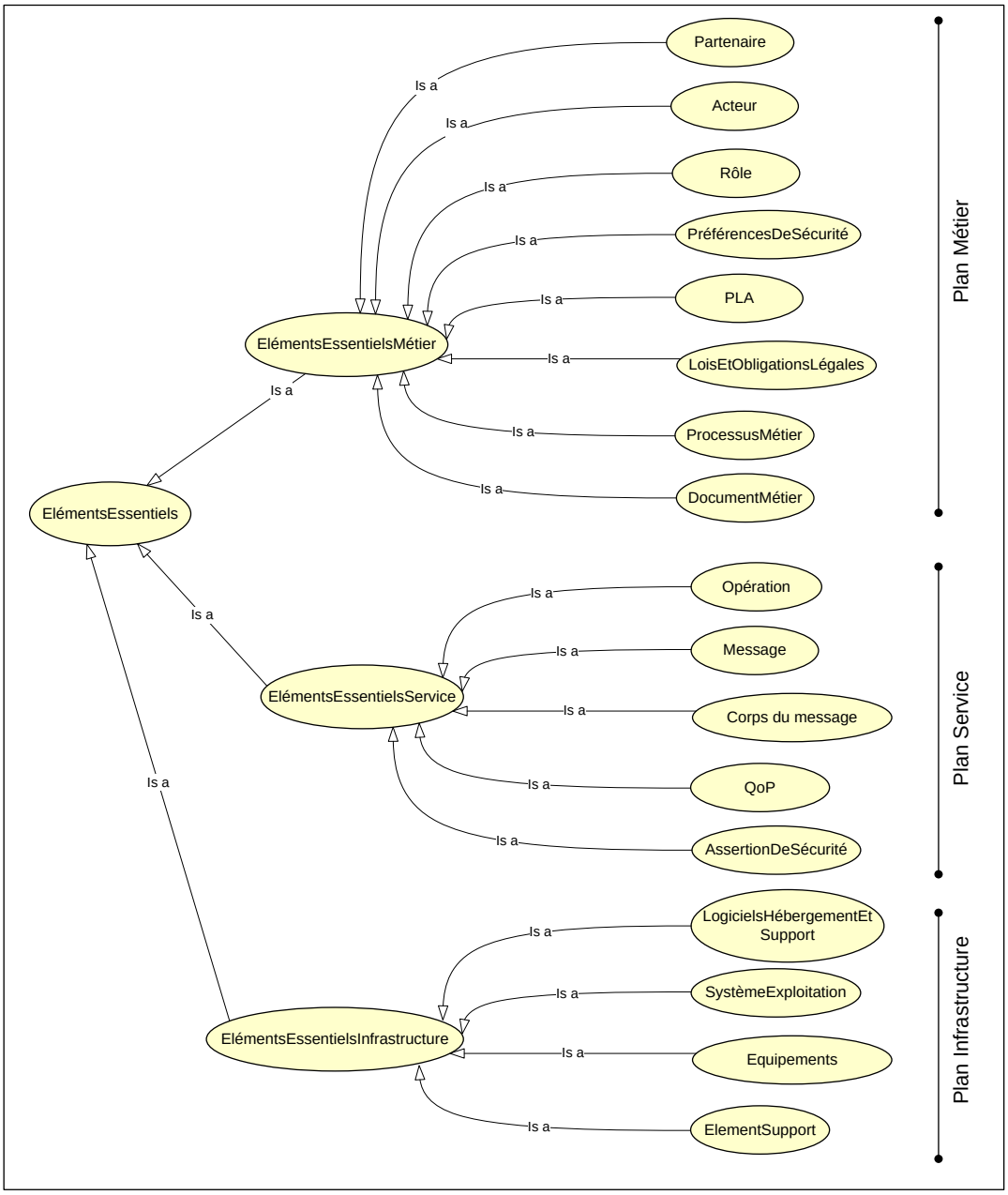


Figure 5-10 : L'ontologie « éléments essentiels d'une SOA»

5.4.1.1 Définition des concepts du plan métier :

1. Concept partenaire

Ce concept spécifie les partenaires qui participent à la réalisation des objectifs métier communs de l'entreprise. Ce sont des organisations ou des personnes qui fournissent ou consomment des services et dont les relations sont gérées par des contrats. Un partenaire joue le rôle d'un acteur en accomplissant des actions particulières. A ce titre, il est important que les partenaires soient disponibles pour assurer les objectifs métier de l'entreprise et la réussite d'un projet SOA

collaboratif. Afin de gérer les politiques d'accès aux ressources et les échanges entre les partenaires, un réseau de confiance doit être établi. Les attributs pris en compte sont :

- ✓ Nom : Cet attribut identifie le partenaire (relativement au réseau de confiance)
- ✓ Domaine : Cet attribut spécifie le domaine métier du partenaire (par exemple domaine bancaire, télécommunication, sécurité informatique, etc.)
- ✓ Type: Cet attribut spécifie le type de partenariat par exemple un partenariat 'entre concurrents' (définissant un accord entre des entreprises ayant un même savoir faire pour atteindre un objectif métier commun) ou un partenariat 'de complémentarité' (définissant un accord entre des entreprises de secteurs différents pour atteindre un objectif métier commun) [105].
- ✓ Réseau de confiance : Cet attribut définit le(s) réseau(x) de confiance auquel(s) appartient le partenaire. Un réseau de confiance regroupe plusieurs partenaires. Les modalités de collaboration (gestion des échanges de l'information, contrôle d'accès aux ressources, etc.) sont définies par des contrats.

Description Formelle :

$Partenaire \subseteq (= 1\ aUnNom.StringData) \wedge$
$(= 1\ appartient.Domaine) \wedge$
$(= 1\ aUnTypePartenariat.TypePartenariat) \wedge$
$(\geq 1\ appartient.RéseauDeConfiance)$

2. *Concept acteur*

Un acteur est une organisation, une personne ou un système qui réalise des actions. Un acteur a un ou plusieurs rôles bien définis dans l'architecture orientée services. L'identification des acteurs est indispensable pour identifier ceux qui consomment ou fournissent les activités métier du domaine. Les attributs pris en compte sont :

Nom : Cet attribut identifie l'acteur.

- ✓ Type : Cet attribut spécifie le type acteur (organisation, personne ou service)
- ✓ Rôles : Cet attribut spécifie le ou les rôles de l'acteur.

Description Formelle :

$Acteur \subseteq (= 1\ aUnNom.StringData) \wedge$
$(= 1\ aUnTypeActeur.TypeActeur) \wedge$
$(\geq 1\ aDesRôles.Rôles)$

3. *Concept rôle*

Ce concept est utilisé pour définir les droits, l'autorité, la qualification, et les responsabilités de l'acteur. Ces sous-éléments indispensables pour la gestion de la sécurité sont cités dans l'architecture de référence de l'OASIS [22]. Les attributs pris en compte sont :

- ✓ Le nom identifie le rôle
- ✓ Les droits sont des permissions prédéfinies autorisant un acteur à exécuter une action particulière.

- ✓ La délégation permet à un agent (une personne ou un système) d'agir au nom d'une organisation pour des actions précises.
- ✓ La qualification représente une certification donnée par une autorité affirmant que l'acteur a atteint un certain état ou bien une certaine compétence.
- ✓ Les responsabilités représentent les obligations associées à un rôle pour exécuter les actions.

Description Formelle :

$$Rôle \subseteq$$
$$(= 1\ aUnNom.StringData) \wedge$$
$$(\geq 1\ aDesDroits.Droits) \wedge$$
$$(= 1\ aUneDélégation.Délégation) \wedge$$
$$(= 1\ aUneQualification.Qualification) \wedge$$
$$(\geq 1\ aDesResponsabilités.Responsabilités)$$

4. Concept préférences de sécurité

Les préférences de sécurité permettent de décrire les mécanismes de sécurité et les mesures de protection que l'utilisateur désire appliquer pendant et après l'interaction avec le service [63]. Ces préférences concernent à la fois le stockage et l'utilisation des données et permettent de définir les exigences de sécurité et le niveau exigé pour chacune. Les attributs pris en compte sont :

- ✓ Type : Cet attribut spécifie le type des préférences (stockage ou utilisation des données personnelles)
- ✓ Niveau de protection : Cet attribut spécifie l'exigence de sécurité et le niveau de protection requis (exemple : niveau de confidentialité élevé)

Description Formelle :

$$PréférencesDeSécurité \subseteq$$
$$(\geq 1\ aUnType.TypePréférencesDeSécurité)$$
$$(= 1\ aUnNiveauDeProtection.NiveauDeProtection)$$

5. Concept PLA– Protection Level Agreement (Accord du niveau de protection)

Un Protection Level Agreement « PLA » est un contrat spécifiant les paramètres de la qualité de protection « QoP » déterminant les critères de sécurité qu'un fournisseur de service garantit. Garantir les PLA dans une architecture orientée services est très important surtout quand plusieurs partenaires doivent se mettre d'accord sur la QoP globale. Au niveau du plan métier, nous considérons le PLA global lié aux processus interentreprises. Les attributs pris en compte sont :

- ✓ Le nom du PLA.
- ✓ Les partenaires impliqués dans le contrat.
- ✓ Les politiques : définissant les politiques de sécurité à appliquer.

Description Formelle :
$ProtectionLevelAgreementGlobal \subseteq$
$(= 1\ aUnNom.StringData) \wedge$
$(\geq 2\ aDesPartenaires.Partenaires) \wedge$
$(\geq 1\ aDesPolitiquesPLA.PolitiquesPLA)$

6. *Concept lois et obligations légales*

Afin de gérer la sécurité dans un écosystème de services, il est nécessaire de respecter les lois et obligations légales comme par exemple :

- ✓ La directive 95/46/CE qui constitue le document de référence, au niveau européen, en matière de protection des données à caractère personnel,
- ✓ La loi « ECPA-Electronic Communications Privacy » qui prévoit des pénalités pour l'interception des communications électroniques,
- ✓ La loi « Privat Act » canadienne qui précise les obligations des institutions dans le traitement des informations privées des citoyens
- ✓ La loi Sarbanes-Oxley (SOX) qui impose plusieurs règles aux entreprises parmi lesquelles, fournir des informations concernant les transactions financières pour qu'elles se fassent d'une façon transparente et par la suite permettre un meilleur audit.

Il est donc capital d'identifier les lois et obligations réglementaires à appliquer d'une part et d'autre part de savoir comment les appliquer. Pour répondre à ce problème, on peut fournir une plateforme de gouvernance permettant d'intégrer les lois et les obligations dès la conception, d'adapter le processus au changement de la loi et en cas de changement dans un processus de vérifier la conformité vis-à-vis de la loi. Les attributs pris en compte sont :

- ✓ Nom : Le nom de la loi.
- ✓ Les politiques : définissant les politiques issues de cette loi.

Description Formelle :
$LoisEtObligationsLégales \subseteq$
$(= 1\ aUnNom.StringData) \wedge$
$(\geq 1\ aDesPolitiquesLoisEtObligationsLégales.PolitiquesLoisEtObligationsLégales)$

7. *Concept processus métier*

Un processus métier est une séquence d'activités ordonnées suivant un ensemble de règles. Chaque activité porte sur un objectif bien déterminé et utilise les ressources de l'entreprise afin de fournir des résultats précis.

Les attributs pris en compte sont :

- ✓ Nom : Cet attribut identifie le processus métier.
- ✓ Portée: Cet attribut spécifie la portée du processus, par exemple un processus interne à l'entreprise ou un processus inter entreprises.
- ✓ Partenaires : Dans le cas des processus collaboratifs, cet attribut spécifie les partenaires.
- ✓ Acteurs : Cet attribut spécifie les acteurs réalisant les instances d'activités du processus. Ces acteurs peuvent être des personnes ou des services.

- ✓ Niveau de sensibilité : Cet attribut spécifie le niveau de sensibilité du processus : noir pour les processus privés, gris pour les processus qui nécessitent un niveau de protection intermédiaire et blanc pour les processus publics.

. *Description Formelle :*

$ProcessusMétier \subseteq$
$(= 1\ aUnNom.StringData) \wedge$
$(= 1\ aUnePortéePM.PortéeProcessusMétier) \wedge$
$(\geq 0\ aUnPartenaire.Partenaire) \wedge$
$(\geq 1\ aUnActeur.Acteur) \wedge$
$(= 1\ aUnNiveauDeSensibilité.NiveauDeSensibilité)$

8. Concept document métier

Un document métier est la représentation sémantique des données consommées ou produites par un processus métier. Les attributs pris en compte sont :

- ✓ Nom : Cet attribut identifie le document métier.
- ✓ Niveau de sensibilité : Cet attribut spécifie le niveau de sensibilité associé au document métier (noir, gris ou blanc)

Description Formelle :

$DocumentMétier \subseteq$
$(= 1\ aUnNom.StringData) \wedge$
$(= 1\ aNiveauDeSensibilité.NiveauDeSensibilité)$

Les services métier identifiés lors de la modélisation des processus sont mis en œuvre par des services composites ou atomiques.

5.4.1.2 Définition des concepts du plan service :

1. Concept opération

Cet élément spécifie les fonctions (opérations) qui sont offertes par un service. La définition d'une opération inclut également les messages d'entrée et de sortie. Cette spécification est abstraite dans le sens où elle est indépendante des langages de programmation, du codage des messages et des protocoles utilisés. Les attributs pris en compte sont :

- ✓ Nom : Cet attribut identifie le nom de l'opération.
- ✓ Message : Cet attribut spécifie les messages d'entrée et de sortie.

Description Formelle :

$Opération \subseteq$
$(= 1\ aUnNom.StringData) \wedge$
$(= 2\ aUnMessage.Message)$

2. Concept message

Un message est représenté par une 'enveloppe' : Il peut être de deux types : requête ou réponse. Un message comporte une entête pouvant inclure les informations de sécurité (par exemple les

mécanismes pour intégrer la confidentialité, le contrôle d'accès, l'intégrité et la non-répudiation) et un corps du message. Les messages échangés entre les services sont au format XML (utilisent le standard SOAP [40] pour communiquer). Les attributs pris en compte sont :

- ✓ Nom : Cet attribut identifie le message.
- ✓ Type : Cet attribut spécifie le type du message requête ou réponse.
- ✓ Niveau de sensibilité: Cet attribut spécifie le niveau de sensibilité des messages (noir, gris ou blanc)
- ✓ Protocole : Cet attribut spécifie le protocole utilisé pour échanger les messages, par exemple le protocole SOAP.

Description Formelle :

$$Message \subseteq$$
$$(= 1\ aUnNom.StringData) \wedge$$
$$(= 1\ aUnTypeMessage.TypeMessage) \wedge$$
$$(= 1\ aUnNiveauDeSensibilité.NiveauDeSensibilité) \wedge$$
$$(= 1\ aUnProtocoleTransport.ProtocoleDeTransport)$$

3. Concept corps du message

Le corps du message encapsule les données échangées par le service. La nature de ces données dépend du type du service et de sa granularité. Par exemple, un service centré sur des tâches de forte granularité manipule des documents métier alors qu'un service de données de fine granularité, manipule une donnée simple (entier, chaines de caractères, etc). Le niveau de sensibilité permet de définir la protection à apporter aux données. Les attributs pris en compte sont :

- ✓ Type : Cet attribut identifie le type des données (document métier ou données)
- ✓ Niveau de sensibilité : Cet attribut spécifie le niveau de sensibilité du document / des données (noir, gris ou blanc)

Description Formelle :

$$CorpsDuMessage \subseteq$$
$$(= 1\ aUnTypeMessageCM.TypeMessageCorpsDuMessage) \wedge$$
$$(= 1\ aUnNiveauDeSensibilité.NiveauDeSensibilité)$$

4. Concept qualité de protection - QoP (Quality of protection).

La qualité de protection est définie comme étant l'ensemble des propriétés et caractéristiques d'un service lui permettant de satisfaire les exigences de sécurité (capacité à utiliser des algorithmes de chiffrement, de hachage, etc.). Dans notre travail, il faut assurer une QoP globale dans une chaîne de service en se conformant au PLA défini au niveau métier. Les attributs pris en compte sont :

- ✓ Les différents types de mesures de sécurité supportés par le service (protocoles de sécurité, mécanismes de sécurité, etc
- ✓ Le niveau de sécurité garantit par exigence de sécurité (ex : niveau de confidentialité et niveau d'intégrité des données).

Description Formelle :
$QualitéDeProtection \subseteq$
$(\geq 1\ aUnTypeMesuresSécurité.TypeMesuresSécurité) \wedge$
$(\geq 1\ aUnNiveauSécurité.NiveauSécuritéGarantit)$

5. Concept assertion de sécurité

Une assertion de sécurité permet de définir, dans une politique, les éléments mis en œuvre pour satisfaire une exigence de sécurité. Les attributs pris en compte sont :

- ✓ Type : Cet attribut spécifie le type de l'assertion de sécurité (assertion de confidentialité, d'intégrité, de jeton de sécurité, etc.)
- ✓ Eléments de l'assertion : Cet attribut spécifie les éléments de l'assertion (ex : Type de jetons : non d'utilisateur, certificats X.509, kerberos – Algorithmes de chiffrement : 3DES, AES – les Algorithmes d'intégrité: RSA-SHA, MD5, etc.)

Description Formelle :
$AssertionDeSécurité \subseteq$
$(= 1\ aUnTypeAssertionDeSécurité.TypeAssertionDeSécurité) \wedge$
$(\geq\ 1\ aUnélémentAssertion.ElémentAssertion)\ \wedge$

5.4.1.3 Définition des concepts du plan infrastructure :

1. Concept logiciels d'hébergement et de support

Cet élément spécifie les éléments logiciels qui supportent les services : serveurs d'application et logiciels de support (ESB, serveurs web, serveurs de messagerie électronique, serveurs de base de données, etc.) Les attributs pris en compte sont :

- ✓ Nom : Cet attribut identifie le logiciel.
- ✓ Type : Cet attribut spécifie le type du logiciel : serveur d'applications, serveur web, etc.
- ✓ Nom du fournisseur : Cet attribut spécifie le nom du fournisseur.
- ✓ Version : Cet attribut spécifie la version du logiciel.

Les éléments nom du fournisseur et version permettent d'améliorer la gestion des vulnérabilités à partir de bases de connaissances spécialisées (Voir section 6.4 pour une application à la vulnérabilité des logiciels)

Description Formelle :
$ElémentLogiciel \subseteq$
$(= 1\ aUnNom.StringData) \wedge$
$(= 1\ aUnTypeLogicielHébergementSupport.TypeLogicielHébergementSupport) \wedge$
$(= 1\ aUnNomFournisseur.StringData) \wedge (= 1\ aUneVersion.StringData)$

2. Concept systèmes d'exploitation

Cet élément spécifie les systèmes d'exploitation installés sur chaque équipement utilisé (incluant donc les équipements d'interconnexion, tels que les routeurs qui font partie des éléments essentiels de l'infrastructure). Les attributs pris en compte sont :

- ✓ Nom : Cet attribut identifie le système d'exploitation.
- ✓ Nom du fournisseur : Cet attribut spécifie le nom du fournisseur.
- ✓ Version : Cet attribut spécifie la version du système d'exploitation.

Description Formelle :

$$SystèmeExploitation \subseteq (= 1\ aUnNom.StringData) \wedge (= 1\ aUnNomFournisseur.StringData) \wedge (= 1\ aUneVersion.StringData)$$

3. Concept équipements

Cet élément spécifie les équipements matériels de l'infrastructure (serveurs d'hébergement et équipements d'interconnexion). Les attributs pris en compte sont :

- ✓ Nom : Cet attribut identifie l'équipement.
- ✓ Type : Cet attribut spécifie le type de l'équipement : serveur d'hébergement, équipement d'interconnexion tel que les routeurs, les commutateurs (Switch), les pare-feu, etc.

Description Formelle :

$$Equipement \subseteq (= 1\ aUnNom.StringData) \wedge (= 1\ aUnTypeEquipement.TypeEquipement)$$

4. Concept éléments de support

Cet élément spécifie les éléments de support tels que les connexions réseau et les alimentations électriques, etc. Les attributs pris en compte sont :

- ✓ Nom : Cet attribut identifie l'élément de support.
- ✓ Type : Cet attribut spécifie le type de l'élément.

Description Formelle :

$$ElementSupport \subseteq (= 1\ aUnNom.StringData) \wedge (= 1\ aUnTypeElementSupport.TypeElementSupport)$$

5.4.2 Ontologie « Profil de sécurité »

Afin de satisfaire les exigences de sécurité relatives à un élément essentiel, il faut choisir les mesures de sécurité adaptées au contexte. Pour cela, nous utilisons l'ontologie NRL-SO [53]. Cette ontologie définit trois types de mesures de sécurité : les mécanismes, les protocoles et les politiques de sécurité. A ces concepts, nous avons ajouté 'le service de sécurité' dans notre ontologie 'Profil de sécurité' (Figure 5-11).

Un service de sécurité est un service autonome, capable de fournir des fonctionnalités de sécurité. Il satisfait une exigence de sécurité spécifique (ex : un service d'authentification).

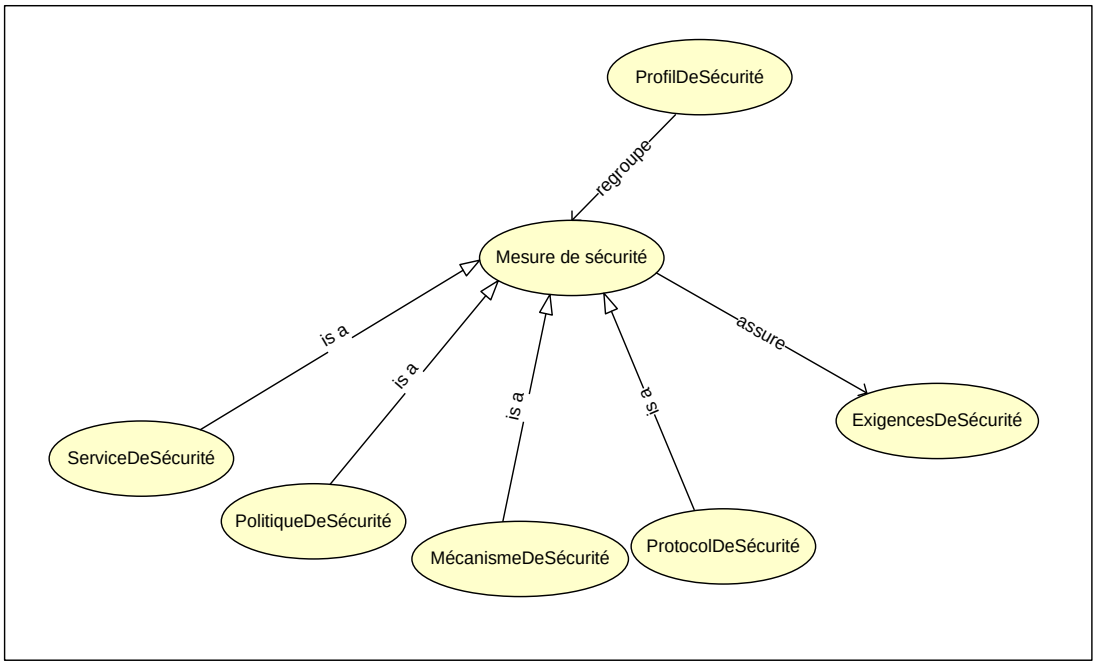


Figure 5-11 : Ontologie Profil de sécurité

1. *Concept protocole de sécurité*

Un protocole est un ensemble de règles définissant le mode de communication entre deux systèmes [110]. Un protocole de sécurité spécifie les règles de sécurité à appliquer et peut porter sur le transport de l'information, l'échange et la gestion des clés, etc. Les attributs pris en compte sont :

Description Formelle :

$ProtocoleDeSécurité \sqsubseteq$

$(= 1\ aUnNom.StringData) \wedge$

$(= 1\ aUnAlgorithmeDeSécurité.AlgorithmeDeSécurité) \wedge$

$(= 1\ aUnTypeProtocoleDeSécurité\ .TypeProtocoleDeSécurité) \wedge$

$(\exists\ AssureDesExigencesDeSécurité.ExigencesDeSécurité)$

2. *Concept politique de sécurité*

Une politique de sécurité regroupe les règles qui spécifient les exigences de sécurité d'un élément essentiel. Nous distinguons deux types de politiques de sécurité :

- ✓ Les politiques de sécurité métier qui concernent les éléments essentiels métier et organisationnel (définir des droits d'accès aux ressources, classification de l'information).
- ✓ Les politiques de sécurité technologiques qui concernent les éléments essentiels des plans service et infrastructure et qui contiennent les assertions de sécurité (algorithmes de chiffrement, longueurs des clés, l'ordre dans lequel se fait le chiffrement, signature ou chiffrement de parties spécifiques des messages, etc.)

Les attributs pris en compte sont :

- ✓ Nom : Cet attribut identifie la politique de sécurité.
- ✓ Type : Cet attribut spécifie la portée de la politique métier ou technologique
- ✓ Règles : Cet attribut spécifie les règles de sécurité.
- ✓ Action : Cet attribut spécifie les actions à accomplir.
- ✓ Exigences de sécurité assurées: Cet attribut précise les exigences de sécurité satisfaites par la politique de sécurité.

Description Formelle :

$PolitiqueDeSécurité \subseteq$
$(= 1\ aUnNom.StringData) \wedge$
$(\geq 1\ aUnType.TypePolitiqueDeSécurité) \wedge$
$(\geq 1\ aDesRègles.RèglesDeSécurité) \wedge$
$(\geq 1\ aUneAction.ActionDePolitiqueDeSécurité) \wedge$
$(\exists\ AssureDesExigencesDeSécurité \cdot ExigencesDeSécurité)$

3. *Concept service de sécurité*

Comme tout service, un service de sécurité est un service autonome. Il encapsule une fonctionnalité de sécurité qu'il pourra offrir aux applications (authentification, autorisation, chiffrement, signature et vérification de signature, etc.).

En définissant les services de sécurité autonomes, les services et applications sont sécurisés de manière non intrusive. Les attributs pris en compte sont :

- ✓ Nom : Cet attribut identifie le service de sécurité.
- ✓ Type : Cet attribut spécifie le type du service transparent ou non.
- ✓ Exigences de sécurité assurées: Cet élément précise les exigences de sécurité satisfaites par le service de sécurité.

Description Formelle :

$ServiceDeSécurité \subseteq$
$(= 1\ aUnNom.StringData) \wedge$
$(= 1\ aUnTypeServiceDeSécurité.TypeServiceDeSécurité) \wedge$
$(\exists\ AssureDesExigencesDeSécurité.ExigencesDeSécurité)$

4. *Concept mécanisme de sécurité*

Un mécanisme de sécurité est un mécanisme conçu pour détecter, prévenir ou 'rattraper' une attaque ou une faille de sécurité. Comme dans le cas d'un protocole de sécurité, un mécanisme de sécurité peut couvrir une défaillance de différents éléments des plans services et infrastructure.

Description Formelle :

$MécanismeDeSécurité \subseteq (= 1\ aUnNom.StringData) \wedge$
$(= 1\ aUnTypeMécanismeDeSécurité.TypeMécanismeDeSécurité) \wedge$
$(\exists\ AssureDesExigencesDeSécurité.ExigencesDeSécurité)$

5. Concept exigence de sécurité

Ce concept spécifie les exigences de sécurité qui devront être satisfaites par la mise en place des mesures de sécurité. Les exigences de sécurité sont définies à partir des objectifs de sécurité définis dans le modèle conceptuel de politique de sécurité (la gestion, la classification, la confiance, l'identification, l'authentification, l'autorisation, la confidentialité, l'intégrité, la disponibilité, la non-répudiation, et l'audit) selon le niveau métier, service ou infrastructure auquel l'exigence s'applique.

Description Formelle :

$ExigenceDeSécurité \subseteq (= 1\ aUnNom.StringData) \wedge$
$(= 1\ aUnTypeExigenceDeSécurité.TypeExigenceDeSécurité)$

5.5 Conclusion

Dans ce chapitre, notre contribution a porté sur la préparation d'un projet SOA sécurisé. Pour cela, nous avons développé un modèle de service sécurisé ainsi qu'une ontologie support pour la conception d'une SOA sécurisée (OCSS).

Le développement du modèle conceptuel de service sécurisé a été réalisé en unifiant trois modèles conceptuels: celui des services, des politiques de sécurité et des risques. Nous avons utilisé les éléments communs tels que le contexte, les contraintes et les politiques de sécurité pour réaliser l'unification de ces modèles.

Pour garantir l'ouverture et l'interopérabilité, nous avons défini l'ontologie OCSS en unifiant deux ontologies : celle des éléments essentiels et celle du profil de sécurité :

- La première définit les concepts des biens à protéger, que nous avons classés selon trois niveaux d'abstraction :
  - ✓ Le plan métier liste les éléments essentiels métier et organisationnels tel que les partenaires, les acteurs et les processus métier.
  - ✓ Le plan service liste les éléments essentiels au niveau du service tel que les opérations, les messages et les politiques de sécurité.
  - ✓ Le plan Infrastructure liste les éléments essentiels de l'infrastructure tels que les logiciels d'hébergement, les systèmes d'exploitation et les équipements.
- La seconde définit les concepts associés aux exigences de sécurité et aux mesures de sécurité.

Le travail réalisé dans ce chapitre sera utilisé comme base de la conception d'une SOA sécurisée. Dans le chapitre suivant, nous développerons la méthodologie de conception d'une SOA sécurisée. Nous détaillerons les étapes à suivre au niveau de la conception en proposant les directives et les bonnes pratiques à suivre.

Nous utiliserons le modèle conceptuel de service sécurisé comme un cadre de référence pour la conception puisque ce modèle met en évidence le lien entre la sécurité et les services. De plus, nous utiliserons les ontologies développées pour établir le contexte de conception et mener une étude de gestion des risques sur les biens à protéger.

Chapitre 6. Cadre méthodologique de gestion de la sécurité dans une SOA

Résumé

Dans ce chapitre, nous développons une méthodologie de conception d'une SOA sécurisée puis abordons la construction de processus métier sécurisés avant de proposer un service de gestion des vulnérabilités de l'infrastructure.

Sommaire

6.1 Introduction

Nombreuses sont les organisations qui ont migré vers les architectures orientées services du fait de leur capacité à assurer l'alignement métier et système d'information pour optimiser l'agilité de l'entreprise. Toutefois, l'adoption de ces technologies soulève de nouveaux défis de sécurité, notamment lorsque l'entreprise fournit un accès externe à son patrimoine informationnel.

Dans le chapitre précédent, nous avons présenté des modèles pour définir un cadre de référence pour entamer la conception d'une SOA sécurisée. Dans ce chapitre, nous développons une Méthodologie de Conception d'une SOA Sécurisée (MCSS) qui nous permettra d'atteindre deux objectifs complémentaires : assurer l'alignement métier, la flexibilité et la réutilisation des services d'une part et identifier les risques potentiels présents au niveau des plans métier, service et infrastructure d'autre part.

Pour ce qui concerne la phase d'exécution, nous proposons une architecture de sélection dynamique des services sécurisés. Cette architecture permet la découverte et la sélection des services sécurisés en se basant sur les annotations de sécurité et les préférences de sécurité des utilisateurs. Au cœur de cette architecture, nous plaçons un service de sécurité intermédiaire – Security Broker – pour publier les annotations de sécurité dans l'annuaire et permettre la sélection des services sécurisés selon les préférences des utilisateurs.

Enfin, dans la phase de supervision, nous proposons un service de gestion des vulnérabilités. Ce service vérifie auprès des bases de vulnérabilités publiques la présence de vulnérabilités liées aux composants du plan infrastructure (logiciel d'hébergement, systèmes d'exploitation).

6.2 La méthodologie MCSS : La phase de conception

6.2.1 Aperçu de la méthodologie MCSS

La méthodologie MCSS (Figure 6-1) permet de concevoir des SOA sécurisées en :

- ✓ exposant les fonctionnalités de l'entreprise en tant que services métier réutilisables dans des processus métier. Pour cela, nous abordons la modélisation des processus.
- ✓ prenant en compte l'interopérabilité (syntaxique et sémantique) dans l'échange d'information. Pour cela, nous détaillons la modélisation des documents métier.
- ✓ étudiant la dépendance entre le service et les éléments essentiels (éléments métier et éléments de l'infrastructure) définissant le contexte. Pour cela, nous développons un modèle de dépendance créant les liens entre les éléments essentiels formant le contexte.
- ✓ intégrant un cycle de gestion des risques au sein de la phase de conception des services.

Nous détaillons les différentes étapes de la méthodologie en proposant la création d'un dialogue commun entre les responsables métier et les responsables techniques de l'entreprise. En effet, comme nous l'avons vu dans le chapitre 4, il est crucial que ces responsables communiquent entre eux afin de mener à bien une étude de gestion des risques. De ce fait, notre méthodologie MCSS s'apparente aux méthodes CORAS et OCTAVE.

Pour ce qui concerne l'identification des risques, nous procédons par une approche de modélisation des menaces (approche de la méthode CORAS) et nous faisons référence aux méthodes EBIOS et OCTAVE en cas de besoin ce qui rend MCSS adaptée à la gestion des risques dans un environnement de services.

Dans ce qui suit, nous commençons par présenter les trois phases de la méthodologie :

- ✓ Phase 1: Identification et spécification des services
- ✓ Phase 2: Intégration du cycle de gestion des risques
- ✓ Phase 3: Annotation de sécurité

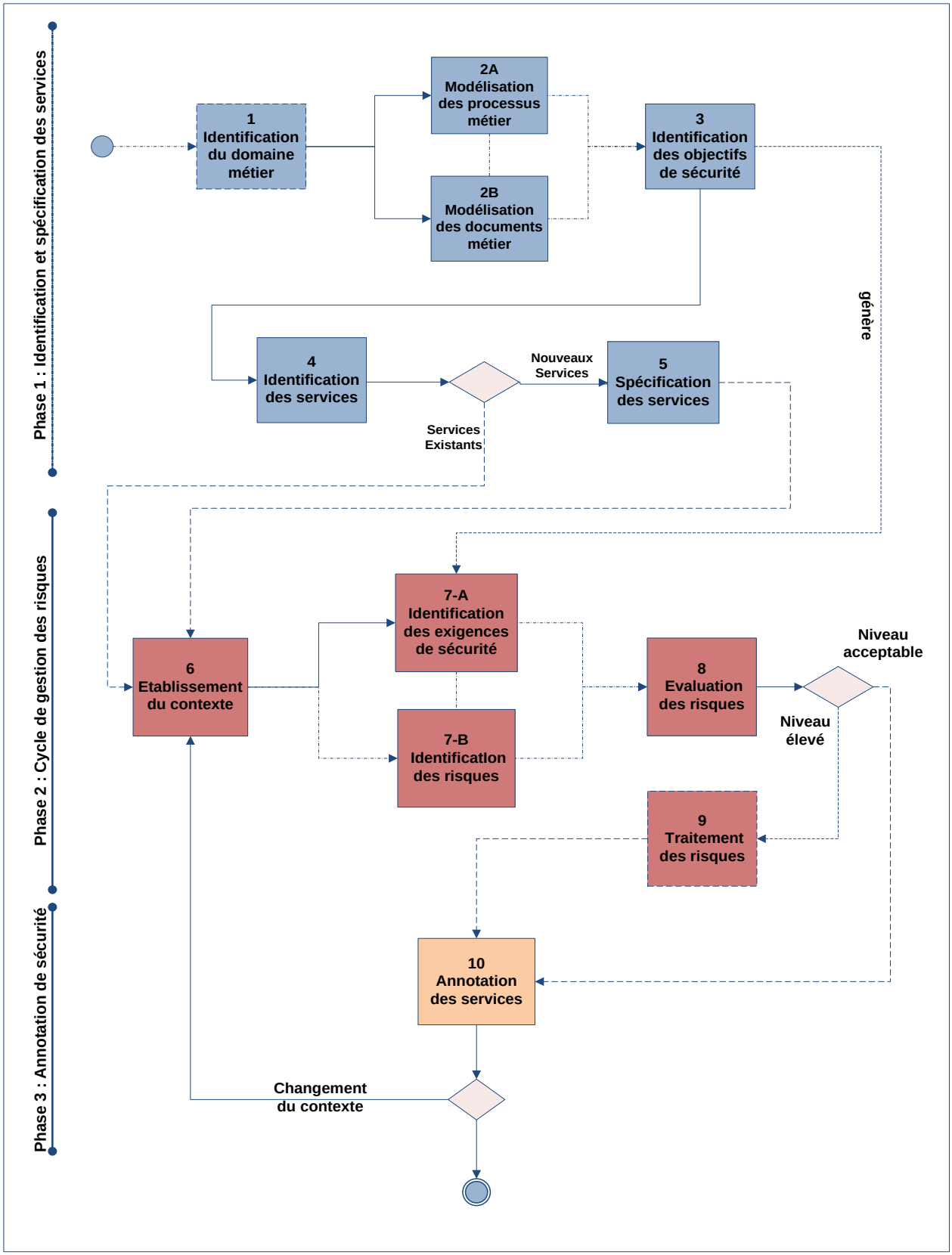


Figure 6-1 : Méthodologie de conception d'une SOA sécurisée

Phase 1: Identification et spécification des services

La première phase représente la définition du contexte métier et la démarche d'identification et de spécification des services qui répondent aux besoins métier. Cette phase comporte des étapes destinées à :

- identifier et synthétiser les informations permettant l'alignement métier et applicatif (identification des objectifs métier, de la stratégie de l'entreprise et des activités primaires et secondaires)
- modéliser les processus métier internes et externes à l'entreprise
- établir un modèle sémantique de l'information pour assurer une meilleure interopérabilité entre les services internes et externes à l'entreprise.

Ces étapes nous permettent d'identifier les objectifs de sécurité métier. Ces derniers vont permettre de générer les exigences de sécurité technologiques. La spécification des processus métier et des services qu'ils orchestrent permet de faire le lien entre visions métier et technologique. Outre l'alignement métier, cette vision apporte une cohérence globale au niveau de la sécurité des processus métier.

Phase 2 : Gestion des risques

La deuxième phase de la méthodologie représente l'intégration d'un cycle de gestion des risques dans la conception des services en fonction des objectifs de sécurité métier. Nous entamons le cycle de gestion des risques par une étape d'établissement du contexte : toutes les informations correspondant aux éléments essentiels, en particulier la spécification des services existants ou à créer sont regroupées avant d'identifier les exigences de sécurité, en nous basant sur le contexte de conception et les objectifs de sécurité définis dans la phase 1.

Nous identifions ensuite les risques portant sur les biens à protéger. Nous évaluons ces risques en fonction de leur impact, probabilité et du coût de la non-sécurité (étape 'évaluation des risques'). Enfin, nous étudions le traitement des risques dans la dernière étape de cette phase.

Phase 3 : Annotation de sécurité

La dernière phase de la méthodologie représente la synthèse du travail effectué en produisant l'annotation des services par des paramètres de sécurité. Ces éléments d'annotation enrichissent la description des services, présentent une garantie de sécurité et sont utilisés pour améliorer la sélection des services à l'exécution. Pour supporter l'amélioration continue de la conception des services sécurisés, nous avons construit la méthodologie pour une exécution itérative (Figure 6-1) en permettant le retour sur le cycle de gestion des risques en cas de changement du contexte de conception. Ainsi les mesures de sécurité mises en place et l'annotation de sécurité restent valides tant que le contexte de conception ne change pas (il n'y a pas de limitation de durée). Par contre, il faut entamer à nouveau le cycle de gestion des risques et adapter la sécurité en cas de changement du contexte. Par exemple, lorsqu'une entreprise donne accès à ses services dans le cadre d'un nouveau partenariat, il va falloir étudier ce nouveau contexte et adapter les mesures de sécurité ce qui aboutit à de nouvelles annotations de sécurité.

6.2.2 La méthodologie MCSS

Etape 1: Identification du domaine métier

Un domaine métier représente une division logique de l'entreprise où les services affectés peuvent être conçus et/ou acquis indépendamment de ceux appartenant à d'autres domaines [111]. Donner une vue d'ensemble du domaine métier suppose de répondre à quatre questions : le qui, le quoi, le comment et le pourquoi. Ces questions sont à la base de l'identification des acteurs, des objectifs métier, des activités et de la stratégie de l'entreprise. Dans cette étape, nous définissons les éléments à protéger du domaine métier et les règles permettant de les identifier. Les éléments sont identifiés dans un atelier de '*brainstorming*' en présence des responsables métier du domaine en question.

Définition 1 : Le *Qui* est la liste des acteurs du domaine métier. Notons qu'il faut prendre en compte l'aspect « *qui* fait *quoi* » en d'autres termes : qui est en train de consommer ou de fournir des activités métier dans le domaine ?
Pour identifier ces éléments, nous nous appuyons sur l'architecture conceptuelle métier (Figure 6-2), qui est une représentation de haut niveau et qui permet de présenter à la fois :

- ✓ Les *activités primaires* : identifiées en listant les activités représentant le cœur du métier du domaine. Ces activités sont à la base de la modélisation des processus métier.
- ✓ Les *activités secondaires* : identifiées en listant les activités de support qui supportent le bon fonctionnement du domaine métier mais qui ne sont pas liées au métier lui-même (ex : gestion des ressources humaines dans le domaine métier 'formation à distance' au sein d'une université)
- ✓ Les *acteurs* : identifiés en listant les acteurs qui interagissent avec les activités du domaine métier. Ces acteurs (humains ou systèmes) fournissent ou consomment les services du domaine et peuvent avoir des préférences de sécurité spécifiques.

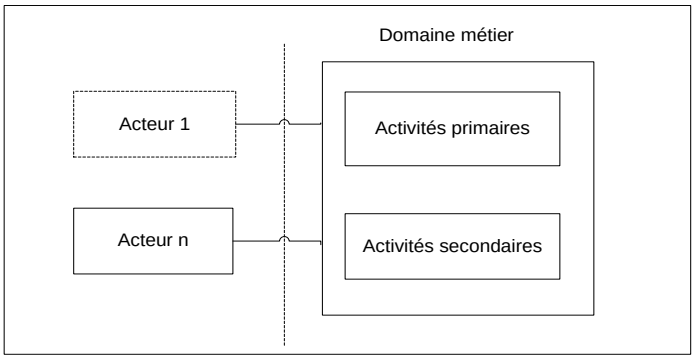


Figure 6-2 : Architecture conceptuelle (1)

Définition 2 : Le *Quoi* est la définition des objectifs métier et des moyens mis en œuvre pour les atteindre, c'est-à-dire la définition des activités du domaine métier et de leur portée.

Définition 3 : Le Comment est la liste des interactions globales se déroulant entre les acteurs ou entre les acteurs et les activités primaires.
Pour identifier cette liste, nous ajoutons à l'architecture conceptuelle les documents métier échangés entre les acteurs (Figure 6-3). Cette liste est le résultat d'une première phase d'identification des documents métier et sera raffinée dans l'étape 2A.

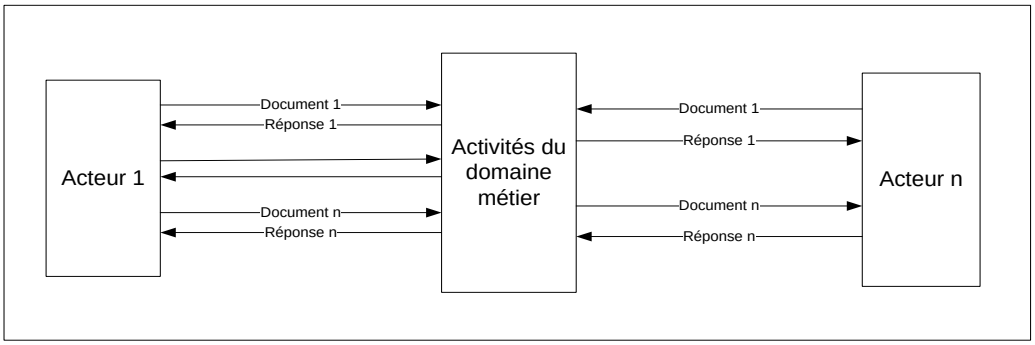


Figure 6-3 : Architecture conceptuelle (2)

Définition 4 : Le *Pourquoi* permet d'identifier les stratégies et les motivations en déterminant pourquoi les activités du domaine métier sont offertes et pourquoi les acteurs interagissent avec ces activités. Le modèle de motivation métier (BMM) de l'OMG (détaillé dans l'Annexe 2) [112] nous fournit un cadre pour mener le dialogue avec les responsables métier et identifier les stratégies de l'entreprise pour atteindre ses objectifs. Le modèle détaille à la fois les stratégies et les motivations métier : les éléments définis par le modèle BMM sont regroupés en quatre catégories :

- ✓ La finalité (end) est ce que l'entreprise veut accomplir ou atteindre par exemple le développement d'une nouvelle activité économique.
- ✓ Les moyens définissent ce que l'entreprise décide de mettre en œuvre pour atteindre la finalité.
- ✓ Les influenceurs sont ceux qui peuvent causer des changements affectant l'entreprise dans la réalisation de ses objectifs.
- ✓ L'évaluation est l'appréciation de l'impact des influenceurs sur la capacité de l'entreprise.

Le tableau suivant récapitule les outils et modèles utilisés dans cette étape ainsi que les résultats.

Outils utilisés / Modèles de référence	Sortie (Résultat)
✓ Architecture conceptuelle ✓ Modèle de motivation métier	✓ Le *Qui* : Les acteurs du domaine métier. ✓ Le *Quoi* : Les objectifs métier. ✓ Le *Comment* : Les interactions globales. ✓ Le *Pourquoi* : Les stratégies et les motivations.

Tableau 6-1 : Tableau récapitulatif - Identification du domaine métier

Etape 2A: Modélisation des processus métier

Dans une architecture orientée services, les fonctionnalités de l'entreprise sont exposées en tant que services métier réutilisables dans des processus métier (que nous désignons sous le terme 'processus'). Ces processus sont définis comme une séquence d'activités ordonnées suivant un ensemble de règles. Chaque activité porte sur un objectif bien déterminé et utilise les ressources de l'entreprise afin de fournir des résultats précis.
Un processus métier peut combiner des activités automatiques qui seront exécutés par des services métier, des activités semi-automatiques qui nécessitent des interventions humaines et des activités manuelles. D'après [8], il existe différentes approches pour décrire les processus en fonction de la nature des applications prises en compte. Nous pouvons distinguer :

- ✓ Les processus coopératifs ou créatifs permettant la médiation entre des personnes engagées dans des activités créatives à forte valeur ajoutée (ex : processus de conception)
- ✓ Les processus interactifs organisant les interactions entre des personnes qui saisissent des informations modifiant le système d'information (ex : processus administratifs)
- ✓ Les processus automatiques organisant l'enchaînement d'opérations totalement automatisés (ex : processus de production en back-office)

Dans [12], P. Bonnet complète la classification des processus en ajoutant une perspective selon le cycle «projet de conception des SOA» :

- ✓ Les processus conceptuels sont associés à une modélisation ne prenant en compte ni les acteurs ni les systèmes utilisés.
- ✓ Les processus macro organisationnels sont associés à une modélisation tenant compte des types d'acteurs et des types d'outils informatiques impliqués dans ce processus
- ✓ Les processus organisationnels sont associés à une modélisation tenant compte de la réalité de l'organisation et des contingences des outils informatiques.

Lors de l'inventaire des processus métier, nous les caractérisons à l'aide de ces deux classifications (nature des applications et type de processus) auxquelles nous ajoutons la définition de la portée (intra ou inter entreprise) et du niveau de confidentialité (privé, publique ou semi-publique). Après les avoir caractérisé, la modélisation des processus constitue la première étape dans l'identification des services permettant d'atteindre les objectifs métier. La modélisation peut se faire en utilisant des langages de modélisations comme :

- ✓ Le Business Process Modeling Notation « BPMN » dont la version 1.0 a été publiée par la BPMI (Business Process Management Initiative) en 2004 et la version 2.0 de 2010
- ✓ Le Unified Modeling Language « UML » dont la version 2.0 a été publiée par l'OMG en 2004

L'alliance entre l'OMG et le BPMI a fait en sorte que ces deux notations deviennent complémentaires. Toutefois :

- ✓ BPMN s'adresse aux analystes métier alors qu'UML s'adresse aux développeurs.

- ✓ Les outils BPMN présentent l'avantage de pouvoir générer du langage d'exécution opérationnel [8] (tel que BPEL : Business Process Execution Langage) ce qui permet (sous réserve d'avoir les outils nécessaires) de transformer ces modèles en workflow exécutables.

Nous listons dans le Tableau 6-2 les éléments de base d'un modèle de processus métier :

Eléments	Description
Activités	Les étapes du processus dont le nom reflète les activités métier. Ces activités peuvent être manuelles, semi-automatiques ou automatiques.
Acteur	Un acteur est une organisation, une personne ou bien un système qui accomplit des actions et qui a un rôle bien défini dans l'architecture orientée services.
Rôles	Les rôles spécifient les droits et les responsabilités des acteurs.
Le flux de contrôle	Représente l'ordre dans lequel les activités et l'évaluation des règles de transition entre les activités.
Le flux de donnée	Représente la façon dont les données sont produites ou consommées par le processus. En d'autres termes, le flux de données prend en charge les variables qui permettent de conserver les documents et de gérer l'état d'un processus.
Les documents métier	Représentent un regroupement d'objets métier et sont à la base de la spécification des messages échangés entre les activités composant le processus. En particulier, l'identification des documents durant la conception des processus métier doit être liée au modèle sémantique de l'information (ce qui est réalisé dans l'étape '2B-Modélisation de l'information sémantique')

Tableau 6-2 : Les éléments du processus métier

Pour modéliser les processus métier, nous nous basons sur les éléments du domaine métier (le *Qui* et le *Quoi*) de l'étape précédente afin de lister les éléments de base d'un modèle des processus : les acteurs et les activités. Nous modélisons en UML les diagrammes des cas d'utilisation possibles ce qui répond à la question « *Qui fait Quoi* » (Figure 6-4) en recueillant les exigences sur la base des scénarios d'utilisation en se focalisant sur les interactions entre les acteurs et le système.

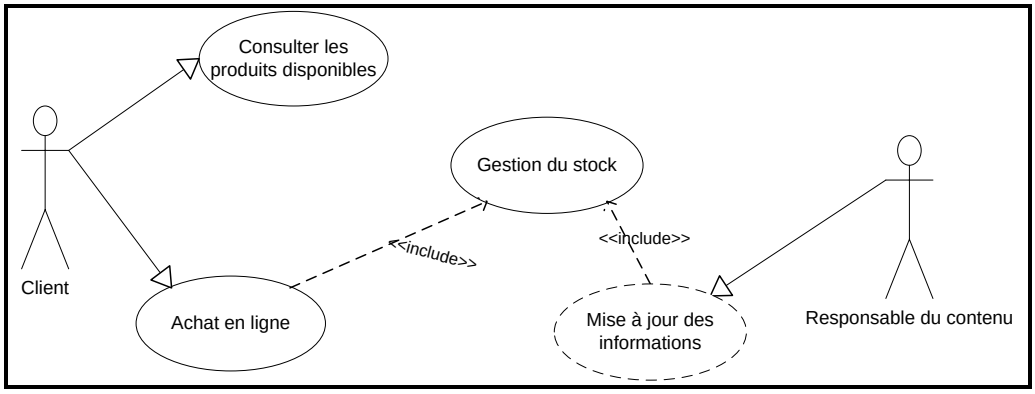


Figure 6-4 : Exemple de cas d'utilisation

Pour chaque cas d'utilisation, nous élaborons le modèle du processus métier qui définit la séquence des activités à réaliser et les documents métier à échanger entre les activités (ces documents sont modélisés dans l'étape 2B).

Nous avons choisi le BPMN (voir Annexe 3) comme langage de modélisation pour que le travail effectué soit assimilé par tous les utilisateurs de l'entreprise, depuis les responsables métier jusqu'aux développeurs de la solution technologique en passant par les utilisateurs de ces processus. BPMN propose deux catégories d'objets:

- ✓ La première caractérise la description du flot de contrôle. On y trouve
  1 Les événements de début, intermédiaire et de fin affectant la vie du processus.
  2 Les activités qui peuvent être atomique ou composites.
  3 Les décisions contrôlant la convergence ou la divergence de plusieurs flots.
- ✓ La seconde catégorie concerne les objets de connexion. Elle distingue : les flots de connexion, les flots de séquence, les flots conditionnels, les flots de messages et les associations.

La Figure 6-5 représente un extrait d'un processus métier « Achat en ligne » incluant la vérification du payement pour approbation. Cette figure illustre le partitionnement du processus métier par rôle définissant la collaboration entre deux organisations, la plateforme de commerce électronique, qui assure la gestion de la relation client, le marketing et la création de devis, d'une part, et d'autre part, une plateforme logistique chargée de la livraison. Un client demande un devis en soumettant ses préférences à la 'plateforme commerce électronique' et soumet la commande après réception du devis. La 'plateforme commerce électronique' se charge de la création du devis, de la vérification des informations clients et de la notification de la 'plateforme logistique' pour livrer le produit dans le cas d'un achat approuvé ou de la notification du client dans le cas du refus.

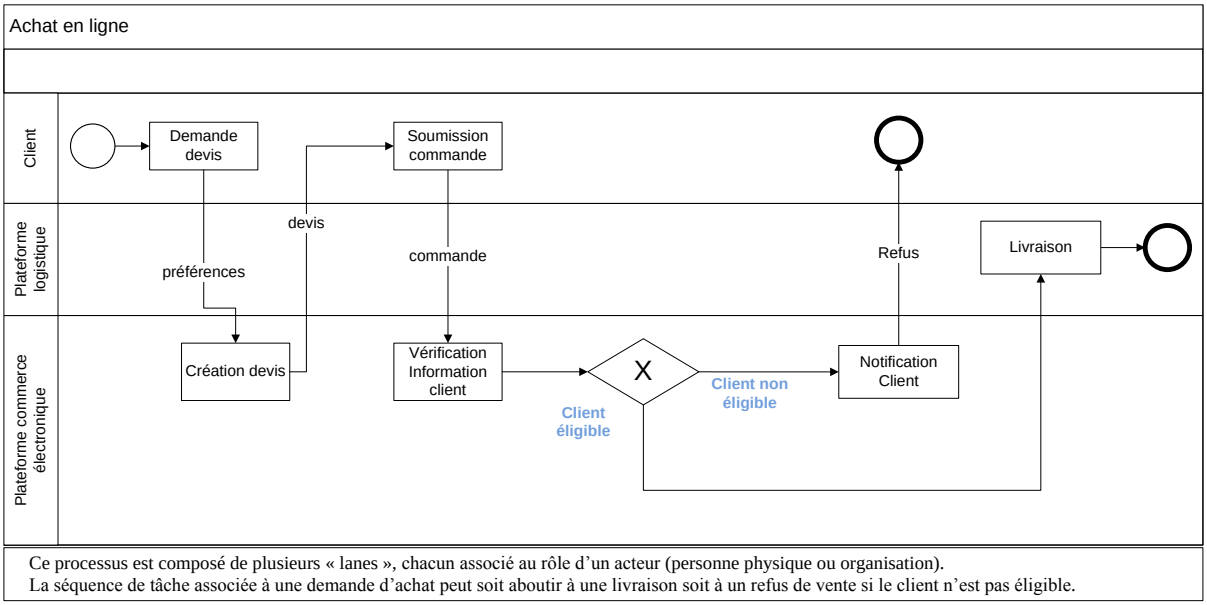


Figure 6-5 : Exemple d'un processus métier : Achat en ligne

La modélisation différencie les activités atomiques (ne se décomposent pas) et les activités composites formant un sous-processus décomposables [8]. Lorsque les processus intègrent des activités composites, ils doivent être décomposés en sous processus afin d'identifier les services. A titre d'exemple, l'activité 'création devis' peut être définie comme un sous-processus qui se compose des activités suivantes : recueil des informations produit, calcul du prix, calcul des frais de transport, vérification et modification des prix ou des frais et soumission devis (Figure 6-6)

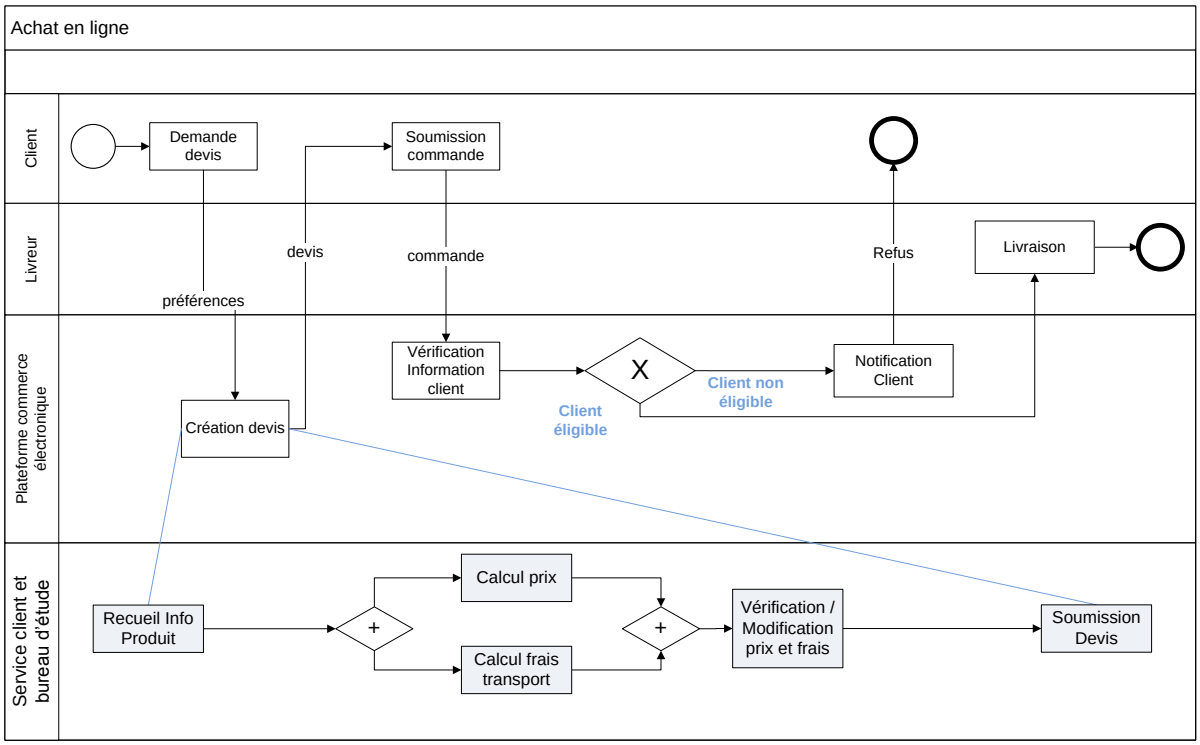


Figure 6-6 : Sous-processus métier création devis

Nous notons que les deux étapes 2A et 2B de la méthodologie sont initiées simultanément après l'étape 1 (Figure 6-1). En outre, nous utilisons les processus métier comme cadre pour la modélisation des documents métier. En effet, la modélisation des processus/sous-processus raffine l'identification des documents métier à modéliser dans l'étape 2B.

Entrées	Langages de Modélisation	Sorties (Résultats)
✓ Le *Quoi* : Liste des activités correspondant au domine métier ✓ Le *Qui* : Liste des acteurs du domaine métier ✓ Le *comment* : Liste des documents métier	Langages de modélisation UML et BPMN	✓ Diagrammes des cas d'utilisation en UML ✓ Modèles de processus métier en BPMN

Tableau 6-3 : Tableau récapitulatif - Modélisation des processus métier

Etape 2B: Modélisation des documents métier

Pour permettre la compréhension des informations échangées, il faut prendre en compte l'interopérabilité (syntaxique et sémantique) dans la modélisation des documents métier. Au niveau syntaxique, une solution est de recourir au standard de fait XML comme langage commun de communication pour échanger les documents encapsulés dans les messages. Au niveau sémantique, nous développons un modèle structuré de données métier à partir duquel nous modélisons les documents métier et nous identifions les schémas correspondants. Cette démarche nous permettra de cadrer le contexte sémantique du domaine métier et d'assurer l'interopérabilité syntaxique et sémantique évitant les ambiguïtés et mauvaises interprétations des données échangées. Pour cette étape, nous proposons la démarche suivante :

1. Identification de la sémantique du domaine métier :

Nous recherchons auprès des bases de données publiques telles que l'ANSI [113] et le W3C Semantic Web Activity [114] l'existence de standards d'interopérabilité définissant la sémantique du domaine métier et le vocabulaire à utiliser dans l'échange, l'intégration et le partage de l'information. Comme par exemple:

- ACORD [115] est défini pour décrire le domaine de l'assurance,
- HL7 [116] est défini pour décrire le domaine de la santé,
- NIEM [117] est défini pour décrire le domaine de la justice.

A ces standards s'ajoutent ceux utilisés par les partenaires eux-mêmes. Si aucune recommandation n'est trouvée, nous utiliserons le vocabulaire interne de l'entreprise.

2. Création du modèle structuré de données métier :

La modélisation de l'information du domaine sous forme de diagramme de classes et donc de la structure des documents métier est réalisée en se focalisant sur les interactions globales de l'architecture conceptuelle de l'étape 1. Une classe est un type de données complexe. Elle est identifiée à partir des objets métier (un objet physique, un fait tel qu'un évènement ou une action) qui forment le domaine métier. Le diagramme est enrichi en ajoutant les attributs et les traitements aux classes (Voir Annexe 4: Création des diagrammes de classe UML). Lorsque le modèle devient trop complexe, nous identifions des clusters (des groupements de classes) en créant des partitions là où se trouvent le moins d'associations entre les classes.

La Figure 6-7 illustre un exemple du modèle structuré de données correspondant au domaine « Achat en ligne » incluant les classes commande, sélection, produit, fournisseur, prix, devis, client, compte crédit et payement.

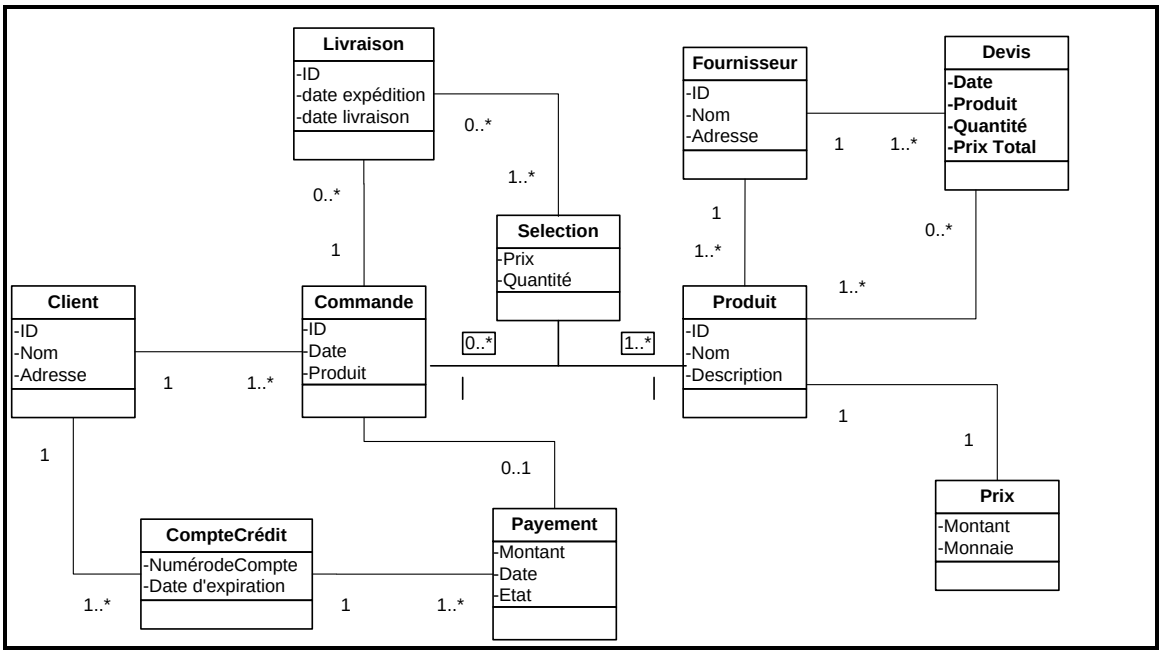


Figure 6-7 : Exemple modèle structuré de données : Achat en ligne

3. Modélisation des documents métier :

Pour modéliser les documents métier, nous reprenons le modèle structuré de données et nous regroupons les classes qui ont un rôle dans une interaction et qui caractérisent l'information à échanger. Ce regroupement forme le document métier, que nous nommons aussi objet métier, sujet métier ou catégorie selon le vocabulaire retenu par l'entreprise. Nous réalisons ce travail en suivant la méthodologie de cartographie en catégories décrite dans [12], où une catégorie présente les propriétés suivantes :

- ✓ Stable : La catégorie n'est pas propre à un projet particulier. Sa frontière ne varie pas en fonction des évolutions des traitements.
- ✓ Consistante : La catégorie décrit un concept métier spécifique. Elle est composée d'une classe pivot qui représente ce concept et de classes dérivées qui la décrivent.
- ✓ Mono Préoccupation : La catégorie est singulière. Elle ne contient que les classes qui décrivent un seul et unique concept métier
- ✓ Contiguë : Les classes qui composent la catégorie sont toutes en relation entre elles. Il n'y a pas de classes isolées.
- ✓ Nommée : Chaque catégorie dispose d'un nom unique la définissant.

Afin de prendre en compte la totalité des classes, la modélisation des documents doit être faite d'une façon itérative. L'amélioration de la sémantique des documents est faisable en alimentant les attributs des classes ou en ajoutant de nouveaux liens entre des classes. Enfin, tout changement majeur, tel que l'ajout d'éléments ou d'attributs obligatoires à un document, est nécessaire avant de publier le schéma du document puisque ces changements ne sont pas rétro-compatibles.

Dans notre exemple d'achat en ligne, le document métier 'Commande' inclut les informations du client, le compte crédit du client, les produits sélectionnés et le prix des produits. La classe 'Commande' représente la classe pivot de ce document (Figure 6-8). Nous encadrons cette classe et nous marquons les autres classes qui doivent faire parti du document métier et qui doivent être associées à la classe pivot (classes en bleu).

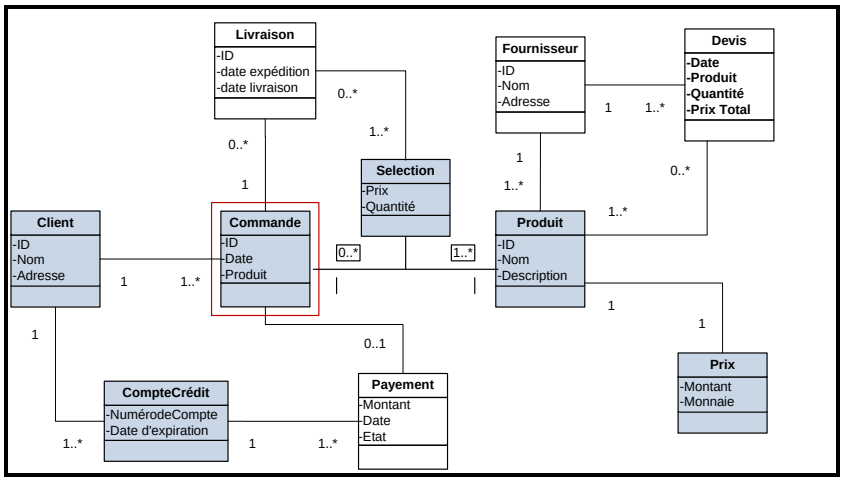


Figure 6-8 : Technique d'identification des documents

4. Création du schéma XML des documents métier :

Pour supporter les échanges en XML, le standard 'de facto' utilisé dans les infrastructures de services, nous créons le schéma XML de chaque document pour contraindre le document à un vocabulaire et une structure spécifiques. Les balises du schéma suivront la structure du document : éléments, sous-éléments, références et attributs. (Voir Annexe 5). L'utilisation de XML design patterns tels que : Russian Doll, Venetian Blind [1] permettra la création de meilleurs documents XML. Au cours d'une interaction, les documents métier reçus sont validés en les confrontant à leurs schémas. Cette validation permet de repérer différents types d'erreurs :

- ✓ les erreurs structurelles qui se rapportent à la structure hiérarchique du document métier.
- ✓ les erreurs sémantiques qui se rapportent au vocabulaire et aux attributs définissant les concepts du document métier.

Entrées	Langage de modélisation / Patrons de conception	Sorties (Résultats)
A partir de l'étape 1 et 2A : La liste des documents métier à modéliser	✓ Langage de modélisation UML ✓ Patrons de conception XML	✓ Modèle structuré de données métier ✓ Modèles des documents métier et schéma XML de ces documents métier

Tableau 6-4 : Tableau récapitulatif - Modélisation de l'information sémantique

[1] http://www.xmlpatterns.com/

Etape 3: Identification des objectifs de sécurité

Afin de concevoir une SOA sécurisée, les objectifs de sécurité doivent être pris en compte dès les premières phases de conception en considérant à la fois les aspects métier et technologiques. L'objectif de cette étape est de déterminer les objectifs de sécurité qui correspondent à la sécurisation des éléments du domaine métier. A ce stade de la méthodologie, il faut identifier et spécifier les objectifs de sécurité stratégiques qui correspondent aux acteurs, activités, processus et documents métier ainsi que sur la stratégie métier qui permet d'anticiper des objectifs de sécurité métier (le pourquoi) : par exemple une stratégie métier de développement de partenariat s'accompagnera d'un objectif de sécurité métier portant sur la création d'accord de protection pour sécuriser les échanges. L'identification de ces objectifs de manière globale permet de les prioriser et de cadrer le périmètre de l'étude. Pour cela, nous recourons à des ateliers de 'brainstorming' dans lesquels, nous réunissons les responsables métier et les responsables technique.

Les responsables métier sont les personnes qui maîtrisent les objectifs, la mission et le fonctionnement de l'organisation. Les responsables techniques sont les développeurs, les administrateurs systèmes et réseaux. Le(s) expert(s) de sécurité de l'entreprise sont les personnels impliqués dans la gestion des risques et / ou la gestion de la sécurité du SI. Si cette dernière fonction n'existe pas au sein de l'entreprise, les responsables techniques pourront acquérir les connaissances nécessaires en se référant aux méthodes de gestion des risques que nous recommandons au cours de la démarche.

Les ateliers permettent de cadrer le périmètre efficacement puisque les participants qui ont des compétences différentes voient la cible de l'étude à partir de perspectives différentes. Durant la séance de brainstorming, nous récapitulons les éléments métier de la première étape, en particulier :

- ✓ Les objectifs métier de l'organisation et les moyens à mettre en place pour atteindre ces objectifs (les stratégies, les politiques pour gouverner, contrôler ou guider)
- ✓ Les partenaires qui influent sur les stratégies et les objectifs métier de l'organisation.
- ✓ Les acteurs ainsi que leurs préférences de sécurité

A partir de ces éléments, nous définissons les objectifs de sécurité en se référant aux méthodes de gestion de risques [3] [95] qui présentent des catalogues et de bonnes pratiques et qui permettent de définir les objectifs de sécurité d'une façon systématique. Ces catalogues représentent un outil d'aide pour mener les discussions dans la séance de brainstorming. A titre d'exemple, la méthode EBIOS permet de rédiger une Fiche d'Expression Rationnelle des Objectifs de Sécurité (FEROS) présentant la démarche et l'argumentation qui permet d'identifier les objectifs de sécurité du système d'information en étude. La démarche d'élaboration d'une FEROS, illustrée dans l'Annexe 6 peut être suivie en se focalisant à ce niveau sur les éléments métier.

Les objectifs de sécurité stratégiques peuvent être définis en termes de :

- ✓ La création ou le respect du cadre de coopérations entre les acteurs (ex : mise en commun d'informations sensibles nécessitant des accords au niveau des échanges sécurisés)
- ✓ La vérification des exigences en matière de responsabilités des acteurs.
- ✓ La disponibilité et la continuité de l'ensemble des activités / processus métier ainsi que leur confidentialité
- ✓ La protection des documents métier confidentiels.
- ✓ La prise en compte des lois, règles ou règlements pouvant affecter l'environnement, l'accomplissement des missions ou influencer l'organisation.

En outre, la spécification des objectifs de sécurité permet de définir les exigences de sécurité à appliquer sur les éléments essentiels (des trois plans métier, service et infrastructure) formant le contexte. Cette spécification est réalisée dans l'étape 7A en suivant une approche descendante 'Top-Down' et en identifiant les :

- objectifs métier de sécurité : gestion, classification, confiance
- objectifs de base de sécurité : confidentialité, intégrité, disponibilité
- objectifs de support de sécurité : identification, authentification, contrôle d'accès, non-répudiation, audit.

Par exemple, un objectif de sécurité « Assurer la disponibilité des processus métier collaboratifs » permettra de dériver les exigences de sécurité suivantes :

1- Sur le plan métier :
   - ✓ Mettre en place un PLA avec les partenaires garantissant la disponibilité de leurs services
2- Sur le plan service :
   - ✓ Assurer la disponibilité des services composant les processus métier
3- Sur le plan infrastructure :
   - ✓ Assurer la disponibilité des éléments essentiels de l'infrastructure hébergeant les services.

Entrées	Méthodes / Outils	Sorties (Résultats)
A partir de l'étape 1 : Identification du domaine métier ✓ Le Quoi : Liste des domaines métier et des activités correspondantes ✓ Le Qui : Liste des acteurs et leurs rôles ✓ Le Pourquoi : Liste des motivations et des objectifs métier A partir des étapes 2A et 2B : La liste des processus métier et des documents métier manipulés.	Méthode : Les objectifs de sécurité sont identifiés dans des ateliers de brainstorming entre les responsables métier et techniques ---- Méthodes d'aide à l'identification des objectifs de sécurité : EBIOS, OCTACE	Liste des objectifs de sécurité

Tableau 6-5 : Tableau récapitulatif - Identification des objectifs de sécurité

Etape 4: Identification des services

Dans cette étape, notre objectif est d'identifier les services selon une approche 'Outside-In' pour assurer un alignement métier et applicatif et optimiser la réutilisation des services :

- ✓ La réutilisation des services permet de les partager entre différents processus métier.
- ✓ Le couplage lâche permet de réduire les dépendances entre les différents services, ce qui nous permet de créer des services autonomes.
- ✓ La 'responsabilité du service' permet d'intégrer dans le service les opérations qui font partie uniquement de son domaine de responsabilité. A titre d'exemple, le service « Gestion Client » dans le domaine métier « Achat en ligne » est le seul service qui est *responsable* de gérer, maintenir et mettre à jour les informations concernant les clients. Tout processus métier nécessitant des informations relatives aux clients utilisera ce service, ce qui assure la cohérence d'accès à l'information.

L'approche 'Outside-In' consiste à combiner les approches descendante et ascendante. Pour cela, nous réalisons simultanément deux démarches d'identification des services :

- ✓ une démarche d'ingénierie selon une approche top-down : à partir des processus métier modélisés dans l'étape 2A et des documents métier modélisés dans l'étape 2B, l'identification se déroule selon les étapes suivantes :
  1- Analyse des activités et identification des opérations des services métier
  Après décomposition des processus en activités atomiques, nous identifions les activités automatiques et semi-automatiques puisque chacune de ces activités sera liée à une opération d'un service métier [11].
  2- Identification des services composites
  Nous identifions d'abord les services composites qui répondent au contexte métier comme des compositions de services atomiques [118]. Pour cela, chaque processus/sous processus ne comportant que des activités automatiques/semi-automatiques est associé à un service composite.
  3- Identification des services atomiques :
  Une fois les services composites identifiés, nous passons à l'identification des services atomiques en utilisant à nouveau le modèle des processus/sous-processus. Pour cela, nous identifions les opérations des services atomiques (services de données, services centrées sur des tâches et services utilitaires) qui devront être invoquées afin de réaliser la logique métier du processus/sous-processus comme suit :
  a. Identification des services de données
  A partir des documents métier modélisés dans l'étape 2B correspondant au processus métier étudié, nous définissons les services de gestion de ces documents i.e. les services « CRUD » (CREATE, READ, UPDATE, DELETE). Par exemple, les services « Produit » et « Devis » sont définis pour gérer les documents métier correspondants.

b. Identification de services centrés sur les tâches.

Les activités atomiques qui encapsulent une logique métier feront l'objet de services centrés sur les tâches. Ces activités pourront être regroupées dans un même service pour réduire les dépendances entre les services. Pour cela, nous regroupons les opérations qui devront être sous la responsabilité du même service [119]. Ces opérations sont fonctionnellement liées et ne seront pas utilisées séparément dans d'autres processus.

c. Identification des services utilitaires

Certaines activités n'encapsulent aucune logique métier et peuvent être présentes au sein d'un ou plusieurs processus métier. Ces activités sont associés à des services utilitaires (ex : soumission, copie).

✓ une démarche d'ingénierie ascendante permet de prendre en compte l'existant du SI. Pour cela, il convient de rechercher auprès des répertoires de l'entreprise les solutions logicielles existantes qui répondent au besoin (services, composants d'applications, middleware et code source)

Dans ce qui suit, en l'absence de convention de nommage reconnue permettant de différencier les services de données des services centrées sur les tâches, nous proposons que le nom des services centrés sur les tâches reflètent les tâches à accomplir (ex : gestion prix) et que les services de données soient nommés par le nom de la classe pivot du document métier (ex : produit, devis). Dans ce qui suit, nous illustrons cette démarche au sous-processus création devis de la Figure 6-9 :

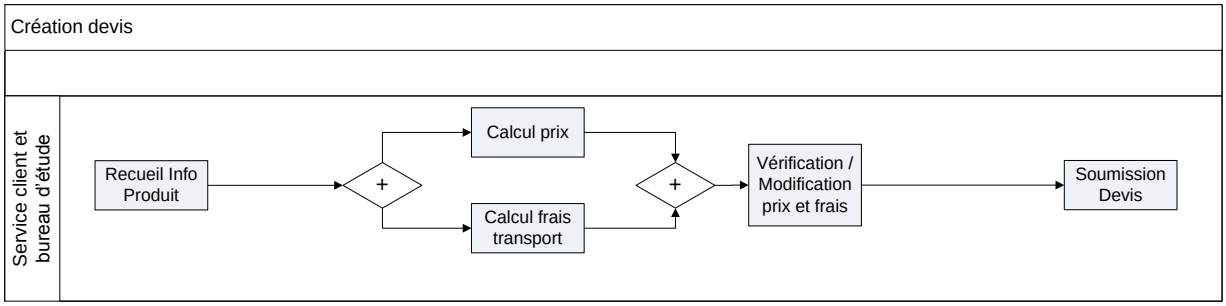


Figure 6-9 : Sous-processus métier création devis

1. Les activités identifiées dans ce sous-processus pourront être automatisées et feront l'objet d'opérations de services métier.
2. Les activités du sous-processus métier 'création devis' peuvent être complètement automatisées. Par conséquent, 'création devis' fera l'objet d'un service composite.
3. Nous identifions les services atomiques suivants :
    a. Nous identifions les services de données « Produit » et « Devis » pour la gestion des documents métier. Ces services sont de type CRUD.
    b. Nous identifions des opérations qui peuvent être offerte par un même service. Les opérations calcul prix et modification prix pourront être regroupées sous la responsabilité d'un seul service centré sur les tâches : 'Gestion prix'. De même,

les opérations 'calcul frais transport' et 'modification frais transport' sous la responsabilité du service 'Gestion frais transport'. Ce regroupement respecte la propriété 'réutilisation' étant donné que ces opérations ne seront pas utilisées séparément dans d'autres processus.

c. Le service soumission devis est un service utilitaire étant donné qu'il n'encapsule aucune logique métier.

Nous récapitulons dans le Tableau 6-6 la liste des services :

Nom du service	**Granularité**	**Type**	**Opération** nomOpération (nomParamètre : type) : typeDeRetour	**Services composants**
Produit	Atomique	service de données	✓ create (donnéeProduit : Produit) : string ✓ read (produitID : string) : Produit ✓ update (donnéeProduit : Produit) : Notification ✓ delete (produitID : string) : Notification	X
Devis	Atomique	service de données	✓ create (donnéeDevis : Devis) : string ✓ read (devisID : string) : Devis ✓ update (donnéeDevis : Devis) : Notification ✓ delete (devisID : string) : Notification	X
Gestion prix	Atomique	Service centré sur les tâches	✓ calculPrix (produitID : string, préférences : Préférence) : Prix ✓ modificationPrix(devisID : string, nouveauPrix, long) : Notification	X
Gestion frais transport	Atomique	Service centré sur les tâches	✓ calculFraisTransport(villeID : string, valeurs: long) : FraisTransport ✓ modificationFraisTransport(devisID : string, nouvellesvaleurs : long) : long	X
Soumission	Atomique	Service utilitaire	✓ soumission (données : string) : Notification	X
Création Devis	Composite	X	créationDevis (préférences : Préférence) : Devis	Produit, Facture, Prix, Frais de transport et soumission

Tableau 6-6 : Inventaire des services

Entrées	**Méthode / Caractéristiques**	**Sorties (Résultats)**
A partir des étapes 2A et 2B : ✓ La liste des processus métier ✓ La liste des documents métier à manipuler	Démarche d'identification des services	inventaire des services

Tableau 6-7 : Tableau récapitulatif - Identification des services

Etape 5 : Spécification des services

Avant d'entamer cette étape, nous rappelons que dans le cadre de la conception d'une SOA sécurisée, nous avons suivi une démarche orientée métier en nous focalisant sur les besoins de l'entreprise ainsi que sur les objectifs de sécurité métier. Nous avons modélisé les processus métier à partir desquels nous avons pu identifier des services.
Cette étape vise à documenter les spécifications des services (concernant leurs capacités et leurs exigences) tout en dissimulant les détails de l'implémentation.
Pour cela, nous nous basons sur le méta modèle SOA de CBDI [111] et sur la contribution de J. Amsden [120]. Ces travaux définissent les propriétés métier et techniques des services que nous récapitulons dans le Tableau 6-8.

	Description
Propriétés métier	Nom du service
	Objectifs métier
	Domaine métier
	Fournisseur
	Consommateurs ciblés
	Processus métier supportés par les services
Propriétés techniques	Liste des opérations (listant les capacités du service)
	Les dépendances: - Services desquels dépend ce service pour un bon fonctionnement - Services qui dépendent du service
	Séquence d'opérations : L'ordre dans lequel les opérations sont effectuées.
	Le SLA définit les paramètres de la qualité de service (performance, coût, etc.)
	Définition des messages échangés à partir du modèle d'information sémantique (schéma et emplacement d'un document XML).
	L'interface du service représente le groupement des opérations du service. Une opération peut appartenir à une seule interface.
	Le fichier de description définit au moins tous les noms d'opération et tous les messages d'entrée et de sortie pour chaque opération. Dans un environnement de service web, ce serait un le fichier WSDL (Web Services Definition Language)

Tableau 6-8 : Propriétés du service

Dans cette étape, nous nous focalisons sur les caractéristiques de l'interface des services sans aborder les technologies et les patterns d'implémentation (servlet, JBI, javabeans,..) pour garder notre étude indépendante de ces technologies. Toutefois, ces technologies et patterns d'implémentation devront être pris en compte avant la mise en œuvre du projet.

Note : nous ne définissons pas les paramètres de la sécurité dans cette étape (QoP, assertions de sécurité). Ils seront définis lors du cycle de gestion des risques pour une meilleure visibilité. Dans le Tableau 6-9, nous définissons les détails des opérations. Cette spécification est indépendante du langage de programmation, du format des messages et des protocoles.

Opération	Description	
Signature	De la forme « nomOpération (nomParamètre : type) : typeDeRetour »	
But	Intérêt de cette opération	
Style	Style d'interaction avec le service (ex : passage par document)	
Transmission	Unidirectionnelle ou requête/réponse	
Messages d'entrées	Messages / documents d'entrée de l'opération	
Messages de sorties	Messages / documents de sortie de l'opération	
Messages d'erreurs	Exceptions que peut lancer l'opération en cas d'erreur	
Pairs de pré-et-post-conditions	Pré-conditions	Conditions que doit satisfaire l'utilisateur avant d'invoquer l'opération du service.
	Post-conditions	Liste des conditions qui seront validées après l'exécution de l'opération.

Tableau 6-9 : Les opérations

Entrées	Méthode	Sorties (Résultats)
Inventaire des services	Etude des caractéristiques fonctionnelles et non fonctionnelles des services. Les paramètres de sécurité (QoP, assertions de sécurité) sont exclus de l'étude.	Spécification des services

Tableau 6-10 : Tableau récapitulatif - Spécification des services

Etape 6 : Etablissement du contexte de conception

D'après [121], l'établissement du contexte représente la définition des paramètres externes et internes à prendre en compte lors de la définition du périmètre de l'étude et de la gestion des risques. Dans notre travail, nous nous focalisons sur l'identification des biens à protéger dans un environnement de services dynamiques pour intégrer la gestion de la sécurité. De plus, nous nous intéressons à l'adaptation continue de la sécurité en cas de changement de contexte (changement des besoins organisationnels, implication de nouveaux acteurs, ajout de nouveaux services, changement au niveau de l'infrastructure technique, externalisation des services existants, etc.) La gestion de cette adaptation continue est un défi majeur pour le maintien du niveau de sécurité. Afin de répondre à ce défi, nous plaçons la gestion des risques au cœur de la conception et nous

développons notre méthodologie de façon itérative permettant d'adapter la sécurité en fonction des changements. L'objectif de cette étape est :

1. d'assurer une compréhension du contexte de conception commune entre les responsables métier et techniques en expliquant l'impact de la sécurité et les liens de dépendance entre les éléments essentiels des trois plans.
2. d'utiliser les processus métier comme cadre de référence pour identifier les éléments essentiels pertinents des trois plans d'abstraction.

Afin d'établir le contexte de conception, nous utilisons une démarche « Top-down » en allant du plan métier au plan infrastructure. La Figure 6-10 illustre le modèle de dépendance développé et qui représente cette démarche descendante (flèche à droite). Ce modèle met en évidence le lien de dépendance entre les éléments essentiels des trois plans. Les éléments essentiels du plan métier (éléments organisationnels, lois et obligations légales) cadrent les processus métier qui se composent d'activités manuelles et de services métier (associés aux activités automatiques/semi-automatiques). Les opérations des services métier sont réalisées par des services qui échangent des messages encapsulant des documents métier ou des données. Enfin, le service opère sur les éléments essentiels de l'infrastructure (les logiciels d'hébergement et de support, les systèmes d'exploitation, les équipements d'hébergement et les éléments de support). Nous soulignons le fait que le service n'est pas isolé et que la gestion de la sécurité dans un environnement de services ne pourra se faire sans la prise en compte des liens de dépendance. Une fois le modèle dépendance expliqué, nous définissons le périmètre de l'étude de gestion des risques. En effet, nous utilisons le modèle de dépendance pour identifier les éléments essentiels des plans métier, service et infrastructure associés à chaque processus métier modélisé dans l'étape 2A. Les éléments de l'infrastructure sont identifiés en se référant à l'inventaire des logiciels et à l'architecture réseau.

Cette étape implique de nombreux acteurs (décideurs, responsables métier, responsables techniques et analystes) qui doivent s'accorder précisément sur la délimitation du périmètre de l'étude et la description du contexte de conception. Les acteurs devront rassembler les différents éléments gouvernant les plans métier, service et infrastructure et devront créer les liens de dépendance entre ces différents éléments. Le résultat attendu est une description complète du contexte de conception (différent du contexte d'exécution qui regroupe les informations concernant l'invocation et l'utilisation des services comme la localisation des utilisateurs ou le moment de l'accès, etc.) ainsi qu'une identification précise des éléments essentiels à protéger.

Lors de cette étape, il faut rappeler aux responsables techniques que ce sont les objectifs métier qui cadrent le contexte de conception. Assurer une sécurité maximale revient à se focaliser sur les objectifs de sécurité puis en dériver les exigences de sécurité sur les plans métier, service et infrastructure. De manière duale, l'importance des plans services et infrastructures dans la

réalisation des objectifs métier et l'atteinte des objectifs de sécurité est mise en évidence pour justifier vis-à-vis des responsables métier tout investissement fait au titre de la sécurité.

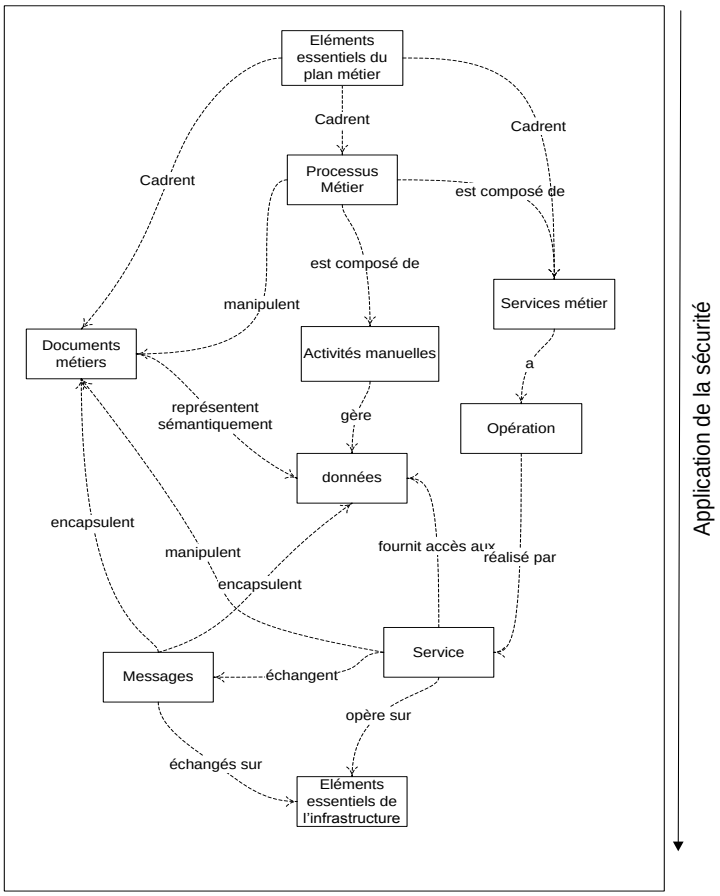


Figure 6-10 : Modèle de dépendance

Prenons à titre d'exemple, le processus 'achat en ligne' modélisé dans l'étape 2A, nous identifions les éléments essentiels métier associés à ce processus (partenaires, acteurs, rôles, documents métier, lois et obligations légales). Nous identifions ensuite :

- les éléments essentiels du plan service en choisissant les services composant le processus métier et en reprenant les résultats des étapes 4 et 5 pour chaque service (spécification des opérations, messages, documents métier/données).
- les paramètres de sécurité (assertions de sécurité, QoP) des services existants.

Une fois identifié les éléments des plans métier et service, nous identifions les composants de l'infrastructure (logiciels d'hébergement et de support, systèmes d'exploitation, équipements d'hébergement et équipements réseau) qui hébergent ces services et qui sont indispensables à leur exécution. Pour cela, nous nous référons à l'inventaire des logiciels et à l'architecture réseau pour construire la chaîne de liaison.

La Figure 6-11 illustre l'exemple d'une architecture réseau où un pare-feu découpe le réseau en deux sous-réseaux. La DMZ (de l'anglais demilitarized zone) est le sous-réseau qui sera accessible depuis le réseau public (Internet) en utilisant des règles de filtrages et des politiques de sécurité. A partir de cette architecture et de l'inventaire des logiciels (Tableau 6-11), nous pouvons établir le lien de dépendance entre les services et les éléments essentiels de l'infrastructure.

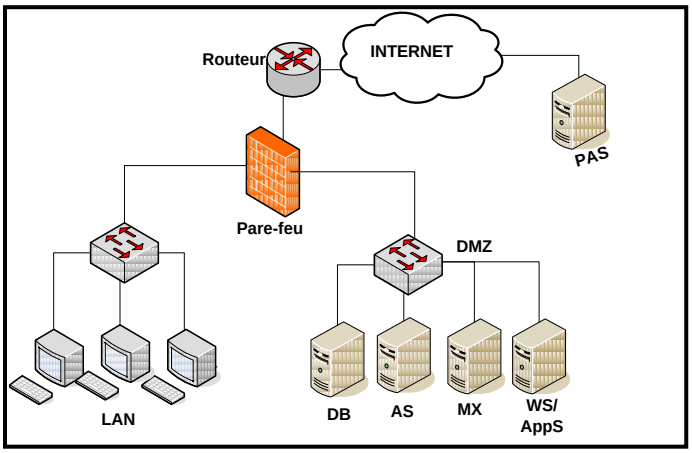


Figure 6-11 : exemple d'architecture réseau

Logiciels d'hébergement et de support	✓ Serveur d'application Apache Tomcat ✓ ESB : Petals [122] ✓ MO : Moteur d'orchestration ✓ BD : Serveur de base de données
Systèmes d'exploitation	✓ Debian Squeeze
Equipements	✓ Serveur d'hébergement redondant (double processeurs, disques RAID) ✓ Equipements d'interconnexion : Un routeur un pare-feu et un switch
Eléments de support	✓ Connexion Internet H-DSL de débit 10Mbits/s ✓ Alimentation redondante (Onduleur)

Tableau 6-11 : Exemple inventaire des logiciels

A titre d'exemple, nous construisons le lien de dépendance du service 'création devis' montrant que ce service dépend :

- Des logiciels d'hébergement et de support listés dans le Tableau 6-11.
- Du système d'exploitation 'Debian squeeze'
- Du serveur matériel qui l'héberge
- Des équipements réseaux et de la connexion Internet

Une fois ces éléments listés, nous demandons aux responsables métier de classer les éléments métier selon leur importance afin de les prioriser dans l'étude de gestion des risques (étape suivante) en adoptant l'échelle suivante :

- ✓ Critique : pour les biens qui nécessitent un haut niveau de protection.
- ✓ Important : pour les biens qui nécessitent un niveau de protection standard.
- ✓ Sans importance : pour les biens qui ne nécessitent pas de protection.

A leur tour, les responsables techniques utiliseront la classification des éléments métier et le lien de dépendance pour classer les services et les essentiels de l'infrastructure selon la même échelle.

Entrées	Méthodes	Sorties
Les éléments essentiels des plans métier et service	✓ Utiliser le modèle de dépendance pour déterminer les éléments essentiels pertinents pour les processus métier ✓ Se référer si nécessaire à l'architecture réseau pour déterminer et identifier les équipements et les éléments de support	Liste des éléments essentiels formant le contexte de conception

Tableau 6-12 : Tableau récapitulatif - Etablissement du contexte

Etape 7A : Identification des exigences de sécurité

Nous avons déterminé dans l'étape 3 les objectifs de sécurité métier qui couvrent l'architecture SOA dans sa globalité et les utilisons dans cette étape pour en déduire les exigences de sécurité pour les trois plans (métier, service et infrastructure) en suivant le modèle de dépendance et en allant du niveau métier au niveau infrastructure pour avoir une cohérence globale au niveau de la sécurité et optimiser l'investissement en implémentant les mesures de sécurité les plus adaptées au contexte de conception.

Comme pour la détermination des objectifs de sécurité, l'identification des exigences de sécurité se fait dans des ateliers de brainstorming regroupant responsables métier et responsables technique en utilisant des catalogues génériques comme ceux fournis dans la méthode EBIOS pour cette tâche (voir référence : Annexe 7). Dans ce qui suit, nous donnons une liste non exhaustive d'exemples d'exigences de sécurité dans les trois plans.

7A-1 Exigences de sécurité au niveau du plan métier :

Au plan métier, les exigences sont identifiées à partir des objectifs de sécurité comme par exemple :

- ✓ L'établissement d'un réseau de confiance et l'attribution d'un niveau de confiance aux différents partenaires ainsi que la propagation de l'identité au niveau des processus métier collaboratifs.
- ✓ La gestion des stratégies de collaboration entre les partenaires.
- ✓ La gestion des préférences de sécurité des utilisateurs (les droits et les obligations)
- ✓ Le contrôle d'accès au patrimoine de l'organisation. Qui peut accéder à l'information ?
- ✓ La classification des documents métier. Quels sont les documents confidentiels ?
- ✓ La gestion des 'Protection Level Agreement' PLA.
- ✓ La gestion des lois et des obligations légales.

Pour spécifier les exigences de sécurité au niveau des processus métier et des documents métier, nous utilisons la méthode d'annotation de [105]. Dans cette méthode, les processus et les données sont différenciés selon trois niveaux de sécurité.

- ✓ Blanc pour les processus ou les données ne nécessitant pas de protection.
- ✓ Gris pour les processus ou les données nécessitant un contrôle d'accès.
- ✓ Noir pour les processus ou les données nécessitant une confidentialité et une intégrité.

Cette stratégie d'annotation permet d'assurer une cohérence maximale surtout quand il s'agit d'une collaboration entre les organisations. En outre, elle permet de mettre en évidence les contraintes de la sécurité liées aux documents métier noirs manipulés par des processus métier blancs.

7A-2 Exigences de sécurité au niveau du plan service :

Les exigences de sécurité du plan service sont spécifiées à partir des exigences de sécurité du plan métier. A cette fin, nous reprenons la liste de services composant les processus métier et pour chacun de ces services, il faut documenter les exigences de sécurité en rapport avec les éléments essentiels.

- ✓ Au niveau des opérations du service, il faut préciser s'il existe un besoin de disponibilité ou de contrôle d'accès au service. En ce qui concerne le contrôle d'accès, nous considérons plusieurs cas comme décrit dans [105] :
    a. Le contrôle d'accès portant sur le service appelant.
    b. Le contrôle d'accès portant sur l'utilisateur faisant appel au processus métier, ce qui pourra être exprimé en fonction de l'identité de l'utilisateur ou bien de son rôle.
    c. Le contrôle d'accès portant à la fois sur le service appelant et l'utilisateur faisant appel au processus métier.
    d. Le contrôle d'accès portant sur l'utilisateur faisant appel au processus métier et sur l'ensemble des différents services invoqués dans le processus métier avant de faire appel au service en question.
- ✓ Au niveau des messages échangés, il faut préciser les besoins de confidentialité, d'intégrité et de non-répudiation.
- ✓ Au niveau des documents métier et des données, il faut préciser s'il existe un besoin de confidentialité et d'intégrité nécessitant des mécanismes de chiffrement ou de signature.

Dans le cas de processus métier collaboratifs, les exigences de sécurité identifiées pour les services externes devront être communiquées aux partenaires pour obtenir les garanties de sécurité correspondantes. A titre d'exemple, l'entreprise peut demander à un partenaire la garantie de la confidentialité des données privées de ses clients. Ceci fera l'objet d'une politique associée à un PLA à établir.

7A-3 Exigences de sécurité au niveau du plan infrastructure :

Une fois les exigences de sécurité identifiées au niveau des services, nous dérivons celles qui correspondent aux éléments de l'infrastructure. Nous utilisons plusieurs critères afin d'identifier les éléments critiques :

- ✓ La classification de l'élément de l'infrastructure (niveau critique, important ou sans importance) spécifiée dans l'étape 6 qui nous permet de prioriser les éléments à étudier.
- ✓ L'emplacement de l'élément au niveau de l'architecture : serveur sécurisé par un pare-feu, exposé directement sur le réseau, etc.
- ✓ Les mécanismes de sécurité mis en place (comme les règles de filtrage d'un pare-feu)
- ✓ L'accès logique aux logiciels d'hébergement et aux systèmes d'exploitation, l'accès physique aux équipements et au local technique et les mécanismes de contrôle d'accès qui leur sont associés.

Pour une meilleure visibilité, on peut se référer à l'architecture globale qui permet d'identifier les éléments sensibles de l'infrastructure avant de définir des exigences de sécurité sur ces éléments comme :

- ✓ Le contrôle d'accès et la mise à jour périodique de sécurité (application de patchs aux applications) au niveau des logiciels d'hébergement.
- ✓ Le durcissement des systèmes d'exploitation
- ✓ La mise en place de matériels redondants
- ✓ La mise en place d'une connexion réseau redondante

A la fin de cette étape, il faut synthétiser les exigences de sécurité identifiées sur la totalité des éléments essentiels utilisés par le processus métier étudié. Cette étape est cruciale dans l'identification des risques puisqu'elle nous permet de mettre en évidence les exigences de sécurité sur les éléments essentiels classés comme prioritaire dans l'étape 'Etablissement du contexte'. Un exemple de génération des exigences de sécurité à partir des objectifs de sécurité sera détaillé dans le cas d'usage développé dans le chapitre suivant.

Entrées	Méthodes /Catalogues	Sorties (Résultats)
Les objectifs de sécurité Les éléments essentiels du contexte de conception	Méthode : brainstorming / processus de transformation d'exigences Catalogue d'exigences de sécurité générique de EBIOS	Liste des exigences de sécurité sur les trois plans

Tableau 6-13 : Tableau récapitulatif - Identification des exigences de sécurité

Etape 7B : Identification des risques

Comme nous l'avons indiqué dans le chapitre relatif à la gestion des risques dans l'état de l'art, nous n'avons pas trouvé d'approche de modélisation des menaces dédiée aux SOA. Nous avons proposé d'utiliser la méthode de gestion des risques CORAS conjointement avec les catalogues génériques des méthodes EBIOS et OCTAVE (des extraits de ces catalogues sont fournis dans l'Annexe 7 et l'Annexe 8). En effet, CORAS offre un langage de modélisation et utilise une stratégie d'analyse basée sur les arbres pour l'identification des risques. Sa flexibilité et son pouvoir d'adaptation aux changements du contexte de l'étude rendent son application adéquate à un environnement de services dynamiques.

Pour cette étape, nous proposons toujours d'utiliser des ateliers de 'brainstorming' dans lesquels sont réunis les responsables métier et techniques. Ces ateliers débutent par le rappel des concepts liés aux éléments autour du risque avant de passer à l'identification de ces éléments, les évènements redoutés, les menaces, les scénarios de menace et les vulnérabilités. Un risque est associé à la probabilité d'occurrence d'un événement redouté et de son impact sur les éléments essentiels. Dans ce qui suit, nous détaillons cette démarche.

Dans un premier temps, il faut réaliser une analyse de risques de haut niveau en redéfinissant le risque par rapport au contexte des biens essentiels qui ont été identifiés :

1. Les événements redoutés sont les scénarios génériques que l'on souhaite éviter concernant le périmètre de l'étude.
2. Les menaces et leurs sources sont déterminées en se focalisant sur la cause initiale d'un événement redouté, ceci inclut:
    a. Le *Qui* : les menaces peuvent être créées par des humains ou des non humains (systèmes, virus, …)
    b. Le *Quoi* : la cause de la menace, accidentelle ou volontaire
3. Les scénarios de menaces sont les scénarios menant à ces événements redoutés.
4. Les vulnérabilités sont les faiblesses du système permettant aux menaces de causer les événements redoutés et par conséquent aboutissent aux risques.

1. Détermination des événements redoutés

Nous déterminons pour chaque élément essentiel, les événements redoutés associés en déterminant les événements qui peuvent nuire à ou réduire la valeur de cet élément essentiel. Les réflexions sont menées à partir de la vision métier puis dérivée sur les plans techniques. Le Tableau 6-14 liste des exemples d'événements redoutés de niveau métier et ceux qui en ont été dérivés dans les plans service et infrastructure :

Elément Essentiel	Evénement redouté (plan métier)	Evènement redouté dérivé (plan service)	Evènement redouté dérivé (plan infrastructure)
Partenaire	Non respect des accords de la QoP	Mauvaise application des politiques de sécurité	Indisponibilité du PEP (Policy enforcement point)
Rôles	Mauvaise gestion des rôles	Attribution de faux privilèges pour l'accès aux services	Attribution de faux privilèges pour l'accès aux logiciels d'hébergement
Document métier	Mauvaise classification des documents métier	Atteinte aux données stockées.	Atteinte aux données en échange.
Processus métier	Indisponibilité du processus métier	Indisponibilité des services composant le processus métier	Indisponibilité des logiciels d'hébergement / de la connexion réseau.

Tableau 6-14 : Identification des évènements redoutés

2. Détermination des menaces

Pour chaque événement redouté, nous déterminons la source de la menace et si cette menace est accidentelle ou volontaire. Nous rappelons que :

1. Toutes les sources possibles de menaces (humaines ou non humaines) sont prises en compte (un attaquant extérieur, un client, un partenaire, un employé, un virus, un système, un service, etc.)
2. Pour chaque événement redouté, il faut impérativement identifier toutes les menaces et leurs causes avant de créer le lien entre ces éléments.

Les catalogues de menaces des méthodes EBIOS (Annexe 8) et OCTAVE (Annexe 9) sont de bonnes références pour assister les responsables lors de l'identification des menaces sur les éléments essentiels des plans métier et infrastructure. L'identification des menaces au niveau du plan service est réalisée en identifiant celles liées à la technologie d'implémentation. Le guide du NIST [123] est une référence pour l'identification des menaces génériques portant sur les services web et les architectures orientées services. Dans ce guide, la sécurité est analysée en termes de besoins d'authentification, d'autorisation, de confidentialité, d'intégrité et de mesures de sécurité à mettre en œuvre pour assurer ces besoins. Ce guide intègre également les éléments liés à la sécurité des portails d'accès aux services, la confidentialité et l'intégrité des messages, le contrôle d'accès aux services et la sécurité de l'annuaire.

L'exploitation du modèle de dépendance permet l'identification des menaces sur les services issues du plan métier et les menaces sur les composants de l'infrastructure issues de celles des plans métier et service. Par exemple, une mauvaise gestion des droits d'accès au niveau du plan métier peut mener à l'attribution de faux privilèges au niveau des services et des composants de l'infrastructure.

Dans le Tableau 6-15, nous listons à titre d'exemple des menaces sur les plans service et infrastructure :

Menaces sur le plan service	**Menaces sur le plan infrastructure**
Attaque par déni de service sur un service (ex : envoi de documents de grande taille dans des messages SOAP)	Attaque par déni de service sur un serveur web, un ESB (ex : attaque TCP flood)
Détournement de l'usage prévu d'un service (ex : modification des opérations du service)	Scan des ports ouverts sur un serveur d'hébergement (ex : scan des ports TCP/UDP)
Attaque du type 'XML injection' ou 'XPath' (ex : attaque sur un service d'authentification résultant une élévation de privilèges)	Détournement de trafic réseau (ex : attaque sur un routeur et modification de sa table de routage)
Scan des interfaces des services pour découvrir des éventuelles failles (ex : wsdl scanning et découverte d'opérations privées déclarées dans l'interface par erreur)	Écoute du canal de transmission et collecte d'information. (ex : attaques du type man in the middle)

Tableau 6-15 : Identification de menaces

Une fois les menaces et les sources de menaces déterminées, nous créons un lien avec la ressource concernée en utilisant le langage CORAS (Voir le langage CORAS en Annexe 10) pour créer les diagrammes de menaces. Dans notre exemple de la Figure 6-12, nous illustrons deux menaces d'origine humaine dont la première, due à une erreur d'un personnel du fait du manque de compétence, est accidentelle et la deuxième, venant d'un intrus ayant accédé au système pour y conduire une attaque lui permettant d'écouter le trafic sur le canal, est délibérée. Ces menaces portent sur la compromission et l'altération d'un document métier confidentiel.

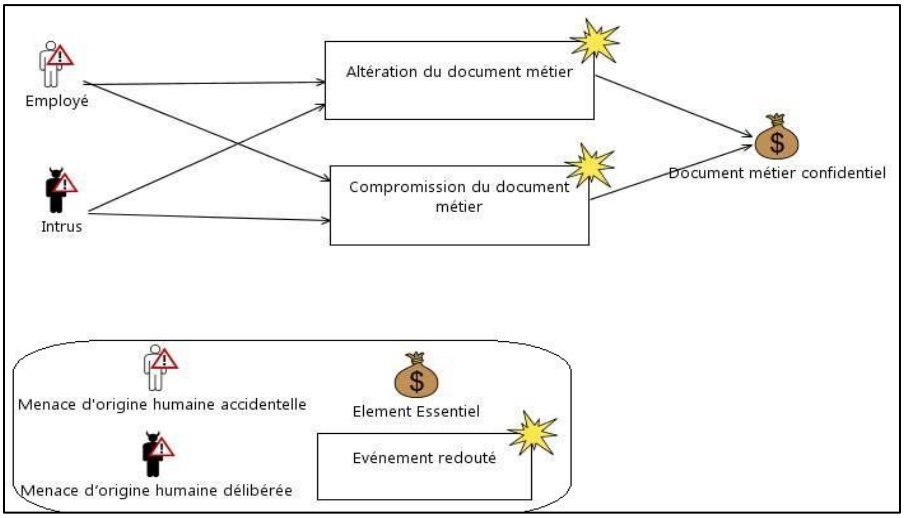


Figure 6-12 : Exemple diagramme de menaces

3. Détermination des scénarios de menaces

Les diagrammes des menaces sont utilisés lors de séances de brainstorming menées par les responsables techniques pour déterminer les scénarios de menaces correspondant. Les scénarios de menaces peuvent entrainer soit directement les événements redoutés soit d'autres scénarios de menaces dont résulte l'événement redouté. La Figure 6-13 illustre le scénario de menace 'écoute du canal de transmission' correspondant à l'événement redouté 'compromission d'un document métier confidentiel'.

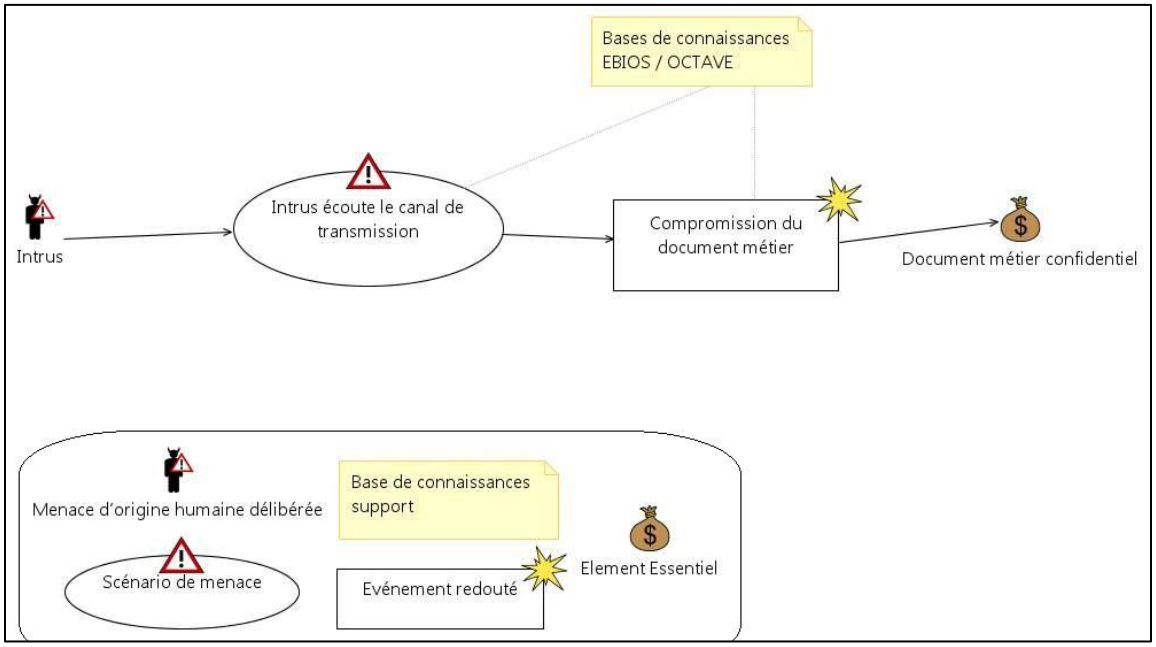


Figure 6-13 : Exemple de modélisation des scénarios de menaces

4. *Détermination des vulnérabilités*

La dernière tâche de cette étape est l'identification des vulnérabilités qui permettent de décrire la façon dont les menaces et les scénarios de menaces deviennent possibles. Les vulnérabilités sont :

a. Des éléments utilisés intentionnellement ou exploités par une menace
b. Des éléments empêchant le rétablissement suite à une attaque [97]

Pour chaque événement redouté et pour chaque scénario de menace, nous recherchons de manière systématique les vulnérabilités correspondantes en utilisant :

- ✓ les catalogues des vulnérabilités génériques des méthodes EBIOS, OCTAVE pour déterminer les vulnérabilités au niveau des plans métier et infrastructure.
- ✓ l'étude faite dans [59] pour déterminer les vulnérabilités génériques au niveau des services.
- ✓ les bases de vulnérabilités du CERT et du MITRE pour déterminer les vulnérabilités spécifiques aux éléments de l'infrastructure.

Enfin, nous alimentons les digrammes par les vulnérabilités identifiées. Dans notre exemple de la Figure 6-14, l'écoute du canal de transmission est due à un canal de transmission non sécurisé.

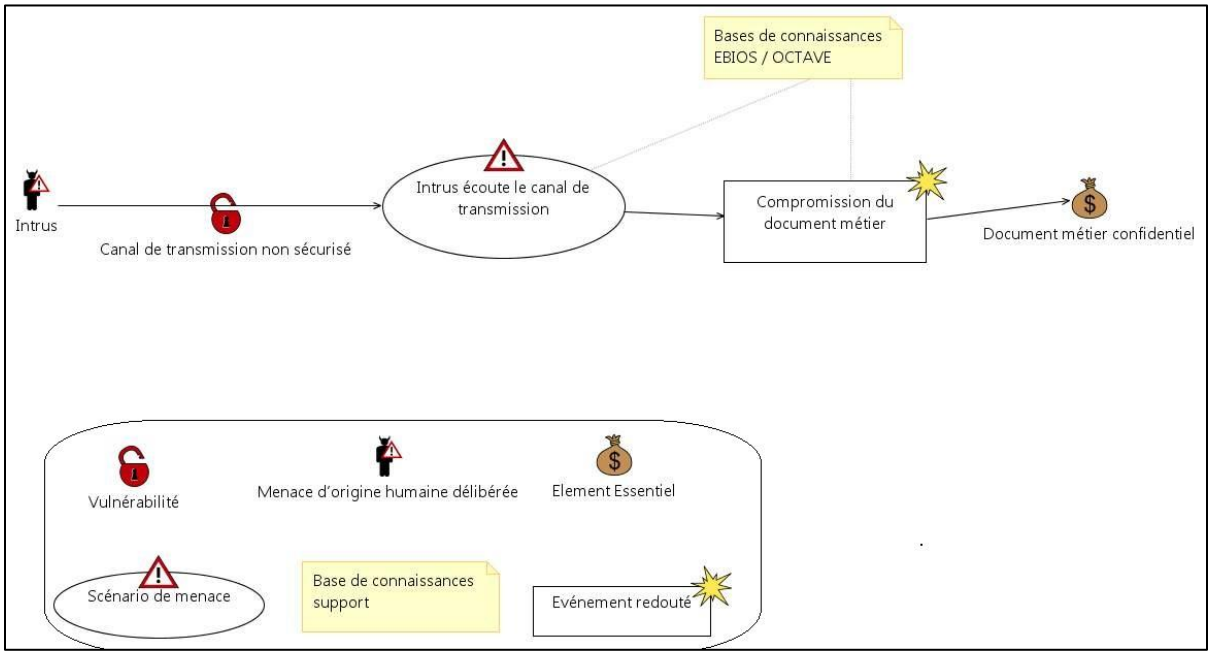


Figure 6-14 : Exemple de modélisation des vulnérabilités

Entrées	Méthodes / Langage	Sorties (Résultats)
- Les éléments essentiels du contexte - Les exigences de sécurité sur les différents plans	- Catalogues de menaces et de vulnérabilités génériques de EBIOS et d'OCTAVE - Bases de vulnérabilités du CERT et du MITRE - Langage de modélisation CORAS	Diagrammes des éléments autour du risque (évènements redoutés, menaces, scénario de menaces, vulnérabilités)

Tableau 6-16 : Tableau récapitulatif - Identification des risques

Etape 8 : Evaluation des risques

Dans cette étape, notre objectif est d'estimer le niveau des risques qui ont été identifiés et de déterminer leur gravité afin d'établir des priorités et de déterminer les risques qui doivent être traités. Dans un premier temps, nous établissons les échelles de l'évaluation de l'impact et des probabilités d'occurrence avant de définir les fonctions de tolérance aux risques et d'évaluer les risques. Cette tâche est réalisée dans une séance de brainstorming entre les responsables techniques et métier.

1. *Échelle de l'impact des évènements redoutés*

L'échelle de l'impact définit les valeurs qualitatives ou quantitatives utilisées pour estimer l'impact des évènements redoutés. Cette échelle est établie en termes d'atteinte à ou de réduction de valeur pour un élément essentiel et/ou pour l'organisme (perte d'image, perte financière, impact sur la productivité, la crédibilité, etc.) de manière similaire à ce qui est fait dans d'autres méthodes (EBIOS, OCTAVE…). Une échelle quantitative permet une caractérisation précise de l'atteinte de la valeur des ressources, mais exige de disposer de l'expertise et des données nécessaires pour estimer l'impact. L'utilisation des valeurs qualitatives, telles que significatif ou important peut souvent être plus appropriée, mais exige une description précise des valeurs de cette échelle et qu'il y ait une différence significative entre ces valeurs afin de pouvoir distinguer les niveaux d'impact et pouvoir ensuite prioriser les risques.

Pour chaque élément essentiel, nous proposons une échelle d'impact de 4 niveaux : {insignifiant, mineur, majeur et catastrophique} et évaluons les risques selon les exigences de sécurité. Par exemple : si un portail web a été identifié comme un élément essentiel exigeant une forte disponibilité, on peut définir l'échelle suivante mesurant l'impact de l'indisponibilité (Tableau 6-17):

Impact	**Description**
Catastrophique (valeur = 4)	Indisponibilité de [1 jour, ∞]
Majeur (valeur = 3)	Indisponibilité de [1 heure, 1 jour]
Mineur (valeur = 2)	Indisponibilité de [1 minute, 1 heure]
Insignifiant (valeur = 1)	Indisponibilité de [0, 1 minute]

Tableau 6-17 : Exemple impact indisponibilité du portail web

2. *Échelle de la probabilité d'occurrence des évènements redoutés*

Comme l'échelle de l'impact, l'échelle de la probabilité d'occurrence peut être quantitative ou qualitative. Une échelle quantitative peut être appropriée lorsque nous avons accès à des données précises sur une période significative concernant les fréquences des événements redoutés. Les valeurs qualitatives sont quant à elles adaptées aux cas où il est difficile voire impossible de trouver des estimations de probabilité exactes. Dans ce cas-là, le plus important serait de garder la cohérence dans les décisions prises tout au long de l'étude.

Nous établissons une échelle de probabilité à 4 niveaux (Tableau 6-18): {rare, possible, probable et certain} tout en fixant la durée de référence pour estimer les probabilités d'occurrence. Par exemple, nous fixons une durée de vie de 5 ans pour un processus métier et nous utilisons la notation [*min*, *max*] : *période*

Probabilité d'occurrence	**Description**	
Certain (valeur = 4)	[20, ∞[:5 ans	4 ou plus par an
Probable (valeur = 3)	[6,19] :5 ans	1 à 4 fois par an
Possible (valeur = 2)	[2,5] :5 ans	Moins d'une fois par an
Rare (valeur = 1)	[0,1] :5 ans	Moins d'une fois dans les 5 ans

Tableau 6-18 : Exemple probabilité d'occurrence d'un événement redouté

3. *La fonction de tolérance au risque*

Le niveau du risque est défini en fonction de la probabilité d'occurrence d'un événement redouté et de son impact or un événement redouté peut impacter différents éléments essentiels. Le niveau du risque associé doit donc être évalué pour chaque élément essentiel en tenant compte de la probabilité d'occurrence et de l'impact sur l'élément considéré. Ensuite, une fonction de tolérance est définie. Pour cela :

1- Nous définissons la fonction du risque pour chaque élément essentiel pour chaque couple impact/probabilité (selon une échelle à trois valeurs)
2- Nous définissons pour chaque couple impact/probabilité le niveau de tolérance ajusté en fonction du coût de la sécurité (budgets associés, rémunération des responsables techniques, coût des produits de sécurité, des formations, veille technologique en matière de sécurité, gestion de la crise et de la reprise des activités etc.) qui est en suite comparé au coût de 'non-sécurité' (coûts induits par des pertes d'image, impacts au niveau des clients, partenaires, baisse de productivité, perte de part de marché, etc.)

Dans le Tableau 6-19, nous avons défini la fonction du risque liée au 'portail web' en définissant trois niveaux de risque :

- ✓ le niveau faible (couleur bleue) : valeur du risque entre 1 et 3 (impact x probabilité d'occurrence)
- ✓ le niveau moyen (couleur jaune) : valeur du risque entre 4 et 8.
- ✓ le niveau élevé (couleur rouge) : valeur du risque entre 9 et 16.

		Impact			
		Insignifiant (1)	**Mineur (2)**	**Majeur (3)**	**Catastrophique (4)**
Probabilité d'occurrence	**Rare (1)**				
	Possible (2)				
	Probable (3)				
	Certain (4)				

Tableau 6-19 : Exemple fonction du risque liée au 'portail web'

Dans le Tableau 6-20, nous définissons la fonction de tolérance au risque lié au 'portail web'. Nous définissons les risques faibles et moyens comme des risques acceptables. Dans cette matrice, un événement redouté « *Rare* » et dont l'impact est « *Insignifiant* » présente un risque

acceptable. Un événement redouté « *Certain* » et dont l'impact est « *Majeur* » présente un risque inacceptable donc nécessite un traitement.

		Impact			
		Insignifiant (1)	**Mineur (2)**	**Majeur (3)**	**Catastrophique (4)**
Probabilité d'occurrence	**Rare (1)**	Risque Acceptable	Risque Acceptable	Risque Acceptable	Risque Acceptable
	Possible (2)	Risque Acceptable	Risque Acceptable	Risque Acceptable	Risque Acceptable
	Probable (3)	Risque Acceptable	Risque Acceptable	Risque Inacceptable	Risque Inacceptable
	Certain (4)	Risque Acceptable	Risque Acceptable	Risque Inacceptable	Risque Inacceptable

Tableau 6-20 : Exemple fonction de tolérance au risque lié au 'portail web'

4. *Évaluation des risques*

Une fois la fonction de tolérance au risque définie pour chaque élément essentiel, nous procédons à l'évaluation effective des risques.

En utilisant les échelles établies précédemment, nous annotons les diagrammes des risques par :

- Les probabilités d'occurrence des événements redoutés qui sont calculées en fonction des probabilités d'occurrence des scénarios de menaces et des probabilités que les scénarios de menaces mènent aux événements redoutés. Les responsables métier et techniques utilisent l'information du contexte (métier et technique) pour estimer ces probabilités.
- L'impact des événements redoutés sur les biens essentiels est estimé en fonction de la gravité de l'événement redouté.

Enfin, nous nous référons à la fonction de tolérance au risque pour l'évaluation du risque.

Pour illustrer notre démarche, nous considérons un exemple simplifié de l'atteinte à la disponibilité du portail web à cause de problèmes au niveau du serveur d'hébergement et du routeur (Figure 6-15) :

- ✓ La probabilité d'occurrence du scénario 'Crash du serveur d'hébergement' est 'rare' ([0, 1] :5) à cause de la redondance du serveur.
- ✓ La probabilité d'occurrence du scénario 'Crash du routeur' est 'possible' ([2, 5] :5) à cause des erreurs affichés dans les logs (problème au niveau des cartes réseaux)

Nous attribuons une probabilité 0.9 que ces deux scénarios de menace mènent à l'indisponibilité du portail web.

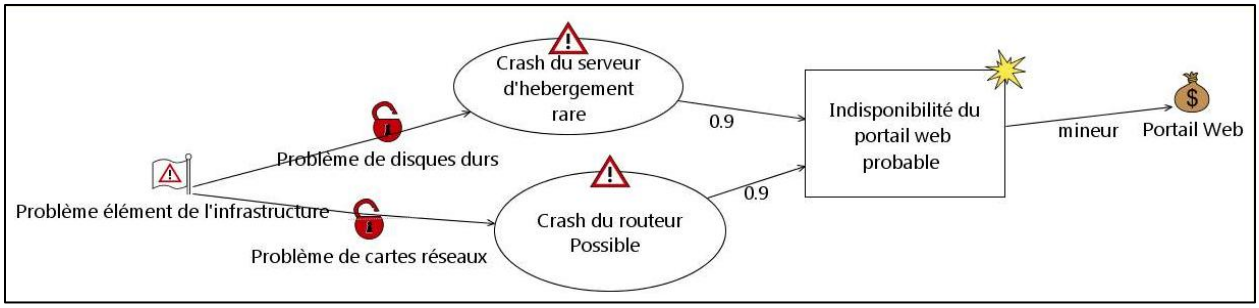


Figure 6-15 : Exemple évaluation des risques

Nous calculons la probabilité d'occurrence de l'indisponibilité du portail web en considérant que ces deux scénarii sont indépendants (Tableau 6-21). La valeur globale de la probabilité d'occurrence est [1.8, 5.4] :5 et donc 'Probable'.

	Scénario de menace (1) : Crash du serveur d'hébergement	Scénario de menace (2) : Crash du routeur
Probabilité d'occurence	Rare : [0, 1] :5	Possible : [2, 5] :5
Probabilité que le scénario conduit à l'évènement redouté	0.9	0.9
Valeur combinée	[0, 1] :5 x 0.9 = [0, 0.9] :5	[2, 5] :5 x 0.9 = [1.8, 4.5] :5
Valeur globale de la probabilité d'occurrence de l'évènement redouté	[0, 0.9] :5 + [1.8, 4.5] :5= [1.8, 5.4] :5 [1.8, 5.4] :5= Probable	

Tableau 6-21 : Exemple probabilité d'occurrence de 'l'indisponibilité du portail web'

Une fois la probabilité d'occurrence calculée, il faut estimer l'impact de cet événement redouté. Dans cet exemple, l'indisponibilité est due à des défaillances d'équipements d'infrastructure auxquelles on peut remédier rapidement ce qui conduit à évaluer un impact mineur en termes de réduction de valeur du portail web et d'image de l'organisme. Comparé à la matrice de tolérance au risque, ce risque est donc acceptable.

Entrées	Méthodes	Sorties (Résultats)
- Les évènements redoutés - Les diagrammes des éléments autour du risque	Identification des critères d'évaluation du risque - Echelle de l'impact - Echelle de la probabilité d'occurrence - Fonction de tolérance au risque Evaluation des risques	Liste des risques acceptables et non-acceptables

Tableau 6-22 : Tableau récapitulatif - Evaluation des risques

Etape 9 : Traitement des risques

Dans l'étape précédente, nous avons évalué les risques et déterminé ceux qui sont acceptables et ceux qui ne le sont pas. Dans cette étape, nous nous focalisons sur le traitement des risques évalués 'inacceptables'. Notre objectif est de les ramener à un niveau acceptable en étudiant les moyens à mettre en œuvre pour réduire la probabilité d'occurrence et/ou l'impact.
Pour cela, il faut examiner attentivement toutes les informations concernant les menaces, les vulnérabilités et les scénarios de menace conduisant à l'événement redouté et essayer d'identifier les moyens pour éliminer ces éléments ou réduire leur impact. Le traitement des risques se base sur des catégories de traitement qui représentent des actions à entamer pour réduire ou éliminer un risque en l'évitant, le transférant, le réduisant ou l'acceptant. Le choix de la catégorie se fait en fonction du coût de la sécurité confronté au coût de non sécurité.

1. *Le transfert et le partage du risque*

Le transfert du risque consiste à transférer le risque, ou une partie du risque, à un tiers ce qui conduit de fait à la conclusion de partenariat et/ou à une externalisation partielle aux niveaux services et infrastructures (ex : hébergement cloud)

2. *L'évitement du risque*

Éviter le risque revient à changer le contexte de conception de telle sorte qu'on ne soit plus exposé à l'événement redouté. Cela peut se traduire par des mesures de sécurité telles que le changement de localisation géographique, le fait de ne pas commencer ou poursuivre l'activité porteuse du risque, etc. [3]

3. *La réduction du risque*

La réduction du risque consiste à prendre des mesures de sécurité pour diminuer l'impact et/ou la probabilité d'occurrence d'événements redoutés en fonction des exigences de sécurité pesant sur l'élément essentiel considéré. Comme nous l'avons détaillé dans l'ontologie OCSS, les mesures de sécurité peuvent être de différents types :

- les protocoles de sécurité définissent une série d'étapes pour accomplir une tâche
- les mécanismes de sécurité représentent la mise en œuvre ou l'implémentation des protocoles
- les politiques définissent les règles pour assurer un état sécurisé des systèmes
- les services de sécurité encapsulent une logique de sécurité qu'ils pourront offrir aux applications.

4. *La prise du risque*

La prise du risque est la décision d'accepter un risque lorsque la réduction du risque, son transfert ou son évitement s'avèrent impossible ou non rentables. C'est à ce stade de la méthodologie qu'on décide que la non-sécurité est la solution et que le risque ne nécessite aucun investissement.

La réduction des risques peut se faire sur les différents plans : métier, service ou infrastructure. Les moyens à mettre en œuvre peuvent être par exemple :

- ✓ Au niveau du plan métier : Classification des partenaires et organisation de l'accès au patrimoine informationnel.
- ✓ Au niveau des services : Chiffrement des messages, mise en place d'une stratégie de redondance des services critiques.
- ✓ Au niveau de la couche infrastructure : Mise en place d'un équipement de sécurité (pare-feu, IDS, etc.) ou nouvelle configuration d'un équipement existant (nouvelles règles de filtrage).

Dans le Tableau 6-23, nous donnons des exemples de mesures de sécurité:

Mesures de sécurité	Description
Propagation de l'identité (SSO[2])	Mettre en place un mécanisme permettant d'assurer la gestion et la propagation de l'identité des utilisateurs
Chiffrement et signature partiels des messages	Utilisation des standards WS-Security pour chiffrer ou signer des parties des messages qui passent par des services intermédiaires avant d'atteindre la destination.
Serveur redondant	La disponibilité des services dépend aussi de la disponibilité du matériel d'hébergement. Des disques miroirs (RAID) peuvent être mis en place.

Tableau 6-23 : Exemples mesures de sécurité

Entrées	Méthode	Sorties (Résultats)
La liste des risques inacceptables	Evaluation du coût de la sécurité	Liste des mesures de sécurité adéquate au contexte.

Tableau 6-24 : Tableau récapitulatif - Traitement des risques

Etape 10 : Annotation des services

L'objectif de cette étape est d'enrichir la description des services par des paramètres de sécurité qui pourront améliorer la sélection des services à l'exécution. Évidement, pour un consommateur, un service conçu d'une façon sécurisée peut être plus avantageux qu'un autre offrant les mêmes fonctionnalités, mais n'offrant pas de garantie de sécurité. Cette étape a comme entrée les résultats de l'étape 8, dans le cas où le niveau du risque est acceptable ou bien les résultats de l'étape 9 dans le cas où le niveau du risque est inacceptable. Nous annotons le

[2] L'identification unique (en anglais *Single Sign-On* : SSO) est une méthode permettant à un utilisateur de ne procéder qu'à une seule identification pour accéder à plusieurs applications.

service avec des paramètres qui permettront de garantir que le service a été conçu d'une façon sécurisée et que ce service est lié/utilise des éléments essentiels eux-mêmes sécurisés. En d'autres termes, l'annotation reflète le niveau de sécurité global suite au cycle de gestion des risques et à la définition des mesures de sécurité mises en place. Toutefois, nous n'aborderons pas la garantie du niveau d'assurance. Ce dernier point est en rapport avec la fiabilité des mesures de sécurité elles-mêmes (voir les EAL (Evaluation Assurance Level) des critères communs (Common Criteria) [106]) A titre d'exemple, nous considérons que la mise en place d'un serveur redondant améliore la disponibilité du serveur indépendamment de la fiabilité du mécanisme de sécurité (l'objectif est d'avoir une mesure de sécurité quand cela est nécessaire).
Afin d'annoter les services, il est important de répondre aux questions suivantes :

1. Comment peut-on profiter de l'étude de gestion des risques effectuée à la conception ?
2. Comment l'annotation va-t-elle pouvoir garantir la satisfaction des contraintes de sécurité sans divulguer les faiblesses du service ?
3. Comment représenter l'annotation, quels sont les standards que nous pouvons utiliser et quel est le format de l'annotation ?

Nous abordons les deux premières questions dans la partie suivante 'Annotation de sécurité' et la troisième dans la partie '6.2.4 Utilisation des politiques pour représenter l'annotation de sécurité' (p. 168)

6.2.3 Annotation de Sécurité

La dernière étape de la méthodologie représente la synthèse du travail de gestion des risques effectué en produisant l'annotation des services par des paramètres de sécurité. En effet, l'annotation de sécurité a un double rôle :

- Pour le fournisseur de service, elle représente un niveau d'une sécurité globale.
- Pour un consommateur de services, elle peut être utilisée pour améliorer la sélection entre différents services qui répondent aux exigences fonctionnelles (surtout dans un environnement ouvert)

Dans ce qui suit, nous développons l'Ontologie Annotation de Sécurité « OAS » en mettant en évidence les concepts suivants :

a. La *disponibilité* : Cet élément garantit la disponibilité du service et des autres éléments essentiels dont le service dépend.
b. La *confidentialité*: Cet élément garantit que les données échangées ou stockées par le service sont protégées contre tout accès non autorisé.
c. La *supervision* : Cet élément garantit que les opérations du service fonctionnent comme prévu et qu'il existe une garantit des paramètres de la qualité de protection.

Ces concepts, que nous intitulons 'éléments de l'annotation', permettent au service de garantir que :

- ✓ Les services sont disponibles pour être réutilisés à tout moment.
- ✓ Les données sensibles échangées ou au repos sont sécurisées.
- ✓ Les opérations et les paramètres de la qualité de protection des services sont contrôlés.

En outre, pour garantir que l'annotation de sécurité ne divulgue pas les faiblesses du service (en donnant des informations favorisant les attaques), nous proposons *une annotation de sécurité calculée* à partir des éléments essentiels et des mesures de sécurité mises en place.

Pour choisir les éléments de l'annotation, nous avons utilisé la dépendance entre les éléments essentiels et les travaux exposés dans [124] et [125] dans lesquels Microsoft et IBM abordent la sécurité des services selon deux axes : la sécurité physique qui se base sur la construction d'une infrastructure sécurisée et la sécurité logique relative aux logiciels d'hébergement et à la protection des données privées des consommateurs. Nous nous sommes limités à ces trois éléments comme étant les besoins de sécurité communs et essentiels des consommateurs. Toutefois, l'ajout d'autres éléments à l'annotation pourra être réalisé en suivant une démarche similaire par exemple, le concept 'intégrité' peut permettre de demander une garantie sur l'intégrité de l'information et le concept 'gestion des vulnérabilités des éléments logiciels' (logiciels d'hébergement et systèmes d'exploitation) pour garantir que la gestion des vulnérabilités de ces logiciels est prise en compte.
Nous notons que chaque service pourra être annoté par un ou plusieurs éléments de l'annotation. Par exemple, nous annotons le service par l'élément confidentialité uniquement si ce service gère des données confidentielles.

La Figure 6-16 illustre l'ontologie OAS dans laquelle nous spécifions les trois éléments de l'annotation. Ces éléments représentent les exigences de sécurité à satisfaire. Par exemple, l'élément de l'annotation 'disponibilité' représente la disponibilité (exigence de sécurité pertinente pour la disponibilité) des éléments essentiels suivants : opérations du service, logiciels d'hébergement et de support, système d'exploitation, équipements et éléments de support. Ces éléments essentiels sont pertinents pour évaluer la disponibilité du service.

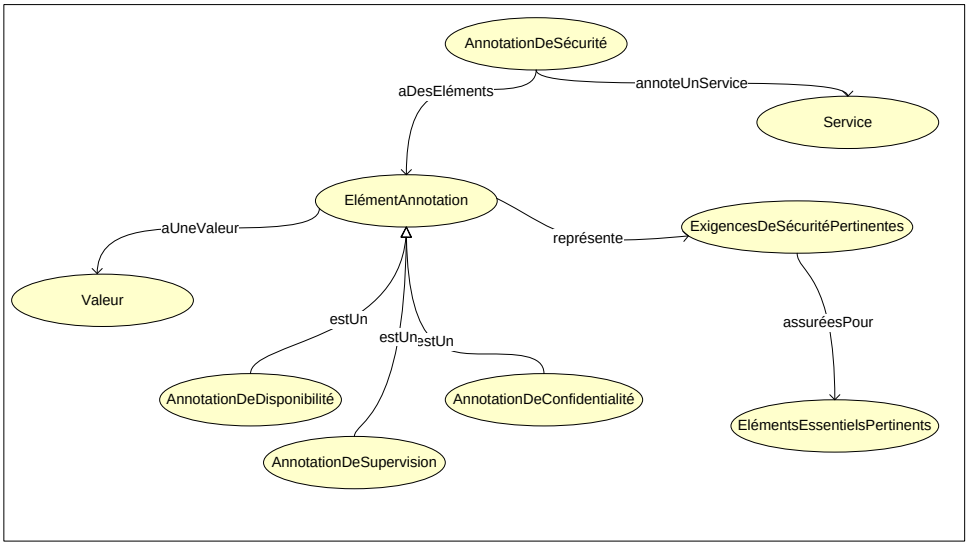


Figure 6-16 : Ontologie Annotation de Sécurité

A chaque élément de l'annotation est attribuée une valeur globale qui est le résultat d'un travail d'audit de la sécurité. Le calcul de la valeur se fait à la base de l'Equation 1 :

$$\text{VEA} = \sum_{i=1}^{n} \frac{x_i}{N}$$

Equation 1 : Valeur de l'élément de l'annotation

où VEA est la valeur de l'élément de l'annotation.
x_i est le nombre d'Instances des Eléments Essentiels Pertinents Sécurisés (IEEPS).
N est le total des Instances des Eléments Essentiels Pertinents (IEEP).
Les éléments essentiels pertinents sont identifiés à partir du modèle de dépendance pour spécifier les éléments de l'annotation.
Dans ce qui suit, nous définissons pour chaque élément de l'annotation les éléments essentiels pertinents à évaluer.

6.2.3.1 L'élément de l'annotation 'disponibilité'

Pour pouvoir déterminer la disponibilité des services, il est primordial de se baser sur le modèle de dépendance qui nous permet de calculer une valeur de disponibilité globale. En d'autres termes, la disponibilité des services est remise en question si les logiciels d'hébergement présentent des failles de sécurité ou bien s'il existe un problème réseau qui rend le service inaccessible. En se focalisant sur les éléments essentiels des plans service et infrastructure, nous en déduisons que la disponibilité d'un service représente la disponibilité de ses opérations et par conséquent dépend de celle de tous les éléments de l'infrastructure. Ci-dessous la description formelle de l'annotation de disponibilité :

$$AnnotationDeDisponibilité \subseteq$$
$$(\forall\ aUneDisponibilitéAssurée.Opération) \wedge$$
$$(\forall\ aUneDisponibilitéAssurée.LogicielD'HébergementEtSupport) \wedge$$
$$(\forall\ aUneDisponibilitéAssurée.SystèmeExploitation) \wedge$$
$$(\forall\ aUneDisponibilitéAssurée.Equipement) \wedge$$
$$(\forall\ aUneDisponibilitéAssurée.ElémentSupport)$$

Le Tableau 6-25 illustre un exemple simplifié permettant de calculer le niveau de disponibilité du service 'création devis' ne comportant que quatre instances des éléments essentiels de l'infrastructure : un serveur d'application, un serveur web, un routeur et la connexion Internet.

Service 'création devis'	Eléments essentiels	Mesures de sécurité	Disponibilité Assurée
Plan service	Opération	Aucune	-
Plan Infrastructure	Serveur d'application	Mise à jour de sécurité périodique	+
	Serveur Web	Mise à jour de sécurité périodique	+
	Routeur	Aucune	-
	Connexion Réseau	Redondance	+

Tableau 6-25 : Exemple Valeur calculée de la disponibilité

La valeur '+' signifie qu'une mesure de sécurité assure la disponibilité des éléments essentiels, la valeur '-' signifie que la disponibilité n'est pas assurée. Cet exemple montre que parmi les instances des éléments essentiels des plans service et infrastructure, cinq instances sont concernées par la disponibilité dont trois leur ont été attribués des mesures de sécurité et par la suite la valeur calculée de la disponibilité est de 3/5 = 60%.

6.2.3.2 L'élément de l'annotation 'confidentialité'

La confidentialité des données repose à la fois sur la protection des données au repos et sur la protection de celles en transit. Nous nous intéressons alors à l'évaluation des aspects suivants:

- ✓ La confidentialité des documents métier, des données et des messages pour assurer la protection des données privées en transit.
- ✓ La confidentialité des données stockées qui pourra être assurée par des mécanismes de contrôle d'accès voire de chiffrement

Dans l'évaluation de la confidentialité, nous nous intéressons uniquement aux mesures de sécurité mises en place et non pas à l'évaluation du niveau de sécurité procuré par ces mesures. Au niveau des éléments du plan service, nous nous intéressons au mécanisme de chiffrement de l'information privée et au niveau des éléments de l'infrastructure, nous nous intéressons aux mécanismes de contrôle d'accès. Ceci nous conduit à définir la classe ontologique de l'annotation de confidentialité comme :

$$AnnotationDeConfidentialité \subseteq$$
$$(\forall\, aUneConfidentialitéAssurée.Message) \wedge$$
$$(\forall\, aUneConfidentialitéAssurée.DocumentMétier) \wedge$$
$$(\forall\, aUneConfidentialitéAssurée.Données) \wedge$$
$$(\forall\, aUneConfidentialitéAssurée.LogicielD'HébergementEtSupport) \wedge$$
$$(\forall\, aUneConfidentialitéAssurée.SystèmeExploitation) \wedge$$
$$(\forall\, aUneConfidentialitéAssurée.Equipement)$$

6.2.3.4 L'élément de l'annotation 'supervision'

La supervision porte sur l'audit des opérations des services pour s'assurer de leur bon fonctionnement et sur l'audit des paramètres de la qualité de protection. En effet, les utilisateurs cherchent à s'assurer que les services garantissent bien ce qu'ils annoncent dans leur description, surtout concernant l'application des paramètres de la QoP et la gestion des préférences de sécurité. Ceci nous conduit à définir la classe ontologique de l'annotation de supervision comme :

$$AnnotationDeSupervision \subseteq$$
$$(\forall\ aUneSupervisionAssurée.Opération) \wedge$$
$$(\forall\ aUneSupervisionAssurée.QoP)$$

6.2.4 Utilisation des politiques pour représenter l'annotation de sécurité

Pour exploiter l'annotation, il faut la représenter sous un format exploitable pour l'insérer dans la description du service (tel que les fichiers WSDL par exemple) Cependant, pour donner plus de flexibilité dans la gestion des paramètres de sécurité suite au changement du contexte, il est plus intéressant de séparer les descriptions des éléments fonctionnels et non fonctionnels.

Dans notre travail - inspiré de la contribution de [126] - nous proposons une plateforme de services web dans laquelle nous annotons les services selon le standard de la W3C qui est le WS-Policy [45]. En définissant une politique WS-Policy, nous séparons les descriptions non fonctionnelles (ex : paramètres de la sécurité, QoS) des descriptions fonctionnelles (WSDL). De plus, WS-Policy peut être étendu pour intégrer les descriptions sémantiques des annotations de sécurité, et donc permettre un matching utilisant les ontologies. Après une présentation rapide du standard WS-Policy, nous explicitons son utilisation dans la sélection dynamique des services.
Le standard WS-Policy est un langage extensible développé pour les services web. La spécification WS-Policy définit une politique (Policy) comme un ensemble d'alternatives. Ces dernières sont définies comme une collection d'assertions qui permettent, elles-mêmes de définir les capacités et les exigences d'un service web. En outre, les assertions peuvent inclure de sous-assertions et un ensemble d'attributs. La Figure 6-17 représente la forme d'une politique WS-Policy dans laquelle nous trouvons trois opérateurs:

- 'Policy' permettant de délimiter une politique.
- 'ExactlyOne' définissant un ensemble d'alternatives.
- 'All' contenant toutes les assertions d'une alternative.

```
<wsp:Policy ... >
  <wsp:ExactlyOne>
      ( <wsp:All> ( <Assertion ...> ... </Assertion> )* </wsp:All> )*
  </wsp:ExactlyOne>
</wsp:Policy>
```

Figure 6-17 : Forme d'une politique WS-Policy

Une fois cette politique définie, nous la référençons depuis le fichier WSDL en utilisant le standard WS-PolicyAttachment [44]. Ce standard permet de référencer les politiques depuis quatre éléments du WSDL qui sont : *service*, *endpoint*, *operation*, et *message*. Nous avons choisi l'élément « service » puisque l'annotation concerne le service dans son ensemble.
La Figure 6-18 illustre l'exemple d'un service de bourse annoté par des paramètres de sécurité définis dans une politique WS-Policy (ligne 3 de la description). Cette politique est présentée dans la Figure 6-19.

```
1  <wsdl:service name="ServiceDeBourse"
2      <wsdl:port name="ServiceDeBourseport" binding="tns:SecureBinding"/>
3      <wsp:PolicyReference URI="http://www.auf.org/services/policy/WSBoursePolicy.xml"
4       wsdl:required="true"/>
5  ...
```

Figure 6-18 : Référence à la politique depuis WSDL

```
1  <wsp:Policy
2  xmlns:wsp="http://www.w3.org/ns/ws-policy"
3  xmlns:as="http://www.auf.org/soa/annotationdesécurité>
4      <wsp:ExactlyOne>
5      <wsp:All>
6          <wsp:name="Annotation De Sécurité Service De Bourse">
7          <as:annotationdesecurite>
8          <as:elementannotation> disponibilité </as:elementannotation>
9          <as:valeur> 76.92 </as:valeur>
10         <as:elementannotation> confidentialité </as:elementannotation>
11         <as:valeur> 84.82 </as:valeur>
12         <as:elementannotation> supervision </as:elementannotation>
13         <as:valeur> 66.66 </as:valeur>
14         </as:annotationdesecurite>
15     </wsp:All>
16     </wsp:ExactlyOne>
17 </wsp:Policy>
```

Figure 6-19 : Exemple WS-Policy service de Bourse

Ces annotations peuvent être utilisées pour sélectionner des services pour composer dynamiquement des processus sécurisés en phase d'exécution.

6.3 Construction du processus métier sécurisé : la phase d'exécution

Outre l'exécution des processus métier défini lors de la conception, les architectures orientées services ont offert la possibilité d'adapter rapidement les applications et les systèmes d'information de l'entreprise aux changements métier et organisationnel. A condition que les entités à connecter puissent partager une même interface de service [127], il est possible d'organiser un processus ad hoc en sélectionnant, composant et invoquant les services dynamiquement à l'exécution. C'est la prise en compte des annotations de sécurité que nous présentons ici pour construire un processus sécurisé en fonction des exigences des utilisateurs.

La découverte et la sélection des services dans une SOA sont des étapes importantes dans la phase d'exécution du cycle de vie des processus métier et des services. La découverte représente

la localisation des services qui répondent aux critères fonctionnels. La sélection des services correspond à l'évaluation des services trouvés afin d'identifier ceux qui sont les plus adaptés aux préférences de l'utilisateur [1]. La méthodologie de conception d'une SOA sécurisée (MCSS) que nous avons développée permet d'évaluer la sécurité globale des services développés et de calculer les valeurs des annotations de sécurité. Pour construire dynamiquement des processus sécurisés, nous proposons d'intégrer ces annotations aux critères de sélection purement fonctionnels utilisés généralement pour ne conserver que les services répondant aux exigences des utilisateurs. Par exemple, dans une collaboration commerciale, un processus métier de gestion de commande met en jeu plusieurs partenaires : le fournisseur, différents sous traitants et agents de livraison. Le service de commande permet d'ajouter des préférences de sécurité (ex : confidentialité des données personnelles des clients) aux critères fonctionnels usuels (produit, prix, information client, etc.) pour ne conserver que les services répondant aux besoins.

Dans la littérature, S. Chaari a proposé l'intégration des aspects non fonctionnels (la QoS) dans la sélection des services [126]. Dans cette même perspective, nous avons choisi de faire une composition de service utilisant les annotations de sécurité ce qui permettra de construire des processus métier sécurisés.

L'architecture de sélection dynamique que nous proposons utilise les composants de base de l'architecture des services web : le fournisseur de service, le client et l'annuaire (la structure de l'annuaire UDDI est décrite dans l'Annexe 11). Nous étendons cette architecture en introduisant un service de sécurité intermédiaire 'Security Broker' pour :

- ✓ Publier les annotations de sécurité auprès de l'annuaire en obtenant les annotations de sécurité à partir des politiques des services. En effet, nous trouvons que les annotations qui se trouvent au niveau des politiques (pour spécifier les paramètres de sécurité et permettre de faire des publications automatiques) devront être publiées au niveau de l'annuaire pour éviter l'envoi des requêtes aux services afin de récupérer ces paramètres lors de la sélection. Notre proposition s'inspire des travaux de [126] [128] sur la sélection des services en fonction des paramètres de la qualité de service.
- ✓ Sélectionner dynamiquement les services web en fonction des aspects fonctionnels et des préférences de sécurité de l'utilisateur.

Etant donné que l'annuaire peut être sujet à des attaques mettant en cause l'intégrité des annotations de sécurité, nous proposons que le 'Security Broker' stocke dans un registre les annotations de sécurité afin de les vérifier avant la sélection des services. Notre proposition s'inspire du travail [129] sur la vérification de l'intégrité des paramètres de la qualité de service.

Le 'Security Broker' se compose des modules suivants (Figure 6-20):

1. Le module de publication qui gère la publication de la description des services (aspects fonctionnels et paramètres de sécurité) auprès de l'annuaire, alimente le registre 'Annotation de sécurité' avec les paramètres de sécurité.
2. Le module de sélection qui réalise la recherche dans l'annuaire de services, effectue la sélection du service qui répond aux critères fonctionnels et aux critères de sécurité après vérification des annotations de sécurité auprès du registre des annotations de sécurité.

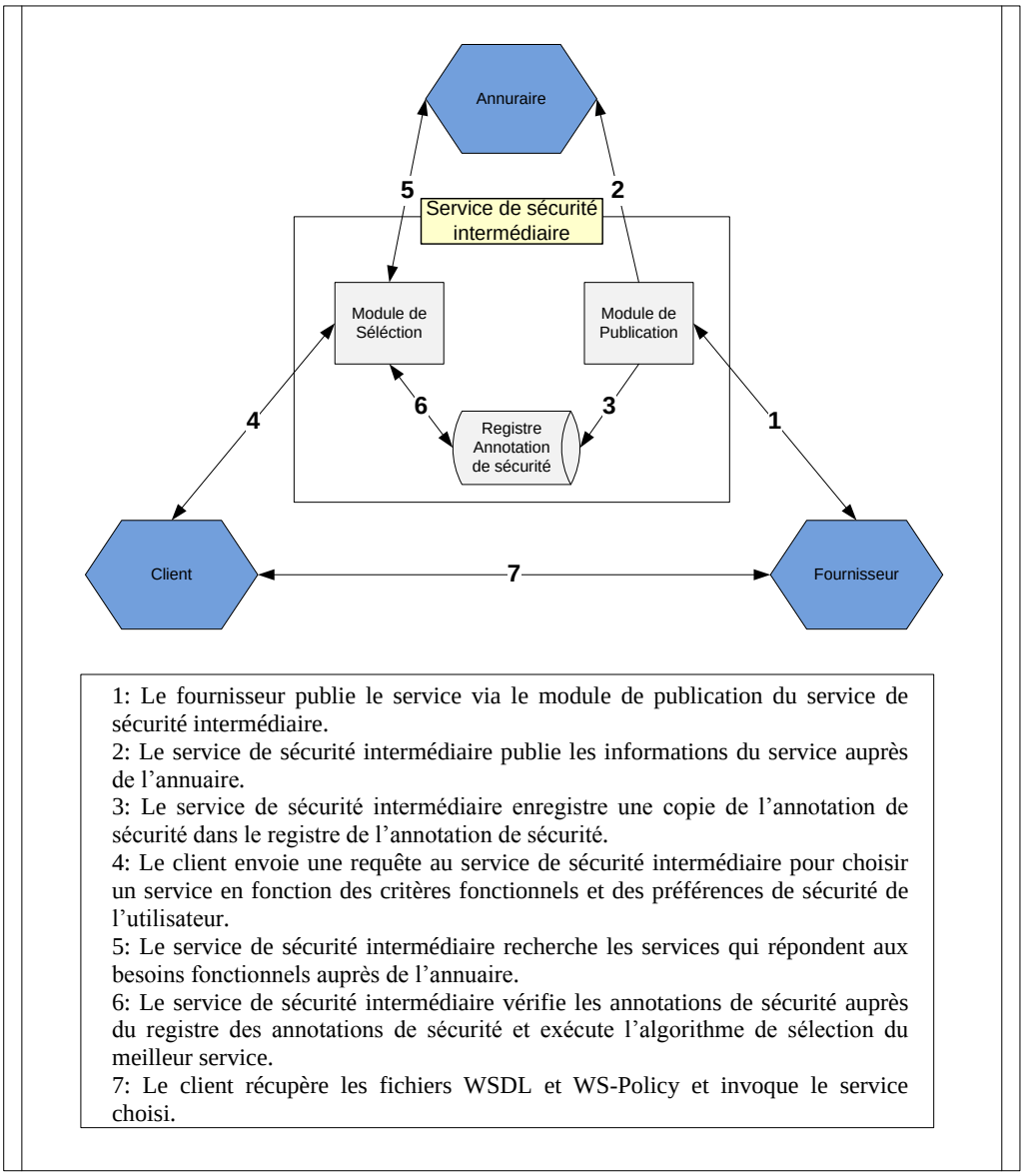


Figure 6-20: Architecture de sélection dynamique de services sécurisés

6.3.1 La publication

A la réception d'une requête de publication de service, le service de sécurité intermédiaire va parcourir le fichier WS-Policy. Il recherche les valeurs des éléments de l'annotation et les publie au niveau de l'annuaire en utilisant l'élément tModel (Figure 6-21). Ces annotations seront

référencées par la structure de donnée ‘bindingTemplate‘ qui liste les informations nécessaires à l’invocation du service. Dans l’élément tModel, chaque élément de l’annotation de sécurité est représenté par l’élément KeyedReference, qui contient le nom de l’élément de l’annotation définit dans l’ontologie ‘OAS’ et sa valeur calculée à partir du rapport VEA = $\sum_{i=1}^{n} \frac{x_i}{N}$ et en se basant sur le modèle de dépendance.

```
<tModel tModelKey = "auf.org:ServiceBourse:AnnotationdeSécurité">
    <name>Annotation de Sécurité Service de Bourse</name>
    <categoryBag>

    <keyedReference
    tModelKey="uddi:auf.org:AS:Confidentialité"
    keyName="Confidentialité "
    keyValue=" 84.82" />

    <keyedReference
    tModelKey="uddi:auf.org:AS:Disponibilité"
    keyName="Disponibilité"
    keyValue="76.92" />

    <keyedReference
    tModelKey="uddi:auf.org:AS:Supervision"
    keyName="Supervision"
    keyValue="66.66" />
    <keyedReference

    </categoryBag>
</tModel>
```

Figure 6-21 : publication de l'annotation - UDDIv3 tModel

6.3.2 La sélection

Pour sélectionner un service en fonction des critères de sécurité, nous utilisons le travail proposé dans [130] dans lequel, Y. Badr propose une sélection des services en utilisant une moyenne pondérée intégrant les propriétés non fonctionnelles des services. Ceci permet de s’adapter aux préférences des utilisateurs en donnant plus d'importance à certaines propriétés, les propriétés restantes seront considérées comme ayant une importance égale.

Pour ce qui concerne la sécurité, les préférences sont définies en termes de confidentialité, disponibilité et supervision : l'utilisateur peut soit leur donner une importance égale, soit définir un poids relatif ou un poids nul. L’utilisateur aura simplement à préciser l'importance de ces éléments. Si l'utilisateur considère tous les éléments de même importance, alors les poids seront répartis de manière égale. Si l'utilisateur considère que certains éléments de l’annotation sont plus importants, nous leur attribuons les poids spécifiés par l’utilisateur et le poids restant est répartis équitablement entre les autres éléments de l’annotation de sécurité. En outre, si l’utilisateur ne désire pas prendre en compte certains critères (étant donné que la sécurité a un coût en qualité de service), il peut le mentionner en spécifiant un poids nul pour un ou plusieurs élément(s) de l’annotation. Le poids restant est alors réparti proportionnellement aux poids

spécifiés selon leur importance. Enfin, si l'utilisateur annule tous les poids, les services seront sélectionnés uniquement selon leurs propriétés fonctionnelles.

L'Equation 2 nous permet de calculer pour chaque service la moyenne pondérée en fonction des critères de l'utilisateur.

$$\sum_{i=1}^{n} w_i \; x_i$$

Equation 2 : Calcul de la moyenne pondérée

où

w_i représente le poids attribué par l'utilisateur à l'élément de l'annotation de sécurité x_i

x_i représente la valeur de l'élément de l'annotation

avec :

$0 \leq w_i \leq 1$

$\sum_{i=1}^{n} w_i = 1$ (D'après les concepts de l'optimisation multicritère et quand les poids sont normalisés)

A titre d'exemple, nous considérons le service de bourse (S1) de l'étape précédente. Les valeurs des éléments de l'annotation de ce service sont les suivantes :

S1$\langle Confidentialité\ 84.82 | Disponibilité\ 76.92 | Supervision\ 66.66 \rangle$

Supposons que :

- Il existe un autre service de bourse S2 qui présente les mêmes aspects fonctionnels et a les annotations de sécurité suivantes :
  S2$\langle Confidentialité\ 87.3 | Disponibilité\ 86.32 | Supervision\ 76.12 \rangle$
- Après avoir choisi les trois éléments à prendre en compte, l'utilisateur donne ses préférences :
  $\langle Confidentialité\ 70\% | Disponibilité\ 20\% | \rangle$
  Par conséquent, la supervision aura une importance de 10%.

La valeur globale de la sécurité (VGS) de chaque service est déterminée en se basant sur la moyenne pondérée.

- VGS (S1) = 0.7*84.82+0.2*76.92+0.1*66.66 = 81.424
- VGS (S2) = 0.7*87.3+0.2*86.32+0.1*76.12 = 85.986

S2 sera donc considéré comme le meilleur service pour cette requête.

Si l'utilisateur donne les préférences $\langle Confidentialité\ 70\% | Disponibilité\ 20\% | Supervision\ 0\% \rangle$
la supervision sera exclue du calcul de la moyenne pondérée et les préférences de l'utilisateur seront : $\langle Confidentialité\ 77.777\% | Disponibilité\ 22.222\% | \rangle$

Dans la partie suivante, nous abordons les concepts liés à la gestion des vulnérabilités et nous proposons un service de gestion des vulnérabilités logicielles au niveau des éléments de l'infrastructure.

6.4 Service de gestion des vulnérabilités : la phase de supervision

Dans le chapitre 4, nous avons défini les vulnérabilités comme étant des faiblesses ou des failles pouvant être exploitées pour nuire aux éléments essentiels d'une SOA. Les vulnérabilités peuvent être de niveaux différents :

- ✓ Les vulnérabilités au niveau métier comme par exemple l'absence d'une politique définissant les droits d'accès aux documents métier confidentiels et aux processus métier privés.
- ✓ Les vulnérabilités au niveau service comme par exemple les vulnérabilités liées à XML menant à des attaques de type XML-injection
- ✓ Les vulnérabilités au niveau du plan infrastructure comme les vulnérabilités logicielles (logiciels d'hébergement et systèmes d'exploitation) et les vulnérabilités matérielles (problème de fiabilité)

Pour couvrir la phase de supervision, il est nécessaire d'apporter une garantie lors de l'exécution en considérant à la fois les aspects métier et technologiques :

- Au niveau du plan métier, il est nécessaire de faire des contrôles d'usage et monitoring des activités.
- Au niveau du plan service, il faut gérer la traçabilité des 'enforcement points'
- Au niveau du plan infrastructure, il faut gérer les vulnérabilités portant sur les composants identifiés lors de la conception.

Dans notre travail, nous nous sommes focalisés sur la gestion des vulnérabilités (tâche dans le périmètre des responsabilités techniques) au niveau des logiciels d'hébergement et des systèmes d'exploitation pour l'automatiser et créer un service de gestion de vulnérabilités. Lors de la définition des concepts des éléments essentiels de l'infrastructure dans l'ontologie OCSS (p. 121), nous avons mis en évidence l'importance du triplet {nom, vendeur, version} pour les logiciels d'hébergement et les systèmes d'exploitation. Ce triplet nous permet d'identifier auprès des bases de vulnérabilités publiques, qui sont mises à jour régulièrement, les vulnérabilités qui leur sont associées.

Parmi les bases de données de vulnérabilités les plus utilisées, nous trouvons :

a. La base NVD : National Vulnerability Database [131] du NIST
b. La base de données OSVDB : Open Source Vulnerability DataBase [132]
c. La base US-CERT : United States Computer Emergency Readiness Team [133]

Ces bases de données utilisent un identifiant unique pour désigner les vulnérabilités. Cet identifiant est nommé CVE (Common Vulnerabilites and Exposures). Il est géré par la société MITRE en partenariat avec nombreuses CNA (CVE Numbering Authority) comme Apple, Oracle, Microsoft, Ubuntu etc.

L'objectif du CVE est d'attribuer un identifiant unique aux vulnérabilités reconnues et de faciliter le partage d'information concernant ces vulnérabilités. Dans le Tableau 6-26, nous donnons deux exemples de CVE listé dans la base NVD :

CVE	Description	Détails
CVE-2011-1571	Unspecified vulnerability in the XSL Content portlet in Liferay Portal Community Edition (CE) 5.x and 6.x before 6.0.6 GA, when Apache Tomcat is used, allows remote attackers to execute arbitrary commands via unknown vectors.	Published: 05/07/2011 CVSS Severity: 9.3 (HIGH)
CVE-2011-2688	SQL injection vulnerability in mysql/mysql-auth.pl in the mod_authnz_external module 3.2.5 and earlier for the Apache HTTP Server allows remote attackers to execute arbitrary SQL commands via the user field.	Published: 07/28/2011 CVSS Severity: 7.5 (HIGH)

Tableau 6-26 : Exemple de vulnérabilités de la base NVD

La mise en place de CPE (Common Platform Enumeration) par MITRE a pour objectif de formaliser le nommage des plateformes et de rendre possible l'automatisation de tâches de gestion des vulnérabilités. Dans la dernière spécification 2.0 [134], la structure d'un CPE est la suivante :

cpe:/ *{part}* : *{vendor}* : *{product}* : *{version}* : *{update}* : *{edition}* : *{language}*

Avec :

Part: la partie de la plateforme spécifiée (h = pour matériel/o = pour système d'exploitation/a pour application).

Vendor : nom du vendeur, nom du produit, version, mises à jour, édition et langage utilisé}.

A titre d'exemple, le cpe:/o:microsoft:windows_xp:::pro représente la plateforme WindowsXP professionnel.

Enfin, ces bases de vulnérabilités associent à chaque base de CVE un "CVSS" (Common Vulnerability Scoring System) annotant la vulnérabilité identifiée par une valeur de sévérité (comprise entre 0 et 10)

Comme le montre la Figure 6-22, le Service de Gestion des Vulnérabilités (SGV) notifie le responsable technique des vulnérabilités correspondant aux logiciels d'hébergement et aux systèmes d'exploitation de l'infrastructure.

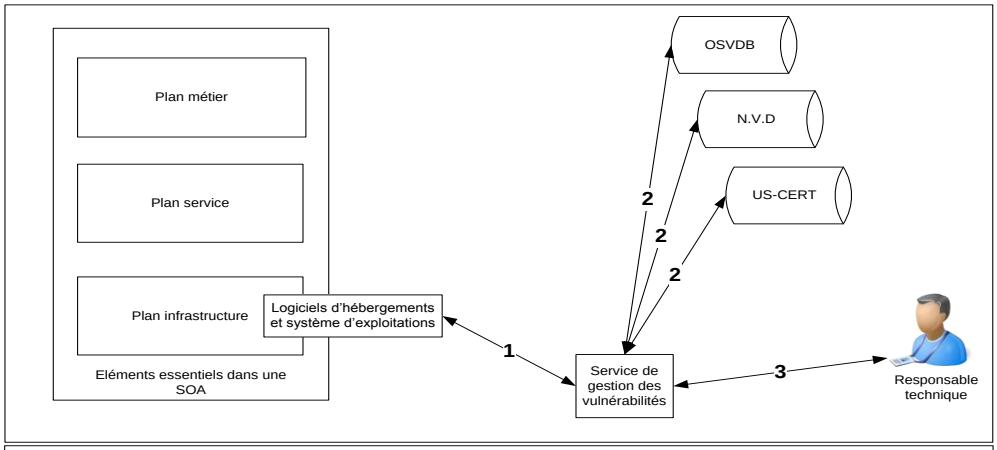


Figure 6-22 : Service de gestion des vulnérabilités

6.5 Conclusion

Dans le cadre de la conception des SOA, plusieurs modèles de référence et méthodologies ont été proposés. Toutefois, l'intégration de la sécurité lors de la conception de services ou d'application utilisant des services est limitée. Afin de combler ce manque, nous avons développé une méthodologie de conception d'une SOA sécurisée (MCSS). Dans cette méthodologie, nous avons intégré un cycle de gestion des risques. Ce travail, nous a permis de mieux:

1. Évaluer les besoins ainsi que le coût de la sécurité par rapport au coût de la « non sécurité ».
2. Identifier les mesures de sécurité adéquates permettant la réduction des risques.
3. Assurer de futures collaborations sécurisées entre les partenaires.

La méthodologie développée est formée de trois phases:

1. La phase d'identification et de spécification des services : permet d'identifier les biens à protéger et leurs dépendances.
2. La phase de la gestion des risques : intègre un cycle de gestion des risques pour chaque bien.
3. La phase de synthèse : permet de récapituler les informations de sécurité afin d'annoter les services pour de futures réutilisations.

La sélection dynamique de services durant la phase d'exécution a été enrichie en proposant une architecture de sélection dynamique de services sécurisés. Au sein de cette architecture, nous avons proposé un service de sécurité qui prend en charge la publication et la sélection des services sécurisés en utilisant l'annotation de sécurité pour enrichir la description des services. Enfin, nous avons présenté le service de gestion des vulnérabilités qui permet de détecter les vulnérabilités logicielles des composants de l'infrastructure.

Dans le chapitre suivant, nous développons un outil d'aide à la conception d'une SOA sécurisée et nous appliquerons la méthodologie MCSS dans la conception d'une plateforme e-learning à base de services.

Chapitre 7. Outil de conception d'une SOA sécurisée et application de la méthodologie MCSS

Résumé

Dans ce chapitre, nous présenterons l'outil de conception d'une SOA sécurisée et nous appliquons la méthodologie MCSS sur un cas d'illustration : une plateforme de e-learning orientée services.

Sommaire

7.1 Introduction

Dans les chapitres précédents, nous avons abordé la sécurité dans :

- ✓ La *phase de préparation* : nous avons développé un méta modèle conceptuel de service sécurisé permettant de définir des patrons de sécurité et une ontologie de conception d'une SOA sécurisée permettant la définition des concepts des biens essentiels à protéger.
- ✓ La *phase de conception* : nous avons proposé une méthodologie de conception en intégrant la sécurité à la conception de services réutilisables.
- ✓ La phase d'exécution : nous avons présenté une architecture de sélection dynamique des services sécurisés.
- ✓ La phase de supervision : nous avons proposé un service de gestion des vulnérabilités liées aux composants du plan infrastructure (logiciels d'hébergement, systèmes d'exploitation).

Dans ce chapitre, nous présentons l'outil de conception d'une SOA sécurisé que nous avons développé pour faciliter la gestion des éléments essentiels et des liens de dépendances entre eux. En outre, cet outil permet de calculer les annotations de sécurité reflétant le niveau de sécurité globale des services en recensant les éléments concernant la sécurité. Enfin, nous montrerons comment appliquer la méthodologie de conception d'une SOA sécurisée MCSS sur un cas d'illustration : une plateforme de e-learning orientée services.

7.2 Outil de conception d'une SOA sécurisée

Pour faciliter la mise en œuvre de la méthodologie MCSS, nous avons développé un outil support pour :

- ✓ faciliter la gestion des dépendances entres les éléments essentiels formant le contexte
- ✓ recenser la sécurité globale des services en générant les annotations de sécurité à la base de l'information saisie par le concepteur suite au cycle de gestion des risques (mise à jour des profils de sécurité des éléments essentiels)

A ce titre, cet outil supporte les activités :

- ✓ d'identification et de spécification des services (phase 1 de la méthodologie MCSS)
- ✓ d'établissement du contexte (phase 2) en guidant l'identification des éléments d'infrastructure
- ✓ d'annotation de sécurité (phase 3)

Les étapes du cycle de gestion des risques (étapes 7A, 7B, 8 et 9) se déroulant principalement lors de séances de brainstorming n'ont pas fait l'objet de développement de composants supports.

7.2.1 Architecture du prototype

L'architecture de notre prototype se compose de quatre modules (Figure 7-1)

1. Le module de gestion des éléments essentiels

Ce module se préoccupe de la gestion des éléments essentiels formant le contexte. Il permet d'alimenter la base de connaissances en y intégrant les éléments essentiels identifiés dans la phase 1 de la méthodologie MCSS pour les éléments essentiels des plans métier et service ainsi que les éléments essentiels du plan de l'infrastructure identités dans l'étape 'établissement du contexte' de la phase 2.

2. Le module d'évaluation de la sécurité

Ce module permet de faire une évaluation de la sécurité en permettant aux responsables métier et techniques de récapituler pour chaque élément essentiel les exigences de sécurité puis inventorier la mise en place de mesures de sécurité identifiées lors du cycle de gestion des risques.

3. Le module de calcul des annotations

Ce module permet de calculer les valeurs globales des éléments de l'annotation de sécurité en se basant sur l'inventaire des mesures de sécurité mises en place, les exigences de sécurité assurées et le lien de dépendance établis entre les éléments essentiels correspondant aux services.

4. Le module de gestion des dépendances

Ce module permet de gérer les liens de dépendance entre les éléments essentiels des trois plans (métier, service et infrastructure) lors de l'alimentation de la base de connaissances par de nouveaux éléments essentiels. En outre, il permet de générer les liens de dépendance lors de la conception de services à partir des éléments essentiels existants dans la base de connaissances.

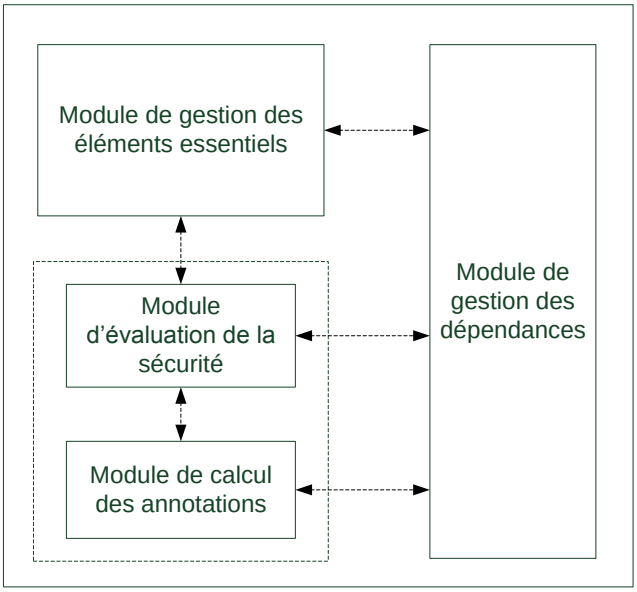


Figure 7-1 : Architecture du prototype

7.2.2 Technologies utilisées et prise d'écran du prototype

Cet outil a été développé avec le langage Java et l'environnement de développement Eclipse Helios (Eclipse IDE for SOA Developers). L'interface principale du prototype présentée dans la Figure 7-2 correspond à la gestion des éléments essentiels. Elle est composée de deux parties : la première (à gauche) permet aux analystes d'alimenter la base de connaissance par les instances des éléments essentiels des plans métier, service et infrastructure, la deuxième partie (à droite) permet d'afficher les instances des éléments essentiels sélectionnés.

La « création d'un processus métier sécurisé » déclenche le processus de conception d'un processus métier, intra ou inter entreprise implémentant le modèle de dépendance (Figure 7-2). Le concepteur est invité à alimenter la base par les instances des éléments essentiels (des trois plans) relatifs à ce processus métier soit en utilisant des instances existantes dans la base soit en créant de nouvelles. A l'issue de ce processus de création, un lien de dépendance entre les instances des éléments essentiels des plans métier, service et infrastructure est créé.

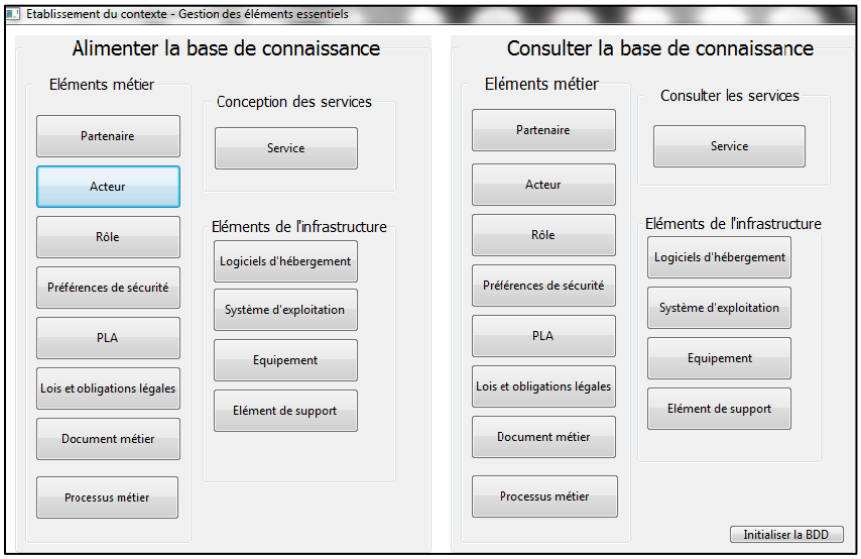


Figure 7-2 : Outil de conception d'une SOA sécurisée: Etablissement du contexte

On note que la création des processus métier commence par le choix du type de processus (interne /inter entreprises) pour réduire les éléments du plan métier à saisir. Par exemple, nous ne spécifions pas les partenaires dans le cas des processus internes à l'entreprise. Ensuite on procède à l'ajout ou la création des instances des éléments du plan métier (Figure 7-3).

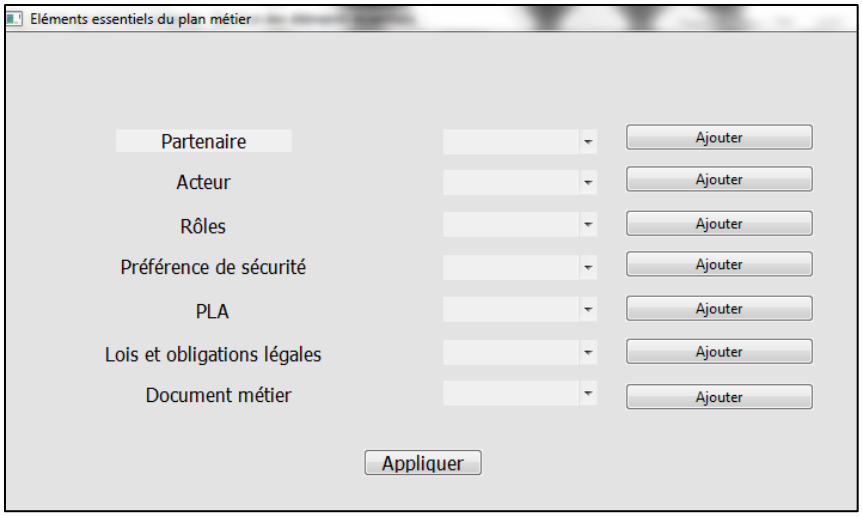


Figure 7-3 : Alimentation des éléments du plan métier

En suivant le modèle de dépendance, l'outil permet d'ajouter les services appartenant au processus métier ou de créer de nouveaux services. L'ajout d'un nouveau service (atomique ou composite) est suivi par l'alimentation des éléments essentiels du plan service et du plan infrastructure qui lui correspondent (Figure 7-4, Figure 7-5).

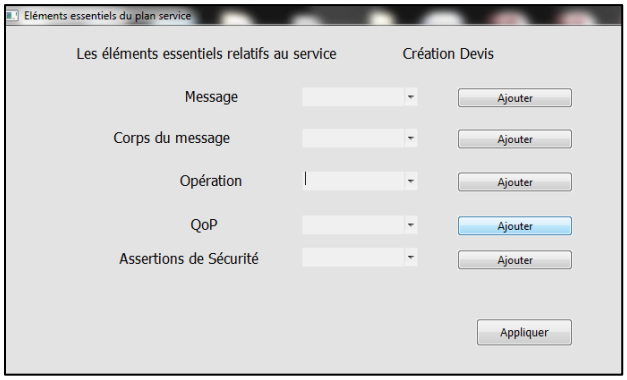


Figure 7-4 : Alimentation des éléments du plan service

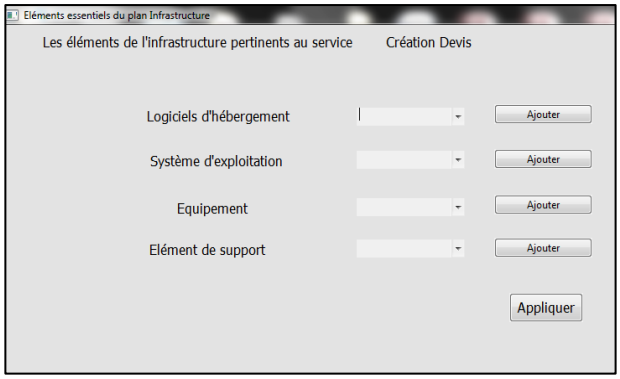


Figure 7-5 : Alimentation des éléments du plan infrastructure

L'ajout ou la création des instances est possible à tout moment de l'alimentation de la base de connaissance. Pour un service composite, le concepteur devra lui associer les services (atomiques ou composites) qui le composent (Figure 7-6). L'objectif est d'aboutir à la définition de l'ensemble des services atomiques.

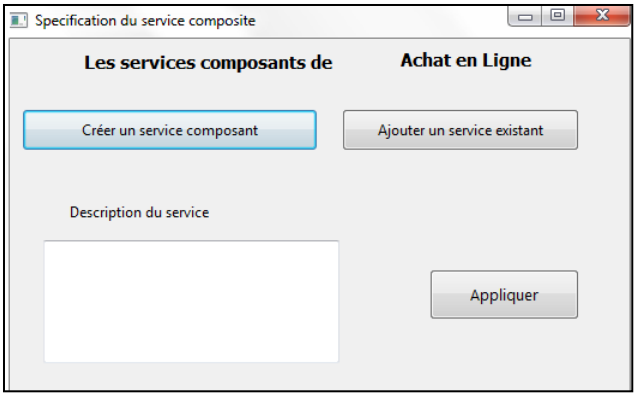


Figure 7-6 : Spécification des services composant un service composite

Evaluation de la sécurité des éléments essentiels (support à la phase 3 – étape 10)

Après avoir réalisé le cycle de gestion des risques (étapes 7A, 7B, 8 et 9 de la méthodologie MCSS), le concepteur doit alimenter le profil de sécurité de chaque instance des éléments essentiels. Le concepteur spécifie le type des mesures de sécurité mises en place ainsi que les exigences de sécurité satisfaites liés aux risques et aux objectifs de sécurité fixés. Nous précisons qu'à ce niveau, il est nécessaire de faire une évaluation des mesures de sécurité mises en place (ce qui est équivalent à un travail d'audit de la sécurité des éléments essentiels)

La Figure 7-7 illustre un exemple de la mise à jour du profil de sécurité de l'instance « Linux/Debian » de type système d'exploitation. L'analyste évalue les mesures de sécurité mises en place, il associe par la suite la disponibilité et l'autorisation comme exigences de sécurité assurées au niveau du système d'exploitation.

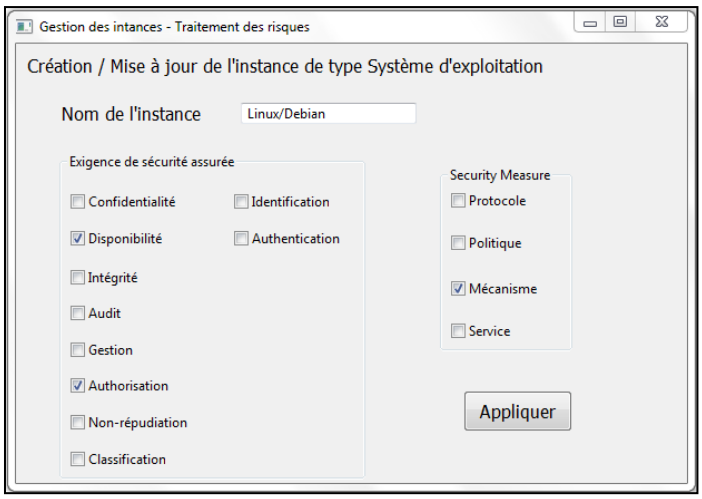


Figure 7-7 : Evaluation de la sécurité

Enfin, la Figure 7-8 illustre l'affichage des annotations de sécurité après la mise à jour des profils de sécurité des éléments essentiels. Il suffit d'appuyer sur « consulter les services » pour afficher les éléments de l'annotation de sécurité.

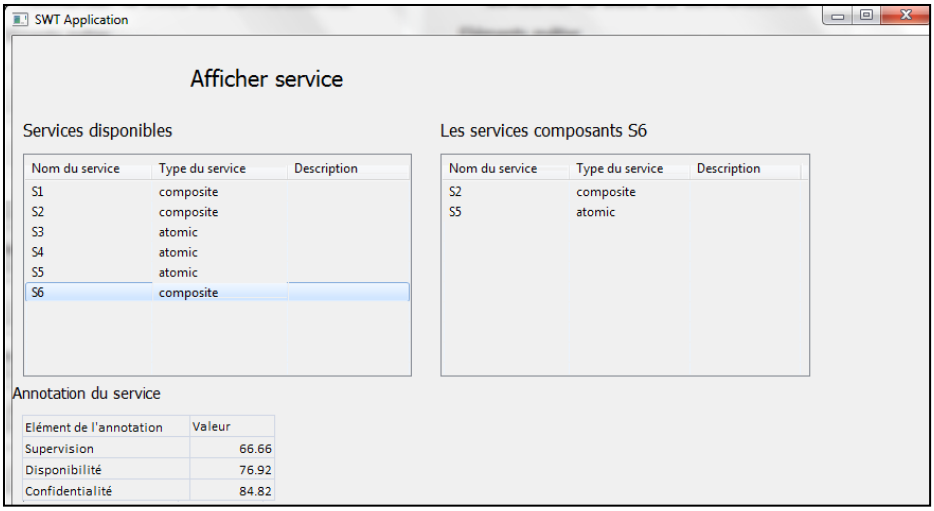


Figure 7-8 : Calcul des éléments de l'annotation

Dans ce qui suit, nous détaillons les algorithmes développés pour gérer les dépendances et calculer les valeurs des éléments de l'annotation.

7.2.3 La gestion des dépendances et le calcul des valeurs des éléments de l'annotation.

Comme nous l'avons dit, le processus de conception de processus métier sécurisé implémente le modèle de dépendance. Un processus métier dépend des services (composites/atomiques) qui le composent. Les services dépendent quant à eux des composants de l'infrastructure d'hébergement. Conformément à la stratégie 'top down' de modélisation, le lien de dépendance est défini en suivant une approche descendante allant du plan métier au plan infrastructure. Outre ces liens de dépendances entre niveaux, il faut aussi gérer les dépendances dans le cas de services composites. Pour cela, il faut fusionner les instances qui se répètent dans les liens de dépendance des services composants pour générer le lien de dépendance du service composite. Prenons comme exemple la composition du service de payement S1 à partir de deux services, un service de vérification des comptes S2 et un service de transfert d'argent S3 (Figure 7-9). Supposons que les services composants sont hébergés sur un serveur d'application 'AT : Apache Tomcat' et installés sur un système d'exploitation 'LU : Linux/Ubuntu' (même équipement). La création du lien de dépendance du service composite S1 doit se faire en fusionnant les instances qui se répètent pour éviter de surestimer les risques (Par exemple, si les instances du serveur d'application 'AT' n'ont pas été fusionnées, les menaces et vulnérabilités portant sur cet élément logiciels seront prises en compte deux fois lors de l'estimation du risque portant sur le service composite S1).

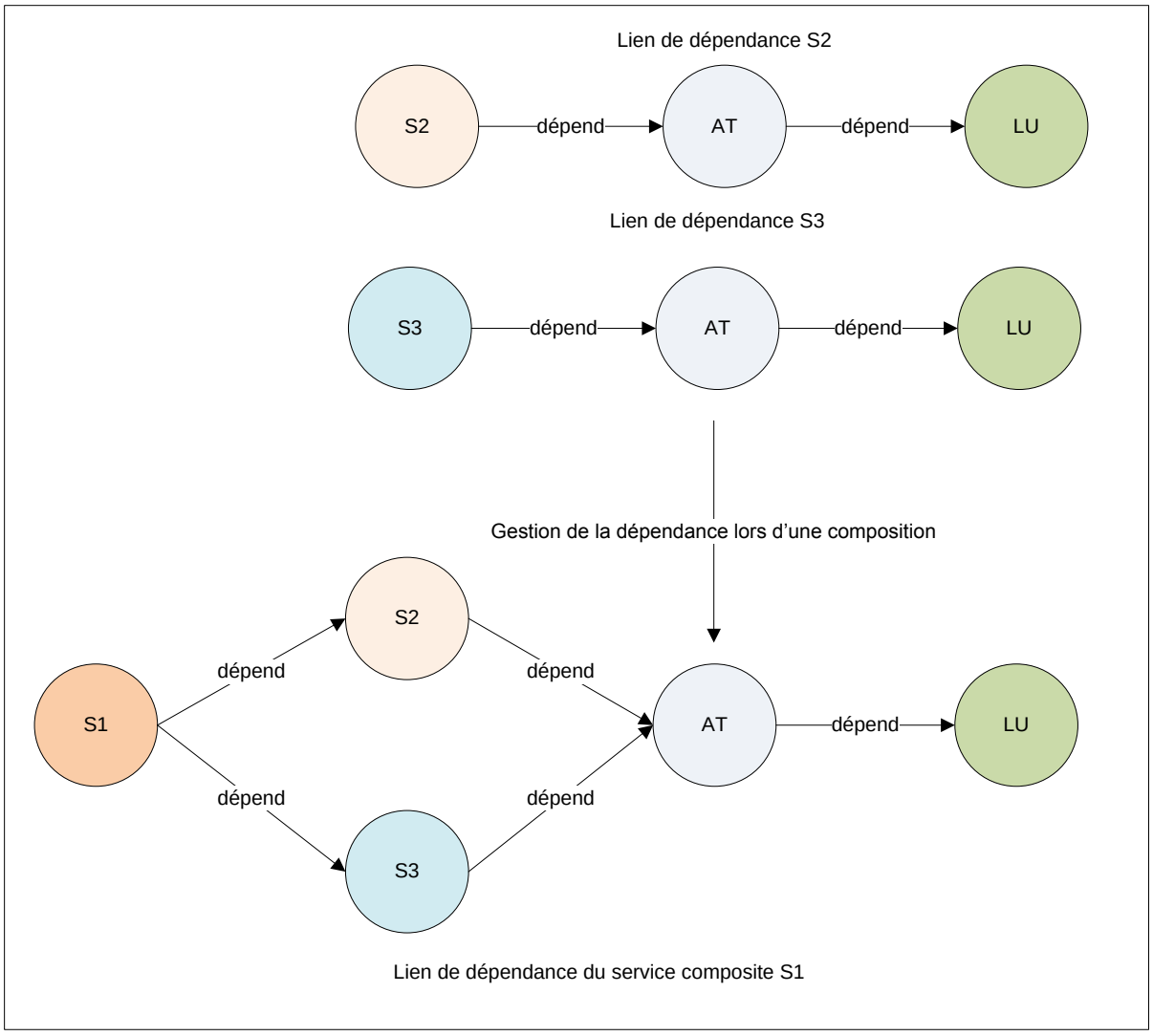


Figure 7-9 : Gestion de la dépendance lors d'une composition

Pour synthétiser la totalité des instances relatives à un service composite et éliminer les redondances, nous utilisons une représentation en arbre où le service composite est le sommet de l'arbre et les services composants sont les nœuds de l'arbre. Cette organisation est répétée récursivement puisque dans une SOA, les services composants peuvent eux-mêmes être des services composites comme le montre la Figure 7-10.

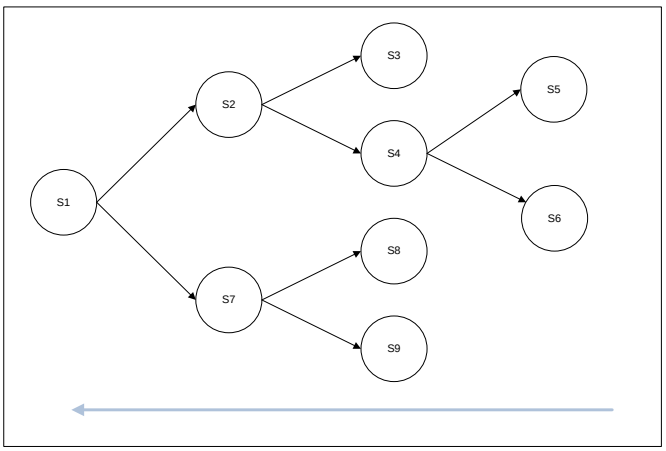


Figure 7-10 : Exemple d'un service composite

Pour trouver la totalité des instances des éléments essentiels relatifs au service S1, nous parcourons l'arbre des feuilles à la racine en commençant par les services composites dont les services composants sont atomiques. Dans notre exemple de la Figure 7-10, c'est le cas des services S4 et S7. Le traitement d'élimination des redondances est alors réalisé sur l'ensemble de ces services sans retenir d'ordre particulier (S4 peut être traité avant S7 ou inversement). Si nous appliquons le traitement sur le service S4, nous associons à ce service les instances de S5 et S6 en éliminant les redondances. Après cette tâche, S4 sera marqué comme un service atomique (annotation virtuelle dans la base de données) et nous traitons ensuite les autres services de la liste qui est modifiée au fur et à mesure (par exemple après avoir traité S4, ce dernier est marqué comme virtuellement atomique. Cela implique d'ajouter le service S2 à la liste puisque ce dernier est composé du service atomique S3 et du service S4 qui vient d'être marqué comme atomique). Nous procédons de la même manière pour atteindre le service S1 auquel sera attribué la totalité des instances non dupliquées des services qui le composent.

Les étapes du pseudo-algorithme sont les suivantes :

1- Rechercher les services composites dont tous les services composants sont atomiques (ligne5). (La structure de la base de données est décrite en Annexe 12)
2- Pour chaque service composite, regrouper les instances pertinentes non dupliquées, et attribuer des copies logiques de ces instances au service composite (ligne6).
3- Annoter le service composite comme étant un service atomique (ligne7).
4- Une fois l'arbre traité complètement, initialiser l'annotation (ligne10).

```
1  Function FusionDesInstances()
2      {
3          While (ExisteServiceComposite)
4              {
5              SelectionServiceComposite();
6              RegrouperInstancesNonDupliquées(R1);
7              AnnoterService(R1);
8              }
9          end while;
10         InitialiserAnnotation();
11     }
```

Figure 7-11 : Pseudo-algorithme de fusion des instances

Dans ce qui suit, nous détaillerons les fonctions de l'algorithme (Figure 7-12):

```
Function ExisteServiceComposite()
    {
        if compositeservice == 1
            return true;
        else
            return false;
    }
Function SelectionServiceComposite()
{
    ResultSet   R1 <- SQLquery(Select SCA);
        // SCA sont des services composites dont les services
        // fils sont atomiques. Cette requête sélectionne les
        // services composites, leurs services composants

    Return R1;
}
Function RegrouperInstancesNonDupliquées(ResultSet R)
{
    for each tuple in R
        ResultSet R2 <- SQLquery(Select IND);
        // IND sont les instances non dupliquées de SCA
            for each tuple in R2
                if(R2.Intance appartient R.ServiceComposant)
                    Attribuer copie logique à R.ServiceComposite;
                    // la copie logique représente l'attribution
                    // des IND au service composite
                end if;
            end for;
    end for;
}
Funtion AnnoterService(ResultSet R);
{
    for each tuple in R
        Set R.ServiceComposite to atomic;
    end for
}
Funtion InitialiserAnnotationService();
{
    // Initialiser les annotations
    // Initialiser les copies logiques
}
```

Figure 7-12 : Algorithme fusion des instances

Pour automatiser l'étape « Annotation de sécurité », nous avons développé un module de calcul des valeurs associées aux éléments de l'annotation: la disponibilité, la confidentialité et la supervision. Ce module pourra être étendu à tout moment pour permettre d'ajouter de nouveaux éléments à l'annotation de sécurité. Le calcul des valeurs des éléments de l'annotation est effectué à la base du rapport VEA = $\sum_{i=1}^{n} \frac{x_i}{N}$ (cf. Annotation de sécurité p. 164):

Une fois la fusion des instances effectuée, nous calculons les éléments de l'annotation en suivant l'algorithme de la Figure 7-13 :

```
Function CaculDesElementsAnnotation
{
ResultSet R1 <- SQLQuery (Selection des éléments de l'annotation);
ResultSet R2 <- SQLQuery (Selection des services composants un service);
  for each tuple in R1
  {
    for each tuple in R2
    {
        I1 <- (Cacluler la somme de IEEPS);
        // Instances des Eléments Essentiels Pertinents Sécurisés
        I2 <- (Cacluler la somme de IEEP);
        // Instances des Eléments Essentiels Pertinents
        PourcentageD'annotation = (I2/I1)*100;
    }
  }
}
```

Figure 7-13 : Calcul des éléments de l'annotation

Après avoir abordé l'outil de conception d'une SOA sécurisée, nous abordons dans ce qui suit l'application de la méthodologie MCSS sur un cas d'usage : une plateforme de e-learning orientée services.

7.3 Cas d'usage : FOAD-OS : Une plateforme e-Learning Orientée Services

De nos jours, les formations ouvertes et à distance ont été adaptées pour répondre aux besoins des apprenants, des universités et de différents instituts de formation. Le succès d'un tel service de formation à distance est évalué selon plusieurs critères comme un accès facile à l'information, la réduction des coûts de déplacement et la collaboration continue entre les participants et les partenaires de formation.

Les architectures orientées services présentent une solution intéressante qui répond au besoin d'agilité et de la mutualisation de différents services de e-Learning (services d'enseignement, processus d'apprentissage, environnements virtuels d'apprentissage et simulations en temps réel)

Nous présentons la conception d'une nouvelle plateforme FOAD [3] (Formation Ouverte et A Distance) pour l'Agence Universitaire de la Francophonie (AUF). Cette nouvelle plateforme FOAD-OS (FOAD – Orientée Services) a comme objectif de créer des formations personnalisées à partir de services d'apprentissage distribués offerts par les partenaires universitaires de l'Agence. Nous détaillons l'utilisation de notre méthodologie pour la conception de la plateforme FOAD-OS sécurisée en insistant sur les points principaux mais sans donner de manière exhaustive tous les détails.

[3] La plateforme FOAD est accessible sur http://foad.refer.org/

7.3.1 L'AUF, les plateformes FOAD et FOAD-OS

L'Agence Universitaire de la Francophonie [4] est présente sur tous les continents, avec plus de 427 agents répartis dans 710 implantations rattachées à neuf bureaux régionaux. Depuis 1989, cette association d'universités est un opérateur de la Francophonie Institutionnelle. Partenaire des établissements d'enseignement supérieur et de recherche qui ont choisi le français comme langue d'enseignement, l'AUF propose plusieurs programmes de coopération visant notamment à soutenir la recherche et l'enseignement en français [135]. Parmi ses actions, l'AUF propose 87 formations ouvertes et à distances (FOAD) et offre des allocations d'études afin de suivre ces licences et ces masters proposés par des universités partenaires. Le cas présenté ici est une plateforme de nouvelle génération FOAD-OS qui permet la personnalisation des formations en combinant les enseignements exposés par les universités partenaires selon les préférences des étudiants en matière de cours, outils et fonctionnalités.

Pour la réussite du projet FOAD-OS, l'AUF doit créer un nouveau département comprenant une équipe métier (un directeur et un coordinateur des programmes) et une équipe technique (un architecte logiciel, un responsable technique, un développeur et un webmestre) en charge de la mise en œuvre et de la maintenance de la plateforme.

Dans ce qui suit, nous supposons que les universités partenaires ont déjà exposé les fonctionnalités de leurs plateformes de e-learning comme des services. Nous appliquons la méthodologie de conception d'une SOA sécurisée MCSS afin de concevoir le projet FOAD-OS en se focalisant sur l'identification et le traitement des risques métier et technologique.

7.3.2 Application de MCSS

Etape 1: Identification du domaine métier

La première étape de la méthodologie consiste à identifier les éléments du domaine métier, 'approvisionnement de formation ouverte à distance'. Nous commençons par lister les activités primaires qui reflètent le cœur de métier. L'objectif métier de FOAD-OS est de faciliter l'accès aux formations à distance et d'offrir des formations personnalisées selon les préférences des étudiants. Nous identifions :

[4] Pour plus d'information : http://www.auf.org

1) Des activités primaires :
   - ✓ La gestion des étudiants permet l'inscription des étudiants, la mise à jour de leur profil, la gestion de leurs droits d'accès, etc.
   - ✓ La gestion des formations permet de mettre à disposition et de gérer les formations offertes par les partenaires (base des offres de formation)
   - ✓ La gestion des partenaires permet aux modérateurs de la plateforme de gérer les contrats de partenariats.

2) Des activités secondaires sont celles qui devront être mises en place pour maintenir le bon fonctionnement du projet, sans apporter directement de valeur ajoutée aux services comme par exemple :
   - ✓ La gestion des ressources humaines.
   - ✓ La gestion du système d'information.

La Figure 7-14 montre l'architecture conceptuelle métier qui permet de lister au centre les activités (le Quoi) et les acteurs du système (le Qui).

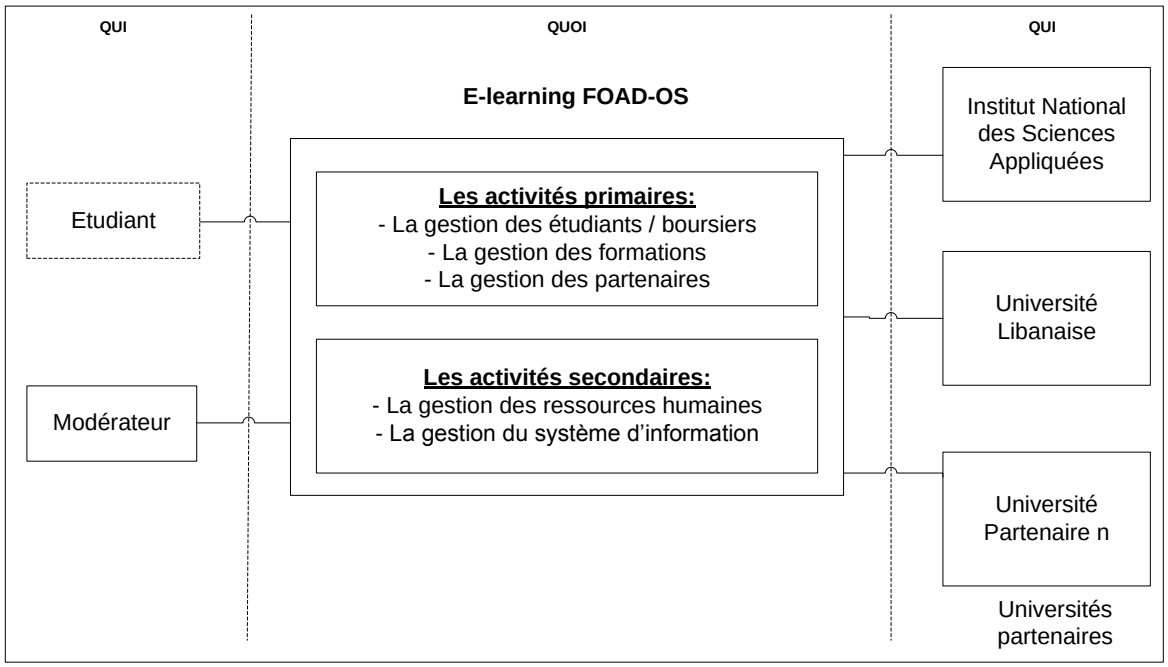


Figure 7-14 : Architecture Conceptuelle (1)

Nous enrichissons cette architecture conceptuelle en identifiant les interactions entre les différents acteurs pour identifier le « comment » qui reflète les documents métier à échanger. (Figure 7-15)

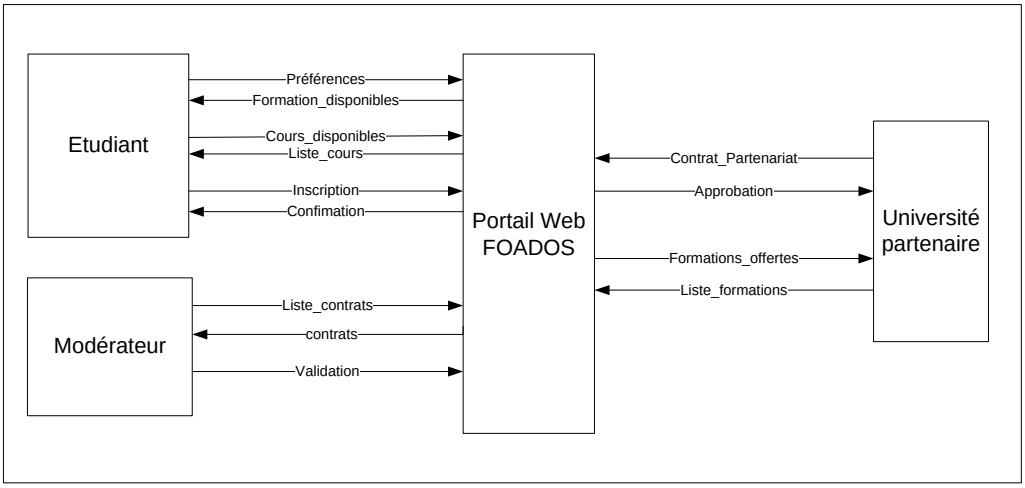


Figure 7-15 : Architecture Conceptuelle (2)

En se référant au modèle de motivation métier de l'OMG, nous avons procédé à l'identification, dans un atelier de brainstorming, des stratégies et des motivations métier de l'AUF dans la mise en place du projet FOAD-OS (le Pourquoi) :

1- Être l'agence francophone de référence assurant des formations ouvertes et à distance.
2- Offrir des formations multidisciplinaires et personnalisées.
3- Enrichir le réseau de partenariat entre les universités partenaires.

Les processus et documents métier sont ensuite modélisés à partir de ces éléments dans les étapes 2A et 2B. Pour illustrer l'utilisation de notre méthode, nous nous réduisons, dans ce qui suit, à la présentation du processus métier 'Inscription'

Etape 2A: Modélisation des processus métier

Après avoir abordé le domaine métier et les interactions entre les différents acteurs, nous examinons et modélisons les processus métier correspondant. Nous procédons de la manière suivante : à partir des listes des acteurs et des activités issues de l'étape précédente, les diagrammes des cas d'utilisation sont modélisés en UML avant de décrire (en BPMN) précisément les processus métier correspondant aux cas d'utilisation retenus. Comme nous l'avons précisé dans le chapitre 6, ces activités ont été réalisées au cours de plusieurs réunions impliquant les responsables métier et techniques.

Dans ce qui suit, nous présentons quelques exemples résultant de cette étape :

- un extrait des cas d'utilisation que peut réaliser un étudiant (Figure 7-16).
- le processus métier «Inscription» en utilisant la notation BPMN (Figure 7-17).

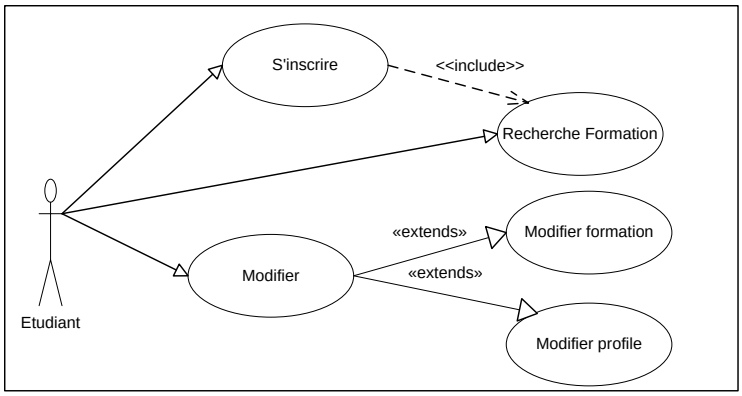


Figure 7-16 : Etudiant: Cas d'utilisation

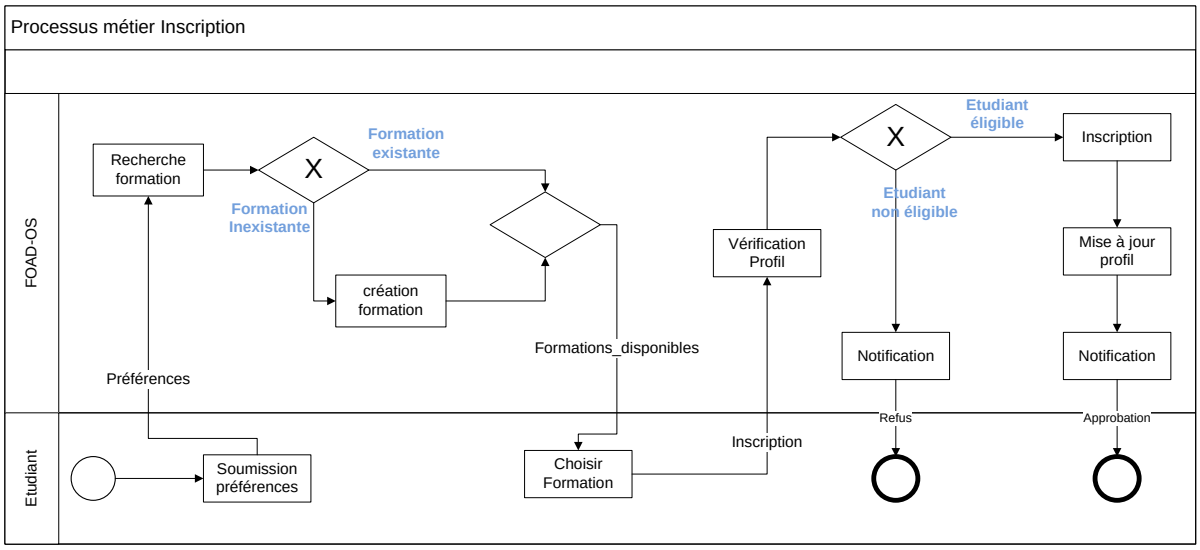


Figure 7-17 : Processus métier inscription

Comme le montre la Figure 7-17, la plateforme FOAD-OS permet de créer des formations personnalisées à partir de services distribués fournis par les universités partenaires. La plateforme offre des formations existantes et permet également de créer des formations selon les préférences de l'étudiant.

Nous décomposons les activités composites du processus métier afin d'avoir des activités atomiques. A titre d'exemple, l'activité « création formation » est une activité décomposable et sera modélisée comme un sous-processus formé des activités: 'calcul nombre crédit', 'calcul frais scolarité', 'vérification nombre crédit / frais de scolarité' et 'copier la liste des formations'.

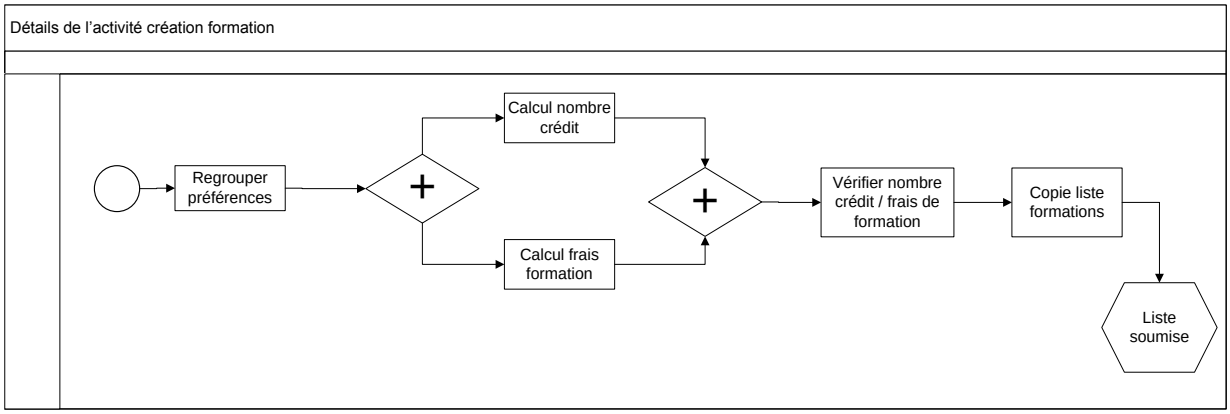


Figure 7-18 : sous-processus : création formation

Etape 2B: Modélisation des documents métier

Pour modéliser les documents métier, nous recherchons dans un premier temps, l'existence de standards d'interopérabilité définissant la sémantique du domaine métier. Nous passons ensuite à la création du modèle structuré de données métier en modélisant l'information du domaine sous forme de diagrammes de classes. Enfin, nous modélisons les documents métier en regroupant les classes qui jouent un rôle dans une interaction et qui caractérisent l'information à échanger.

Pour illustrer cette démarche, nous avons retenu la spécification du document métier inscription. Celui-ci comporte à la fois des informations administratives sur l'étudiant et des informations concernant le parcours de formation. Dans notre exemple, nous avons utilisé le vocabulaire interne de l'Agence pour créer le modèle structuré de données métier étant donné qu'il n'existe pas un standard dans le domaine de la formation à distance.

Dans la Figure 7-19, illustrant le modèle structuré de données simplifié, nous avons choisis la classe 'Inscription' comme étant la classe Pivot du document métier 'Inscription' à laquelle nous avons associé les classes étudiant, compte crédit, sélection et cours qui font partie du document métier 'Inscription'. Ce document est échangé notamment entre les activités 'choisir formation' et 'vérification profil' dans le processus métier inscription.

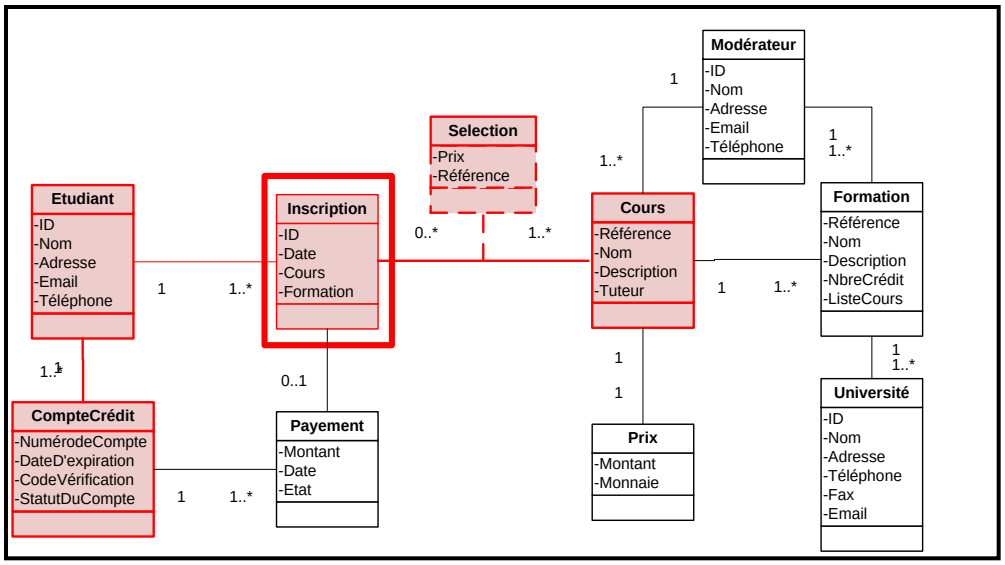


Figure 7-19 : Modèle de l'information sémantique

Etape 3: Identification des objectifs de sécurité

Dans cette étape, nous identifions les objectifs de sécurité stratégiques qui correspondent à la sécurisation des éléments du domaine métier. Ces objectifs sont identifiés au cours d'une séance de brainstorming en présence des responsables métier et des responsables techniques. La présence des responsables techniques est nécessaire puisqu'ils pourront assister les responsables métier à identifier et prioriser ces besoins en se référant aux catalogues génériques des méthodes de gestion des risques EBIOS et OCTAVE.

En se focalisant sur les éléments du domaine métier, nous avons retenu trois objectifs de sécurité stratégique pour l'Agence :

1. Objectif de sécurité 1 : Assurer la continuité du service de formation à distance. Ceci assure la protection de l'image de l'Agence en plus de la satisfaction des étudiants.
2. Objectif de sécurité 2 : Assurer la protection des données personnelles des étudiants (information personnelle : numéro de carte d'assurance sociale, adresse, etc.)
3. Objectif de sécurité 3 : Créer un réseau de confiance entre les différents partenaires.

Etape 4: Identification des services

Après avoir identifié les objectifs de sécurité, nous identifions les services qui répondent aux caractéristiques suivantes : responsabilités, réutilisabilité et couplage lâche. Nous procédons selon une approche 'Outside In' combinant les approches descendante et ascendante afin d'assurer un alignement métier et applicatif et d'optimiser la réutilisation des services. Pour cela, nous identifions :

- ✓ à partir des processus métier modélisés dans l'étape 2A et des documents métier identifiés dans l'étape 2B :
    1- les activités automatiques et semi-automatiques qui seront liées aux opérations des services métier.
    2- les services composites pour mettre en œuvre les processus/sous-processus métier.
    3- les services atomiques :
        a. les services centrés sur les tâches pour mettre en œuvre la logique métier des activités (automatic/semi-automatic) des processus.
        b. les services de données pour manipuler les documents métier.
        c. les services utilitaires pour mettre en œuvre les activités répétitives n'encapsulant pas de logique métier.
- ✓ à partir des solutions logicielles existantes celles qui répondent au besoin (Approche ascendante)

Dans ce qui suit, nous appliquons cette démarche au sous-processus « création formation » (Figure 7-20) :

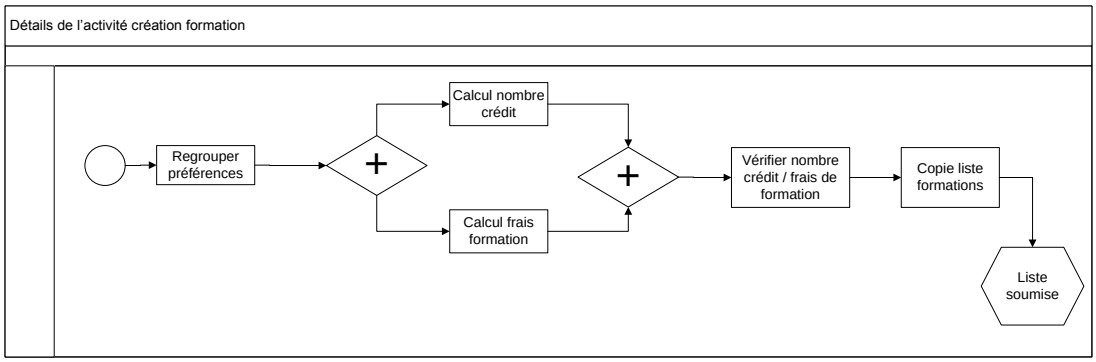


Figure 7-20 : sous-processus : création formation

1- Les activités du sous-processus sont des activités automatiques et seront liées aux opérations des services métier.
2- Le service composite 'création formation' est identifié pour mettre en œuvre le processus 'création formation'.
3- Quatre services atomiques sont identifiés :
    a. Les opérations 'calcul nombre crédit' et 'vérifier nombre crédit' sont regroupés pour former le service 'gestion crédit' centré sur les tâches. Ces activités sont regroupées pour réduire les dépendances entre les services (couplage lâche) et puisqu'elles font partie de la responsabilité du même service. En outre, ces opérations ne peuvent pas être utilisées séparément dans d'autres processus métier, ce qui garantit que le regroupement ne nuit pas à la réutilisation des services. De même, les opérations 'calcul frais scolarité' et 'vérifier frais scolarité' sont regroupés pour former le service 'gestion frais scolarité' centré sur les tâches.

b. Le service de données formation est identifié pour manipuler le document métier formation. Toutefois, la base des formations existe au sein de l'Agence, donc elle sera utilisée pour être interfacée par le service 'formation' (Approche ascendante).
c. Étant donné que l'opération 'copie liste formations' n'encapsule pas de logique métier. Ce service copie est identifié comme un service utilitaire.

Le service composite 'création formation' est donc composé des services atomiques : 'gestion crédit', 'formation', 'gestion frais scolarité' et copie.
Nous suivons la même démarche pour identifier la totalité des services issus de la modélisation des processus métier.

Etape 5: Spécification des services

A partir des résultats de l'étape précédente, nous procédons à la spécification des services selon le méta-modèle SOA de CBDI. Il s'agit de spécifier les propriétés métier tels que l'objectif métier pour lequel le service a été conçu, le domaine métier auquel le service appartient, etc. Ces informations sont déduis à partir des éléments du domaine métier. Les caractéristiques de l'interface sont définies sans entrer dans les détails des technologies et des patterns d'implémentation (qui ne font pas parti du périmètre de notre étude). En se référant aux modèles des processus métier, nous identifions les dépendances, la liste des opérations, les documents en entrée et en sortie. Le cas du service 'création formation' est présenté dans le Tableau 7-1

	Description
Propriétés métier	Nom du service : création formation
	Objectifs métier : Ce service permet de créer des formations personnalisées correspondant aux préférences des étudiants. Il accepte en entrée les préférences de l'étudiant et retourne la liste des formations créé à partir des cours proposés par les universités partenaires.
	Domaine métier : Formation à distance
	Fournisseur : AUF
	Consommateurs ciblés : Etudiants
	Processus métier supportés par les services : Inscription
Propriétés techniques	Liste des opérations : créerFormation (), afficheNombreCrédit (), afficheFraisFormation()
	Les dépendances: Services sur lesquels dépend ce service pour un bon fonctionnement : Gestion crédit, Gestion frais formation, copie. Services qui dépendent du service : inscription
	SLA : Le service supporte jusqu'à 30 requêtes simultanées.
	Document en entrée : Préférences Document en sortie : ListeFormation
	Fonctionnement en mode document
	Protocole de transport utilisé : SOAP over HTTP

Tableau 7-1 : Propriétés du service 'Formation'

Etape 6: Etablissement du contexte

Pour établir le contexte de conception, nous récapitulons les éléments essentiels des trois plans métier, service et infrastructure en prenant les processus métier modélisés comme cadre de référence. Ce travail permet d'une part d'alimenter la base de connaissances du contexte et d'autre part, de créer les liens de dépendance entre les éléments essentiels. Dans les étapes précédentes, nous avons explicité la façon d'identifier les éléments essentiels métier ainsi que les spécifications des services. Dans ce qui suit, nous identifions les éléments essentiels de l'infrastructure en identifiant dans un premier temps, les éléments à acquérir pour répondre aux besoins fonctionnels du projet.

Pour simplifier l'intégration des services hétérogènes des partenaires, le responsable technique a décidé de mettre en place un bus de service pour simplifier la gestion, l'adressage et le routage entre les services. En outre, le bus retenu intègre plusieurs fonctionnalités de sécurité et permet la mise en place d'une solution SSO fédérée nécessaire pour la propagation de l'identité des étudiants entre les différents partenaires.

Pour identifier les éléments essentiels de l'infrastructure, nous travaillons à partir des documents existants, notamment la représentation de l'architecture réseau (Figure 7-21). Ce type de document nous permet de simplifier l'identification des éléments de l'infrastructure et des liens de dépendances entre ces éléments. Pour chaque élément ainsi identifié, on dresse l'inventaire des logiciels d'hébergement et support, des systèmes d'exploitation, des équipements et des éléments de support. Le Tableau 7-2 récapitule les éléments essentiels de l'infrastructure :

Infrastructure	Logiciels d'hébergement et de support	BDD : Serveur base de données (Mysql 5.5)
		AS : Serveur d'application (Apache Tomcat 7.0.22)
		MO : Moteur d'orchestration (Apache ODE 1.3.5)
		SW : Serveur Web (Apache 2.2.21)
		ESB (Petals 3.1.3)
	Systèmes d'exploitation	D/L : Debian/Linux (Squeeze 6.0.3)
	Equipement	5 serveurs (Double Processeur 2.4GHZ / 8GB RAM Disques SCSI + RAID)
		3 switchs : Cisco catalyst 2900
		1 routeur : Cisco 2600
	Elément de support	Connexion au réseau Internet (débit dédié de 20 Mbits/s)

Tableau 7-2 : Eléments de l'infrastructure

Pour illustrer la recherche des dépendances, nous nous intéressons au service 'création formation'. Grâce au schéma d'architecture (Figure 7-21) et à l'inventaire, on peut construire une chaîne de liaison (Figure 7-22) montrant que l'exécution de ce service dépend :

- ✓ Des logiciels d'hébergement et de support.
- ✓ Du système d'exploitation « Debian/Linux »

- ✓ Du serveur matériel qui l'héberge
- ✓ Des équipements réseaux et de la connexion au réseau Internet.

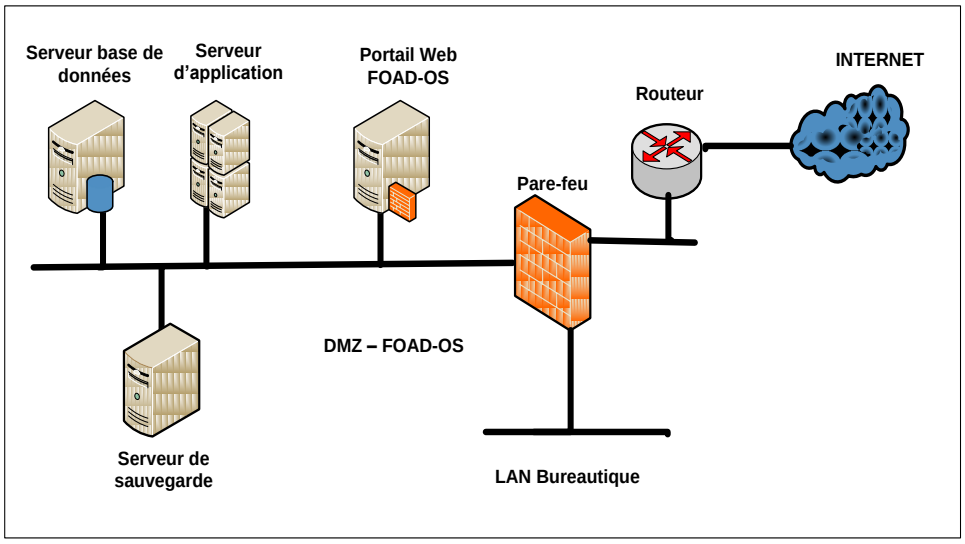


Figure 7-21: Architecture réseau – AUF

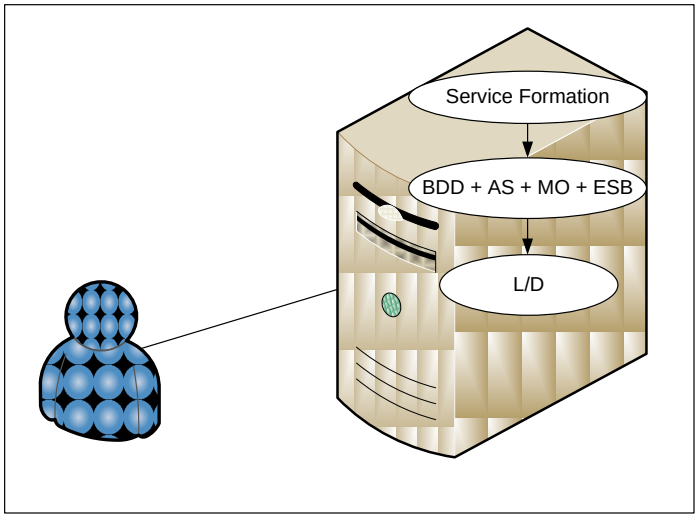


Figure 7-22 : Lien de dépendance

Une fois les éléments listés, nous classons les éléments essentiels afin de les prioriser. Dans ce qui suit, nous nous réduisons à l'étude de deux éléments essentiels : le portail web classé comme un élément 'critique' et les données confidentielles classées comme un élément 'important'.

Etape 7A: Identification des exigences de sécurité

Dans cette étape, nous utilisons les objectifs de sécurité métier identifiés dans la troisième étape pour déterminer les exigences de sécurité sur les éléments essentiels des trois plans en suivant le modèle de dépendance et en allant du plan métier au plan infrastructure. Nous donnons ci-après

une liste (non exhaustive car il s'agit de montrer les points clefs de la méthode) des exigences de sécurité:

- ✓ Objectif de sécurité 'Assurer une continuité du service FOAD-OS' : sera assuré en exigeant une garantie de disponibilité des services externes et en assurant la disponibilité des services internes et des composants de l'infrastructure. Par conséquent, nous définissons les exigences suivantes sur les différents plans :

Plan métier	✓ Garantir la disponibilité des services des partenaires *(L'exigence de disponibilité des services des partenaires devra être communiquée et garantie par les universités elles-mêmes)*
Plan service	✓ Assurer la disponibilité des services de la plateforme FOAD-OS
Plan infrastructure	✓ Assurer la disponibilité du portail web de la plateforme FOAD-OS ✓ Assurer la disponibilité des éléments essentiels de l'infrastructure (remédier aux attaques du genre DoS et vérifier la fiabilité du matériel)

- ✓ Objectif de sécurité 'Assurer la protection des données privées' : sera assuré en identifiant les données sensibles (niveau métier) et en protégeant ces données au niveau des services et des éléments de l'infrastructure. Ceci impose la définition des exigences de sécurité suivantes sur les différents plans :

Plan métier	✓ Classer les données / processus sensibles : attribuer un degré de sensibilité 'noir' aux données personnelles des étudiants et aux processus métier étant donné que ces données et processus nécessitent une protection maximale. Le reste des données ne nécessite pas de protection et on leur attribue un degré de sensibilité 'blanc' ✓ Mettre en place une politique d'utilisation de ces données au sein de l'agence et par les partenaires
Plan service	✓ Assurer la confidentialité des ces données en transit ✓ Assurer la confidentialité des ces données au repos
Plan infrastructure	✓ Assurer le contrôle d'accès aux éléments de l'infrastructure qui stockent ces données sensibles.

- ✓ Objectif de sécurité 'créer un réseau de confiance entre les différents partenaires' : sera assuré en identifiant les types de partenariat et en définissant les droits d'accès (niveau métier), en appliquant/contrôlant ces droits d'accès au niveau des services et des éléments de l'infrastructure. Par conséquent, nous définissons les exigences de sécurité suivantes sur les différents plans :

Plan métier	✓ Classer les partenaires selon un partenariat 'entre concurrent' ou un partenariat 'de complémentarité' ✓ Attribuer les droits d'accès d'une façon concise et cohérente
Plan service	✓ Contrôler les droits d'accès aux services fournis aux partenaires
Plan infrastructure	✓ Mettre en place des solutions pour contrôler et superviser l'application des politiques d'accès (PDP : Policy Decision Point et PEP : Policy Renforcement Point). ✓ Sécuriser les moyens de communication pour que personne ne puisse se faire passer pour un partenaire (Attaque Man-in-the-middle).

Etape 7B: Identification des risques

Dans cette étape, notre objectif est d'identifier les éléments liés au risque (évènements redoutés, menaces, scénarios des menaces et vulnérabilités) correspondant à chaque élément essentiel formant le contexte. Nous utilisons des ateliers de brainstorming en présence des responsables métier et techniques pour identifier les événements qui peuvent nuire aux éléments essentiels des plans métier, service et infrastructure (événements redoutés). Pour chaque événement, nous déterminons les sources et scénario de menace en se référant aux catalogues génériques des méthodes EBIOS et OCTAVE (niveau métier et infrastructure) et du document du NIST 800-95 (niveau service). En outre, le lien de dépendance nous permet d'identifier les menaces qui peuvent être dérivées à partir d'autres menaces (ex : une intrusion sur le réseau interne qui mène à l'accès sur la base de donnée menant à l'atteinte sur les données privées des étudiants). Enfin, nous identifions les vulnérabilités en nous référant aux catalogues génériques d'EBIOS, d'OCTAVE ainsi qu'aux bases de vulnérabilités du CERT et du MITRE pour les vulnérabilités spécifiques aux éléments de l'infrastructure.

Au cours de la séance de brainstorming, nous élaborons les diagrammes de menaces en utilisant le langage CORAS. Ces diagrammes nous permettent de simplifier l'identification des risques en illustrant la dépendance entre les éléments liés au risque et les enchainements lors d'une attaque. Par exemple, un scénario de menace 'serveur infecté par un ver informatique (worm)' peut mener à un autre un scénario de menace 'diffusion du ver et surcharge du réseau' qui mène à l'événement redouté 'Plateforme FOAD-OS inaccessible. Nous rappelons dans la Figure 7-23 les symboles du langage CORAS.

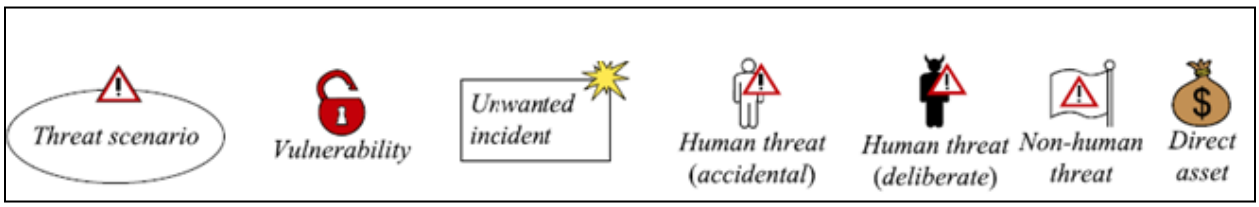


Figure 7-23 : Symboles du langage de modélisation des risques CORAS [95] p. 27

Pour illustrer cette étape, nous retenons deux éléments essentiels portail web et le service 'Profil'. Les autres éléments essentiels devront être traités avec la même attention. En effet, dans une étude de gestion des risques, la classification des biens essentiels est importante pour attribuer des priorités aux éléments à étudier.

Dans la Figure 7-24, nous nous focalisons sur l'atteinte à la confidentialité des données privées au niveau de la base de données des étudiants, données accessibles par le service 'Profil'. Ce service n'est accessible qu'à partir des autres services de la plateforme et du réseau interne de l'AUF. Les scénarios de menaces identifiées sont liés à : (1) l'injection d'un code malveillant au niveau du serveur web qui divulgue les données privées, (2) l'accès non autorisé au local technique, (3) l'accès non autorisé au service profil depuis le réseau de l'AUF. Bien que d'autres scénarios de menaces puissent être identifiés, nous nous limitons à ces trois scénarios qui nous permettent de mettre en évidence les étapes ultérieures.

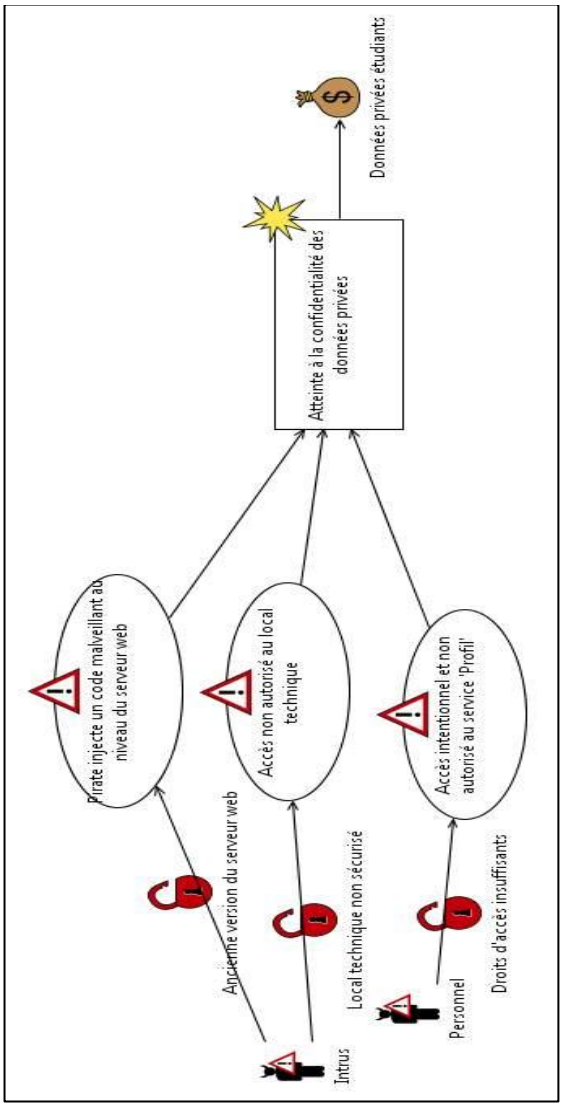


Figure 7-24 : Modélisation des menaces - Données privées étudiants

Dans la Figure 7-25, nous nous focalisons sur la disponibilité du Portail web. Nous identifions les évènements redoutés pouvant rendre ce portail indisponible. Nous identifions trois évènements redoutés liés à : (1) une attaque par déni de service, (2) une faille logicielle, (3) des problèmes sur les éléments de l'infrastructure (problème de matériel et de connexion réseau).

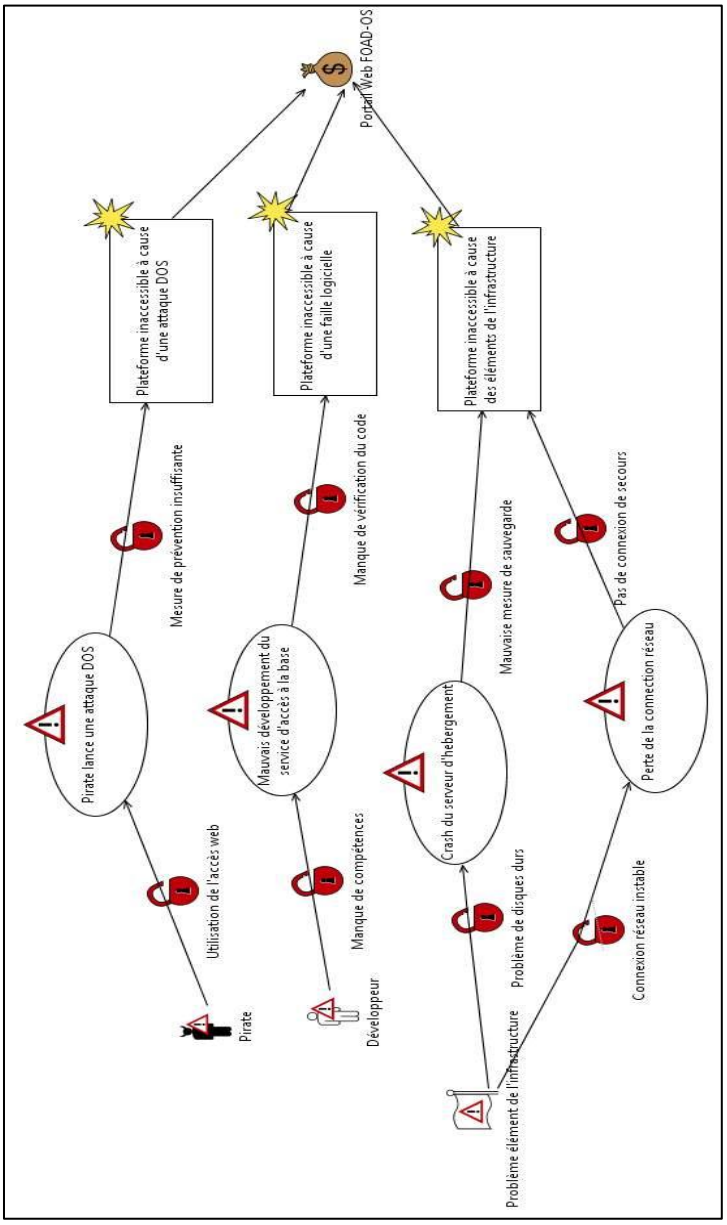


Figure 7-25 : Modélisation des menaces - Portail Web FOAD-OS

Etape 8 : Évaluation des risques

Pour évaluer les risques, nous établissons les échelles de l'évaluation de l'impact et des probabilités d'occurrence. Ces échelles définissent les valeurs que nous utiliserons dans l'attribution des probabilités d'occurrence aux événements redoutés et dans l'estimation des impacts de ces événements redoutés. Ensuite, nous définissons les fonctions du risque et les

fonctions de tolérance au risque relatives à chaque élément essentiel. Enfin, nous reprenons les diagrammes de menaces et nous déterminons les probabilités d'occurrence des événements redoutés et leurs impacts sur les biens essentiels. Ce travail est réalisé dans des séances de brainstorming entre les responsables technique et métier. Les probabilités d'occurrence des événements redoutés sont calculées en fonction des probabilités d'occurrence des scénarios de menaces et des probabilités que ces scénarios de menaces mènent à l'événement redouté. L'impact est estimé en fonction de la gravité de l'événement redouté et de ses conséquences sur l'Agence (perte d'image, productivité, crédibilité, etc.). Pour illustrer cette démarche, nous avons choisi le portail web et les données personnelles comme les deux éléments essentiels à étudier dans le reste de notre cas d'usage. Afin d'évaluer les risques, nous établissons l'échelle de l'impact des événements redoutés sur ces éléments essentiels (Tableau 7-3) et l'échelle de la probabilité d'occurrence (Tableau 7-4) de ces événements en fixant une durée de vie de 10 ans pour ce projet. Les fonctions du risque sont définis pour chaque élément essentiel pour chaque couple impact/probabilité (selon une échelle à trois niveaux : couleur bleu est un risque faible, couleur jaune est un risque moyen et couleur rouge est un risque élevé). Nous définissons pour chaque couple impact/probabilité le niveau de tolérance ajusté en fonction du coût de la sécurité confronté au coût de non sécurité (Tableau 7-5 et Tableau 7-6).

Impact	**Indisponibilité portail FOAD-OS**	**Atteinte à la confidentialité des données de la base**
Catastrophique	Indisponibilité de [2 heures, ∞[(*)	Atteinte à la confidentialité de [50%, 100%] des enregistrements de la base. (**)
Majeur	Indisponibilité de [5 minutes, 2 heures]	Atteinte à la confidentialité de [20%, 50%] des enregistrements de la base.
Mineur	Indisponibilité de [30 secondes, 5 minutes]	Atteinte à la confidentialité de [1%, 20%] des enregistrements de la base.
Insignifiant	Indisponibilité de [0 minute, 30 secondes]	Atteinte à la confidentialité de [0%, 1%] des enregistrements de la base.
(*) [min, max] = durée d'indisponibilité		(**) [min, max] = % des enregistrements atteints

Tableau 7-3 : Echelle de l'impact des évènements redoutés sur les éléments essentiels

Probabilité d'occurrence	**Description**	
Certain	[50, ∞[: 10 ans *	5 ou plus dans une période d'un an
Probable	[20, 49] :10 ans	2 à 5 fois par an
Possible	[10, 19] : 10 ans	1 à 2 fois par an
Rare	[2, 9] :10 ans	Moins d'une fois par an
*[min, max] : 10 ans = probabilité d'occurrence min et max pendant la durée de vie du projet		

Tableau 7-4 : Echelle de la probabilité d'occurrence

		Impact			
		Insignifiant (1)	**Mineur (2)**	**Majeur (3)**	**Catastrophique (4)**
Occurrence	**Rare (1)**	Risque Acceptable	Risque Acceptable	Risque Acceptable	Risque Inacceptable
	Possible (2)	Risque Acceptable	Risque Inacceptable	Risque Inacceptable	Risque Inacceptable
	Probable (3)	Risque Acceptable	Risque Inacceptable	Risque Inacceptable	Risque Inacceptable
	Certain (4)	Risque Inacceptable	Risque Inacceptable	Risque Inacceptable	Risque Inacceptable

Tableau 7-5 : Fonction de tolérance au risque : Portail FOAD-OS

		Impact			
		Insignifiant (1)	**Mineur (2)**	**Majeur (3)**	**Catastrophique (4)**
Occurrence	**Rare (1)**	Risque Acceptable	Risque Acceptable	Risque Acceptable	Risque Acceptable
	Possible (2)	Risque Acceptable	Risque Acceptable	Risque Acceptable	Risque Acceptable
	Probable (3)	Risque Acceptable	Risque Acceptable	Risque Inacceptable	Risque Inacceptable
	Certain (4)	Risque Acceptable	Risque Acceptable	Risque Inacceptable	Risque Inacceptable

Tableau 7-6 : Fonction de tolérance au risque : Données privées

Une fois que les échelles et les fonctions de tolérance au risque sont établies, nous passons à l'évaluation des risques. La méthode que nous utilisons repose sur les informations collectées dans l'étape 'établissement du contexte', en particulier le profil de sécurité de chaque élément essentiel. Nous procédons alors de la façon suivante :

- ✓ Nous estimons la probabilité d'occurrence des scénarios de menaces et la probabilité que les scénarios de menaces conduisent aux événements redoutés. Ceci est réalisé en évaluant les mesures de sécurité mises en place pouvant contrer ces menaces et les vulnérabilités pouvant être exploités par ces menaces.
- ✓ Nous calculons une valeur globale de la probabilité d'occurrence.

Les probabilités sont déterminées suite aux discussions avec les responsables métier et techniques. Dans la Figure 7-26, nous avons attribué :

- La probabilité 'rare' au scénario 'accès au local technique' étant donné que le local est verrouillé à clé et qu'il n'existe une procédure de contrôle d'accès à ces clefs.
- La probabilité 'rare' au scénario 'accès intentionnel au service 'Profil' étant donné que l'accès au service 'Profil' est bien contrôlé.
- La probabilité 'rare' à l'injection d'un code malveillant au niveau du serveur web par un pirate étant donné la veille de sécurité et la mise à jour périodique des logiciels d'hébergement.
- Une probabilité 0.4 (niveau moyen) à la divulgation des données privées interfacées par le service profil par ce code malveillant.
- Une probabilité 0.8 (niveau élevé) que l'accès au local technique conduise à l'atteinte à la confidentialité des données privées en raison d'un accès au local technique. (Il s'agit d'un accès physique aux équipements/accès logique via un terminal ouvert)
- Une probabilité 0.2 (niveau faible) à l'atteinte à la confidentialité des données privées à partir du service profil étant donné que le service n'est pas accessible depuis l'extérieur, que des mesures de sécurité sont mises en place contre les intrusions (filtrage, pare-feu, etc), qu'il existe une politique d'utilisation des données à l'Agence et que son personnel est formé à l'utilisation de ce service.

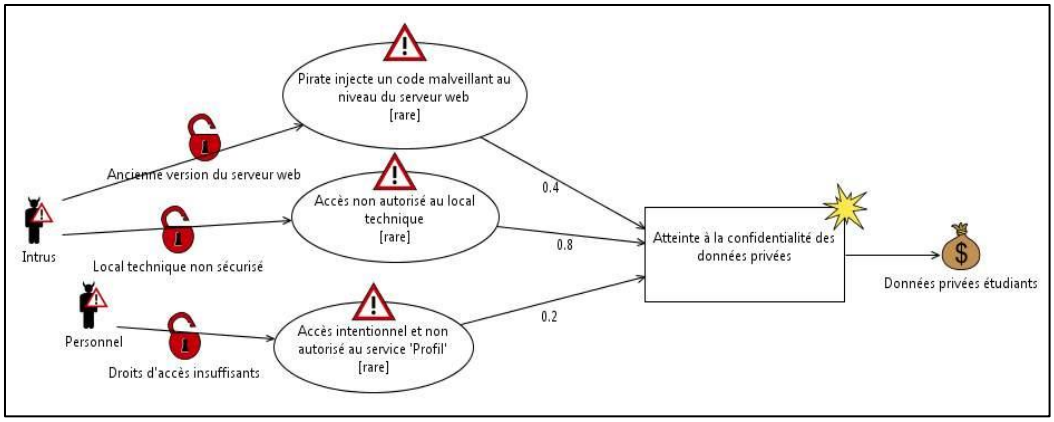


Figure 7-26 : Estimation des probabilités d'occurrence

Dans le Tableau 7-7, nous calculons la valeur combinée de la probabilité d'occurrence de l'évènement redouté. Cette valeur représente la probabilité d'occurrence de l'événement redouté.

	Scénario de menace (1)	Scénario de menace (2)	Scénario de menace (3)
Probabilité d'occurence	Rare : [2, 5] :10	Rare : [2, 5] :10	Rare : [2, 5] :10
Probabilité que le scénario conduit à l'évènement redouté	0.4	0.8	0.2
Valeur combinée	[2, 5] :10 x 0.4 = [0.8, 2.0]	[2, 5] :10 x 0.8 = [1.6, 4.0]	[2,5] :10x 0.2 = [0.4, 1] :10
Valeur globale de la probabilité d'occurrence de l'évènement redouté	[0.8, 2.0] :10 + [1.6, 4.0] :10 + [0.4, 1.0] :10 = [2.8, 7.0] :10 [2.8, 7.0] :10 = Rare moins d'une fois par an		

Tableau 7-7 : Calcul de la valeur globale de la probabilité d'occurrence

Nous passons ensuite à l'estimation de l'impact de l'événement redouté sur l'élément essentiel et sur l'Agence. Pour cela, nous utilisons les diagrammes de menaces pour estimer la sévérité de l'événement redouté. Les menaces et sources de menaces sont prises en compte lors de l'estimation de l'impact de l'événement redouté sur l'élément essentiel en utilisant l'échelle associée à cet élément. Dans notre cas d'illustration, les scénarios de menaces identifiés (accès physique / logique sur l'équipement / intrusion à distance) peuvent aboutir à une atteinte à la base des étudiants en donnant accès à tous ses enregistrements. D'après l'échelle associée, ceci sera considéré comme un impact catastrophique. Enfin, nous déterminons la valeur du risque en utilisant la fonction de tolérance au risque (Tableau 7-6). Dans notre exemple, un impact 'catastrophique' et une probabilité d'occurrence rare conduisent à un risque acceptable. En revanche, compte tenu des faiblesses au niveau de l'infrastructure (portail web hébergé sur un serveur non redondant, aucune solution de partage de trafic, qualité de service de la connexion non garantie) ces vulnérabilités seront exploitées par des menaces de type déni de service et conduiront à l'inaccessibilité du portail web. En estimant selon le même processus décrit ci-dessus la probabilité d'occurrence et l'impact de l'événement redouté 'portail web inaccessible'

et en se référant à la fonction de tolérance au risque (Tableau 7-5), nous trouvons que le risque 'Portail web inaccessible' est inacceptable. Tous les risques sont évalués selon le même processus.

Etape 9: Traitement des risques

Après avoir inventorié les risques inacceptables, cette étape permet de définir le traitement le plus adéquat au contexte de conception. Nous procédons selon la démarche suivante :

1- Nous dressons la matrice des risques inacceptables (risques qui devront être impérativement traités). Cette matrice nous permet de prioriser le traitement en fonction de la gravité de ces risques (probabilités d'occurrence des événements redoutés/impacts) déterminée dans l'étape précédente. Selon le niveau de chaque risque et les mesures de sécurité possibles, nous choisissons le traitement (transfert, évitement, réduction, prise) en fonction du coût de la sécurité (investissements et ressources humaines, temps, budget que pourra faire l'Agence)
2- Si une décision de réduction du risque est prise, nous procédons par une approche descendante qui intègre les plans métier, service et infrastructure pour identifier les mesures de sécurité. Nous nous référons aux catalogues des méthodes EBIOS et OCTAVE pour rechercher les mesures de sécurité adéquates aux niveaux métier et infrastructure et utilisons des patrons de sécurité pour traiter les risques au niveau du plan service en se référant au guide du NIST [123] et à [59].

Nous supposons dans ce qui suit que trois risques inacceptables ont été documentés dans la matrice du risque (Tableau 7-8) :

- ✓ IPU : Indisponibilité des Partenaires Universitaires
- ✓ CBDD : Atteinte à l'intégrité de la base de données de la plateforme.
- ✓ PWI : Portail web inaccessible

Dans cette matrice la couleur 'bleu' représente le niveau de risque acceptable et la couleur rouge est celle du niveau inacceptable. En se basant sur les niveaux de risque calculés, nous traitons successivement les risques PWI, CBDD et IPU.

		Impact			
		Insignifiant	Mineur	Majeur	Catastrophique
Occurrence	Rare				
	Possible			IPU	
	Probable			CBDD	PWI
	Certain				

Tableau 7-8 : Matrice du risque

D'après cette matrice, nous trouvons que la réduction du risque 'Portail web inaccessible' est inévitable étant donné que cet élément essentiel est d'une extrême importance pour le bon fonctionnement du projet. Pour réduire ce risque, nous procédons par une approche descendante qui intègre les plans métier, service et infrastructure. Nous récapitulons dans le Tableau 7-9 les mesures de sécurité à mettre en place :

Au niveau du plan métier	Mettre en place un PLA avec les universités partenaires garantissant la disponibilité des services et définissant un plan de rétablissement après l'indisponibilité des services de formation à distance suite à l'indisponibilité de la plateforme
Au niveau du plan service	Nous trouvons qu'il va falloir remédier aux attaques de type déni de service XML, (XML-DoS) qui peuvent conduire au crash du serveur d'application (intensive XML parsing – buffer overflow). Pour réduire ce risque, nous utilisons le pattern 'Message inspector gateway pattern' [59] qui permet d'intercepter le trafic pour détecter des éventuelles attaques DOS.
Au niveau du plan infrastructure	Nous trouvons qu'il va falloir remédier aux attaques de déni de service sur les applications d'hébergement et les systèmes d'exploitation, nous pensons alors à : ✓ Améliorer et vérifier les règles de filtrage au niveau du pare-feu ✓ Superviser régulièrement les mises à jour de sécurité, tester et installer les correctifs nécessaires des logiciels d'hébergement et de support ainsi que des systèmes d'exploitation. ✓ Vérifier la fiabilité du matériel et la configuration des disques RAID au niveau des serveurs matériel. ✓ Améliorer l'architecture réseau en mettant en place deux serveurs d'applications et en faisant un partage de trafic : 'Load Balancing' entre les deux serveurs. ✓ Mettre en place une connexion Internet de sauvegarde par l'intermédiaire d'un deuxième fournisseur d'accès.

Tableau 7-9 : Traitement des risques

Etape 10: Annotation de sécurité

Comme nous l'avons évoqué, l'annotation de sécurité a un double rôle :

- ✓ L'AUF, en tant que fournisseur de services pourra indiquer le niveau de sécurité globale de ses services après leur conception.
- ✓ L'AUF en tant que consommateur de services externes, pourra utiliser l'annotation de sécurité des services des partenaires dans la sélection dynamique des services. À titre d'exemple, deux services fournis par deux universités différentes offrant les mêmes propriétés fonctionnelles pourront être choisis selon leur disponibilité, étant donné que cette exigence est primordiale pour le bon fonctionnement de la plateforme.

L'annotation des services est réalisée en deux étapes :

1- Nous déterminons les éléments de l'annotation de sécurité (disponibilité, confidentialité, supervision) à utiliser dans l'annotation en fonction des exigences de sécurité à satisfaire.

2- Pour chaque service et élément de l'annotation, nous reprenons le lien de dépendance établit dans l'étape 6 et nous évaluons la sécurité des éléments pertinents définis dans l'ontologie OAS en identifiant les mesures de sécurité mises en place permettant de satisfaire les exigences de sécurité.

Pour illustrer l'annotation des services, nous nous réduisons à l'annotation du service 'profil' par l'élément 'confidentialité' étant donné que ce service gère les données confidentielles et que la confidentialité est une exigence à satisfaire. Par conséquent, nous évaluons :

a. La confidentialité des données émises/stockées par le service
b. Le contrôle d'accès aux composants de l'infrastructure hébergeant ce service (logiciels d'hébergement et de support, système d'exploitation, équipement)

Dans notre cas d'usage, nous n'avons pas identifié des mesures de chiffrement permettant de garantir la confidentialité des données transmises/stockées. Toutefois, nous avons identifié qu'il existe des mécanismes de contrôle d'accès sur la totalité des éléments de l'infrastructure. Par conséquent, nous annotons le service 'profil' par une confidentialité de 71.4% (Tableau 7-10).

	Service profil	**Eléments essentiels**	**Mesures de sécurité (Confidentialité assurée)**	$VAC = \sum_{i=1}^{n} \frac{x_i}{N}$ = 5/7 = 0.714
Confidentialité	Plan service	Données en transit	Aucune (-)	Avec: VAC = Valeur de l'annotation de Confidentialité. x_i = Instances des éléments essentiels pertinents dont la disponibilité est assurée. N = Instances des éléments essentiels pertinents à la confidentialité.
		Données au repos	Aucune (-)	
	Plan Infrastructure	Serveur d'application	Contrôle d'accès (+)	
		Serveur base de données	Contrôle d'accès (+)	
		ESB	Contrôle d'accès (+)	
		Système d'exploitation	Contrôle d'accès (+)	
		Equipement	Contrôle d'accès (+)	

Tableau 7-10 : Annotation Service Profil: Confidentialité

Changement du contexte de conception

Pour mettre en évidence l'influence du changement du contexte dans l'adaptation de la stratégie de sécurité, nous supposons que l'AUF a décidé d'exposer le service 'profil' à ses partenaires pour qu'ils puissent avoir un accès facile aux profils des étudiants. Suite à ce changement, il va falloir entamer à nouveau le cycle de gestion des risques puisque le service profil a été conçu pour une utilisation interne. Il faut donc adapter les mesures de la sécurité à ce nouveau contexte en prenant en compte la transmission sécurisée des données personnelles sur le réseau public ainsi que l'usage de ces données par les universités partenaires. Pour cela, il faut mettre en place un PLA garantissant la protection des données privées chez les universités partenaires (Plan métier) et la mise en place de mécanismes de chiffrement des données lors de la transmission des données (Plan service).

7.4 Conclusion

Dans ce chapitre, nous avons aussi décrit notre outil de conception d'une SOA sécurisée. Cet outil intègre quatre modules support : le module de gestion des éléments essentiels, le module d'évaluation de la sécurité, le module de calcul des annotations et le module de gestion des dépendances. Cet outil supporte la première phase 'Identification et spécification des services' ainsi que la troisième phase 'Annotation de sécurité' de la méthodologie MCSS et pourra être amélioré pour couvrir la phase de gestion des risques (intégration des bases de connaissances support pour l'identification et le traitement des risques).

Nous avons présenté ensuite un cas d'usage : une plateforme de Formation Ouverte et à Distance – Orientée Services (FOAD-OS). Cette plateforme représente une nouvelle génération de la plateforme FOAD actuelle de l'Agence Universitaire de la Francophonie. Elle permet de fournir des formations multidisciplinaires personnalisées selon les préférences des étudiants. Le travail réalisé dans le cas d'usage illustre la démarche suivie pour la conception d'une SOA sécurisée, en particulier la conception de services réutilisables, l'utilisation de services existants, la gestion des risques et enfin l'adaptation de la sécurité au changement du contexte.

Conclusion Générale et Perspectives

L'environnement distribué et dynamique créé par les architectures orientées services a ajouté de nouveaux défis au niveau de la sécurité tant au niveau individuel des services qu'au niveau de la composition. Dans cet environnement, la sécurité ne doit pas se limiter à fournir des solutions technologiques mais à trouver une stratégie de sécurité prenant en compte les dimensions métier, organisationnelle et technologique. En outre, la sécurité doit être appréhendée comme un processus continu qui vise l'optimisation des investissements de sécurité et assure la pérennité des mesures de sécurité mises en œuvre. Or les modèles et architectures de référence du domaine des services ont sous-estimé l'identification des risques métier et organisationnels et la définition des biens à protéger. De même, les méthodologies de développement des SOA se focalisent sur les dimensions fonctionnelles et prêtent moins d'attention à la sécurité.

Afin de répondre à cette problématique, nous avons défini un cadre méthodologique permettant d'identifier les biens (métier et technologiques) et les liens de dépendance entre ces eux, d'identifier les risques portant sur ces biens et enfin de proposer les mesures de sécurité les plus adéquates au contexte en intégrant la gestion de la sécurité dans les phases de préparation, conception, exécution et supervision du cycle de vie des services. Notre contribution est résumée par les points suivants.

I- Au niveau de la phase de préparation, nous avons développé :

1- Un métamodèle de service sécurisé qui permet de définir des patrons de conception. Ce métamodèle est l'union de :

a) Un modèle de service qui met en évidence les éléments essentiels (éléments à protéger) liés au service. Dans ce modèle, nous faisons le lien entre le service et son environnement métier et technologique.

b) Un modèle de politique de sécurité qui met en évidence les objectifs et les besoins de sécurité.

c) Un modèle de risque qui met en évidence les risques métier et technologiques.

2- Un modèle de classification des biens à protéger. Nous avons modélisé la sécurité comme un plan orthogonal à trois plans :

1) le plan métier décrit le cadre métier, organisationnel et légal. Ce cadre est constitué de l'ensemble des processus métier et des documents métier, des partenaires, des acteurs et de leurs rôles, des accords du niveau de protection (PLA global), des préférences de sécurité et des lois et des obligations légales.

2) le plan service regroupe les concepts associés aux services : opérations, messages, corps des messages, assertions de sécurité et qualité de protection.

3) le plan infrastructure décrit le cadre technologique et l'environnement d'hébergement des services : logiciels d'hébergement et de support, systèmes d'exploitation, équipements et éléments de support.

3- Une ontologie de conception d'une SOA sécurisée qui permet de définir les concepts des éléments essentiels des trois plans d'abstraction.
Tous ces modèles et ces concepts sont à la base de la conception d'une SOA sécurisée.

II- Au niveau de la phase de conception, nous avons développé une méthodologie de conception d'une SOA sécurisée qui définit le contexte de conception en identifiant les biens essentiels et les liens entre eux avant d'intégrer un cycle de gestion des risques nous permettant d'identifier les risques et les mesures de sécurité à mettre en œuvre. Dans la dernière étape de la méthodologie, nous récapitulons les informations de sécurité pour annoter les services par des paramètres de sécurité qui seront utilisés dans la phase d'exécution. Ces annotations de sécurité ont un double rôle. Pour le fournisseur de service, elles représentent un recensement d'une sécurité globale. Pour un consommateur de services, elles peuvent être utilisées pour améliorer la sélection entre différents services qui répondent aux exigences fonctionnelles.

III- Au niveau de la phase d'exécution, nous avons proposé une architecture de sélection dynamique des services sécurisés en utilisant l'annotation de sécurité. Au cœur de cette architecture, nous avons placé un service intermédiaire – Security Broker – qui permet de publier les annotations de sécurité dans l'annuaire et de sélectionner des services sécurisés selon les préférences des utilisateurs.

IV- Au niveau de la phase de supervision, nous avons proposé un service de gestion des vulnérabilités liées à l'infrastructure. Ce service vérifie auprès des bases de vulnérabilités publiques la présence de vulnérabilités associées aux logiciels d'hébergement et aux systèmes d'exploitation mis en place au niveau de l'infrastructure.

Nous n'avons pas abordé les phases de construction, de déploiement et de test du cycle de vie des services. Ces phases sont abordées dans le cadre d'une autre thèse au sein de notre équipe.

Les travaux réalisés dans le cadre de cette thèse ouvrent diverses perspectives et plusieurs travaux futurs peuvent être envisagés :

1. Développement de modèles permettant de définir des patrons métier pour prendre en compte les différents modes de collaboration entre les partenaires. Ces patrons seront utilisés pour traiter les risques métier et seront complémentaires des patrons de sécurité technologiques.

2. Amélioration de l'annotation de sécurité :
   L'annotation de sécurité proposée est un premier pas dans la spécification des paramètres de sécurité. Ce concept pourra être étendu:

- ✓ En passant d'une annotation de veille de sécurité à une annotation d'assurance de sécurité.
- ✓ En améliorant le calcul de la valeur globale en donnant des poids aux éléments essentiels.
- ✓ En définissant d'autres éléments de l'annotation en rapport avec les besoins de sécurité (ex : intégrité des données) et/ou les éléments de l'infrastructure (ex : gestion de la vulnérabilité)

3. Certification des annotations de sécurité :
   L'annotation de sécurité permet la sélection de services sécurisés parmi d'autres services. Toutefois, la question de la véracité de l'annotation est primordiale dans la sélection des services. Pour cela, nous proposons de créer une infrastructure de certification, qui s'apparente à l'architecture des PKI (public key infrastructure). Ceci suppose un tiers de confiance pour certifier les annotations de sécurité. Nous proposons également d'étendre :
   - ✓ Le service de sécurité intermédiaire ('security broker') définit dans notre architecture de sélection dynamique pour gérer les requêtes de certification auprès du tiers de confiance avant de publier les annotations.
   - ✓ le module de sélection du service de sécurité intermédiaire pour vérifier la signature des annotations de sécurité.

   En autre alternative, nous pouvons étendre le service de sécurité intermédiaire afin qu'il assure les fonctions du tiers de confiance. Ceci suppose d'étendre le module de publication pour certifier les annotations de sécurité et le module de sélection pour vérifier les signatures.

Bibliographie

[1] CHAARI S. *Interconnexion des processus Interentreprises : une approche orientée services*. Thèse Doctorat. Lyon : INSA de Lyon, 2008. 177 p.

[2] Schroth C. « The service-oriented enterprise ». *Journal of enterprise architecture*. 2007. Vol. 3, n°4, p. 73–80.

[3] ANSSI. *EBIOS: Expression des Besoins et Identification des Objectifs de Sécurité*. [En ligne]. 2010. Disponible sur : < http://www.ssi.gouv.fr/ > (consulté le 8 juin 2010)

[4] ANSSI. *Référentiel général de la sécurité*. [En ligne]. Disponible sur : < http://www.ssi.gouv.fr/fr/reglementation-ssi/referentiel-general-de-securite/ > (consulté le 15 décembre 2010)

[5] INSEE. *Institut National de la statistique et des études économiques - Définition Services*. [En ligne]. Disponible sur : < http://www.insee.fr > (consulté le 13 novembre 2011)

[6] Gronroos C. *Service management and marketing : managing customer relationships for service and manufacturing firms*. 2nd ed.Chichester : Wiley, 2000. ISBN : 9780471720348.

[7] The Open Group. *Ontologies for SOA*. [En ligne]. 2010. Disponible sur : < http://www.opengroup.org/projects/soa-ontology/ > (consulté le 4 décembre 2010)

[8] Godart C. *Les processus métiers : concepts, modèles et systèmes*. Paris : Hermès science publications-Lavoisier, 2009. ISBN : 9782746223004.

[9] Erl T. *SOA : principles of service design*. Upper Saddle River NJ : Prentice Hall, 2008. ISBN : 9780132344821.

[10] Brown P. *Succeeding with SOA realizing business value through total architecture*. Upper Saddle River, NJ : Addison-Wesley, 2007. ISBN : 9780321508911.

[11] Rosen M. *Applied SOA : service-oriented architecture and design strategies*. Indianapolis IN : Wiley Pub, 2008. ISBN : 9780470223659.

[12] Bonnet P. *Urbanisation des Systèmes d'Information: Cadre de référence SOA, Synthèse Méthode Praxeme SOA*. [En ligne]. 2006. Disponible sur : < http://urba-si.blogspot.com/2006/07/cadre-de-rfrence-soa-synthse-mthode.html > (consulté le 5 juin 2010)

[13] Erl T. *Service-oriented architecture : concepts, technology, and design*. Upper Saddle River NJ : Prentice Hall Professional Technical Reference, 2005. ISBN : 9780131858589.

[14] Booth D., Haas H., McCabe F. *SOA and Web Services Architecture*. [En ligne]. 2004. Disponible sur : < http://www.w3.org/TR/ws-arch/ > (consulté le 16 novembre 2010)

[15] Krafzig D. *Enterprise SOA : service-oriented architecture best practices*. 6. print.Upper Saddle River NJ : Prentice-Hall, 2006. ISBN : 9780131465756.

[16] Colombo M. et al. « Speaking a common language: A conceptual model for describing service-oriented systems ». *Service-Oriented Computing-ICSOC 2005*. 2005. p. 48–60.

[17] Emig C. et al. « Model-driven development of SOA services ». *Cooperation & Management, Universit\ät Karlsruhe (TH), Internal Research Report*. 2008.

[18] C. Matthew M. et al. *OASIS Reference Model for Service Oriented Architecture 1.0.* [En ligne]. 2006. Disponible sur : < http://docs.oasis-open.org/soa-rm/v1.0/ > (consulté le 4 juin 2010)

[19] Kreger H., Jeff E. *Navigating the SOA Open Standards Landscape Around Architecture*. [En ligne]. 2009. Disponible sur : < http://www.opengroup.org/pubs/catalog/w096.htm > (consulté le 13 mars 2010)

[20] OMG. *SOA Modeling Language (SoaML)*. [En ligne]. 2009. Disponible sur : < http://www.omg.org/spec/SoaML/ > (consulté le 22 août 2010)

[21] The Open Group. *SOA Reference Architecture*. [En ligne]. avril 2009. Disponible sur : < https://www.opengroup.org/projects/soa-ref-arch/uploads/40/19713/ > (consulté le 15 août 2010)

[22] Estefan J. A. et al. *OASIS Reference Architecture for Service Oriented Architecture Version 1.0*. [En ligne]. 2008. Disponible sur : < http://docs.oasis-open.org/soa-rm/soa-ra/v1.0/ > (consulté le 4 juin 2010)

[23] The Open Group. *SOA Integration Maturity*. [En ligne]. 2009. Disponible sur : < http://www.opengroup.org/projects/osimm/ > (consulté le 22 mai 2010)

[24] The Open Group. *SOA Governance*. [En ligne]. 2009. Disponible sur : < http://www.opengroup.org/projects/soa-governance/ > (consulté le 27 septembre 2010)

[25] Wall Q. *SOA Service Lifecycle Design*. [En ligne]. 10 avril 2006. Disponible sur : < http://www.oracle.com/technetwork/articles/entarch/soa-service-lifecycle-design-096035.html > (consulté le 4 août 2010)

[26] Papazoglou M. P., Van Den Heuvel W. J. « Service-oriented design and development methodology ». *International Journal of Web Engineering and Technology*. 2006. Vol. 2, n°4, p. 412–442.

[27] Papazoglou M. P. et al. « Service-oriented computing: State of the art and research challenges ». *COMPUTER-IEEE COMPUTER SOCIETY-*. 2007. Vol. 40, n°11, p. 38.

[28] Schmidt M. T. et al. « The enterprise service bus: making service-oriented architecture real ». *IBM Systems Journal*. 2005. Vol. 44, n°4, p. 781–797.

[29] Chappell D. *Enterprise service bus*. Beijing;Cambridge : O'Reilly, 2004. ISBN : 9780596006754.

[30] Broeckelmann R., Triplett R. *Secure identity propagation using WS-trust, SAML2 and WS-security*. [En ligne]. 2011. Disponible sur : < http://thinkmiddleware.com/blog01/wp-content/uploads/2012/04/IdentityPropagationPresentation_v10.0.pdf > (consulté le 4 juillet 2012)

[31] Marks E. A., Bell M. *Service-oriented architecture : a planning and implementation guide for business and technology*. Hoboken, N.J. : Wiley, 2006. ISBN : 0471768944 9780471768944.

[32] Arsanjani A. et al. « SOMA: a method for developing service-oriented solutions ». *IBM Syst. J.* 2008. Vol. 47, n°3, p. 377-396.

[33] Allen P. « CBDI The service oriented process ». *CBDI Journal*. février 2007.

[34] Erradi A., Anand S., Kulkarni N. « SOAF: An Architectural Framework for Service Definition and Realization ». In : *Services Computing, 2006. SCC '06. IEEE International Conference on*. Chicago, IL : IEEE Computer Society, 2006. p. 151 -158.

[35] Jones S., Morris M. « A methodology for service architectures ». *Capgemini UK plc*. 2005. Vol. 201, p. 202–204.

[36] OMG. *CORBA 3.1*. [En ligne]. Disponible sur : < http://www.omg.org/spec/CORBA/3.1/ > (consulté le 24 juillet 2008)

[37] ORACLE. *Enterprise JavaBeans Technology*. [En ligne]. Disponible sur : < http://www.oracle.com/technetwork/java/javaee/ejb/index.html > (consulté le 24 novembre 2009)

[38] *Web Services Glossary*. [En ligne]. Disponible sur : < http://www.w3.org/TR/2004/NOTE-ws-gloss-20040211/ > (consulté le 24 novembre 2010)

[39] Chinnici R. et al. *Web services description language (wsdl) version 2.0 part 1: Core language*. 2007.

[40] Gudgin M. et al. « SOAP Version 1.2 ». *W3C recommendation*. 2003. Vol. 24, p. 12.

[41] Bellwood T. et al. « UDDI Version 3.0 ». *Published specification, Oasis*. 2002. Vol. 5, p. 16–18.

[42] Vedamuthu A., Orchard D., Hirsch F. « Web Services Policy 1.5 ». Disponible sur : < http://www.w3.org/TR/ws-policy/ > (consulté le 16 novembre 2009)

[43] Box D. et al. « Web services policy assertions language (WS-PolicyAssertions) ». *MSDN Library*. 2003. Vol. 1, p. 22-24.

[44] Bajaj S. et al. *Web Services Policy Attachment (WS-PolicyAttachment) version 1.2*. 2006.

[45] Lawrence K. et al. « WS-SecurityPolicy 1.3 ». *OASIS Standard, February*. 2009.

[46] Jordan D. et al. *Web services business process execution language version 2.0*. 2007.

[47] Kavantzas N. et al. « Web services choreography description language version 1.0 ». *W3C Working Draft*. 2004. Vol. 17, p. 10–20041217.

[48] CHARIF Y. *Chorégraphie dynamique de services basée sur la coordination d'agents introspectifs*. Thèse Doctorat. Paris : Université Pierre et Marie Curie Paris VI, 2007. 181 p.

[49] NIST. *Standards for Security Categorization of Federal Information and Information Systems*. [En ligne]. 2004. Disponible sur : < http://csrc.nist.gov > (consulté le 10 septembre 2009)

[50] Alberts C. *Managing information security risks : the OCTAVE approach*. Boston : Addison-Wesley, 2003. ISBN : 9780321118868.

[51] Stallings W. *Sécurité des réseaux : applications et standards*. Paris : Vuibert, 2002. ISBN : 9782711786534.

[52] ANSSI. *Agence nationale de la sécurité des systèmes d'information*. [En ligne]. Disponible sur : < http://www.ssi.gouv.fr/ > (consulté le 5 mai 2012)

[53] Kim A., Luo J., Kang M. « Security ontology for annotating resources ». *On the Move to Meaningful Internet Systems 2005: CoopIS, DOA, and ODBASE*. 2005. p. 1483–1499.

[54] Schneier B. *Applied cryptography : protocols, algorithms, and source code in C*. 2nd ed.New York : Wiley, 1996. ISBN : 9780471128458.

[55] BUECKER A. et al. *Understanding SOA Security Design and Implementation*. [En ligne]. 2007. Disponible sur : < http://www.redbooks.ibm.com/abstracts/sg247310.html > (consulté le 4 septembre 2010)

[56] Samarati P., De Vimercati S. « Access control: Policies, models, and mechanisms ». *Foundations of Security Analysis and Design*. 2001. p. 137–196.

[57] Ferraiolo D., Cugini J., Kuhn D. R. « Role-based access control (RBAC): Features and motivations ». In : *Proceedings of 11th Annual Computer Security Application Conference*. Washington : IEEE Computer Society Press, 1995. p. 241–48.

[58] Hachani wafa. *Patrons de conception à base d'aspects pour l'ingénierie des systèmes d'information par réutilisation*. Thèse Doctorat. GRENOBLE : Université Joseph Fourier, 2006.

[59] Steel C. *Core security patterns best practices and strategies for J2EE, Web services, and identity management*. Upper Saddle River, NJ : Prentice Hall PTR, 2006. ISBN : 9780131463073.

[60] Kanneganti R. *SOA security*. Greenwich CT : Manning Pubns Co, 2008. ISBN : 9781932394689.

[61] Cranor L. *Web privacy with P3P*. 1st ed.Beijing;Sebastopol Calif : O'Reilly, 2002. ISBN : 9780596003715.

[62] *A P3P Preference Exchange Language 1.0 (APPEL1.0)*. [En ligne]. Disponible sur : < http://www.w3.org/TR/2002/WD-P3P-preferences-20020415/ > (consulté le 28 mars 2011)

[63] Sliman L., Biennier F., Badr Y. « A security policy framework for context-aware and user preferences in e-services ». *Journal of Systems Architecture*. avril 2009. Vol. 55, n°4, p. 275-288.

[64] Agrawal R. et al. « XPref: a preference language for P3P ». *Computer Networks*. 2005. Vol. 48, n°5, p. 809–827.

[65] *P3P Using the Semantic Web (OWL Ontology, RDF Policy and RDQL Rules)*. [En ligne]. Disponible sur : < http://www.w3.org/P3P/2004/040920_p3p-sw.html > (consulté le 28 mars 2011)

[66] Trabelsi S., Gomez L., Roudier Y. « Context-Aware Security Policy for the Service Discovery ». In : *International Conference on Advanced Information Networking and Applications*. AINAW : IEEE, 2007. p. 477 -482.

[67] Preibusch S. « Privacy negotiations with p3p ». In : *W3C Workshop on Languages for Privacy Policy Negotiation and Semantics-Driven Enforcement*. Ispra/Italy : W3C, 2006.

[68] Della-Libera G. et al. « Security in a web services world: A proposed architecture and roadmap ». *Online whitepaper, IBM/Microsoft*. 2002. Vol. 7, p. 1-25.

[69] Nadalin A. et al. « Web Services Security ». *Standard specification, OASIS*. 2004. p. 59.

[70] Eastlake D. et al. « Xml encryption syntax and processing ». *W3C Recommendation*. 2002. p. 46.

[71] Hafner M. *Security engineering for service-oriented architectures*. Berlin : Springer, 2009. ISBN : 9783540795384.

[72] Hughes J., Maler E. « Security Assertion Markup Language (SAML) V2. 0 ». *OASIS SSTC Working Draft*. 2005. p. 61.

[73] Anderson A. « Core and hierarchical role based access control (RBAC) profile of XACML v2. 0 ». *OASIS Standard*. 2005. p. 2005.

[74] Anderson S. et al. « Web services trust language (ws-trust) ». *Public draft release, Actional Corporation, BEA Systems, Computer Associates International, International Business Machines Corporation, Layer*. 2005. Vol. 7, p. 68.

[75] Kaler C. et al. « Web Services Federation Language (WS-Federation) V1.2 ». *OASIS Standard*. 2009. p. 140.

[76] Anderson S. et al. « Web services secure conversation language (WS-SecureConversation) ». *Actional Corporation/BEA Systems Inc./Computer Associates International Inc./IBM Corporation/Layer*. 2005. Vol. 7, p. 35.

[77] *EEC, Information Technology Security Evaluation Criteria (ITSEC), Rapport Technique*. [En ligne]. Disponible sur : < http://csrc.nist.gov/publications/secpubs/itsec.txt > (consulté le 15 septembre 2009)

[78] MATHIEU H. *Modélisation conjointe de l'infrastructure et des processus pour l'administration pro-active de l'entreprise distribuée*. Thèse Doctorat. Lyon : INSA de Lyon, 2004. 252 p.

[79] ISO/IEC. *Common Criteria for Information Technology Security Evaluation - Part 1: Introduction and general model*. [En ligne]. 1998. Disponible sur : < www.commoncriteriaportal.org > (consulté le 9 décembre 2009)

[80] ISO/IEC, The International Organization for Standardization and The International Electrotechnical Commission. *ISO/IEC 27002:2005*. [En ligne]. 2005. Disponible sur : < http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=50297 > (consulté le 13 octobre 2009)

[81] NIST. *Federal Information Security Management Act (FISMA) Implementation Project*. [En ligne]. Disponible sur : < http://csrc.nist.gov/groups/SMA/fisma/index.html > (consulté le 7 décembre 2010)

[82] Biennier F., Mathieu H. « Technical Solutions vs. Global BPR Investment ». *Schedae Informaticae*. 2005. Vol. 14, p. 13-34.

[83] Vacca J. *Managing information security*. Burlington MA : Elsevier, 2010. ISBN : 9781597495332.

[84] Clusif. *Plan de continuité d'activité, stratégie et solutions de secours du S.I.* [En ligne]. 2003. Disponible sur : < http://www.clusif.asso.fr/ > (consulté le 8 mars 2010)

[85] ISO. *ISO Guide 73:2009 - Risk management Vocabulary*. [En ligne]. 2009. Disponible sur : < http://www.iso.org/iso/catalogue_detail?csnumber=44651 > (consulté le 14 décembre 2010)

[86] Harry M. « Portfolio selection ». *Journal of Finance*. 1952. Vol. 7, n°1, p. 77–91.

[87] Robillard L. *Integrated risk management framework*. [En ligne]. 2001. Disponible sur : < http://interuniversity.ns.ca > (consulté le 17 mai 2009)

[88] Gourc D. *Vers un modèle général du risque pour le pilotage et la conduite des activités de biens et de services*. Habilitation à diriger des recherches. Toulouse : Institut National Polytechnique de Toulouse, 2006. 122 p.

[89] Hopkin P. *Fundamentals of risk management : understanding, evaluating, and implementing effective risk management.* London; Philadelphia : Kogan Page, 2010. ISBN : 9780749459420 0749459425.

[90] Sienou A. *Proposition d'un cadre méthodologique pour le management intégré des risques et des processus d'entreprise*. Thèse Doctorat. Toulouse : Université de Toulouse, 2009. 263 p.

[91] Arthur J. D. et al. « Mitigating Security Risks in Systems that Support Pervasive Services and Computing: Access-Driven Verification, Validation and Testing ». In : *IEEE International Conference on Pervasive Services*. Virginia : IEEE, 2007. p. 109 -117.

[92] Lowis L. « Towards automated risk identification in serviceoriented architectures ». *Proceedings of the Multikonferenz Wirtschaftsinformatik (MKWI)*. 2008. p. 253–254.

[93] Yee G. « Visual Analysis of Privacy Risks in Web Services ». In : *ICWS IEEE International Conference on Web Services*. Council Canada, Ottawa : IEEE, 2007. p. 671 -678.

[94] Kokash N. « Risk Management for Service-Oriented Systems ». *Web Engineering*. p. 563–568.

[95] Alberts C. et al. *OCTAVE: Information Security Risk Evaluation*. [En ligne]. 2001. Disponible sur : < http://www.cert.org/octave/ > (consulté le 10 janvier 2009)

[96] Mead N. R. et al. *Survivable network analysis method*. [En ligne]. 2000. Disponible sur : < www.cert.org/archive/pdf/00tr013.pdf > (consulté le 15 septembre 2009)

[97] Lund M. *Model-driven risk analysis : the CORAS approach*. Berlin : Springer, 2010. ISBN : 9783642123221.

[98] Clusif. *MEHARI : Méthode Harmonisée d'Analyse de Risques*. [En ligne]. 2010. Disponible sur : < http://www.clusif.asso.fr/ > (consulté le 26 juin 2010)

[99] Siemens/Insight. *CRAMM : CCTA Risk Analysis and Management Method*. [En ligne]. Disponible sur : < http://www.cramm.com/ > (consulté le 13 septembre 2010)

[100] Microsoft. *Microsoft Threat Analysis and Modeling*. [En ligne]. 2009. Disponible sur : < http://archive.msdn.microsoft.com/tam > (consulté le 9 mars 2010)

[101] Schneier B. « Attack trees ». *Dr. Dobb's journal*. 1999. Vol. 24, n°12, p. 21–29.

[102] Bouti A., Ait Kadi D. « A state-of-the-art review of FMEA/FMECA ». *International Journal of reliability, quality and safety engineering*. 1994. Vol. 1, n°4, p. 515–543.

[103] Heckerman D. « A tutorial on learning with Bayesian networks ». *Innovations in Bayesian Networks*. 2008. p. 33–82.

[104] Ouedraogo W. F., Biennier F., Ghodous P. « Adaptive security policy model to deploy business process in cloud infrastructure. ». Disponible sur : < http://liris.cnrs.fr/Documents/Liris-5558.pdf > (consulté le 5 janvier 2012)

[105] Badr Y., Biennier F., Tata S. « The Integration of Corporate Security Strategies in Collaborative Business Processes ». *IEEE Transactions on Services Computing*. 2010.

[106] ISO/IEC. *Common Criteria for Information Technology Security Evaluation*. [En ligne]. 2007. Disponible sur : < www.commoncriteriaportal.org > (consulté le 12 septembre 2009)

[107] Herrmann P., Herrmann G. « Security requirement analysis of business processes ». *Electronic Commerce Research*. 2006. Vol. 6, n°3, p. 305–335.

[108] Nassar P. B. et al. « Towards integrating security services in e-learning platforms ». In : *ACTEA 09 International Conference on Advances in Computational Tools for Engineering Applications, 2009*. Zouk Mosbeh : IEEE, 2009. p. 573 -577.

[109] Keita A. K., Roussey C., Laurini R. « Un outil d'aide à la construction d'ontologies pré-consensuelles: le projet Towntology ». *Actes du 24ème congrès INFORSID*. 2006. Vol. 31, p. 911–926.

[110] Larousse - Dictionnaire Français. *Définition : protocole*. [En ligne]. Disponible sur : < http://www.larousse.fr > (consulté le 26 mai 2011)

[111] CBDI-SAE. *Meta Model for SOA Version 2*. [En ligne]. 2007. Disponible sur : < http://www.cbdiforum.com/public/meta_model_v2.php > (consulté le 4 juin 2010)

[112] OMG. *BMM Business Motivation Model*. [En ligne]. Disponible sur : < http://www.omg.org/spec/BMM/1.1/ > (consulté le 12 mars 2010)

[113] ANSI. *American National Standards Institute*. [En ligne]. Disponible sur : < http://www.ansi.org/ > (consulté le 13 octobre 2010)

[114] Koivunen M. R. « W3C semantic web activity ». *Semantic Web KickOff in Finland*. 2001. p. 27–41.

[115] ACORD. *Insurance Data Standards*. [En ligne]. Disponible sur : < http://www.acord.org > (consulté le 10 mars 2011)

[116] HL7. *Health Level Seven International*. [En ligne]. Disponible sur : < http://www.hl7.org/ > (consulté le 16 janvier 2011)

[117] NIEM. *National Information Exchange Model*. [En ligne]. Disponible sur : < https://www.niem.gov > (consulté le 13 janvier 2010)

[118] Kohlborn T. et al. « Identification and analysis of business and software services—a consolidated approach ». *Services Computing, IEEE Transactions on*. 2009. Vol. 2, n°1, p. 50–64.

[119] Chen F., Li S., Chu W. C.-C. « Feature Analysis for Service-Oriented Reengineering ». In : *Proceedings of the 12th Asia-Pacific Software Engineering Conference*. Washington, DC, USA : IEEE Computer Society, 2005. p. 201–208.

[120] Amsden J. *Modeling with SoaML, the Service-Oriented Architecture Modeling Language: Part 2. Service specification*. [En ligne]. 14 janvier 2010. Disponible sur : < http://www.ibm.com/developerworks/rational/library/09/modelingwithsoaml-2/index.html > (consulté le 7 septembre 2010)

[121] *Agence nationale de la sécurité des systèmes d'information*. [En ligne]. Disponible sur : < http://www.ssi.gouv.fr/ > (consulté le 8 juin 2010)

[122] OW2. *Petals ESB, the Open Source ESB for large SOA infrastructures*. [En ligne]. Disponible sur : < http://petals.ow2.org/ > (consulté le 13 mai 2010)

[123] NIST. *Guide to secure web services*. [En ligne]. 2007. Disponible sur : < http://csrc.nist.gov > (consulté le 10 juillet 2009)

[124] Buecker A., Lodewijkx K. *Cloud Security Guidance IBM Recommendations for the Implementation of Cloud Security*. février 2009.

[125] Microsoft. *Security in the Business Productivity Online Suite*. [En ligne]. août 2009. Disponible sur : < www.microsoft.com > (consulté le 15 décembre 2009)

[126] Chaari S., Badr Y., Biennier F. « Enhancing web service selection by QoS-based ontology and WS-policy ». In : *Proceedings of the 2008 ACM symposium on Applied computing*. New York, NY, USA : ACM, 2008. p. 2426–2431.

[127] Parigot D., Boussemart B. *Architecture Orienté Service Dynamique: D-SOA*. [En ligne]. 2008. Disponible sur : < http://hal.inria.fr/inria-00342310/ > (consulté le 13 décembre 2009)

[128] Garcia D. Z. G., De Toledo M. B. F. « A web service Architecture providing QoS Management ». In : *Web Congress, 2006. LA-Web'06. Fourth Latin American* [En ligne]. [s.l.] : [s.n.], 2006. p. 189–198. Disponible sur : < http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4022109 > (consulté le 2 août 2012)

[129] Rajendran T., Balasubramanie P., Cherian R. « An efficient WS-QoS broker based architecture for web services selection ». *International Journal of Computer Applications IJCA*. 2010. Vol. 1, n°9, p. 75–80.

[130] Badr Y. et al. « Enhancing Web Service Selection by User Preferences of Non-functional Features ». In : *Proceedings of the 2008 4th International Conference on Next Generation Web Services Practices*. Washington, DC, USA : IEEE Computer Society, 2008. p. 60–65.

[131] NIST. *NVD: National Vulnerability Database*. [En ligne]. Disponible sur : < http://nvd.nist.gov/ > (consulté le 16 janvier 2010)

[132] *OSVDB: The Open Source Vulnerability Database*. [En ligne]. Disponible sur : < http://osvdb.org/ > (consulté le 11 septembre 2010)

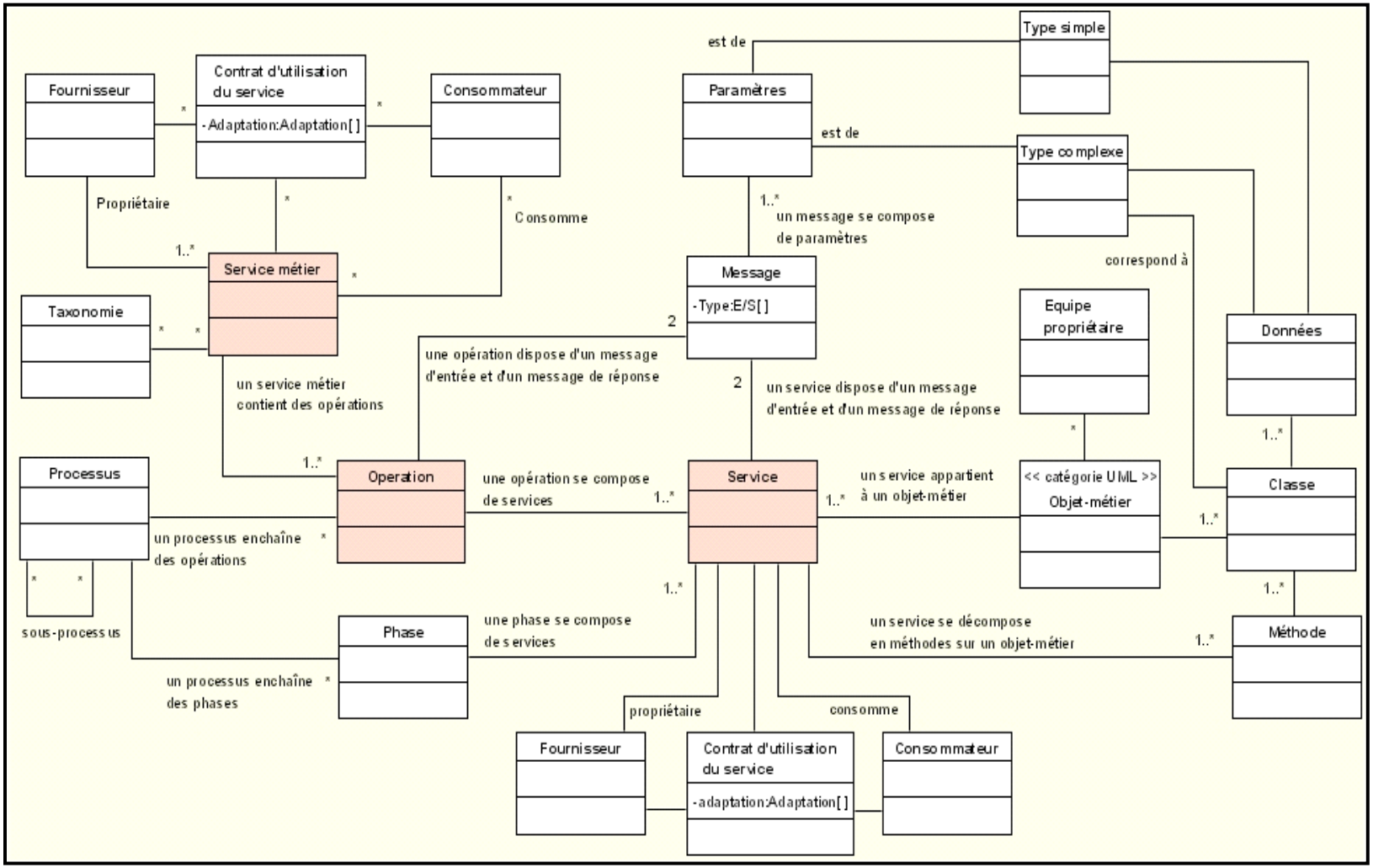
[133] *US-CERT: United States Computer Emergency Readiness Team*. [En ligne]. Disponible sur : < http://www.us-cert.gov/ > (consulté le 11 octobre 2011)

[134] Buttner A., Ziring N. « Common platform enumeration (cpe) ». *specification MITRE*. 2008. p. 37.

[135] *AUF - Agence Universitaire de la Francophonie*. [En ligne]. Disponible sur : < http://www.auf.org/ > (consulté le 16 décembre 2011)

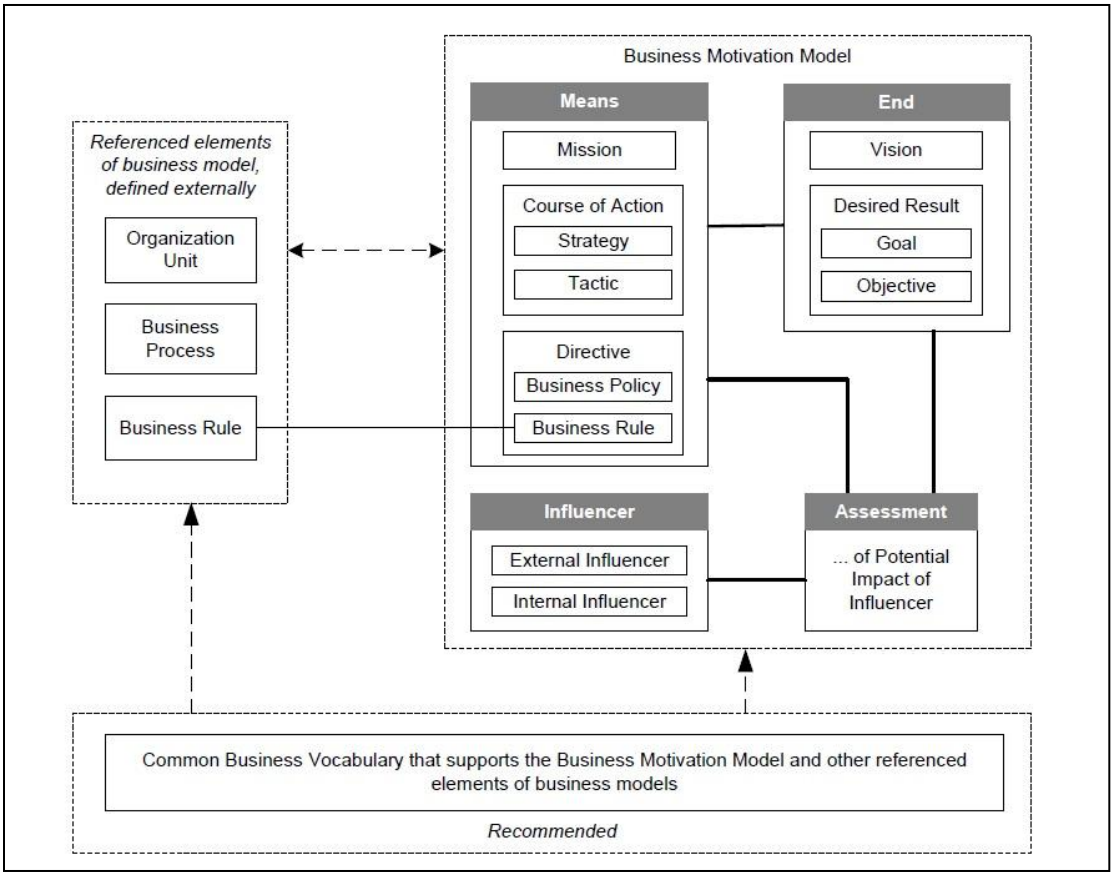
[136] ANSSI. *FEROS : Fiche d'Expression Rationnelle des Objectifs de Sécurité*. [En ligne]. 2005. Disponible sur : < http://www.ssi.gouv.fr > (consulté le 8 juin 2010)

Annexe



Annexe 1: Métamodèle SOA

Annexe 2 : Modèle de motivation métier - OMG



BMM : Business Motivation Model [112] p.12

Le BMM consiste en :

- La finalité (end) est ce que l'entreprise veut accomplir ou atteindre par exemple: Le développement d'un nouveau métier. La finalité consiste en une vision qui est l'image globale de ce que l'organisation veut être ou devenir, et en résultats qu'elle souhaite atteindre (desired results), par exemple, buts et objectifs.
- Les moyens sont ceux que l'entreprise décide de faire pour atteindre la finalité. Un moyen peut être une capacité ou une technique exécutée pour atteindre des buts. La mission indique l'activité en cours de fonctionnement de l'entreprise. Les stratégies et tactiques sont celles que l'entreprise décide de faire pour atteindre les objectifs. Les directives consistent en des politiques de gouverner, contrôler, guider et former les stratégies et tactiques. Ils définissent ce qui peut être fait et ce qui ne doit pas être fait, et peuvent indiquer des limites.
- Les influenceurs sont ceux qui peuvent causer des changements affectant l'entreprise dans la réalisation de ses objectifs ou dans la mise en place des moyens de réalisation des objectifs.
- Une évaluation (assessment) est l'évaluation de l'impact des influenceurs sur la capacité de l'entreprise à implémenter les moyens ou à parvenir à ses finalités.

Annexe 3 : Modélisation en BPMN

Le BPMN propose deux catégories d'objets:

1. La première caractérise la description du flot de contrôle comme le montre la figure ci-dessous :

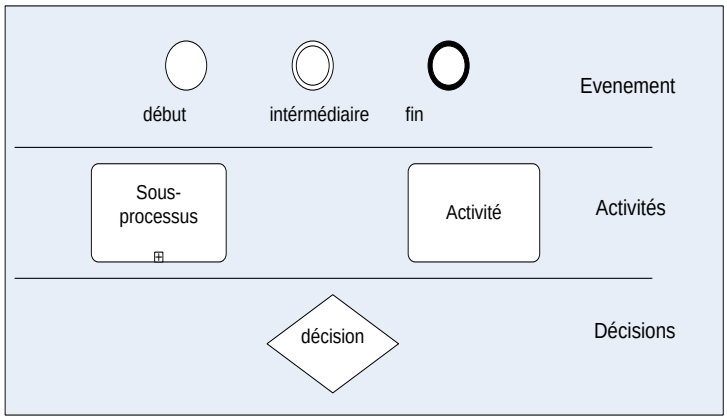


Figure A 1 : Les objets : événements, activités et décision en BPMN

Cette catégorie distingue :

a) Les évènements de début, intermédiaire et de fin affectant la vie du processus
b) Les activités qui peuvent être atomique ou composites (sous-processus)
c) Les décisions contrôlant la convergence ou la divergence de plusieurs flots.

2. La seconde catégorie concerne les objets de connexion, elle distingue : les flots de connexion, les flots de séquence, les flots conditionnels, les flots de message et les associations (Figure A 2)

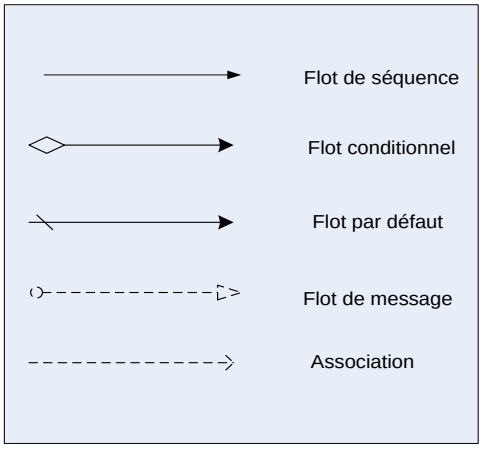


Figure A 2 : Les objets de connexion en BPMN

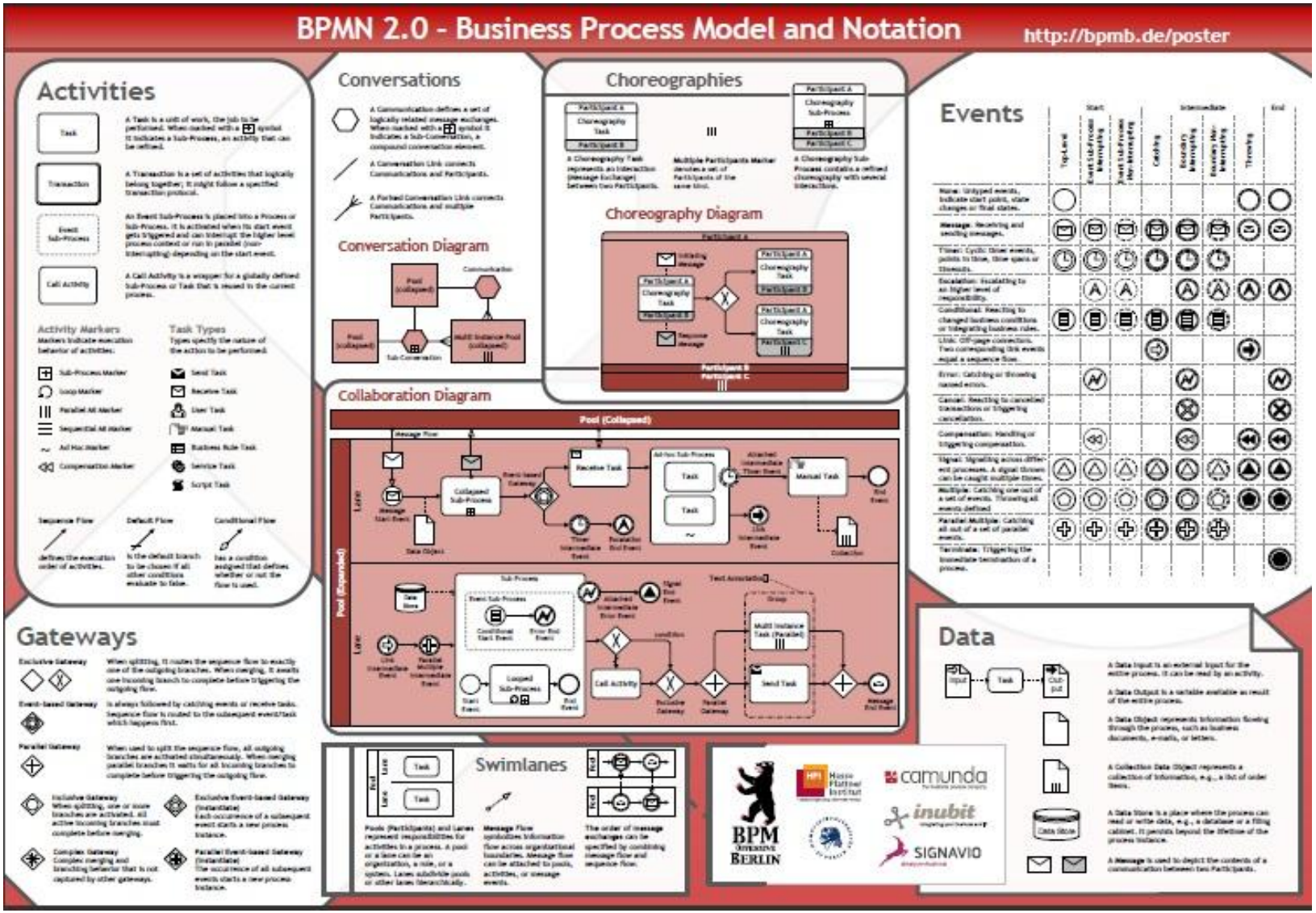


Figure A 3 : Business Process Model and Notation 2.0 [5]

[5] http://bpmb.de/poster

Annexe 4: Création des diagrammes de classe UML

Les diagrammes de classe UML permettent de modéliser les objets métier du domaine, leurs attributs et leurs interactions.

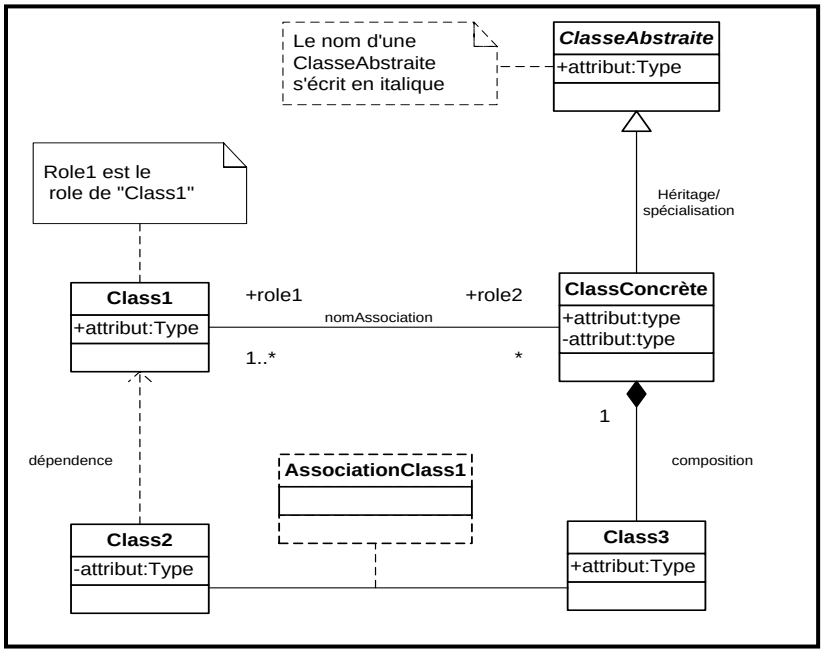


Figure A 4 : Les éléments d'un diagramme de classe

Éléments	Détails
Classe	Les classes représentent les objets métier tangibles mais de plus les objets métier intangibles tel que les rôles et les interactions.
Classe abstraite	Les classes abstraites sont des classes que nous ne pouvons pas instanciés, elles sont réservées pour dériver des attributs.
Classe concrète	Une classe qui pourra être instanciée directement
Classe Association	Une classe association permet de modéliser l'association sous la forme de classe dans le cas où plusieurs attributs caractérisent l'association

Les types des attributs sont d'une importance primordiale pour la cohérence du modèle sémantique, ils représentent les concepts de base d'un domaine particulier. Leurs noms et leur sémantique sont significatives au domaine métier. Nous pouvons avoir deux types complexes ayant le même nom mais représentant des domaines métier différents. Par exemple si nous créons un type complexe « étudiant »

a) Au niveau du domaine métier « Gestion des profils », l'attribut adresse est obligatoire.
b) Au niveau du domaine métier « Gestion des cours », l'attribut adresse est optionnel.

Annexe 5 : Schéma XML d'un document métier

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
    <xs:element name="produit">
        <xs:annotation>
            <xs:documentation>produit achat en ligne</xs:documentation>
        </xs:annotation>
        <xs:complexType>
            <xs:sequence>
                <xs:element name="Fournisseur">
                    <xs:complexType>
                      <xs:sequence>
                        <xs:element name="ID" type="xs:integer"/>
                        <xs:element name="Nom" type="xs:string"/>
                        <xs:element name="Adresse" type="xs:string" />
                      </xs:sequence>
                    </xs:complexType>
                </xs:element>
                <xs:element name="Prix">
                    <xs:complexType>
                      <xs:sequence>
                        <xs:element name="Montant" type="xs:decimal"/>
                        <xs:element name="Monnaie" type="xs:string"/>
                      </xs:sequence>
                    </xs:complexType>
                </xs:element>
            </xs:sequence>
            <xs:attribute name="ID" type="xs:integer" use="required"/>
            <xs:attribute name="Nom" type="xs:string" use="required"/>
            <xs:attribute name="Description" type="xs:string" use="optional" />
        </xs:complexType>
    </xs:element>
</xs:schema>
```

Figure A 5 : Schéma XML du document produit

Annexe 6 : Démarche d'élaboration d'une Fiche d'Expression Rationnelle des Objectifs de Sécurité (FEROS) [136] p.5

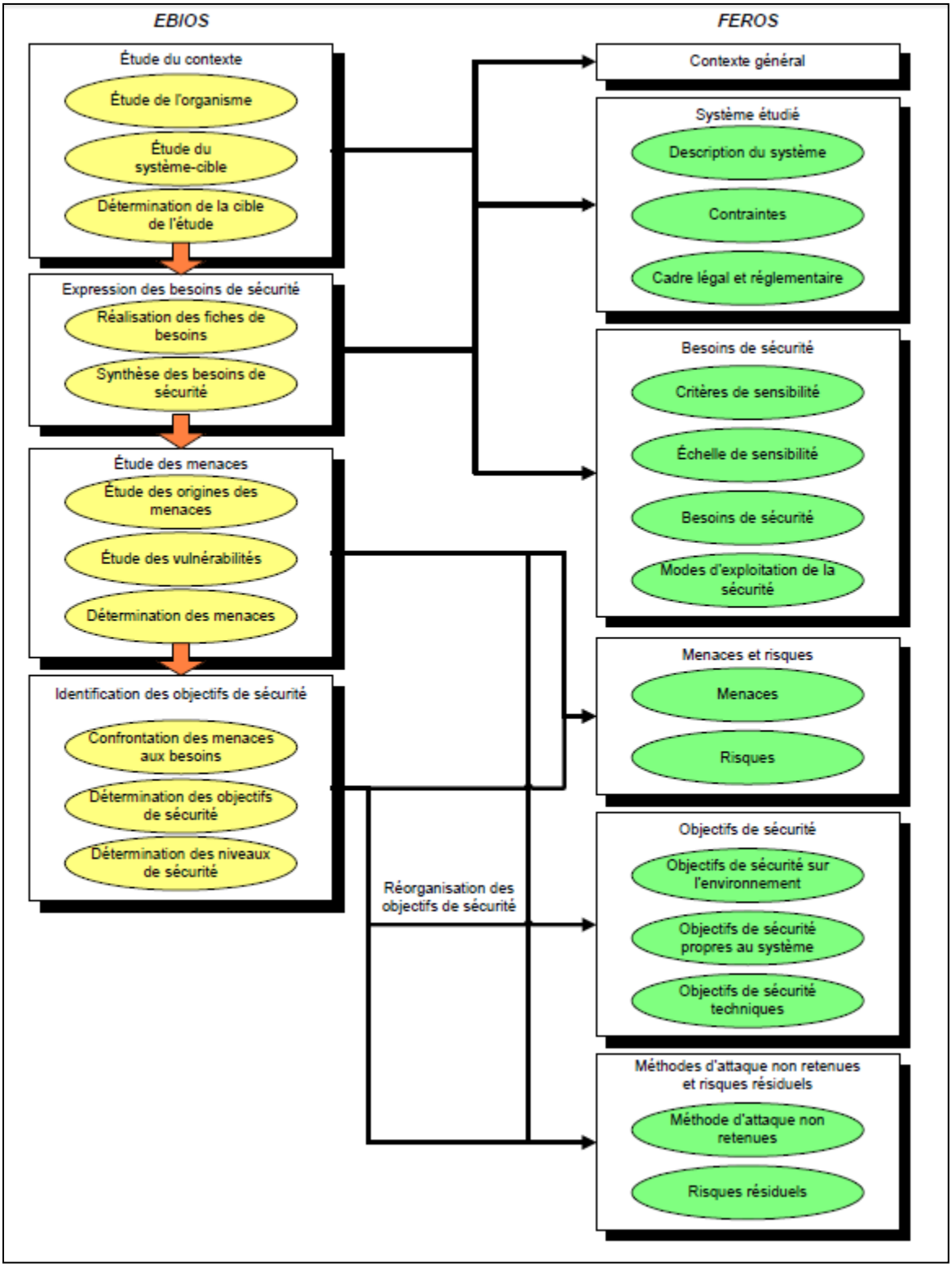


Figure A 6 : Démarche FEROS

Annexe 7 : Extrait du catalogue de la méthode EBIOS : Exigences de sécurité

Code	Libellé
BCM_CLI.1.1	Les classifications et les mesures de protection associées ayant trait à l'information devront tenir compte des besoins de l'entreprise de partager ou de restreindre l'information et des impacts professionnels relatifs à ces besoins
BCM_CLI.1.2	Si possible, la responsabilité de l'attribution d'une classification à une information et de la revue périodique de cette classification doit incombée à l'émetteur de l'information ou à son propriétaire attitré
BCM_CLI.2.1	Un ensemble de procédures doit être défini pour l'étiquetage et le traitement de l'information conformément au système de classification adopté par l'organisme

Code	Libellé
BCO_CEL.1.1	Toutes les exigences légales, réglementaires et contractuelles doivent être définies explicitement et documentées pour chaque système informatique
BCO_CEL.2.1	Des procédures appropriées doivent être appliquées afin d'assurer la conformité aux exigences légales sur l'utilisation de produits pour lesquels il pourrait y avoir des droits de propriété intellectuelle et sur l'utilisation de logiciels propriétaires
BCO_CEL.3.1	Les registres importants de l'organisme doivent être protégés contre toute perte, destruction et falsification
BCO_CEL.4.1	Des mesures de maîtrise doivent être appliquées afin de protéger les renseignements personnels conformément à la législation pertinente
BCO_CEL.5.1	La direction doit autoriser l'utilisation des infrastructures de traitement de l'information et des mesures de maîtrise doivent être appliquées pour empêcher l'utilisation abusive de ces infrastructures

Figure A 7 : Exemples exigences de sécurité métier

Code	Libellé
LOG_01	L'intégrité des logiciels et des données doit être garantie
LOG_02	Les mises à jour logicielles ne doivent dégrader ni la sécurité, ni les fonctionnalités des versions antérieures
LOG_03	Toutes les opérations de mises à jour réalisées sur les logicielles doivent être identifiables et justifiables
LOG_04	La configuration des systèmes et applications doit être conforme aux exigences de la politique de sécurité
LOG_05	Toute malveillance ou négligence pesant sur les applications sensibles ainsi que sur les systèmes les hébergeant doit être détectée

Code	Libellé
RES_01	Les accès aux interfaces de communication doivent être protégés contre une utilisation malveillante ou abusive
RES_02	Les interfaces de communication doivent protéger les transmissions en confidentialité, intégrité et disponibilité
RES_03	L'authentification et la non répudiation des communications doit pouvoir en cas de besoin être établie
RES_04	La compatibilité des éléments interconnectés doit être assurée (langues, fuseaux horaires, normes…)
RES_05	Il doit exister un plan de routage à jour et clair

Figure A 8 : Exemples exigences de sécurité techniques

Annexe 8 : Extrait du catalogue de la méthode EBIOS : Sources de menaces, menaces et vulnérabilités génériques [3] Bases de Connaissances p. 15

SOURCES HUMAINES AGISSANT DE MANIÈRE DÉLIBÉRÉE

- *Source humaine interne, malveillante, avec de faibles capacités*
- *Source humaine interne, malveillante, avec des capacités importantes*
- *Source humaine interne, malveillante, avec des capacités illimitées*
- *Source humaine externe, malveillante, avec de faibles capacités*
- *Source humaine externe, malveillante, avec des capacités importantes*
- *Source humaine externe, malveillante, avec des capacités illimitées*

SOURCES HUMAINES AGISSANT DE MANIÈRE ACCIDENTELLE

- *Source humaine interne, sans intention de nuire, avec de faibles capacités*
- *Source humaine interne, sans intention de nuire, avec des capacités importantes*
- *Source humaine interne, sans intention de nuire, avec des capacités illimitées*
- *Source humaine externe, sans intention de nuire, avec de faibles capacités*
- *Source humaine externe, sans intention de nuire, avec des capacités importantes*
- *Source humaine externe, sans intention de nuire, avec des capacités illimitées*

SOURCES NON HUMAINES

- *Code malveillant d'origine inconnue*
- *Phénomène naturel*
- *Catastrophe naturelle ou sanitaire*
- *Activité animale*
- *Événement interne*

EBIOS : MENACES ET VULNÉRABILITÉS GÉNÉRIQUES

MENACES SUR LES MATÉRIELS

- *M1. MAT-USG – Détournement de l'usage prévu d'un matériel*
- *M2. MAT-ESP – Espionnage d'un matériel*
- *M3. MAT-DEP – Dépassement des limites de fonctionnement d'un matériel*
- *M4. MAT-DET – Détérioration d'un matériel*
- *M5. MAT-MOD – Modification d'un matériel*
- *M6. MAT-PTE – Perte d'un matériel*

MENACES SUR LES LOGICIELS

- *M7. LOG-USG – Détournement de l'usage prévu d'un logiciel*
- *M8. LOG-ESP – Analyse d'un logiciel*
- *M9. LOG-DEP – Dépassement des limites d'un logiciel*
- *M10. LOG-DET – Suppression de tout ou partie d'un logiciel*
- *M11. LOG-MOD – Modification d'un logiciel*
- *M12. LOG-PTE – Disparition d'un logiciel*

MENACES SUR LES CANAUX INFORMATIQUES ET DE TÉLÉPHONIE

- *M13. RSX-USG – Attaque du milieu sur un canal informatique ou de téléphonie*
- *M14. RSX-ESP – Écoute passive d'un canal informatique ou de téléphonie*
- *M15. RSX-DEP – Saturation d'un canal informatique ou de téléphonie*
- *M16. RSX-DET – Dégradation d'un canal informatique ou de téléphonie*

- *M17. RSX-MOD – Modification d'un canal informatique ou de téléphonie*
- *M18. RSX-PTE – Disparition d'un canal informatique ou de téléphonie*

MENACES SUR LES PERSONNES

- *M19. PER-USG – Dissipation de l'activité d'une personne*
- *M20. PER-ESP – Espionnage d'une personne à distance*
- *M21. PER-DEP – Surcharge des capacités d'une personne*
- *M22. PER-DET – Dégradation d'une personne*
- *M23. PER-MOD – Influence sur une personne*
- *M24. PER-PTE – Départ d'une personne*

MENACES SUR LES SUPPORTS PAPIER

- *M25. PAP-USG – Détournement de l'usage prévu d'un support papier*
- *M26. PAP-ESP – Espionnage d'un support papier*
- *M27. PAP-DET – Détérioration d'un support papier*
- *M28. PAP-PTE – Perte d'un support papier*

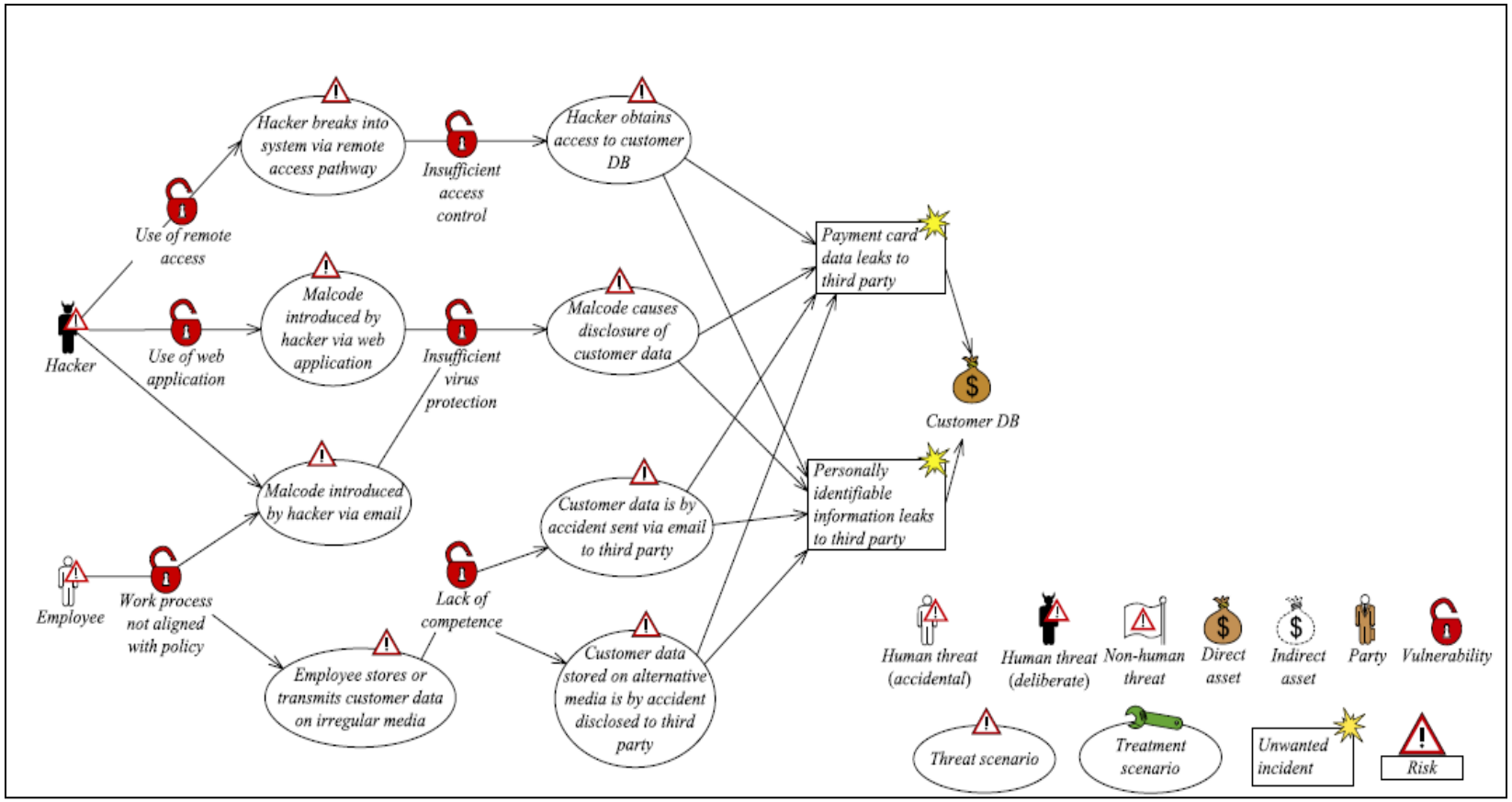
MENACES SUR LES CANAUX INTERPERSONNELS

- *M29. CAN-USG – Manipulation via un canal interpersonnel*
- *M30. CAN-ESP – Espionnage d'un canal interpersonnel*
- *M31. CAN-DEP – Saturation d'un canal interpersonnel*
- *M32. CAN-DET – Dégradation d'un canal interpersonnel*
- *M33. CAN-MOD – Modification d'un canal interpersonnel*
- *M34. CAN-PTE – Disparition d'un canal interpersonnel*

Annexe 9 : Extrait Catalogue OCTAVE (menaces et sources de menaces (octave-s implementation [95] p. 75)

Description	Example*
Problems with services provided by third parties (e.g., maintenance of systems) lead to disclosure of information to unauthorized parties.	A staff member from a third-party service provider views confidential information on a key business system that is maintained by that service provider.
Problems with services provided by third parties (e.g., maintenance of systems) lead to modification of information on a system.	Problems at a third-party service provider lead to the modification of information on a key business system located at that provider's site and maintained by the provider.
Problems with services provided by third parties (e.g., maintenance of systems) lead to loss or destruction of information on a system.	Problems at a third-party service provider lead to the destruction of information on a key business system located at that provider's site and maintained by the provider.
Problems with services provided by third parties (e.g., maintenance of systems) lead to interruption of access to a system.	A system maintained by a third-party service provider and located at the provider's site is unavailable due to problems created by that provider's staff.
Problems with the power supply lead to loss or destruction of information on a system.	A power outage results in loss of any information that was not saved at the time of the outage.
Problems with the power supply lead to interruption of access to a system.	A power outage prevents access to all key business systems.
A hardware defect results in modification of information on a system.	A disk drive develops a hardware problem that affects the integrity of a database that is stored on the disk.
A hardware defect results in the loss or destruction of information on a system.	A disk drive develops a hardware problem that ends up destroying the information on the disk. Files can be retrieved only from backups.
A hardware defect results in a system crash, preventing access to the system.	A disk drive develops a hardware problem, preventing access to any information on the disk until the problem is corrected.
A system is affected by malicious code (virus, worm, Trojan horse, back door) that enables unauthorized parties to view information.	A back door on a system enables unauthorized people to access the system and view customer credit card information on that system.
A system is affected by malicious code (virus, worm, Trojan horse, back door) that modifies information on that system.	A system is infected with a virus that modifies a process control application on the computer's disk drive.
A system is affected by malicious code (virus, worm, Trojan horse, back door) that deletes information on that system.	A system is infected with a virus that deletes all information on the computer's disk drive.
A system is affected by malicious code (virus, worm, Trojan horse, back door) that results in the system crashing.	A system is infected with a virus that is spread via email, slowing network traffic and creating a denial-of-services attack.

Annexe 10 : Symboles du langage de modélisation CORAS [97] p. 27



Annexe 11 : Structure de l'annuaire UDDI

L'annuaire UDDI stocke l'information concernant les services et les entreprises qui les publient dans des :

- "pages blanches" décrivant l'entreprise qui offre le service : nom, adresse, contacts,
- "pages jaunes" comportant les catégories industrielles basées sur des classements standards
- "pages vertes" décrivant les détails techniques des services publiés

Pour décrire ces informations, UDDI définit quatre structures de données :

- 'businessEntity' représente les informations qui concernent les organisations.
- 'businessService' inclut les informations non techniques correspondantes aux services (nom, description, etc.).
- 'bindingTemplate' définissant les points d'entrée du service (URL) et la façon d'accéder au service.
- tModel : définit les liens vers les informations techniques (ex : lien vers le fichier WSDL). Dans sa troisième version, UDDI permet de publier des propriétés non-fonctionnelles telles que les paramètres de sécurité ou de la QoS dans le *tModel.*

Annexe 12 : Schéma conceptuel de la base de données

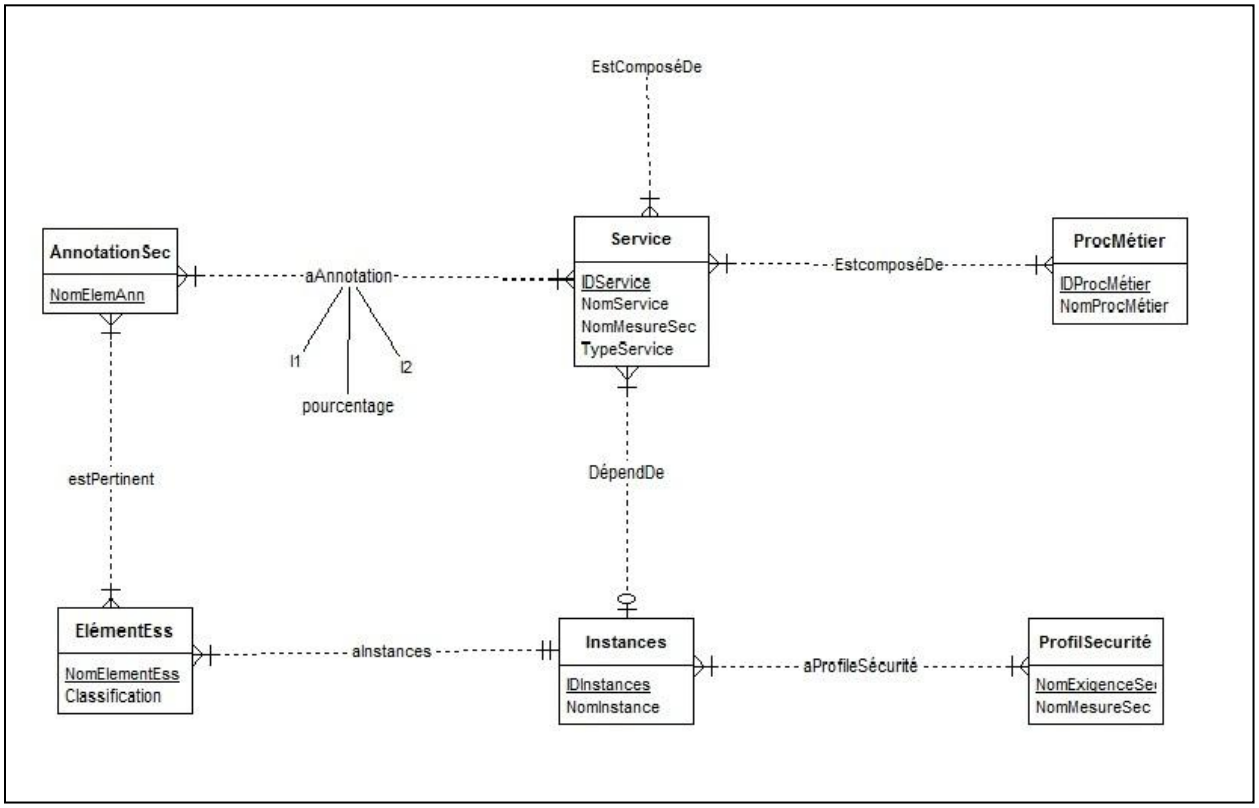


Figure A 9 : Schéma conceptuel de la base de données

La Figure A 9 représente le schéma conceptuel de la base de données de l'outil développé. Dans cette figure :

- ✓ L'entité 'ProcMétier' représente les processus métier. Un processus est identifié par un identifiant 'IDProcMétier' (clé primaire). De plus, il a un nom 'NomProcMétier'. Un processus se compose de services (association EstcomposéDe).
- ✓ L'entité 'Service' représente les services. Un service est identifié par un 'IDService' et il a un nom 'NomService' et un type 'TypeService' (atomique ou composite). La relation récursive m:n 'EstComposéDe' représente la composition de service (Un service composite se compose d'autres services composites ou atomiques). Un service lui sera associé plusieurs instances des éléments essentiels des plans service et infrastructure, ceci est indiqué par la relation 'DépendDe'.
- ✓ L'entité 'ElémentESS' représente les éléments essentiels. Chaque élément est identifié par un nom, et il possède un attribut 'Classification' qui indique le nom du plan auquel il appartient (métier, service ou infrastructure).
- ✓ L'entité 'Instances' représente les instances des éléments essentiels qui ont comme paramètres un identifiant, un nom. Cette entité est liée à 'ElémentESS par la relation 'aInstances' avec une cardinalité 1:1 du côté de l'entité 'Instances'.
- ✓ L'entité 'ProfileSécurité' stocke les éléments du profile de sécurité (mesures de sécurité et exigences de sécurité). Ces éléments du profile seront à évaluer pour déterminer le profile de sécurité de chaque instance des éléments essentiels. 'ProfileSécurité' est liée à 'Instances' avec une cardinalité m:n.
- ✓ L'entité 'AnnotationSec' représente les éléments de l'annotation de sécurité. Chaque élément est identifié par un nom 'NomElemAnn'. Cette entité est liée à 'ElémentESS' par la relation 'estPertinent' avec la cardinalité m:n pour dénoter la pertinence de l'élément par rapport aux exigences de sécurité. En outre, cette entité est liée à l'entité 'service' par la relation 'aAnnotation' qui inclut trois variables {I1, I2, pourcentage}. Ces dernières vont stockées les valeurs globales calculées pour chaque élément de l'annotation.