Des relations entre sûreté et sécurité

Ludovic Piètre-Cambacédès

Thèse

Des relations entre sûreté et sécurité

Présentée à
Télécom ParisTech

pour obtenir
le grade de docteur

École doctorale : ÉDITE de Paris
Spécialité : Informatique et Réseaux

par
Ludovic Piètre-Cambacédès

Soutenue publiquement le 3 novembre 2010

Jury

Pr Marc Bouissou	(École Centrale)	Examinateur
Dr Claude Chaudet	(Télécom ParisTech)	Examinateur
Dr Véronique Delebarre	(SafeRiver)	Examinateur
Pr Nouredine Hadjsaid	(INPG)	Rapporteur
Pr Michel Riguidel	(Télécom ParisTech)	Directeur de thèse
Dr Eric Totel	(Supélec)	Rapporteur
Pr Enrico Zio	(Politecnico di Milano)	Examinateur

Travaux effectués dans le groupe I2D du département Sinetics d'EDF R&D et au sein du département Informatique et Réseaux de Télécom ParisTech.

À Amélia, née en toute sûreté et en toute sécurité.

Remerciements

Singulière page que celle des remerciements. Première parcourue mais dernière écrite ; personnelle, voire intime, dans un mémoire qui se veut neutre et scientifique ; à la fois sans rapport formel avec les travaux présentés, et pourtant si profondément liée à leur déroulement. Singulière mais essentielle, car en effet, cette thèse n'aurait jamais vu le jour sans de nombreux et inestimables soutiens à qui ces quelques lignes vont tâcher de rendre hommage.

Ce travail de thèse a été effectué au sein du département Informatique et Réseaux de Télécom ParisTech, dans le cadre de mes activités d'ingénieur-chercheur à EDF R&D. Je commencerai ainsi par remercier mes responsables hiérarchiques successifs à EDF, qui m'ont permis de me lancer dans cette aventure et m'ont accordé leur confiance pour mener de front mes missions et ces recherches. Je pense en particulier à Vincent Gayrard, Eric Lorentz et Françoise Waeckel, auxquels se sont joints David Bateman et Olivier Morvant par la suite. Je remercie aussi mes collègues du groupe I2D (dont notamment l'équipe sécurité, *i.e.* les Pascaux, les Frédérics, Alia...) pour leur soutien et surtout leur patience vis-à-vis de ma disponibilité « fluctuante », en particulier durant la rédaction de ce mémoire. Je souhaite également exprimer toute ma gratitude à certains collègues « de l'autre côté » (*i.e.* la sûreté) pour leur intérêt et leur appui dans la réalisation de ces travaux. Parmi eux, je remercie tout particulièrement Gilles Deleuze pour avoir attiré mon attention sur la problématique des interactions entre sûreté et sécurité, et Marc Bouissou, pour m'avoir initié et épaulé quant à leurs aspects les plus théoriques. En fait, sans les qualités humaines et pédagogiques de ce dernier, sa maîtrise technique et sa disponibilité (combien d'heures sup' et de discussions *Skype* ?) cette thèse aurait certainement eu un autre visage... et une autre durée. Ce fut aussi à la fois un plaisir et un honneur de le compter parmi les membres de mon jury.

De plus, je souhaite exprimer toute ma reconnaissance à mon encadrement de Télécom ParisTech, c'est à dire Michel Riguidel, pour la confiance qu'il m'a accordée, et Claude Chaudet, pour son suivi attentif, ses conseils, ses encouragements et ses relectures constructives. C'est une grande chance d'avoir pu compter sur Claude durant ces trois années. À noter qu'elles m'ont aussi permis de côtoyer bon nombre d'autres membres du département Infres, aussi bien personnels administratifs, enseignants-chercheurs et doctorants, que je salue et remercie : ils ont su rendre mon séjour dans leur laboratoire, certes intermittent, agréable et enrichissant. J'adresse enfin une mention spéciale à l'équipe de la bibliothèque, souvent sollicitée par mon appétit bibliographique et toujours prompte à donner suite à ses demandes.

Par ailleurs, je tiens à exprimer la plus sincère gratitude à mon jury, et tout particulièrement à Éric Totel et à Nouredine Hadjsaid pour leur analyse minutieuse en tant que rapporteurs. Mes remerciements vont aussi aux examinateurs de ces travaux : si j'ai déjà pu les exprimer concernant Marc, Claude et Michel Riguidel, j'ajoute un chaleureux merci à Véronique Delebarre pour l'intérêt qu'elle a porté à mes recherches, et *mille grazzie* à Enrico Zio pour m'avoir fait l'honneur de présider mon jury avec un savant mélange de solennité, d'humour et d'à-propos.

Enfin, ces trois années passionnantes n'auraient sans doute pas eu le même goût sans l'équilibre et le soutien apportés par mes amis et ma famille (et sur un tout autre plan, les cookies *Pepperidge Farm*). Merci à Guillaume et Marco pour avoir, chacun à leur façon, suivi et partagé l'avancement de ces travaux ; merci aussi à Benoît pour m'avoir inspiré l'idée de mener cette thèse et m'avoir toujours encouragé depuis. Merci à ma belle-famille pour leur générosité et leur inestimable appui logistique dans les phases critiques de la rédaction de ce mémoire. Merci à mon oncle Jean pour ses interventions linguistiques (et ses proverbes africains) appréciées. Merci enfin à mes frères et à mes parents pour leur appui indéfectible dans ce projet.

Je réserve mes derniers et plus profonds remerciements à Aurélie, qui m'a « patiemment » accompagné durant la durée de cette thèse, tout en sachant m'emmener dans de nombreuses autres aventures, à même de relativiser grandement ces activités...

Table des matières

Introduction

Éléments de contexte

L'essor des technologies numériques et la dérégulation du domaine électrique entraînent une mutation profonde de ses infrastructures. Ces évolutions se répercutent directement sur les systèmes informatiques en charge du contrôle et de la supervision de ses composants industriels tels que centrales de production ou réseaux de distribution et de transport. Historiquement isolés et basés sur des technologies propriétaires, ces systèmes informatiques s'interconnectent, s'ouvrent progressivement vers le monde bureautique, voire Internet, et s'appuient de plus en plus sur des technologies standards, favorisant interopérabilité et réduction des coûts. À coté d'apports indéniables, ces changements entraînent cependant de nouveaux risques de malveillance informatique.

La prise en compte de ces nouveaux risques s'inscrit dans une problématique plus large, reconnue et désignée internationalement par « protection des infrastructures critiques » (*critical infrastructure protection*). La Commission européenne définit ces dernières comme « les installations physiques et de technologies de l'information, les réseaux, les services et les actifs qui, en cas d'arrêt ou de destruction, peuvent avoir de graves incidences sur la santé, la sécurité ou le bien-être économique des citoyens ou encore le travail des gouvernements des États membres » [1]. Elles correspondent dans le cadre national aux Secteurs d'Activité d'Importance Vitale (SAIV) [2, 3]. Leur protection contre la malveillance, et notamment contre le risque terroriste depuis les attentats du 11 septembre 2001, est devenue une priorité au niveau national et international [1, 4, 5]. En particulier, les attaques informatiques constituent une préoccupation grandissante, compte tenu de la forte dépendance des infrastructures critiques aux systèmes numériques [6]. Si le système électrique est au cœur de ces préoccupations, la numérisation du contrôle et de la supervision des installations industrielles, et l'ouverture et l'interconnexion des systèmes informatiques associés concernent aussi les autres secteurs industriels des SAIV (par exemple le secteur pétrolier, le transport aérien ou ferroviaire, l'industrie chimique, etc.).

Pour faire face aux nouveaux risques de malveillance informatique, des mesures et architectures de sécurité sont graduellement mises en place [7, 8, 9]. Elles soulèvent de nombreuses nouvelles problématiques liées aux spécificités de l'informatique typique de ces environnements. La première difficulté est en fait d'ordre culturel et humain : la relative nouveauté de la situation et le rythme considérable des évolutions technologiques nécessitent un investissement important des opérateurs de SAIV pour former et sensibiliser leur personnel à ces enjeux. En outre, leur conception historique fondée sur l'emploi de technologies conventionnelles ou de systèmes numériques isolés est un terreau fertile pour différentes croyances erronées entravant une posture de sécurité appropriée [1]. D'autres difficultés sont plus techniques. L'informatique que nous qualifions d'industrielle, c'est-à-dire directement impliquée dans le contrôle ou la supervision de processus industriels, possède des caractéristiques et contraintes propres. Elles la distinguent nettement de l'informatique habituellement rencontrée dans les applications sans rapport direct avec le contrôle ou la supervision d'un processus physique [11, 12] (que nous appellerons par la suite informatique générique). À titre d'exemples, la durée de vie moyenne des systèmes y est bien plus longue, certains automates étant installés pour plusieurs décennies ; leurs capacités de traitement et de communication sont plus contraintes, notamment pour les composants les plus proches du procédé industriel ; ces derniers reposent sur des technologies spécifiques voire propriétaires, même si les standards Internet gagnent du terrain pour les composants de plus haut niveau ; arrêts et mises à jour des ces matériels sont extrêmement délicats : disponibilité et intégrité des systèmes sont une priorité.

1. Indépendamment des travaux de cette thèse, nous avons identifié les principaux « mythes » en la matière dans [10], qui fournit de nombreux éléments de nature à rationaliser la situation.

Dans ce contexte, le passage d'une approche uniquement périmétrique, inadaptée à la diversification et l'intensification des échanges, vers une défense en profondeur, coordonnant plusieurs lignes de protection y compris au plus proche du procédé, est délicat. Technologies propriétaires, nouveaux standards, systèmes « hérités » (dits *legacy*) et « sur étagère » (ou COTS, pour *Commercial-Off-The-Shelf*) se côtoient dans des architectures en pleine mutation, soumises à de fortes contraintes de ressources, de durée de vie, de performance et d'ordre réglementaire. En particulier, dans le contexte des industries dites à risques (nucléaire, chimie, pétrole, transport aérien, etc.), les systèmes numériques contribuent toujours plus directement à la sûreté des installations. Ils sont en effet désormais non seulement incontournables pour leur pilotage et leur supervision, mais aussi impliqués dans les fonctions de sûreté prévenant les conditions accidentelles ou limitant leurs conséquences. Dans la dynamique précédemment évoquée, ces systèmes numériques de sûreté sont eux aussi progressivement interconnectés et s'exposent, malgré toutes les précautions prises dans ce contexte, à de nouveaux risques de sécurité informatique. La convergence d'enjeux de sécurité et de sûreté amène à considérer les relations entre ces deux aspects, et les disciplines associées, sous un nouveau jour.

Sûreté, sécurité et autres notions connexes

Cette thèse s'intéresse aux relations entre sûreté et sécurité, mais avant d'expliciter plus avant la problématique, arrêtons-nous tout d'abord sur les termes mêmes de « sûreté » et « sécurité ». Objets de définitions et d'emplois variant considérablement selon les contextes, il convient en effet de préciser en premier lieu leur signification. Dans ce mémoire, après l'analyse terminologique menée dans le Chapitre I, nous adopterons les conventions [2] suivantes :

- la **sûreté** correspond de façon générique à la démarche, ainsi qu'aux méthodes et dispositions associées, visant à limiter les risques de nature accidentelle, *i.e.* sans malveillance, étant susceptibles d'avoir des répercutions sur l'environnement du système considéré. Nous désignons ici par système l'objet de l'étude, de toute taille et nature (logiciel, organisation, installation industrielle, etc.), et par environnement l'ensemble des autres entités interagissant avec le système étudié et dont les caractéristiques et le comportement sont généralement moins connus et hors de contrôle ;
- la **sécurité** correspond quant à elle à la démarche, ainsi qu'aux méthodes et dispositions associées, visant à limiter les risques de nature malveillante, *i.e.* provenant ou étant exacerbés par une volonté de nuire, indépendamment de la nature de ses conséquences. Il peut donc s'agir de sécurité physique comme de sécurité informatique. Si nous nous intéresserons plus particulièrement à ce dernier aspect dans ce mémoire, nous n'écarterons pas pour autant la sécurité physique de nos considérations. La sécurité informatique vise à se prémunir des atteintes à l'**intégrité** (absence d'altérations indésirables), la **disponibilité** (fait d'être prêt à l'utilisation pour les sollicitations autorisées), ou la **confidentialité** (absence de divulgations indésirables) des données et des systèmes impliqués dans leur traitement informatique.

De plus, afin de mieux situer le sujet et les contributions de cette thèse, il convient aussi de préciser une notion fréquemment associée à la sûreté et à la sécurité, et qui fait également l'objet de différentes interprétations. Pour une partie de la communauté académique en informatique, sûreté et sécurité relèvent en effet de la sûreté de fonctionnement (*dependability*) [14, 15, 16]. Dans cette conception, elle est définie comme la propriété permettant aux utilisateurs d'un système de placer une confiance justifiée dans le service qu'il leur délivre. Une grande partie des concepts et de la taxonomie associés à cette approche fait référence et s'avère pertinente pour la sécurité comme pour la sûreté telles que nous les avons définies. Nous nous appuierons par exemple sur les termes communément admis de **fautes**, qui correspondent aux causes d'**erreurs**, états du système susceptibles d'entraîner des **défaillances**, déviations effectives dans le service attendu. Chacune de ces notions est finement catégorisée ; sûreté et sécurité s'y reflètent en outre dans la distinction entre fautes malveillantes et fautes non-malveillantes (que nous appellerons fautes accidentelles). Plus largement, cette approche intègre aussi la sécurité *via* les attributs de disponibilité et d'intégrité, auxquels s'ajoute la confidentialité. Cependant, à la différence de la sécurité, la sûreté y est

2. Compte tenu de la diversité des définitions existantes, celles proposées ici ne peuvent en effet être considérées que comme des conventions ; elles ne sont pas pour autant arbitraires car cohérentes avec l'usage dans plusieurs domaines, dont notamment l'électronucléaire [13].

considérée comme un attribut de la sûreté de fonctionnement au même titre que la fiabilité, la disponibilité ou l'intégrité. Cette conception ne fait pas l'unanimité et nous semble sujette à caution dans le cadre d'une analyse des relations entre sûreté et sécurité. La Commission électrotechnique internationale (CEI) définit pour sa part la **sûreté de fonctionnement** comme « l'ensemble des propriétés qui décrivent la disponibilité et les facteurs qui la conditionnent : fiabilité, maintenabilité et logistique de maintenance » [17]. Nous préférons nous appuyer sur cette décomposition, qui a l'avantage de rester neutre vis-à-vis des concepts de sûreté et de sécurité. En outre, la sûreté est d'une nature différente, intrinsèquement systémique ; elle s'appuie en partie sur les autres attributs et n'émerge seulement qu'à une échelle globale [18, 19], au même titre que la sécurité. Cependant, si le concept de sûreté de fonctionnement n'est pas suffisant pour caractériser les relations entre sûreté et sécurité, ces notions gardent des relations étroites avec celui-ci. En effet, la sûreté d'un système dépend de la sûreté de fonctionnement de fonctions (et des entités les supportant) en charge d'éviter les comportements dangereux ou de limiter leurs conséquences. De même, la sécurité d'un système dépend de la sûreté de fonctionnement de fonctions de sécurité et des entités assurant ce type de fonctions.

Problématique et objectifs de la thèse

La thèse développée dans ce mémoire repose sur la conviction suivante : sûreté et sécurité doivent être décloisonnées.

Dans les acceptions proposées précédemment, sûreté et sécurité correspondent à des problématiques longtemps considérées comme indépendantes. Les disciplines associées ont historiquement évolué de façon séparée, concernant alors des systèmes distincts [20, 21]. Elles partagent en fait de substantiels points communs, qui progressivement identifiés, ont déjà permis diverses inspirations réciproques, aussi bien d'un point de vue méthodologique que technologique ou architectural. Ainsi, la technique des arbres de défaillances, formalisme graphique largement employé dans les analyses de sûreté, a directement inspiré celle des arbres d'attaque pour l'analyse de risques en sécurité [22]. Autre exemple, l'approche dite de défense en profondeur, amenant à combiner plusieurs types de contre-mesures complémentaires et indépendantes, est aujourd'hui devenue un principe fondamental en sécurité informatique [23] ; inventée dans le domaine militaire, elle a d'abord été réellement formalisée et institutionnalisée par l'industrie du nucléaire à des fins de sûreté, où elle guide encore conception et exploitation des installations. Cependant, malgré certains pas franchis, le rapprochement des disciplines et des communautés est encore très limité. Le paysage normatif en fournit une illustration éloquente : sûreté et sécurité y font, sauf très rares exceptions, l'objet de comités d'étude et de documents complètement séparés. Globalement, le potentiel d'inspiration réciproque, bien qu'avéré, reste encore trop peu exploité.

Par ailleurs, la convergence des exigences et des mesures de sécurité et de sûreté sur les mêmes systèmes, impliquée par l'évolution des menaces et des technologies, est un phénomène récent et encore mal maîtrisé. Il mène à des situations allant potentiellement du renforcement mutuel jusqu'à l'antagonisme frontal [24, 25]. Un exemple simple [26] permet d'illustrer ce dernier cas de figure : un système de fermeture de porte automatisée pourra ainsi être conçu pour laisser la porte ouverte en cas de panne selon des exigences de sûreté (comportement dit *fail-safe*, pour permettre une évacuation en cas de danger) ; il sera conçu pour verrouiller les portes dans le même cas en suivant des exigences de sécurité. Le même genre de conflit se retrouve dans les architectures d'informatique industrielle, où par exemple, la mise en œuvre de mesures d'authentification et de mécanismes cryptographiques impliqués par la sécurité informatique peut s'avérer incompatible avec des contraintes de temps de réponse des systèmes ou des opérateurs, dictées par des considérations de sûreté. Ainsi, tout en cautionnant le rapprochement des communautés et l'intérêt des travaux de fertilisation croisée entre ces deux domaines, il devient également impératif de mieux caractériser et modéliser leurs interdépendances, quand sûreté et sécurité s'appliquent sur les mêmes systèmes. Cette thématique encore peu explorée fait depuis quelques années seulement l'objet d'approches structurées. Elles sont encore très macroscopiques [26, 27, 28, 29, 30], ou au contraire restreintes à l'analyse de systèmes simples ou de situations spécifiques [31, 32, 33, 34]. De nouveaux outils et méthodes sont nécessaires pour appréhender de façon satisfaisante les interdépendances entre sûreté et sécurité : la maîtrise des risques pesant sur les installations industrielles, mais aussi l'optimisation des ressources en conception et en exploitation qui y sont consacrées en dépendent.

Dans ce contexte, nos travaux ont poursuivi trois objectifs :

- pour commencer, le décloisonnement appelé de nos vœux passe nécessairement par une meilleure compréhension réciproque entre les communautés sûreté et sécurité, entravée par la signification et l'emploi changeants de ces termes. Aussi, il nous a paru essentiel de mieux circonscrire sûreté et sécurité sur le plan terminologique, puis d'apporter un éclairage sur leurs grandes différences et similarités ;
- par ailleurs, bien que déjà identifié et exploité à différents égards, le potentiel de synergie entre ces domaines reste significatif du point de vue méthodologique. Nous avons voulu nous inscrire dans cette dynamique et contribuer aux activités de fertilisation croisée entre disciplines, moyen et bénéfice direct du décloisonnement visé ;
- enfin, nous avons tenté de contribuer à une meilleure caractérisation des interdépendances entre sûreté et sécurité, en vue d'améliorer la maîtrise des conséquences de la convergence croissante des deux problématiques sur les mêmes systèmes.

Organisation du mémoire

Dans le Chapitre I, nous nous penchons sur la signification changeante des termes sécurité et sûreté. Nous proposons un nouveau cadre référentiel dénommé SEMA (Système-Environnement/Malveillant-Accidentel) visant à réduire l'ambiguïté de leur emploi et faciliter le dialogue entre communautés hétérogènes. Trois cas d'application sont analysés : le domaine des réseaux électriques, la production électronucléaire et le secteur des télécommunications et réseaux de données. SEMA rend explicite les différences de signification des mots sûreté et sécurité pour chaque domaine et met en exergue les recouvrements et incohérences de certaines définitions.

Le Chapitre II s'éloigne des considérations terminologiques pour chercher à circonscrire sur le fond sûreté et sécurité. Il identifie et commente leurs différences et similarités les plus structurantes, puis brosse un panorama de leurs outils respectifs d'évaluation. Les adaptations fructueuses d'un domaine à l'autre y sont enfin inventoriées, confirmant à la fois le bien-fondé et le potentiel conséquent de telles démarches.

Le Chapitre III s'inscrit dans cette logique en exposant comment nous avons adapté les BDMP (*Boolean logic Driven Markov Processes*), un formalisme graphique employé dans les études de sûreté, au domaine de la sécurité. Le fruit de cette adaptation est une technique de modélisation d'attaque offrant un compromis original et avantageux entre lisibilité, pouvoir de modélisation et capacités de quantification. Elle permet de capturer des aspects dynamiques tels que séquences, détections et réactions, sous une forme proche des arbres d'attaque qui ne peuvent pour leur part les prendre en compte. Le traitement des modèles BDMP, une fois paramétrés dans le cadre d'une analyse de risques, fournit des informations qualitatives et quantitatives pour hiérarchiser menaces et contre-mesures.

Le Chapitre IV met à profit cette adaptation pour caractériser les situations dans lesquelles considérations de sûreté et de sécurité doivent être prises en compte conjointement. L'intérêt des BDMP pour modéliser la diversité des interdépendances entre sûreté et sécurité est souligné à travers un cas d'étude simple. La place de cette approche est précisée vis-à-vis de l'état de l'art : complémentaire à l'existant, plusieurs perspectives de développement intrinsèque mais également d'intégration avec d'autres techniques sont décrites.

Finalement, la Conclusion Générale rappelle les principales contributions développées dans le mémoire, fait le bilan des recherches effectuées vis-à-vis de la problématique et des objectifs de la thèse et présente les perspectives ouvertes par ces travaux.

Chapitre I

SEMA, un référentiel pour différencier les termes sûreté et sécurité

« Quand j'emploie un mot, dit Humpty Dumpty avec un certain mépris,
il signifie ce que je veux qu'il signifie, ni plus ni moins.
– La question est de savoir, dit Alice, si vous pouvez faire que
les mêmes mots signifient tant de choses différentes.
– La question est de savoir, dit Humpty Dumpty, qui est le maître, c'est tout. »
(Lewis Caroll, *De l'autre côté du miroir*, 1871.)

Sûreté et sécurité ont des significations très variables selon le contexte. Le seul secteur de l'industrie électrique permet d'illustrer une telle situation : ainsi, pour l'ingénieur du réseau de transport électrique, le terme sécurité renverra généralement à la capacité du réseau à tolérer des perturbations type court-circuit ou perte d'un composant sans rupture de service aux usagers [35], alors que pour celui du domaine nucléaire, il s'agira principalement de protection contre la malveillance [13]. Le terme sûreté aura aussi des significations différentes. Ainsi, au sein d'un même secteur industriel, la multiplicité des acteurs et des disciplines d'ingénierie impliquées entraîne diverses acceptions et utilisations des termes sûreté et sécurité. Évidemment, la situation ne s'améliore pas quand des parties prenantes issues d'industries distinctes échangent sur ces deux notions. Bien que centrales pour toutes les industries à risques, la signification des mots sûreté et sécurité varie donc considérablement selon leur contexte, alimentant équivoques et incompréhensions.

Dans ce chapitre, nous proposons un nouvel outil conceptuel dénommé SEMA (Système-Environnement / Malveillant-Accidentel) [36, 37] qui offre un cadre de référence positionnant les termes sécurité et sûreté l'un par rapport à l'autre selon leur contexte d'utilisation. Il vise à rendre explicite les différences de signification souvent cachées derrière l'utilisation de ces mots en apparence anodins, mais en fait profondément polysémiques. De plus, les éventuelles incohérences ou recouvrements des définitions existantes peuvent y être soulignés.

Après avoir caractérisé la confusion terminologique de la situation, la Section 1 se concentre sur l'identification de différences discriminant l'emploi de mots sécurité et sûreté. Elle s'appuie sur une analyse bibliographique approfondie. La Section 2 développe cette piste en présentant la contribution principale de ce chapitre, le cadre de référence SEMA. La Section 3 développe différents cas d'utilisation permettant de mieux comprendre l'intérêt et le fonctionnement de SEMA. L'usage des mots sûreté et sécurité y est également précisé et fixé par convention pour le reste de ce mémoire. Enfin, la Section 4 évoque différentes perspectives d'amélioration.

1 Différents usages et distinctions entre sûreté et sécurité

1.1 Une situation très confuse

Étymologiquement, sûreté et sécurité partagent une même racine latine, *securitas*, et l'examen des définitions proposées par les principaux dictionnaires, à commencer par le Larousse et le dictionnaire de l'Académie française, ne permet pas de distinguer clairement ces deux notions. Le Littré va même jusqu'à proposer le terme « sécurité » comme une de ses définitions de sûreté. Hélas, la littérature scientifique et le *corpus* normatif n'éclaircissent pas la situation : ils offrent une surprenante diversité de définitions. En fait, il est possible d'en distinguer plusieurs dizaines [38, 21], variant par de simples nuances jusqu'à de totales inversions selon les sources considérées. La Section 3 analyse en détail plusieurs exemples.

En plus de cette diversité, des pièges linguistiques aggravent la situation. En effet, comme notamment remarqué dans [39, 21], certaines langues ne possèdent qu'un seul mot pour sûreté et sécurité. Si l'on se limite à l'échelle européenne, c'est par exemple le cas de l'espagnol (*seguridad*) ou du portugais (*segurança*), mais aussi du suédois (*säkerhet*) ou du danois (*sikkerhed*). *A contrario*, d'autres langues distinguent comme en français deux mots ; c'est notamment le cas de l'anglais (*safety* et *security*). Par ailleurs, même entre langues utilisant deux mots distincts, le passage de l'une à l'autre reste problématique. D'une part, la frontière entre les deux termes n'est pas toujours la même selon les langues, d'autre part, les traductions d'une langue à l'autre ne sont pas systématiques. On peut ainsi aboutir à des traductions inversées selon les secteurs : le terme anglais *safety* est par exemple directement traduit en français par « sûreté » dans le nucléaire [13], alors que l'Organisation internationale de normalisation (ISO) traduit *safety* par « sécurité » dans de nombreux autres secteurs [40]. Le même constat s'applique au terme *security*, traduit alternativement par « sûreté » ou « sécurité » selon le contexte. Afin de contourner ces pièges, les chercheurs du LAAS [1] préfèrent même éviter le terme sûreté et emploient « sécurité-innocuité » pour désigner l'absence de conséquences catastrophiques pour l'environnement (en traduction de *safety*) et « sécurité-immunité » pour désigner l'association à la confidentialité, la préservation de l'intégrité et de la disponibilité vis-à-vis des actions autorisées (en traduction de *security*) [41, 42].

> NB : Par convention, nous emploierons dans ce chapitre les mots sécurité et sûreté comme traductions systématiques des termes anglais *security* et *safety* tels qu'utilisés dans les références examinées, sans considération pour leurs significations. Cette approche a pour but d'améliorer dans notre cadre cohérence et lisibilité. Une fois les analyses menées, l'emploi des termes sûreté et sécurité en français est arrêté en fin de chapitre pour le reste du mémoire.

1.2 Raisonner par distinction

Dans une telle situation, il semble vain de chercher des définitions génériques et consensuelles ; il s'avère en fait plus constructif de changer de perspective et de nous intéresser aux distinctions entre les deux concepts.

1.2.1 Sources utilisées

Dans cette démarche, nous nous sommes appuyés sur l'analyse :

- de précédents panoramas et états de l'art établis par la communauté académique sur les notions de sûreté et de sécurité. Au final, huit références ont été retenues pour leur pertinence vis-à-vis de la problématique de discrimination entre sûreté et sécurité. Elles sont présentées dans la Table I.1 ;

- de normes génériques et sectorielles, réglementations et guides de mise en œuvre utilisés par l'industrie sur les problématiques de sécurité et de sûreté. La Table I.2 regroupe les documents considérés dans cette étude par secteur et par thématique ; la Figure I.1 illustre leur répartition sectorielle. Les documents ont été sélectionnés vis-à-vis de leur visibilité dans leur secteur d'activité, sur la base de l'expérience de l'auteur pour ceux relevant de l'industrie électrique (nucléaire et réseaux électriques),

1. Laboratoire d'Analyse et d'Architecture des Systèmes (LAAS, http://www.laas.fr) du CNRS (Centre National de la Recherche Scientifique).

TABLE I.1 – Références académiques examinées

Date	Auteurs	Réf.	Pertinence
1986	N. Leveson	[18]	État de l'art sur la sûreté logicielle. En Section 4 de l'article, discussion des différences entre fiabilité, sûreté et sécurité.
1992	A. Burns, J. McDermid et J. Dobson	[39]	Article constatant l'ambiguïté entre sûreté et sécurité. Proposition d'une distinction entre les deux concepts.
1994	J. Rushby	[43]	État de l'art exposant les approches historiquement séparées de la sûreté de fonctionnement (*dependability*), de la sûreté et de la sécurité.
2003	D. G. Firesmith	[44]	Rapport décrivant une taxonomie formalisée en UML des concepts communs à la sécurité, la sûreté et à la survivabilité. Indique une distinction claire entre sûreté et sécurité.
2004	A. Avizienis, J.-C. Laprie, B. Randell et C. Landwehr	[16]	Article complétant et mettant à jour une taxonomie promue par J.C. Laprie dès les années 80. Centrée sur la notion de sûreté de fonctionnement (*dependability*), sûreté et sécurité y sont intégrées dans un même cadre (cf. Intro. Générale).
2004	D. M. Nicol, W. Sanders et K. S. Trivedi	[45]	État de l'art sur les techniques d'évaluation de *dependability* et de sécurité à base de modèles. Définitions implicites.
2006	M. B. Line, O. Nordland, L. Røstad, et I. A. Tøndel	[21]	Papier de conférence comparant sûreté et sécurité sous divers angles. Proposition d'une distinction claire et explicite.
2009	M. Al-Kuwaiti, N. Kyriakopoulos et S. Hussein	[46]	Article comparant les concepts de sûreté de fonctionnement, tolérance de fautes, fiabilité, sécurité et survivabilité. Une taxonomie est proposée pour chaque concept.

de la sécurité de l'informatique industrielle et de l'informatique générique ; par recherche bibliographique pour les autres. Ont été prises en compte notamment de nombreuses références issues d'organismes de normalisation tels que la Commission électrotechnique internationale (CEI ou IEC, pour *International Electrotechnical Committee*), l'Organisation internationale de normalisation (ISO), ou sur le plan national, pour les États-Unis par exemple, le NIST (*National Institute of Standards and Technology*) ou l'ANSI (*American National Standards Institute*). Les agences des Nations-unies, telles que l'AIEA (Agence internationale de l'énergie atomique, IAEA en anglais) ou l'ICAO (*International Civil Aviation Organization*) produisent également une documentation pertinente, considérée dans l'analyse. Par ailleurs, certaines associations industrielles sectorielles ont une influence significative et sont à l'origine de nombreux documents de référence et standards de fait : c'est le cas au niveau international avec, par exemple dans l'aéronautique, l'Association internationale du transport aérien (ou IATA, pour *International Air Transport Association*), la RTCA (*Radio Technical Commission for Aeronautics*) ou son équivalent européen EuroCAE (*European Organization for Civil Aviation Equipment*) ; au niveau national, on peut citer pour le secteur pétrolier l'OLF (*Oljeindustriens Landsforening*) en Norvège ou l'API (*American Petroleum Institute*) aux États-Unis. De plus, agences gouvernementales et ministères émettent également des normes, recommandations et guides de mise en œuvre dans le domaine de la sécurité et de la sûreté : c'est le cas notamment aux États-Unis du département de l'administration fédérale dédié à la sécurité du territoire, le DHS (*Department of Homeland Security*), mais également du département de la défense (DoD, *Department of Defense*) ou de la NRC (*Nuclear Regulatory Commission*) ayant autorité dans le domaine nucléaire. Enfin, certains textes à caractère législatif ou exécutif ont également été considérés : *e.g.* directives présidentielles ou extraits du code fédéral (notés CFR pour *Code of Federal Regulations*) aux États-Unis, réglementations de la Commission européenne (notées EC).

D'une manière générale, la représentativité a été privilégiée sur l'exhaustivité, aussi, certains secteurs n'ont pas été pris en compte (*e.g.* automobile, eau, médical), d'autres l'ont été très partiellement (*e.g.* militaire, ferroviaire). Enfin, certains documents relevant de la sécurité n'ont pu être consultés faute d'habilitation appropriée, comme par exemple dans le domaine de l'aviation civile, le *IATA Security Manual* [47] ou encore celui de l'ICAO [48].

Au total, 89 documents ont été retenus et analysés. Dans tous les cas, nous nous sommes intéressés à la sûreté et à la sécurité des systèmes et installations physiques et informatiques, sans distinction.

TABLE I.2 – Documents analysés (non académiques)

Secteur	Références sécurité (*security*)	Références sûreté (*safety*)
Électronucléaire	**Internationales :** Manuel de référence AIEA (version préliminaire) [49] IEC 62645 (version préliminaire) [52] **Nationales :** (États-Unis) Réglementations fédérales 10 CFR 73 [55] (États-Unis) Guide NRC RG1.152 [57] (États-Unis) Guide NRC RG5.71 [60] (États-Unis) Norme IEEE 692-2010 [63] (Corée) Guide KINS/GT-N09-DR [64]	**Internationales :** Série sûreté AIEA (SF-1 [50], NS-G-1.1 [51], NS-G-1.3 [53], NS-R-1 [54]) Glossaire AIEA [13] Rapport AIEA 75-insag-3 [56] Normes IEC du SC45A (61513 [58], 61226 [59], 60880 [61], 62138 [62]) **Nationales :** (États-Unis) Guide NRC RG1.152 [57] Normes ANSI/IEEE 603-1998 [65] et 7-4.3.2 [66]
Réseaux électriques	**Internationales :** IEC 62351 [67] **Régionales :** (Amérique du Nord) Standards NERC CIP [69] (Europe) Manuel UCTE [35] **Nationales :** (États-Unis) NIST IR 7628 (version préliminaire) [70] (États-Unis) IEEE1402-2000 [71] (États-Unis) IEEE1686-2007 [72] (États-Unis) IEEE1711(version préliminaire) [73]	**Régionales :** (Amérique du Nord) NERC *Reliability Standards* [68]
Aéronautique / Aviation civile	**Régionales :** (Europe) Réglementation (EC) 2320/2002 [74] **Nationales :** (États-Unis) NSPD 47/HSPD 16 [77]	**Internationales :** ICAO Doc 9735 [75] RTCA DO-178B / EuroCAE ED12 B [76] **Régionales :** (Europe) EuroControl ESARRs [78, 79] (Europe) Réglementation (EC) 216/2008 [80]
Ferroviaire	**Nationales :** (États-Unis) 49 CFR parties 1520 et 1580 [81]	**Internationales :** IEC 62278 [82] IEC 62279 [83]
Spatial	**Nationales :** (États-Unis) 14 CFR parties 1203, 1203a, 1203b [84, 85, 86] (États-Unis) NASA EA-STD 0001.0 [88] (États-Unis) NASA NPR 1600.1 [89]	**Régionales :** (Europe) ECSS-P-001B [87] **Nationales :** (États-Unis) NASA-STD-8719.13B [90]
Pétrole et gaz	**Nationales :** (Norvège) OLF Guide 104 [91] (États-Unis) API 1164 [93]	**Internationales :** ISO 10418 [92] ISO 13702 [94] ISO 17776 [95] **Nationales :** (Norvège) NORSOK S-001 [96] et I-002 [97] (Norvège) OLF Guide 70 [98] et 90 [99]
Chimie	**Nationales :** (États-Unis) 6 CFR Partie 27 [100] (États-Unis) DHS CFATS (incl. RBPSG) [102]	**Nationales :** (États-Unis) Glossaire AIChE/CCPS [101]
Militaire	**Internationales :** Glossaire OTAN AAP-6(2009) [103]	**Internationales :** Glossaire OTAN AAP-6(2009) [103] ARMP-7 ed.1 [104] **Nationales :** (États-Unis) DoD MIL-STD-882D [105] (Royaume-Uni) MoD DEF Stan 00-56 [106, 107]
Informatique industrielle (non sectorielle)	**Internationales :** Série de normes IEC 62443 [108, 109] **Nationales :** (États-Unis) NIST SP 800-82 [11] (États-Unis) NIST SP 800-53 (annexe I) [111] (États-Unis) ANSI/ISA99 00.01 [112] (Royaume-Uni) CPNI SCADA GPG [113]	**Internationales :** IEC 61508 [110]
Informatique Générique (non sectorielle)	**Internationales :** ISO/IEC 27000, 27001 [114, 115] ISO/IEC 27002, 27005 [117, 118] ISO/IEC 13335-1 [119] IETF RFC 4949 [120] IUT-T X.1051 [121] **Nationales :** (États-Unis) NIST FIPS 199 [122] NIST IR 7298 [123] Glossaire IAG de la NSA [124]	**Internationales :** IEC 60950 [116]
Générique	**Internationales :** ISO/IEC Guide 81 (version préliminaire) [125]	**Internationales :** ISO/IEC Guide 51 [40] et Guide 2 [126] IEC 60050-191 [17]

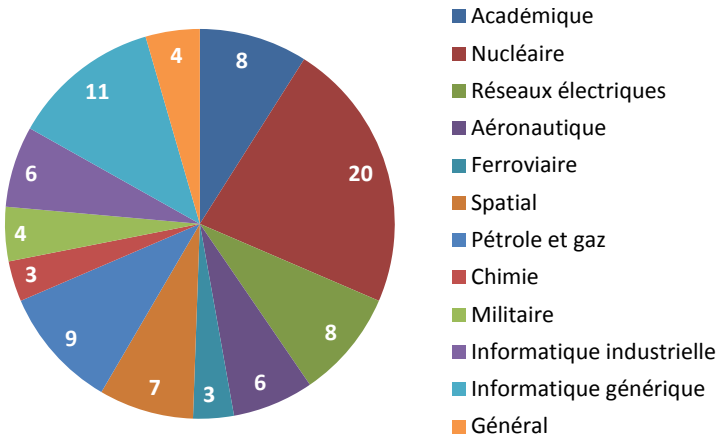


FIGURE I.1 – Répartition sectorielle des documents analysés

1.2.2 Analyse générique

Seulement 14 références comportent des définitions du mot sûreté et du mot sécurité (nous incluons les définitions des notions « contextualisées » au secteur de la référence considérée : *e.g. information security*, *nuclear security*, etc.). La Figure I.2 page suivante indique la répartition de ces références par secteur industriel considéré. Parmi ces 14 références :

- seules trois proposent des définitions des termes sûreté et sécurité permettant de différencier ces notions de façon claire et exclusive. L'article de Line *et al.* (2006) [21] et le rapport de Firesmith (2003) [44] le font de façon explicite. L'article de Burns *et al.* [39] le fait de façon plus indirecte, en indiquant comme définitions discriminantes de sécurité et sûreté celles en fait de *safety critical system* et de *security critical system*. La Table I.3 présente ces trois propositions particulièrement pertinentes pour notre étude ;
- les 11 autres documents donnent des définitions de sûreté et sécurité clairement non-disjointes. Certaines références le reconnaissent explicitement [13], la grande majorité ne le souligne pas. Dans certains cas, le recouvrement des définitions peut être particulièrement important :
  - pour au moins trois des 11 documents, la définition donnée de la sûreté peut inclure la sécurité [108, 112, 101] (pour les deux premiers, « *Safety : freedom from unacceptable risk* »),
  - pour deux d'entre eux, la définition de la sécurité peut inclure la sûreté [46, 112].

TABLE I.3 – Les seules définitions discriminantes de sécurité et sûreté dans le *corpus* étudié

Réf.	***Safety***	***Security***
Line *et al.* (2006) [21]	*The inability of the system to affect its environment in an undesirable way.* (Incapacité du système à nuire à son environnement.)	*The inability of the environment to affect the system in an undesirable way.* (Incapacité de l'environnement à nuire au système.)
Firesmith (2003) [44]	*The degree to which accidental harm is prevented, reduced, and properly reacted to.* (Le degré de prévention, de réduction et de réaction approprié vis-à-vis d'un dommage accidentel.)	*The degree to which malicious harm is prevented, reduced, and properly reacted to.* (Le degré de prévention, de réduction et de réaction approprié vis-à-vis d'un dommage malveillant.)
Burns *et al.* (1992) [39]	*A system is judged to be safety-critical in a given context if its failure could be sufficient to cause absolute harm.* (Un système est jugé critique pour la sûreté dans un contexte donné si sa défaillance est suffisante pour causer un dommage absolu.)	*A system is judged to be security-critical in a given context if its failure could be sufficient to cause relative harm, but never sufficient to cause absolute harm.* (Un système est jugé critique pour la sécurité dans un contexte donné si sa défaillance est suffisante pour causer un dommage relatif, mais ne l'est jamais pour causer un dommage absolu.)

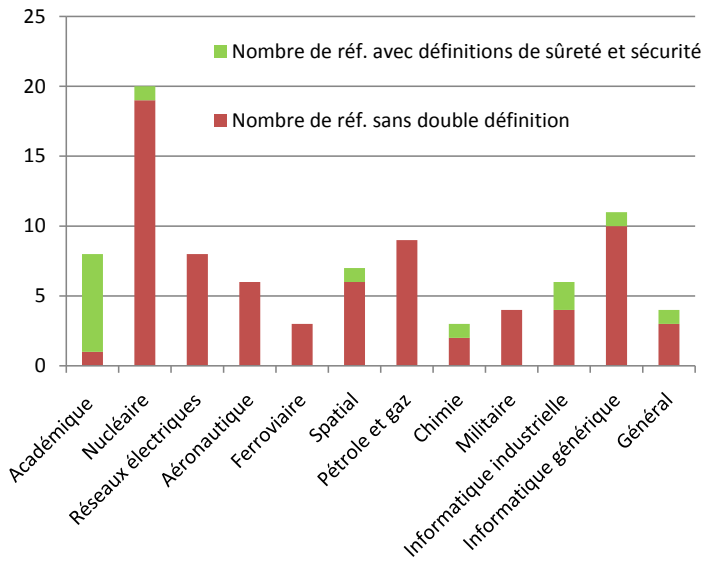


FIGURE I.2 – Répartition des références avec définitions des termes sûreté et sécurité par secteur

Plus globalement sur les 89 documents étudiés, 12 références donnent une ou des définitions des mots sûreté et/ou sécurité avec des recouvrements particulièrement importants (on y retrouve les références mentionnées dans le point précédent) :

- [40, 82, 101, 96, 105, 108, 112, 126] définissent la sûreté dans des termes si larges qu'ils permettent une interprétation pouvant inclure la plupart des définitions de la sécurité ;
- [127, 112, 118, 119] définissent la sécurité dans des termes qui permettent une interprétation incluant la sûreté.

Enfin, 40 de ces documents ne donnent pas directement de définitions explicites de leur propre thématique dominante (*i.e.* sûreté ou sécurité). Ceci-dit :

- d'une part, cette carence est parfois due à un renvoi vers un autre document, de type glossaire ou document de plus haut niveau dans la hiérarchie normative, qui donne la définition manquante (*e.g.* série des documents sûreté AIEA) ;
- d'autre part, de nombreux termes connexes (tels que risque, menace, dommage, etc.) sont souvent définis, et permettent de circonscrire implicitement la signification de sûreté ou sécurité.

1.2.3 Éléments d'analyse lexicale

En complément de l'analyse macroscopique des définitions de sûreté et sécurité relevées dans le *corpus* examiné, nous avons procédé à un examen lexical de leur contenu. L'objectif était de faire émerger ou souligner les dominantes thématiques associées à chaque terme, et caractériser certaines différences rendues éventuellement plus visibles par une analyse lexicale automatisée. Une telle analyse nous amène aux constats suivants :

- les définitions du mot sûreté emploient au total 211 mots différents, celles de sécurité environ le double (411 mots). Il y a plusieurs interprétations possibles ; les deux suivantes, non exclusives, nous paraissent les plus pertinentes :
  - le terme sûreté bénéficie d'un plus grand consensus (comme souligné aussi dans [21]),
  - la définition du terme sûreté est souvent plus générale, et ne nécessite donc pas de vocabulaire spécifique au secteur considéré ;

- la Table I.4 donne le classement par nombre d'occurrences des mots employés dans les définitions de sûreté et sécurité issues du *corpus* étudié [2]. Pour des raisons de pertinence d'analyse, nous ne faisons apparaître que les mots ayant au moins trois occurrences pour les définitions de sûreté, et quatre pour les définitions de sécurité. Nous pouvons alors constater que :
  - les définitions de la notion de sûreté privilégient le registre lexical de l'accident et des dégâts physiques (utilisations répétées de *damage*, *harm*, *injury*, *catastrophic*, *equipment*). On remarque aussi l'appel fréquent à la notion d'environnement, absente des définitions de sécurité. Cette absence est à mettre en relation avec l'utilisation du mot *system*, faite de façon quasiment égale au singulier et au pluriel dans les définitions de *security*, alors que seul le singulier est employé dans les définitions de *safety*. *System* peut alors être mis directement en opposition à la notion d'*environment*,
  - les définitions du terme sécurité s'appuient quant à elles largement sur le registre lexical de la malveillance et de l'action volontaire (cf. utilisation des termes *unauthorized*, *access*, *against*, *sabotage*, *achieving*, *actions*, *malicious*...), avec des spécificités principalement dues au domaine de la sécurité informatique (*e.g.* répétition des mots *confidentiality*, *integrity*, *availability*).

Ainsi, et de façon remarquable, l'analyse lexicale corrobore bien les approches décrites dans les trois seules références identifiées permettant de distinguer explicitement sûreté et sécurité [21, 44, 39] (cf. Section 1.2.2 et Table I.3), soulignant à la fois leur pertinence et leur légitimité, mais également la difficulté de privilégier l'une par rapport aux autres.

1.2.4 Synthèse et proposition

Nous avons entrepris l'analyse précédemment décrite dans le but d'identifier les limites, mouvantes, des notions de sûreté et de sécurité, et mieux caractériser ce qui les distingue. Sur la base des références considérées et des éléments d'analyse discutés, nous proposons de retenir deux distinctions alternatives, en droite ligne avec les définitions données respectivement par Line *et al.* et par Firesmith (nous ne retenons pas celles de Burns *et al.*, qui ne portent pas explicitement sur les termes sûreté et sécurité, et reposent sur des critères de distinction plus subjectifs). Toutes deux distinguent sécurité et sûreté selon les caractéristiques du risque, la première en termes d'origine et de champ de réalisation des conséquences, la seconde en termes de cause, ou plus précisément d'intention. Elles se définissent comme suit :

- la première, que nous appellerons S-E, s'appuie sur une distinction Système vs Environnement. Nous désignons par Système l'objet de l'étude, de toute taille et nature (logiciel, organisation, installation industrielle, etc.), et par Environnement l'ensemble des autres systèmes interagissant avec le système étudié et dont les caractéristiques et le comportement sont généralement moins connus et hors de contrôle. Dans S-E, la sécurité concerne les risques provenant de l'environnement, et dont les conséquences potentielles concernent le système considéré ; la sûreté concerne réciproquement les risques provenant du système, et dont les conséquences potentielles concernent l'environnement ;

- la seconde, que nous appellerons M-A, s'appuie sur une distinction Malveillant vs Accidentel. Par Accidentel, nous entendons associé à des événements arrivant de façon inattendue et sans intention de nuire. Dans cette approche, les risques de nature malveillante relèvent de la sécurité, alors que les risques accidentels relèvent de la sûreté.

Ces distinctions correspondent à des abstractions, et ne sont pas rigoureusement mises en œuvre dans le *corpus* analysé. D'une part, 75 références sur les 89 analysées ne peuvent par nature y être conformes, ne définissant pas conjointement sûreté et sécurité. Quant aux 14 documents les définissant d'autre part, bien peu proposent des définitions disjointes et clairement discriminantes, comme indiqué en Section 1.2.2. Toutefois, la grande majorité des définitions de sûreté et de sécurité considérées individuellement peuvent être associées à une de ces deux distinctions, selon leur prise en compte de l'origine du risque, du champ de réalisation de ses conséquences potentielles, et de son intentionnalité.

2. Ces études textuelles ont été effectuées grâce au logiciel libre (GNU) *TextStat 3.0*.

TABLE I.4 – Classement des mots les plus utilisés dans les définitions de sûreté et de sécurité

Définitions sûreté	
Mot	**Occurrences**
system	17
risk	15
damage	14
environment	13
freedom	11
harm	11
unacceptable	9
property	7
injury	5
acceptable	4
level	4
catastrophic	3
cause	3
conditions	3
consequences	3
equipment	3
illness	3
operating	3

Définitions sécurité	
Mot	**Occurrences**
information	25
system	25
systems	24
unauthorized	19
access	15
availability	13
integrity	13
confidentiality	12
persons	11
against	9
measures	9
protect	9
data	8
condition	7
control	7
reliability	7
accountability	6
authenticity	6
critical	6
disclosure	6
loss	6
protection	6
sabotage	6
achieving	5
actions	5
aspects	5
cyber	5
defining	5
denied	5
destruction	5
harm	5
maintaining	5
modify	5
provide	5
repudiation	5
software	5
acts	4
authorised	4
cause	4
ensure	4
interference	4
malicious	4
safety	4
unwanted	4

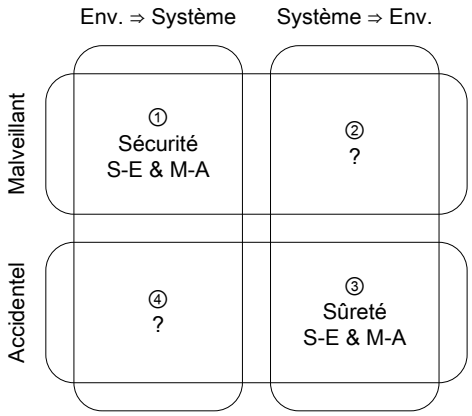


FIGURE I.3 – Représentation graphique du croisement de S-E et M-A

1.2.5 Croisement des distinctions S-E et M-A

Une fois les distinctions S-E et M-A établies, il est alors possible d'analyser les conséquences de leur cohabitation. Elle est en effet représentative d'environnements multidisciplinaires et multiculturels, typiques des grands systèmes industriels où diverses appréhensions des termes sûreté et sécurité coexistent. La Figure I.3 représente graphiquement la superposition des deux distinctions. Elles ne sont heureusement pas complètement incompatibles : il est en effet possible de définir des sous-domaines associés de façon consensuelle à la sécurité ou à la sûreté. Ils correspondent aux quadrants numérotés 1 et 3. Ainsi, un risque de nature malveillante émanant de l'environnement et pesant sur le système est généralement associé à la sécurité, indistinctement du contexte ou de la communauté considérée. De même, un risque accidentel provenant du système et menaçant l'environnement sera très largement vu comme relevant de la sûreté. Hélas, les deux autres quadrants de la Figure I.3 ne peuvent pas être liés sans ambiguïté à la notion de sûreté ou de sécurité : ce qui semblera relever de la sécurité pour certains relèvera de la sûreté pour d'autres, et réciproquement.

En somme, la Figure I.3 illustre le clair potentiel d'incompréhension et d'équivoque lorsque S-E et M-A sont utilisées en même temps. Mais elle suggère également qu'il est possible de décomposer les notions génériques de sûreté et de sécurité en sous-notions épousant le contour des distinctions S-E et M-A, qui donnent le nom à l'approche présentée dans la suite de ce chapitre : SEMA.

2 Le cadre référentiel SEMA

2.1 Description

Sur la base des analyses de la Section 1, nous avons conçu un cadre référentiel dénommé SEMA, qui permet de prendre en compte les distinctions S-E et M-A de façon intégrée. SEMA se veut être un outil neutre permettant de faciliter la communication et la compréhension mutuelle quand les notions de sécurité et sûreté sont en jeu. Il nomme explicitement les sous-notions délimitées par les quadrants de la Figure I.3, augmentés par une dimension capturant les impacts « Système sur Système » complétant l'analyse (par définition, les aspects « Environnement sur Environnement » sont par contre exclus). Ce cadre est représenté dans la Figure I.4. Il découpe sûreté et sécurité en six sous-notions distinctes, désignées en anglais par *Defense*, *Safeguards*, *Self-Protection*, *Robustness*, *Containment Ability* et *Reliability*. Notons que bien que potentiellement traduisible en français, nous préférons conserver la terminologie anglaise qui seule a pu bénéficier de retours et d'analyses extérieures par le biais des publications et débats associés à ce référentiel [128, 36, 37].

Les termes choisis pour les sous-notions se veulent sémantiquement moins ambigus que les termes génériques *safety* et *security*. La Table I.5 résume et complète la description de SEMA.

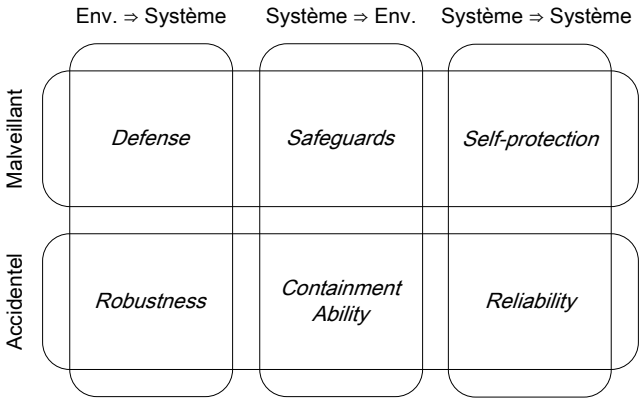


FIGURE I.4 – Le cadre de référence SEMA

TABLE I.5 – Les six sous-notions de SEMA

Sous-notion SEMA	**Risque couvert**			**Remarques**
	M-A	**S-E**		
	Intention	**Origine**	**Cible**	
Defense	Malveillant	Env.	Système	Terminologie militaire et générale
Safeguards	Malveillant	Système	Env.	Terme adapté du domaine industriel
Self-Protection	Malveillant	Système	Système	Protection contre les menaces internes
Robustness	Accidentel	Env.	Système	Employé différemment dans des travaux récents [44] mais considéré comme explicite
Containment Ability	Accidentel	Système	Env.	Terminologie générale
Reliability	Accidentel	Système	Système	Cohérent avec les standards internationaux

2.2 Utilisation, pertinence et limites de SEMA

L'objectif de SEMA et de ses sous-notions n'est évidemment pas de remplacer les termes sûreté et sécurité. Ils sont bien trop historiquement et culturellement implantés, associés aux multiples significations précédemment discutées. SEMA vise à faciliter l'établissement d'une compréhension mutuelle lorsque des communautés disparates échangent sur ces notions, chacune influencée dans leur compréhension des termes sûreté et sécurité par leurs références et schémas de pensée. SEMA constitue alors un outil pratique pour « dessiner » explicitement les contours de leur signification, sur la base des six zones de la Figure I.4. La Section 3 illustre son utilisation sur différents cas d'application. Une telle approche est particulièrement profitable dans les phases amont des analyses de risques, notamment pour définir leur périmètre, mais également à un niveau plus macroscopique dans l'établissement de collaborations ou d'assignation de tâches dans de grands projets multi-équipes et multidisciplinaires. De plus, SEMA permet de rappeler la diversité des facettes du risque d'un point de vue holistique, considérant conjointement la sécurité et la sûreté.

Ceci-dit, il convient de souligner plusieurs limites, ou tout du moins précautions d'usage, concernant SEMA. Tout d'abord, les contours relatifs des sous-notions définies par SEMA ne peuvent pas être considérés comme fixes et hermétiques. En particulier, la frontière entre système et environnement est mouvante et par essence dépendante de la perspective d'analyse adoptée, or elle est cruciale dans la définition et la sélection des sous-notions. L'emploi de SEMA suppose ainsi une délimitation du système claire et explicite. De plus, les sous-notions ne s'excluent pas mutuellement dans le sens où un événement redouté donné ou une contre-mesure peuvent dans certains cas correspondre simultanément à plusieurs sous-notions. Ceci relève largement de l'interprétation de l'analyste. En particulier, des considérations liées à la sous-notion de *Self-Containment* (sûreté dans le secteur nucléaire) impliquent par exemple de fortes contraintes relevant de la *Reliability* sur les systèmes en charge de la protection de l'environnement et/ou de l'installation. Enfin, SEMA ne peut résoudre les problèmes intrinsèques aux différentes définitions existantes des termes sûreté et sécurité, notamment en termes de recouvrements. Comme illustré dans la section suivante, SEMA permet par contre de les identifier et de les caractériser.

3 Exemples et cas d'application

3.1 Les réseaux électriques

Les réseaux de transport et de distribution d'électricité sont de vastes et complexes systèmes techniques, en pleine mutation, et associés à diverses problématiques de sûreté et de sécurité [129, 130]. Dans un tel contexte, les acteurs impliqués ont des parcours et des compétences variés, faisant des réseaux électriques un bon exemple de secteur propice aux pièges terminologiques associés aux termes sûreté et sécurité. En fait, s'agissant du terme sûreté, celui-ci est de façon assez consensuelle dans l'industrie électrique lié à la prévention de dégâts matériels ou des atteintes aux personnes physiques d'origine accidentelle et provoqués par le système [131, 132].

Le terme sécurité est par contre nettement plus ambigu. Du point de vue du génie électrique, la sécurité du réseau correspond à sa capacité à tolérer des perturbations, comme par exemple un court-circuit ou la perte inattendue d'un composant, sans interruption de service aux usagers [35, 133, 134]. La nature des causes de ces perturbations n'est traditionnellement pas considérée et la signification généralement suggérée est représentée grâce à l'utilisation de SEMA en Figure I.5 : le caractère malveillant n'est pas explicitement exclu mais n'est que marginalement traité spécifiquement ; les impacts sur l'environnement sont hors périmètre, plutôt traités comme relevant de la sûreté (en supposant les intervenants humains sur le réseau comme extérieurs au système).

Toutefois, les inquiétudes croissantes vis-à-vis de la vulnérabilité des infrastructures critiques, exacerbées suites aux attentats du 11 septembre 2001, ont conduit à mobiliser des moyens considérables dans une optique de protection contre les risques de type malveillant, en particulier le risque terroriste. Cette mobilisation, désignée internationalement par le terme *Critical Infrastructure Protection* (CIP), a été impulsée par des engagements politiques marqués, structurant encore largement ce domaine aujourd'hui (cf. par exemple [5] pour les États-Unis ou [1] en Europe). Les réseaux électriques y occupent une place fondamentale [135, 136]. Dans cette mouvance, le terme sécurité est clairement associé à une signification différente de celle discutée pour le génie électrique traditionnel, et nettement délimitée par la distinction M-A du référentiel SEMA comme représenté dans la Figure I.5.

Enfin, la place grandissante des technologies numériques soulignée dans l'Introduction Générale devient particulièrement critique dans le domaine des réseaux électriques : l'engouement mondial pour les initiatives de type *Smart Grid* [137, 138, 139] et les déploiements de compteurs intelligents [140] en sont des manifestations directes. Une telle évolution va de pair, comme déjà signalé, avec l'émergence de nouveaux risques malveillants [129]. Pour y faire face, les États-Unis ont ainsi défini un cadre réglementaire contraignant pour protéger leur système électrique des attaques informatiques (cf. notion de *cybersecurity* dans les standards NERC-CIP [69]). Ce contexte implique une autre façon d'appréhender le mot sécurité, également axée sur la composante malveillante de SEMA. Nous la représentons et la comparons explicitement aux autres acceptions des termes sûreté et sécurité du secteur grâce à SEMA dans la Figure I.5. À noter que la couverture partielle de la sous-notion *Self-protection* est liée à la prise en compte variable des menaces d'origine interne (*insider threat*).

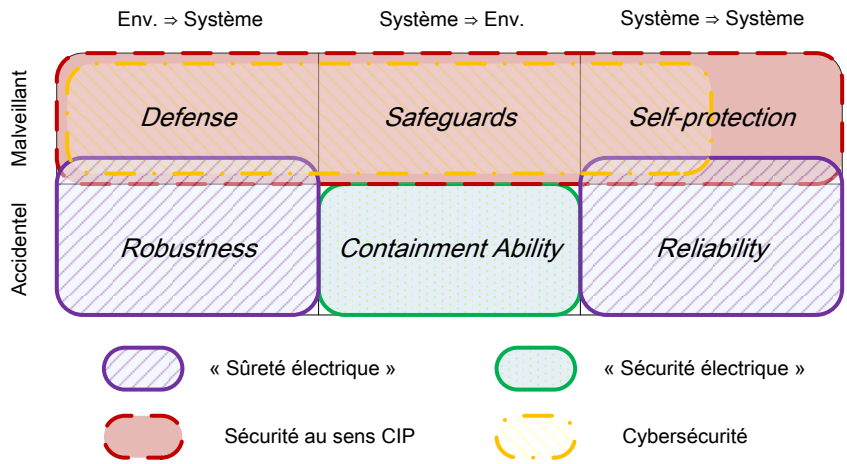


FIGURE I.5 – Sécurité et sûreté dans le domaine des réseaux électriques

3.2 La production électronucléaire

Dans l'industrie électronucléaire, sur le plan international, les notions de sûreté et de sécurité sont généralement utilisées dans l'acception de l'Agence internationale de l'énergie atomique (AIEA) [13] :

- sécurité (nucléaire) : « Mesures visant à empêcher et à détecter un vol, un sabotage, un accès non autorisé, un transfert illégal ou d'autres actes malveillants mettant en jeu des matières nucléaires et autres matières radioactives ou les installations associées, et à intervenir en pareil cas. » ;
- sûreté (nucléaire) : « Obtention de conditions d'exploitation correctes, prévention des accidents ou atténuation de leurs conséquences, avec pour résultat la protection des travailleurs, du public et de l'environnement contre des risques radiologiques indus. ».

Ces définitions peuvent être directement et simplement mises en correspondance avec le référentiel SEMA, comme l'illustre la Figure I.6. La sécurité au sens de l'AIEA recouvre les sous notions de *Defense*, *Safeguards* et *Self-Protection* alors que la sûreté correspond dans ce contexte principalement à la notion de *Containment-Ability* (en supposant les travailleurs comme étant externes au système technique considéré). Les risques liés à la sous-notion *Reliability* de SEMA et qui ne présentent pas d'enjeux pour l'environnement correspondent à des problématiques de disponibilité ou de performance : ils ne sont pas considérés comme relevant de la sûreté. Les aspects *Robustness* ont leurs propres études dédiées, distinctes des études générales de sûreté et de sécurité.

Néanmoins, malgré ces correspondances relativement nettes, équivoques et incompréhensions réciproques restent possibles : certaines utilisations des termes sûreté et sécurité au sein même du secteur nucléaire ne suivent pas ce schéma. C'est ainsi le cas de la loi française dite TSN (Transparence et Sécurité en matière Nucléaire) du 13 juin 2006 [141], qui emploie comme souligné dans [142] le mot sécurité de façon plus large que ne le définit l'AIEA. En effet, le terme sécurité de la loi TSN recouvre tout à la fois la sûreté nucléaire au sens de l'AIEA, la radioprotection mais également la prévention et la lutte contre les actes de malveillance ainsi que les aspects de sécurité civile en cas d'accident [142]. Ce périmètre est représenté en pointillé sur la Figure I.6. Une fois projetées sur les six notions de SEMA, les différences entre les significations des termes sûreté et sécurité tels qu'utilisés dans le domaine nucléaire sont rendues explicites.

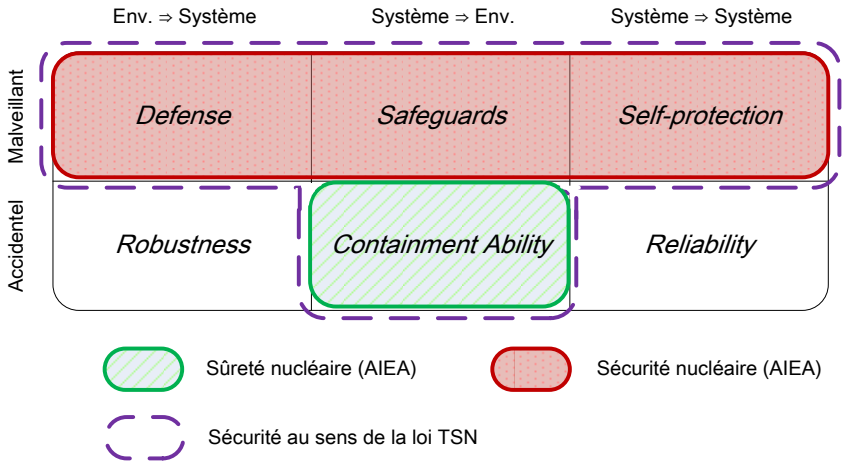


FIGURE I.6 – Utilisation de SEMA pour la production électronucléaire

Finalement, comme pour les réseaux électriques dans la section précédente et plus généralement pour les infrastructures critiques, les risques associés à la malveillance informatique sur les systèmes numériques des installations nucléaires font l'objet d'une attention grandissante des états et de la communauté internationale. Sur ce dernier plan, l'AIEA et plus récemment la CEI travaillent ainsi sur l'élaboration de référentiels internationaux pour la sécurité informatique des installations nucléaires [49, 52] ; aux États-Unis, de nombreuses initiatives et documents structurent déjà le domaine (*e.g.* [55, 143]). Certaines de ces références utilisent le terme sécurité pour la protection des systèmes numériques de façon différente : SEMA pourrait là aussi rendre ces différences explicites et lever certaines ambiguïtés.

3.3 Les télécommunications et réseaux

Les réseaux de télécommunications et de données tiennent, comme les réseaux électriques, une place particulière parmi les infrastructures critiques : ils sont à la fois une infrastructure critique en tant que telle, mais sont aussi constitutifs de toutes les autres infrastructures critiques. En effet, ces dernières dépendent désormais toutes très intimement des capacités de communication de systèmes numériques interconnectés. La protection de telles capacités est parfois désignée par le sigle CIIP (*Critical Information Infrastructure Protection*) [144]. L'utilisation des termes sûreté et sécurité dans ces différents contextes est à l'image de l'utilisation des réseaux de données dans les infrastructures critiques : diverse et omniprésente. Concernant Internet, l'*Internet Engineering Task Force* (IETF), reconnue comme un des principaux architectes et une des organisations techniques de référence en la matière, a publié un glossaire de la sécurité Internet [120]. Dans celui-ci, on trouve les définitions suivantes [3] :

- *security : a system condition in which system resources are free from unauthorized access and from unauthorized or accidental change, destruction, or loss* ;
- *safety : the property of a system being free from risk of causing harm (especially physical harm) to its system entities.*

La distinction M-A n'est pertinente pour aucune des deux définitions. La sûreté est vue comme une problématique de type « système-à-système » alors que la sécurité est définie de façon beaucoup plus large. De plus, une distinction que SEMA ne traite pas spécifiquement est en jeu : le caractère physique des conséquences est en effet évoqué dans la définition de la sûreté, alors que le terme *resources* de la définition de sécurité laisse place à l'interprétation quant à leur nature.

Ouvrant le périmètre au-delà d'Internet, il convient aussi de s'intéresser à la série de normes conjointement éditées par l'ISO et l'IEC sur la sécurité des technologies de l'information (série 27000, dite 27k). Cette série couvre aussi bien risques accidentels que risques malveillants : en outre, la norme ISO/IEC 27005 [118] spécifie que les menaces considérées peuvent être d'origine naturelle ou humaine, et peuvent être accidentelles ou délibérées (« *threats may be of natural or human origin, and could be accidental or deliberate* »). En fait, le domaine couvert est encore plus large, la même référence indiquant qu'une menace peut provenir de l'intérieur ou l'extérieur de l'organisation (« *a threat may arise from within or from outside the organization* »). La notion de sûreté n'est pas couverte dans cette série, ce qui pourrait expliquer l'étendue donnée au domaine de la sécurité.

Hélas, ni les définitions de l'IETF ni celles des normes 27k ne sont en ligne avec celles utilisées dans la majorité des différents secteurs du domaine CIIP, sur la base des références de la Table I.2. C'est le cas de la production électronucléaire et des réseaux électriques, pour lesquels on se reportera aux Sections 3.1 et 3.2, mais également de bien d'autres secteurs, comme celui de la chimie, du pétrole ou de l'aéronautique, où l'utilisation du terme sécurité est généralement restreinte à la malveillance (cf. par exemple les documents du DHS aux États-Unis [145, 146]). La préexistence et l'importance des problématiques de sûreté dans ces industries, historiquement bien établies et structurées dans un *corpus* normatif conséquent, pourraient expliquer cette délimitation plus contrainte du concept de sécurité. Toutefois, une telle préexistence pourrait aussi avoir contribué au fait qu'à l'inverse des définitions IETF ou de la série 27k, c'est la notion de sûreté qui est cette fois définie de façon extrêmement large. Ainsi, l'ISO et l'IEC ont harmonisé pour les standards de plusieurs secteurs industriels leur définition de sûreté en tant qu'absence de risque inacceptable (« *freedom from unacceptable risk* »[40]), alors qu'une des références les plus citées de la littérature académique dans le domaine des systèmes critiques [16] définit la sûreté comme « absence de conséquences catastrophiques sur le(s) utilisateur(s) et l'environnement ».

Globalement, face à la largeur des différentes définitions précédemment considérées, SEMA ne permet pas de tracer des limites claires entre les concepts, mais fournit un moyen efficace de souligner les problèmes de recouvrement et d'incohérence, illustrés par le manque de lisibilité de la Figure I.7 [4]. Dans une telle situation, il peut être utile de repartir du découpage proposé par SEMA pour entamer un dialogue sans équivoques, évitant les pièges inhérents aux contours manifestement trop larges des définitions existantes dans le domaine de télécommunications et réseaux.

3. Nous donnons les définitions d'origine en anglais, elles peuvent être traduites approximativement comme suit : « Sécurité : une condition du système dans laquelle ses ressources ne sont pas accédées de façon non autorisée, et ne sont pas changées, détruites ou perdues de façon non-autorisée ou accidentelle » ; « Sûreté : la propriété d'un système exempt de risque d'endommagement (en particulier physique) d'une de ses entités ».

4. Nous avons conservé dans cette figure les termes anglais *safety* et *security* pour ne pas ajouter à la situation déjà confuse les difficultés liées aux conventions de traduction de l'ISO et de l'IEC évoquées en Section 1.1.

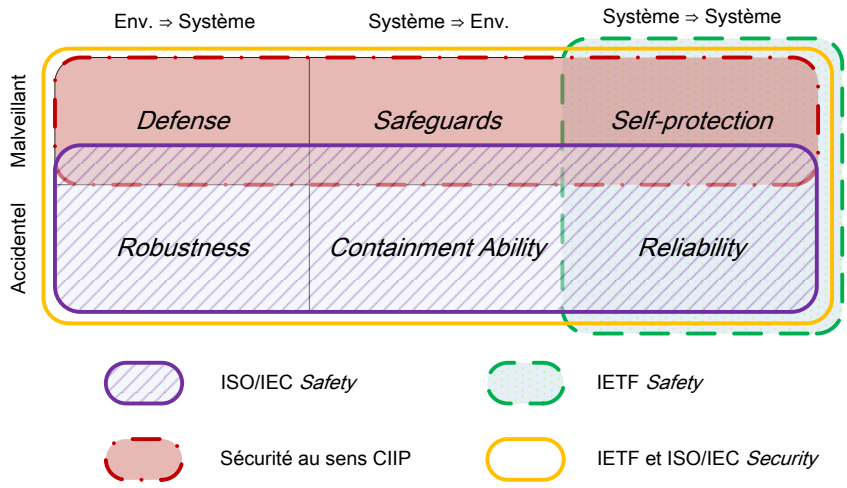


FIGURE I.7 – Utilisation de SEMA pour le domaine des réseaux et télécommunications

3.4 Utilisation des termes sûreté et sécurité dans ce mémoire

Dans ce mémoire, nous emploierons le terme sûreté pour ce qui relève du risque accidentel avec des conséquences sur l'environnement, tandis que le terme sécurité sera employé pour ce qui relève de la malveillance, indépendamment de la dimension S-E. La Figure I.8 illustre notre emploi de ces termes dans le référentiel SEMA. Notons que les pannes internes au système sans effets sur l'environnement (sous-notion désignée par *Reliability*), et les impacts de l'environnement sur le système (*Robustness*), quand eux-mêmes sans conséquences pour l'environnement, sont considérés comme hors périmètre, *i.e.* ne relevant ni de la sécurité, ni de la sûreté.

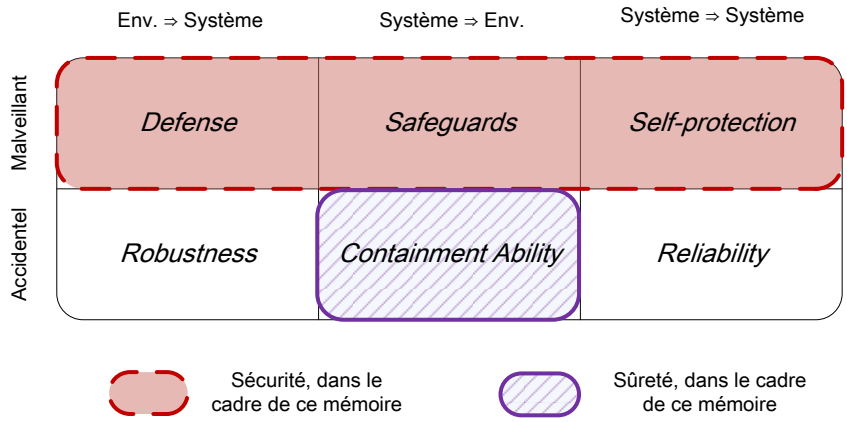


FIGURE I.8 – Définition des termes sûreté et sécurité dans ce mémoire.

4 Perspectives

L'emploi et la promotion de SEMA ont permis d'identifier plusieurs pistes d'amélioration. La plus prometteuse consiste à distinguer la dimension physique de la dimension informatique, impliquées par les problématiques de sûreté et de sécurité. Ceci permettrait une décomposition plus fine, et en particulier, une meilleure prise en compte des spécificités de la sécurité informatique. Dans une même optique, il serait intéressant de distinguer confidentialité, intégrité et disponibilité, ainsi que d'autres propriétés associées aux problématiques de sécurité et de sûreté. Enfin, au-delà des termes sûreté et sécurité, SEMA pourrait être mis à profit pour analyser et rapprocher les taxonomies en cours dans les différentes communautés du risque, qui manipulent nombre de concepts communs sous des dénominations différentes.

5 Conclusion

Dans ce chapitre, nous avons d'abord montré en quoi les termes sûreté et sécurité étaient éminemment ambigus, leurs significations changeant selon leur contexte d'utilisation. Cette situation est source d'équivoques et d'incompréhensions. Sur la base d'une analyse bibliographique approfondie, nous avons construit un outil nommé SEMA, permettant de distinguer et de rendre explicites les différences de signification souvent cachées derrière l'utilisation de ces termes. Nous l'avons mis à profit pour comparer l'emploi des termes sûreté et sécurité dans différents secteurs : les réseaux électriques, l'électronucléaire et le secteur des télécommunications et réseaux. Les deux premiers cas nous ont permis d'illustrer les différences de compréhension des termes sûreté et sécurité dans un même secteur (cf. les Fig. I.5 et I.6 individuellement), mais aussi d'illustrer ces différences entre deux secteurs appartenant pourtant à une même industrie (cf. les Fig. I.5 et I.6 ensemble, décrivant deux secteurs de l'industrie électrique). Dans le troisième cas, SEMA souligne plus particulièrement l'insuffisance des définitions existantes, qui laissent une large place à l'ambiguïté et aux recouvrements. Enfin, nous avons utilisé SEMA pour définir les termes sûreté et sécurité dans le cadre de ce mémoire. Ainsi, malgré les limites et les perspectives d'amélioration identifiées, ce référentiel constitue déjà dans sa forme actuelle une aide efficace pour limiter les ambiguïtés associées à l'emploi des termes sûreté et sécurité.

Chapitre II

Similitudes, différences et inspirations réciproques entre sûreté et sécurité

Seigneur, nous n'avons pas si grande ressemblance,
Qu'il faille de bons yeux pour y voir différence.
(Pierre Corneille, *Nicomède* (1651), Acte IV, Scène 3.)

Il se trouvait entre leurs caractères toute la ressemblance,
et de plus toute la différence qui peuvent servir à former une grande liaison.
(Bernard le Bouyer de Fontenelle, *Éloge des académiciens* (1715), Éloge de Malezieu.)

Dans le chapitre précédent, nous avons souligné l'ambiguïté des termes sécurité et sûreté, avant de proposer un cadre référentiel permettant d'expliciter leur signification selon le contexte. Leur signification a été finalement précisée dans le cadre de ce mémoire. Si l'emploi de ces mots est si confus, c'est en partie car les concepts sous-jacents ont de nombreux points communs. Dans ce chapitre, nous présentons en Section 1 les similitudes entre sûreté et sécurité d'un point de vue général, mais aussi leurs différences fondamentales, afin de mieux circonscrire les deux concepts. Poursuivant ce même objectif, nous changeons d'angle en Section 2, pour nous intéresser plus particulièrement aux techniques d'évaluation, et donnons pour chaque domaine un aperçu des outils et approches les plus répandus. Les différences et similitudes précédemment discutées s'y reflètent implicitement. Enfin, la Section 3 brosse un panorama des techniques et approches ayant la particularité d'être les fruits d'adaptations les ayant fait passer de la sûreté vers la sécurité ou inversement. L'inventaire de telles fertilisations croisées témoigne d'une part du décloisonnement progressif des communautés sûreté et sécurité pour le plus grand bénéfice de tous, et permet de mieux situer d'autre part les contributions développées dans les chapitres suivants.

1 Similitudes et différences d'ordre général

1.1 Similitudes

1.1.1 Le risque comme notion fondamentale

Un premier point commun entre sécurité et sûreté tient dans le fait que leur évaluation et leur gestion s'appuient largement sur la notion de risque. Le risque se définit dans les deux cas de façon macroscopique par une même équation : risque = probabilité [1] × conséquences. Le guide 73 de l'ISO sur le vocabulaire du risque [147] donne ainsi la définition suivante : « combinaison de la probabilité[1] d'un événement et de ses conséquences », cohérente avec les guides de la même organisation, notamment sur l'utilisation du vocabulaire en sûreté [40]. La Table II.1 regroupe d'autres exemples de définitions de la notion de risque, validant une conception commune entre communauté sûreté et communauté sécurité.

Table II.1 – Exemples de définitions de la notion de risque

	Secteur et source	Définition du risque
Sûreté	Nucléaire (Glossaire AIEA [13])	*A multiattribute quantity expressing hazard, danger or chance of harmful or injurious consequences associated with actual or potential exposures. It relates to quantities such as the probability that specific deleterious consequences may arise and the magnitude and character of such consequences.*
	Aviation civile (FAA *Safety Handbook* [148])	*Risk is an expression of possible loss over a specific period of time or number of operational cycles. It may be indicated by the probability of an accident times the damage in dollars, lives, and/or operating units. Hazard probability and severity are measurable and, when combined, give us risk.*
	Chimie (Glossaire CCPS [101])	*Measure of human injury, environmental damage, or economic loss in terms of the incident likelihood and the magnitude of the loss or injury.*
Sécurité	Pétrolier (OLF-104 [91])	*The combination of the probability of an event and its consequence.*
	Internet (IETF RFC 4949 [120])	*An expectation of loss expressed as the probability that a particular threat will exploit a particular vulnerability with a particular harmful result.*
	Informatique générique (NIST SP800-53 [111], NIST FIPS200 [149])	*The level of impact on agency operations (including mission, functions, image, or reputation), agency assets, or individuals, resulting from the operation of an information system given the potential impact of a threat and the likelihood of that threat occurring.*

La démarche d'analyse de risques, en conception d'un nouveau système ou portant sur des systèmes existants, tente de répondre aux mêmes questions aussi bien en sûreté qu'en sécurité. Kaplan et Garrick les ont résumées dans le « triplet » suivant [150] : « Que peut-il se passer ? Quelles en sont les chances ? Quelles en sont les conséquences ? » [2]. L'analyse de risques se structure dans les deux domaines autour de phases similaires : analyse des menaces (ou dangers, selon le contexte), des vulnérabilités (ou faiblesses), identification des conséquences, évaluation des probabilités d'occurrence, hiérarchisation des risques. On peut élargir les similarités à la gestion de risques en général, qui inclut en plus de l'analyse de risques elle-même, l'évaluation du risque en termes de criticité des conséquences pour l'organisation, et les décisions de traitement du risque qui en découle. Quatre options sont alors traditionnellement distinguées, à savoir l'évitement (*risk avoidance)*, la réduction (*risk reduction)*, l'acceptation (*risk acceptance)* ou le transfert (*risk transfer)*, et ce d'un point de vue sûreté comme d'un point de vue sécurité. La Figure II.1 positionne les étapes précédemment discutées les unes par rapport aux autres dans une représentation cohérente avec le guide d'utilisation du vocabulaire du risque de l'ISO/IEC [147] et la norme ISO/IEC 27005 sur la gestion des risques informatiques [118].

Par contre, si la démarche globale d'analyse de risques est similaire, la nature du risque considéré diffère bien sûr selon que l'on adopte un point de vue sécurité ou sûreté. Aussi, les méthodologies, les outils et les formalismes sont adaptés à la nature du risque examiné. La Section 2 développe plus en avant ces considérations.

Il convient enfin d'ajouter deux remarques. D'une part, certaines démarches émergentes de gestion de risques privilégient une approche élargie, considérant risques de sûreté et de sécurité de façon intégrée. Le Chapitre IV présente les motivations, l'intérêt et l'état de l'art de ce type de méthodes. D'autre part, dans certaines industries à risques, notamment dans le nucléaire, les fondements de la gestion du risque diffèrent selon le contexte, en particulier selon le pays ou le type de risques considéré [151]. Ainsi, en France, l'Autorité de sûreté nucléaire privilégie les approches déterministes, où les systèmes sont conçus et exploités

1. Le terme « probabilité » doit être ici compris au sens le plus général, et pas nécessairement au sens mathématique. Le terme *likelihood* est employé dans la définition en anglais mais n'a pas de traduction directe en français.
2. *What can go wrong ? What is the likelihood ? What are the consequences ?*

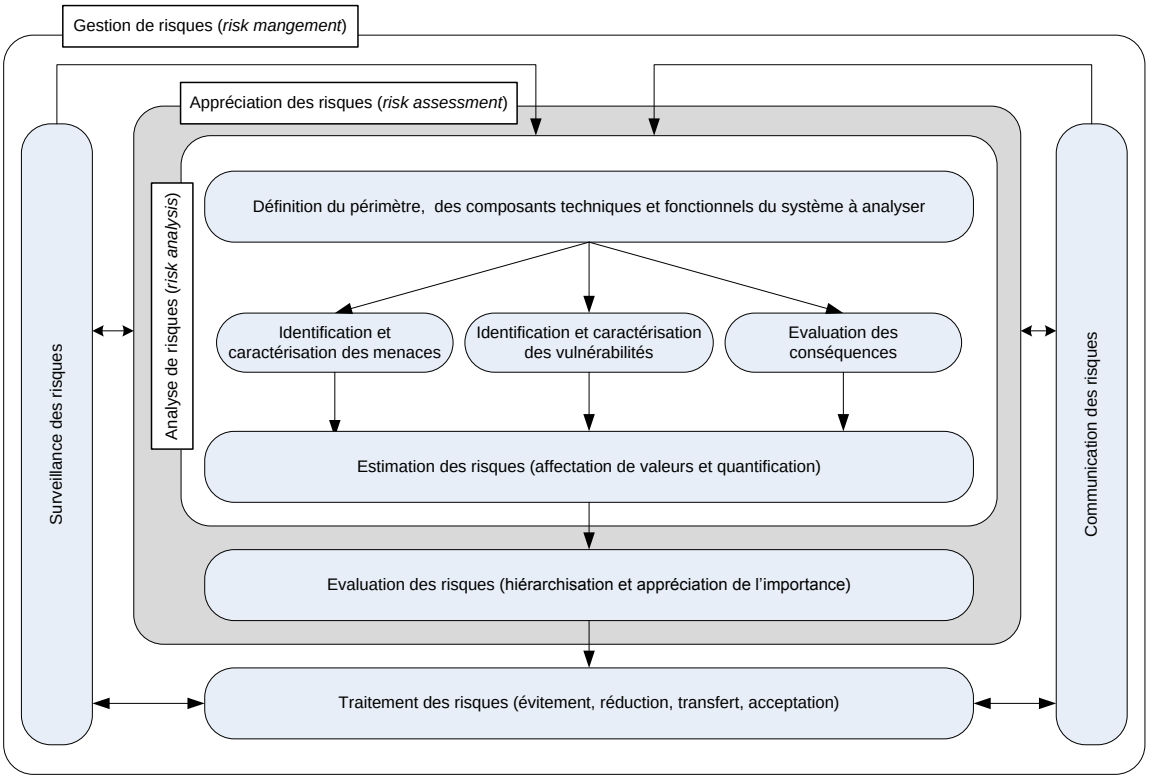


FIGURE II.1 – Gestion et analyse de risques

par rapport à des scénarios accidentels de référence sans faire appel directement à la notion de probabilité d'occurrence, et où les études probabilistes sont des compléments. *A contrario*, dans les pays d'influence anglo-saxonne, les approches probabilistes constituent les fondements des démonstrations de sûreté [3]. Similairement, le domaine de la protection physique des installations nucléaires repose généralement sur la prise en compte d'un document dit DBT (*Design Basis Threat*), établi par les autorités nationales, et définissant les caractéristiques des attaquants potentiels et les scénarios de référence contre lesquels les protections d'une installation doivent être dimensionnées, en dehors de toute considération probabiliste [153, 142]. Dans ces cas, une « approche graduée », proportionnant les dispositifs de sécurité en termes de conséquences potentielles uniquement, est fréquemment adoptée [142, 49].

1.1.2 Similarités dans les grands principes de conception et d'exploitation

En conception, aussi bien en sûreté qu'en sécurité, les dispositions de type préventif sont définies en priorité [142]. Plus les mesures sont considérées en amont de la conception, plus leur mise en œuvre est efficace et économiquement optimale. Les problématiques de sûreté et sécurité peuvent en effet avoir des impacts conséquents en termes d'architecture des systèmes, informatiques ou physiques. À titre d'exemple dans l'industrie nucléaire, les exigences de sûreté, comme le critère de défaillance unique, amènent à des dispositifs redondés, diversifiés et souvent séparés physiquement ; les exigences de sécurité ont aussi des conséquences directes sur la conception, l'implantation et le dimensionnement des ouvrages [142].

Les notions de risque et d'approche graduée évoquées dans la section précédente jouent un rôle central en conception, aussi bien en sûreté qu'en sécurité. En particulier, l'approche dite de défense en profondeur (*defense-in-depth*), initialement mise en œuvre dans le domaine militaire puis en sûreté nucléaire [54], s'applique à d'autres secteurs, en sûreté comme en sécurité informatique [23] ou physique [154]. La Section 3.1.1 revient sur la teneur de cette approche et l'historique des inspirations réciproques entre sûreté et sécurité à ce sujet. On peut cependant d'ores et déjà remarquer, à l'instar de [142], que le

3. Zio discute des origines et des implications de ces deux conceptions notamment pour le domaine nucléaire dans [152], où elles sont qualifiées respectivement de « structuraliste » et « rationaliste ».

domaine de la sécurité met souvent en œuvre un premier niveau, la dissuasion, approprié uniquement face à une menace intelligente. D'une façon générale, si le principe de défense en profondeur est pertinent aussi bien en sûreté qu'en sécurité, il est décliné pour tenir compte des différences de risques et de menaces.

D'un point de vue qualité logicielle, sûreté et sécurité exigent l'utilisation de techniques et de processus de développement spécifiques et souvent coûteux [18], limitant les comportements dangereux et les failles de sécurité (cf. Section 2 pour des exemples).

En exploitation, on retrouve également bon nombre de similitudes. En outre, sûreté et sécurité exigent toutes deux un suivi rapproché et une connaissance en profondeur du système et de ses évolutions [142]. Ainsi, référentiels de sûreté (*e.g.* [110]) et de sécurité (*e.g.* [49, 115, 117]) impliquent la tenue d'inventaires, le suivi des modifications, celui des éventuelles mesures palliatives temporaires, etc. La notion de maintenance préventive joue également un rôle capital, aussi bien en sûreté [151] qu'en sécurité [155]. Le retour d'expérience doit être régulièrement et minutieusement traité, alimentant la mise à jour des référentiels, également stimulée par un suivi rapproché des évolutions réglementaires, scientifiques et techniques. La conformité aux référentiels de sûreté comme de sécurité est réexaminée par des audits et inspections dédiées. Gestion de crise en sûreté et gestion de crise en sécurité sont aussi semblables à plusieurs égards [142] : elles impliquent dans les deux cas l'élaboration préventive de plan d'urgence et la réalisation d'exercices périodiques. Ces derniers permettent de vérifier l'adéquation des plans et des moyens face aux crises, d'évaluer l'entraînement des intervenants et les délais associés aux étapes du plan, de tester les chaînes décisionnelles, ou encore d'améliorer les interfaces et la coordination entre les différentes entités. Ceci-dit, le détail des procédures, et notamment les entités impliquées, peuvent bien sûr changer selon la nature du risque (*e.g.* le rôle de l'État, comme discuté en Section 1.2.5).

Enfin, travaux de recherche et bon sens se rejoignent pour dénoncer la complexité comme ennemi commun de la sécurité et de la sûreté. En sécurité informatique, le nombre de vulnérabilités d'une application peut être grossièrement lié au nombre de lignes de code la constituant [156], mais on peut en fait le lier plus finement à la richesse et la diversité des fonctionnalités [157, 158]. Côté sûreté, on pourra se référer sur le sujet à la première partie de l'ouvrage collectif d'EDF R&D, consacré au risque industriel et à la prise en compte de la complexité [151].

1.1.3 Non-cumulativité et non-composabilité des mesures de sécurité et de sûreté

Contrairement au sens commun, les mesures de sécurité, tout comme les mesures de sûreté, sont rarement cumulatives. À titre illustratif, Deleuze *et al.* citent dans [159] l'exemple de la multiplication de gardiens : plusieurs gardes opérant simultanément relâchent leur vigilance de par leur confiance dans celle des autres gardiens. Dans le domaine de la sécurité informatique, on peut donner un exemple en cryptographie sur l'utilisation de l'algorithme DES (*Data Encryption Standard*). DES a longtemps été le standard de référence de chiffrement symétrique ; son utilisation assurait une protection jugée suffisante pour des informations non classifiées mais considérées comme sensibles jusqu'au début des années 90. La taille limitée de sa clé (56 bits) et des faiblesses cryptographiques ont depuis mené à son remplacement par AES (*Advanced Encryption Standard*) [158]. Un sur-chiffrement effectué par le même algorithme DES mais avec une clé différente (opération appelée double-DES), est bien loin de doubler la solidité du chiffrement : l'amélioration est en réalité tout à fait minime si l'on a assez de mémoire pour mener l'attaque [160]. Dans un autre domaine, Anderson expose un exemple édifiant de mesures successivement mises en échec ayant abouti en octobre 2007 au survol involontaire des États-Unis par six bombes thermonucléaires opérationnelles, laissées ensuite plusieurs heures sans surveillance [158].

D'autre part, ni sécurité ni sûreté ne sont systématiquement composables [19, 161] : l'assemblage de deux composants considérés comme sûrs ou sécurisés ne forme pas un nouveau système héritant de ces qualités, ce qui soulève en outre de grandes difficultés par rapport à la certification de systèmes modulaires [162]. Sécurité comme sûreté doivent *in fine* être évaluées globalement.

1.1.4 Autres similitudes

Sûreté et sécurité, les éternels rabat-joies. Comme justement souligné dans [18, 21, 163], sécurité et sûreté partagent un fardeau commun : elles impliquent toutes deux de se prémunir contre des événements redoutés, de nature « négative » (attaques, accidents), à l'opposé de résultats recherchés, de nature « positive » (service rendu, production de biens). Elles sont souvent perçues comme des freins à la productivité, ou plus généralement comme des entraves aux exigences fonctionnelles et aux objectifs des systèmes ou organisations analysés. Dans ces conditions, leur évaluation est particulièrement délicate.

D'une part, elle nécessite de l'objectivité qu'il est difficile d'attendre des parties-prenantes, d'autre part, elle requiert une connaissance fine de domaines sensibles, peu compatible avec des intervenants extérieurs. De plus, alors qu'une gestion de risques efficace nécessite une bonne circulation des informations annonciatrices d'incidents (« signaux faibles »), elle est entravée par la tentation des responsables à les étouffer ou les gérer localement [164]. La valorisation des efforts en sûreté comme en sécurité est peut-être encore plus problématique ; elle ne peut se faire bien souvent qu'au conditionnel, en termes de situations et de conséquences hypothétiques évitées. En simplifiant, les retours sur investissements concrets sont généralement difficiles à justifier : sûreté comme sécurité partagent par conséquent fréquemment les premières coupes dans des arbitrages budgétaires contraints. De nombreuses catastrophes industrielles appuient ce triste constat (*e.g.* l'explosion de la raffinerie BP en 2005 [165]). Enfin, sécurité et sûreté sont victimes des mêmes biais caractérisant la gestion et les prises de décisions dans toute organisation face au risque : Choo cite des exemples représentatifs dans [166], Schneier [167, 168] et Anderson [169] les caractérisent plus particulièrement pour la sécurité informatique. Globalement, le développement de cultures spécifiques à la gestion des risques sûreté et sécurité, couplées à des obligations réglementaires adaptées, permettent de responsabiliser le management et d'amoindrir les difficultés mentionnées.

Culture sécurité et culture sûreté. On entend par culture sûreté (resp. culture sécurité) l'« ensemble des caractéristiques et des attitudes qui, dans les organismes et chez les personnes, font que les questions de sécurité (resp. de sûreté) bénéficient, en tant que priorité absolue, de l'attention qu'elles méritent en raison de leur importance » [13]. Dans cette acception, toutes deux partagent d'importantes ressemblances : dans les deux cas, l'engagement explicite de la direction, une politique volontariste de formation, et l'assimilation par chacun des enjeux et de sa place à jouer en termes de sûreté et de sécurité sont d'indispensables composantes [142, 170]. Pour cela, la croyance dans la crédibilité des menaces est fondamentale [170] ; les cultures doivent nourrir une attitude de vigilance générale et un questionnement proactif permanent. Le partage et l'échange d'information y sont également centraux (mais dans des modalités différentes, cf. Section 1.2.5). Malgré leurs ressemblances, les deux cultures ne se confondent cependant pas et doivent coexister, s'enrichir, et se renforcer mutuellement [142, 170].

Place du facteur humain. Le facteur humain joue un rôle critique aussi bien en sûreté qu'en sécurité. Cette place a cependant été reconnue plus tardivement en sécurité informatique, notamment suite aux attaques de *social engineering* (ingénierie sociale [4]), popularisées par le pirate Kevin Mitnick [171]. Les références [156, 158] donnent une description actuelle et complète de la problématique, étudiée de façon croissante depuis le début des années 90 [172]. En sûreté, l'incident de la centrale nucléaire de Three Mile Island en 1979 a constitué un tournant, stimulant un effort plus précoce et plus conséquent sur ces aspects, dont [151] donne un bon aperçu.

Cadre réglementaire et législatif. Les grands principes structurant le cadre réglementaire de la sûreté et ceux encadrant la sécurité des installations industrielles à risques sont, de façon macroscopique, très semblables [142, 18] : dans le deux cas, l'État nomme une autorité compétente (*e.g.* en France, l'ASN [5] pour la sûreté nucléaire, l'ANSSI [6] pour la sécurité informatique des infrastructures critiques), met en œuvre un système d'autorisation (sous forme de *licensing* dans les pays anglo-saxons), évalue les dispositions prises par les opérateurs, inspecte les installations. Selon les cas, ce fonctionnement peut reposer sur une même autorité couvrant à la fois sûreté et sécurité, ou sur des autorités différentes, mais coordonnées. Des structures internationales harmonisent les pratiques entre États (*e.g.* l'AIEA [7] pour la sûreté et la sécurité du nucléaire) ; des associations industrielles le font entre acteurs économiques dans certains secteurs (*e.g.* l'IATA [8] dans l'aviation). Les modalités de mise en œuvre des schémas réglementaires et législatifs diffèrent cependant de pays en pays, avec notamment des marges de manœuvre et d'initiative très différentes pour les opérateurs. Notons enfin que l'implication de l'État est généralement encore plus forte pour les problématiques de sécurité (cf. Section 1.2.5).

4. Ensemble de moyens non techniques, de type manipulations et abus de confiance, permettant à l'attaquant d'obtenir des informations utiles à la progression de son attaque. Le terme *social-engineering* est rarement traduit dans la littérature francophone, nous continuerons de l'employer par la suite en anglais.
5. Autorité de Sûreté Nucléaire (ASN, http://www.asn.fr).
6. Agence Nationale de la Sécurité des Systèmes d'Information (ANSSI, http://www.ssi.gouv.fr).
7. Agence International de l'Énergie Atomique (AIEA, http://www.iaea.org).
8. *International Air Transport Association* (IATA, http://www.iata.org).

1.2 Différences

1.2.1 Des risques de nature différente

Si en sûreté comme en sécurité, la notion de risque joue un rôle fondamental, la nature du risque considéré diffère. Dans la convention adoptée en fin de Chapitre I, les risques de nature malveillante sont associés à la sécurité, alors que les risques de type accidentel avec impacts potentiels sur l'environnement relèvent de la sûreté. Soulignons que l'origine des risques en termes de menace joue un rôle clé dans la distinction adoptée, les conséquences pouvant éventuellement être les mêmes. En effet, avec les définitions retenues, si un sabotage ou un accident nucléaire peuvent tous deux aboutir à un rejet de matière radioactive dans l'environnement, le premier relèvera de la sécurité alors que le second sera associé à la sûreté. Cette différence a des implications fondamentales en termes de gestion de risques, de modélisation et d'outils d'évaluation, comme discuté dans les sections qui suivent. En fait, il serait possible de découper encore plus finement le type de risque en s'appuyant sur le référentiel SEMA introduit dans le chapitre précédent. À chacune de ses sous-notions correspond en effet une catégorie de risques dont les spécificités peuvent induire des différences, notamment de modélisation. Ceci-dit, bien que d'une nature différente, les risques et les objectifs associés à la sûreté et ceux de la sécurité ne sont pas indépendants. Le Chapitre IV donne une vue approfondie sur leur interdépendance.

1.2.2 Différence dans la gradation des conséquences

Une autre différence soulignée dans [21], plus subtile mais également plus discutable, concerne la gradation en termes de gravité des conséquences dans les analyses de risques. La gradation serait généralement plus marquée en sûreté qu'en sécurité, distinguant en sûreté des degrés intermédiaires plus nombreux entre simples incidents, qui peuvent être tolérés (limites réglementaires) et ceux associés à des séquences accidentelles potentiellement catastrophiques, systématiquement considérés comme inacceptables. L'examen des références mentionnées dans les débats du Chapitre I tend à le confirmer. À titre d'exemple, la DO-178B [76] encadrant la sûreté logicielle dans l'aéronautique distingue cinq niveaux de criticité ; l'échelle internationale INES [9] notant la gravité des événements relatifs à la sûreté nucléaire en compte sept [173]. Les méthodes d'analyse de risques en sécurité sont d'une part souvent moins prescriptives dans le nombre de degrés à utiliser, et proposent d'autre part en exemple des échelles dépassant rarement les trois à quatre degrés. On peut expliquer une telle différence par la difficulté intrinsèque à évaluer les conséquences d'une attaque, souvent volontairement camouflées par l'attaquant soucieux de limiter les risques de détection. Plus conceptuellement, l'évaluation des conséquences d'un événement relevant de la sécurité s'extrait difficilement d'une vision binaire : il est soit considéré comme indicatif quand il correspond à une action autorisée mais significative d'un point de vue sécurité, et éventuellement suspecte, soit caractérisé comme une infraction et devient alors le signe d'une faillite potentiellement totale de la défense en place. La furtivité souvent recherchée d'une attaque et de ses conséquences, et plus globalement le caractère intelligent de l'attaquant, ne sont pas étrangers à un tel traitement. On trouve une autre illustration de cette différence de gradation, plus binaire en sécurité qu'en sûreté, dans la notion d'algorithme cryptographique « cassé ». Dès le moment où la communauté de la cryptologie identifie une faille théorique dans un algorithme, celui-ci est considéré comme cassé, même si aucune attaque réalisable en pratique n'a été proposée. Seuls les algorithmes n'ayant pas de faiblesses théoriques identifiées sont recommandés. L'adage « *attacks only get better, they don't get worse* », attribué à la NSA [10], sous-tend un tel comportement. Ces précautions sont par ailleurs confortées par la difficulté intrinsèque à connaître les moyens et les compétences des attaquants potentiels.

1.2.3 De l'évaluation de la menace/du danger

L'évaluation de la menace est radicalement différente selon sa nature malveillante ou accidentelle. Dans le premier cas, l'origine des menaces à évaluer est par définition hors de tout contrôle de l'analyste, et couvre un champ des possibles d'une extrême largeur [168, 158]. Dans le second cas, les caractéristiques des dangers sont plus accessibles, et le nombre des scénarios à considérer peut généralement être réduit à un ensemble restreint mais suffisant pour être considéré comme significatif. En sécurité, la menace est potentiellement intelligente et adaptative. En particulier, comme souligné par Deleuze *et al.* [24, 174], elle

9. *International Nuclear Event Scale* (INES), soit en français, échelle internationale des événements nucléaires.
10. *National Security Agency* (NSA, `http://www.nsa.gov`).

peut s'adapter vis-à-vis des vulnérabilités du système considéré, voire des contre-mesures et des réactions d'ordre défensif, alors que menaces et vulnérabilités n'ont pas d'interactions dynamiques en sûreté.

Dans le même ordre d'idée, une fois les dangers identifiés en sûreté, ils sont souvent considérés comme relativement stables dans le temps, faisant de l'utilisation de scénarios de référence une approche adaptée ; dans le cas de la sécurité, les profils, les motivations et les moyens des attaquants évoluent plus rapidement et de façon moins prévisible, ils dépendent de nombreux facteurs, souvent plus larges qu'en sûreté, qui peuvent ainsi inclure le contexte économique voire géopolitique. La logique de « course » entre attaquants et défenseurs contribue aussi à l'instabilité et à la dynamique de ces facteurs. Les référentiels doivent être mis à jour beaucoup plus fréquemment.

De plus, la caractérisation d'une menace de type accidentel peut bien souvent s'appuyer sur une approche statistique. En particulier, la modélisation probabiliste des événements de type panne, impliqués dans des scénarios de sûreté, s'appuie fréquemment sur des banques de données historiques considérables (*e.g.* IEEE pour l'électrotechnique). En sécurité, de telles approches statistiques ne peuvent être adoptées pour caractériser la malveillance : en outre, trop peu de données sont partagées et mutualisées pour des raisons d'image ou de confidentialité (cf. Section 1.2.5). Nous discutons plus largement de l'utilisation des probabilités en sécurité dans la section suivante et dans la Section 6.1 du Chapitre III. Par ailleurs, notons que les approches statistiques ne se prêtent pas pour autant à toutes les situations du domaine de la sûreté : par exemple, les systèmes logiciels présentent de par la nature déterministe de leurs défaillances et la complexité des codes des spécificités appelant des grandes précautions quant à l'utilisation de ces approches ; le secteur spatial a quant à lui des durées cumulées de vol trop limitées pour crédibiliser une approche strictement statistique dans les évaluations de sûreté.

Finalement, les difficultés intrinsèques à l'évaluation de la menace en sécurité laissent aussi place à une grande subjectivité dans leur perception par les populations dans le cas de la sécurité nationale ou d'installations industrielles à risque, et plus généralement dans la perception des utilisateurs du système analysé. Les glissements vers les comportements irrationnels et paranoïaques y sont ainsi plus fréquents que pour les risques de type sûreté [168], plus propices à une argumentation objective, ou pour le moins rationnelle. D'ailleurs, à l'aune de ces considérations, on comprend mieux l'attrait des approches déterministes, simplifiant le terme « probabilité » (*likelihood*) de la macro-équation du risque évoquée en Section 1.1.1, pour préférer une gradation des contre-mesures indexée uniquement sur les gravités de conséquences potentielles.

1.2.4 Des cadres théoriques et méthodologiques d'évaluation distincts

Nous nous intéressons ici aux différences d'ordre général entre les techniques d'évaluation du niveau de sûreté d'un système et celles évaluant sa sécurité. Les techniques elles-mêmes sont présentées en Section 2. Premier constat, également fait par Line *et al.* [21] ou Nicol *et al.* [45] : les méthodes quantitatives sont historiquement plus largement utilisées et industrialisées dans le domaine de la sûreté que dans celui de la sécurité. Les spécificités de cette dernière, décrites dans les sections précédentes, expliquent cet état de fait : la menace y est par nature difficile à caractériser en termes quantitatifs ; le qualitatif, associant prescriptif et « vues d'experts » est donc plus souvent privilégié.

De plus, les méthodes quantitatives s'appuient principalement sur des approches probabilistes. D'une façon générale, l'évaluation des probabilités à caractériser peut se faire par deux approches : par des tests et séries de mesures concrètes, effectués sur le système réel ou un prototype dans l'environnement opérationnel ; ou en s'appuyant sur des modèles abstraits, ne capturant que partiellement, et avec une fidélité très dépendante des formalismes, le comportement du système (approches dites à base de modèles, ou *model-based*). Sûreté et sécurité ne se prêtent pas de la même façon aux deux approches. Ainsi, s'il est possible d'évaluer concrètement la sécurité d'un système par des tests d'intrusion, il est beaucoup plus délicat de tester en grandeur nature des scénarios de sûreté, en particulier quand des conséquences catastrophiques y sont associées [175]. En fait, dans le cas de la sûreté, l'approche par tests et mesures concerne plutôt l'évaluation quantitative d'attributs, de type fiabilité ou disponibilité, de certains composants du système jouant un rôle critique dans les scénarios de sûreté à étudier (cf. Section 2.1.2).

Les approches à base de modèles sont utilisées en préparation des approches par tests ou quand celles-ci ne sont pas praticables ou sont trop chères. Ceci-dit, là encore, il convient de distinguer leur emploi en sûreté et en sécurité. Tout d'abord, ces approches sont communément adoptées pour les études de sûreté dans l'industrie [176], alors qu'elles y restent encore marginales pour la sécurité [45], leurs principales utilisations et développements étant encore, à ce jour, essentiellement académiques. De plus, elles peuvent

elles-mêmes être divisées en deux catégories [176], inégalement adaptées aux problématiques de sûreté et de sécurité. On distingue celles faisant appel à des modèles statiques (dit aussi structurels), qui supposent l'indépendance des composants, de celles impliquant des modèles dynamiques (ou comportementaux) qui permettent de capturer les notions de séquences et plus largement de dépendance des états du système et de ses composants dans le temps. Or si les études de sûreté s'appuient selon les besoins sur l'une ou l'autre catégorie [176], en sécurité, le caractère adaptatif de la menace et l'intelligence de l'attaquant semblent privilégier plus distinctement des modélisations de type dynamique.

D'une façon plus générale, les approches probabilistes en sécurité soulèvent des questions de fond encore largement ouvertes, directement liées à la difficulté de caractériser une menace par définition au moins partiellement hors de portée de toute analyse objective. Pour certains, l'utilisation des probabilités, en tant qu'outil par excellence de caractérisation de l'incertitude, y est donc parfaitement adaptée [177, 178] ; pour d'autres au contraire, la malveillance échappe à toute modélisation probabiliste. De plus, un tel type de modélisation doit couvrir des aspects plus nombreux qu'en sûreté, du fait du plus grand nombre d'inconnues à caractériser : on peut citer notamment la découverte des vulnérabilités, le processus d'exploitation de ces vulnérabilités, le comportement des contre-mesures aussi bien en termes de détection que de prévention et de réaction, et les interactions dynamiques entre ces différents éléments (plus simples voire inexistantes en sûreté). Dans une optique plus fondamentale, l'emploi des probabilités dans chacun des domaines s'inscrit en fait dans une conception différente de cette notion. Épistémologiquement et de manière simplifiée, les probabilités peuvent en effet être considérées de deux points de vue [179]. Le point de vue objectiviste « fréquenciste » (ou fréquentiste) s'appuie sur la recherche de fréquence dans un ensemble, ce qui suppose de pouvoir répéter de façon semblable et potentiellement infinie des expériences caractérisant l'aspect du système ciblé. Dans cette approche, il n'y a pas de probabilité sans ensemble de répétitions. Du point de vue des subjectivistes, au contraire, les probabilités sont pour reprendre les termes de [179] « une mesure de confiance, d'espérance raisonnable, de codage numérique d'un état de connaissance » et peuvent donc en particulier traiter de phénomènes à occurrence unique. Pour les fréquencistes, les probabilités sont caractéristiques de l'événement lui-même, pour les subjectivistes, les probabilités sont conditionnées par le degré de connaissance du phénomène à caractériser et de ses causes, éventuellement changeantes. Si l'utilisation des probabilités en sûreté peut relever aussi bien de l'approche fréquenciste que subjectiviste selon les aspects étudiés [151], elle nous semble clairement plus caractéristique de la seconde pour le domaine de la sécurité [177].

Enfin, aussi bien en sûreté qu'en sécurité, la quantification d'un modèle probabiliste peut se faire analytiquement, c'est-à-dire en utilisant des formules mathématiques liant directement les paramètres du modèle aux valeurs recherchées, ou par simulation de Monte-Carlo [180, 181]. On y procède à un grand nombre de tirages aléatoires des distributions stochastiques définissant le comportement des composants du modèle (on parle d'histoires, ou d'échantillons) pour approcher statistiquement les valeurs globales recherchées. Cette dernière approche est extrêmement polyvalente et permet de quantifier de nombreux types de modèles, mais ses performances sont très changeantes et elle peut s'avérer coûteuse en temps [176]. En fait, on peut souligner ici une nouvelle différence entre sûreté et sécurité. La performance des approches par simulation de Monte-Carlo est directement liée à la valeur des probabilités des variables aléatoires du modèle simulé : plus elles sont faibles, comme pour les pannes des systèmes en charge de la sûreté par conception extrêmement fiables, plus la simulation sera longue. Comme remarqué dans [182, 183], les modèles probabilistes en sécurité s'appuient sur des probabilités comparativement plus grandes que celles employées en sûreté, car associées à la malveillance et à une volonté d'attaquer déjà effective. Dans cette situation, les performances des simulations de Monte-Carlo sont bien meilleures.

1.2.5 Autres différences

Vocabulaire. Les difficultés terminologiques attachées aux termes même de sûreté et sécurité ont été débattues dans le Chapitre I. Les communautés associées ayant évolué séparément, elles ont aussi développé des terminologies propres, parfois différenciées selon les secteurs industriels, maniant pourtant des concepts souvent très proches, voire dans certains cas identiques. Ceci peut être entrevu dans les définitions de la notion de risque de la Table II.1, et confirmé par l'analyse des références de la Table I.2 examinées dans le Chapitre I. Par exemple, le domaine de la sécurité privilégiera le terme menace (*threat*) alors que le domaine de la sûreté utilisera plutôt danger (*hazard*) pour désigner des concepts généralement identiques [21]. *A contrario*, sûreté et sécurité emploieront parfois un même terme, mais dans une acception différente (*e.g.* « incident », événement aux conséquences limitées en sûreté, infraction au sens

large en sécurité). Plus ambigu encore, un même terme peut être employé par les deux communautés, mais dans un sens qui diffère selon les cas, sans pouvoir établir de correspondance systématique (*e.g.* vulnérabilité). Plusieurs efforts d'harmonisation de vocabulaire peuvent être cités, dont notamment ceux de Stoneburner [184], Deleuze [174], ou de façon plus influente, ceux de Laprie *et al.* dont la vue la plus complète et la plus récente est donnée dans [16]. Dans cette dernière approche, évoquée dans l'Introduction Générale, les notions mêmes de sûreté et de sécurité sont définies dans un cadre plus large (la sûreté de fonctionnement, ou *dependability*) qui s'appuie sur une taxonomie complète permettant d'intégrer événements de type malveillant et de type accidentel.

Accès à l'information et au partage du retour d'expérience. La confidentialité de l'information constitue tout à la fois un objectif spécifique à la sécurité et une composante forte de sa culture. La nature malveillante des risques considérés fonde cette différence marquée avec la sûreté, où transparence et large accès à l'information sont le plus souvent recherchés. En sécurité, les référentiels de menaces, les évaluations de risques et la description des contre-mesures sont considérés comme des informations extrêmement sensibles, qui pourraient être exploitées à des fins malveillantes. Ceci-dit, sûreté comme sécurité sont renforcées par l'échange des retours d'expérience et des savoir-faire : seules les modalités associées à ces échanges diffèrent profondément de par l'exigence de confidentialité [142]. Cette exigence reste au cœur d'un débat animé au sein même de la communauté sécurité, car elle est à l'origine d'une conception fallacieuse dans laquelle le secret suffirait à assurer la sécurité (on la désigne souvent par « sécurité par l'obscurité ») [10].

Une implication encore plus large de l'État en sécurité. Dans leur approche comparative entre sûreté et sécurité nucléaire [142], Jalouneix *et al.* font remarquer une implication plus large de l'État en matière de sécurité nucléaire. Ce constat peut être élargi à la sécurité des installations industrielles et des infrastructures critiques. On peut citer plusieurs raisons à cette situation. D'une part, alors que les dispositions de sûreté sont généralement toutes mises en place par les opérateurs, ces derniers ne peuvent assumer entièrement les mesures nécessaires à une bonne gestion du risque malveillant : ils n'en ont ni les moyens, ni la légitimité. L'État intervient à ce titre notamment dans ses missions de renseignement et d'évaluation du risque malveillant ; il peut également intervenir dans la réponse par l'entremise des forces de l'ordre ou des autorités judiciaires ; il édicte les règles de confidentialité pour les informations relevant de la sécurité nationale et contribue aux enquêtes d'habilitation pour les activités et les accès aux informations sensibles. D'autre part, en sûreté, l'opérateur contrôle l'origine du risque puisqu'il est responsable du système, alors qu'en sécurité, les menaces sont par nature moins maîtrisables et potentiellement extérieures : l'État est alors le plus à même d'intervenir.

2 Sûreté et sécurité à travers leurs techniques d'évaluation

Sûreté et sécurité entretiennent des relations subtiles, tissées de ressemblances frappantes mais aussi des différences fondamentales. Elles se reflètent dans les outils et méthodes développés par chacune des communautés. Dans cette section, nous donnons un panorama des techniques les plus communément utilisées pour évaluer le niveau de sûreté ou de sécurité d'un système au sens large (produit, installation, organisation), en conception comme en exploitation. Par évaluation nous entendons la caractérisation qualitative ou quantitative du niveau de sûreté ou de sécurité d'un système, permettant la comparaison avec des systèmes équivalents dans un même environnement mais ayant pris des dispositions différentes. Ainsi, il ne s'agit pas de faire un catalogue des contre-mesures et des approches contribuant à l'amélioration de la sûreté ou de la sécurité, mais bien de se concentrer sur ce qui va permettre leur évaluation. Le champ à couvrir reste néanmoins très large : démarches expérimentales, techniques de modélisation, approches probabilistes... Nous ne prétendons pas à l'exhaustivité : les techniques présentées ici ont été choisies sur la base de leur visibilité dans la littérature normative, scientifique et technique, mais aussi par rapport à leur pertinence pour appuyer le contenu de la Section 3 de ce chapitre, inventoriant les inspirations réciproques entre sûreté et sécurité, et le reste de ce mémoire. Cette sélection donne une image représentative des « boîtes à outils » de chaque domaine, avec en filigrane, les similarités et différences caractérisant sécurité et sûreté telles que discutées précédemment.

2.1 Considérations préliminaires

2.1.1 Des « boîtes à outils » pour l'évaluation ?

Cette expression semble en effet appropriée aussi bien pour la sécurité que pour la sûreté. Dans les deux cas, l'évaluation est un processus complexe et approximatif, étant donné la complexité des phénomènes et des systèmes considérés. Elle implique bien souvent la combinaison de différentes techniques, aux contours parfois flous et aux ramifications souvent nombreuses. Ainsi, certaines techniques font appel implicitement ou explicitement à d'autres, pourtant présentées comme distinctes. D'autres encore changent de désignation selon leur domaine d'application, alors que dans certains cas, une même dénomination cache des outils distincts. Nous tâcherons d'identifier les éventuelles ambiguïtés, et de fournir à chaque fois les références les plus largement admises.

2.1.2 Liens entre sûreté, fiabilité et sûreté de fonctionnement

La sûreté, comme la sécurité, est une propriété émergente, apparaissant au niveau du système alors que tous ses composants interagissent [185]. Elle ne se confond pas avec la fiabilité, qui s'intéresse uniquement aux pannes du système ou à celles de ses composants, sans préoccupation pour les conséquences sur l'environnement du système. Un système peut être fiable sans être sûr (si la réalisation de ses fonctions comporte intrinsèquement un risque sûreté), réciproquement, il peut être sûr sans être fiable (quand les pannes amènent systématiquement le système dans une position de repli sans danger) [18]. Toutefois, une connexion forte existe entre les deux notions. Ainsi, la sûreté d'un système dépend de fonctions de sûreté bien identifiées, en charge d'éviter les comportements dangereux ou de limiter leurs conséquences. Les sous-systèmes portant ces fonctions doivent avoir un haut niveau de fiabilité et/ou de disponibilité : l'évaluation de la sûreté passe alors par des évaluations de fiabilité ou de disponibilité, et plus généralement, par des études de sûreté de fonctionnement au sens précisé dans l'Introduction Générale. Aussi, bon nombre d'outils présentés en Section 2.2 relèvent en fait plus directement du domaine de la fiabilité et de la sûreté de fonctionnement, mais sont mis en œuvre dans les études de sûreté, de nature plus systémique.

2.1.3 Monde numérique et monde physique

Dans le panorama des outils d'évaluation de la sûreté, nous nous intéressons à tout type de systèmes, qu'ils soient informatiques, électrotechniques ou mécaniques. Quand une technique s'applique à un type particulier de système, nous le précisons. Si la sûreté, telle que nous la considérons, s'intéresse *in fine* aux risques accidentels sur l'environnement physique du système, ces risques peuvent avoir pour origine des dysfonctionnements aussi bien physiques que logiciels. Dans le domaine de la sécurité, nous concentrons par contre notre panorama sur les outils relevant principalement de la sécurité informatique, même si certains peuvent être également employés pour la sécurité physique (parfois appelée protection physique).

2.2 Quelques techniques pour l'évaluation de la sûreté

Nous présentons dans cette section une sélection de techniques, dont la Table II.4 p. 41 donne une vue d'ensemble. Des inventaires plus complets peuvent être trouvés par exemple dans les normes [186] et [187]. Des ouvrages de référence en sûreté de fonctionnement (*e.g.* [188, 176]) permettront aussi de compléter ou approfondir ce panorama.

2.2.1 Approches qualitatives structurées

Analyse préliminaire de danger. L'analyse préliminaire de danger est une méthode inductive simple, visant à identifier les dangers d'un système ou d'une installation et ses causes (situations, événements), en amont dans la conception, avec peu d'informations. L'évaluation est macroscopique, elle doit comme son nom l'indique être considérée comme préliminaire à des études plus approfondies. Elle aboutit à une liste de dangers et de risques, ainsi qu'à des recommandations de type acceptation ou déploiement de contre-mesures. Elle s'appuie pour cela sur des tableaux et *check-lists* établis par des experts du domaine considéré. Des évaluations quantitatives sont possibles, on parle alors souvent d'analyse préliminaire de risque. Cette méthode a été mise au point au début des années 60 aux États-Unis pour l'analyse de la sûreté de missiles, et s'est depuis généralisée à tous les secteurs industriels à risques.

AMDE et AMDEC. L'Analyse des Modes de Défaillance et de leurs Effets (AMDE, ou FMEA en anglais) [189] a aussi été mise au point dans les années 60 aux États-Unis dans le domaine de l'aéronautique. Elle reste très appréciée de par sa simplicité, notamment dans l'automobile, l'aéronautique et le ferroviaire, mais aussi le logiciel. Elle se base sur l'utilisation de grands tableaux sur un mode qualitatif et inductif, déclinés selon le secteur. Un examen des défaillances potentielles des composants du système permet d'identifier et caractériser les conséquences. Sa version augmentée, l'AMDEC (le C pour Criticité, FMECA en anglais) prend en compte la notion de gravité des conséquences, sur un mode qualitatif ou éventuellement quantitatif. Revers de leur simplicité, AMDE et AMDEC ne permettent pas de prendre en compte les défaillances multiples.

HAZOP *(HAZard and OPerability studies).* HAZOP est une technique essentiellement qualitative, mais fortement structurée et systématique. Elle permet d'identifier des dangers potentiels sous forme de perturbations et de déviations d'un fonctionnement nominal. Causes et solutions sont aussi caractérisées. HAZOP s'articule autour d'une phase d'examen menée par une équipe multidisciplinaire. L'emploi de mots-guides (*guide words*) permet de systématiser les analyses, structurées sous la forme de tableaux comme illustré par la Table II.2. Mise au point dans l'industrie chimique dans les années 60 [190], elle a depuis été adaptée à de nombreux secteurs et reste prédominante dans le pétrochimique et l'analyse des systèmes thermo-hydrauliques [191]. Elle peut s'inscrire en complément d'autres techniques, notamment pour compenser son incapacité à couvrir les défaillances multiples [186]. La norme [192] donne une description générale, mais de nombreuses déclinaisons sectorielles existent. En outre, les mots-guides sont alors modifiés pour être adaptés au contexte considéré. On trouve par exemple des déclinaisons pour le logiciel comme SHARD [193] ou celles décrites dans [194].

TABLE II.2 – Mots guides typiques [192] et exemple de présentation de résultats HAZOP

Mot-guide	Signification
NE PAS FAIRE	Négation totale de l'intention de conception
PLUS	Augmentation quantitative
MOINS	Diminution quantitative
EN PLUS DE	Modification/augmentation qualitative
PARTIE DE	Modification/diminution qualitative
INVERSE	Contraire logique de l'intention de conception

N°	Déviation	Causes	Conséquences	Préventions	Actions à mener	Probabilité	Gravité

Analyse des conditions insidieuses et Analyse transitoire (*Sneak analysis*). Inventée dans les années 60 par Boeing, l'analyse des conditions insidieuses a rapidement été adoptée par la NASA pour vérifier les circuits électriques de ses appareils, notamment dans le programme *Apollo* [195]. Elle a inspiré une approche au champ d'application plus large, l'analyse transitoire. Cette dernière s'applique au matériel comme au logiciel, et permet de s'intégrer avec d'autres outils comme l'AMDEC, les arbres de défaillances ou HAZOP. L'objectif est de repérer en conception des conditions insidieuses latentes (*sneak conditions*) susceptibles d'échapper aux tests classiques et de provoquer des dysfonctionnements majeurs. Des réseaux arborescents représentent les fonctions du système et les entrées influençant ces fonctions. Des forêts regroupent ces réseaux et représentent les liens entre entrées et sorties du système. Une condition insidieuse est un chemin imprévu qui dans certaines conditions peut provoquer un comportement indésirable. À l'origine de nature électrique, un chemin peut représenter plus largement une séquence logique et se composer d'éléments matériels, logiciels ou humains (actions opérateur). La méthode est surtout employée dans l'industrie spatiale [196, 197], militaire [198] et nucléaire [199] aux États-Unis.

Zonal Hazard Analysis* (ZHA) / *Zonal Safety Analysis (ZSA). Contrairement aux méthodes découpant le système considéré de façon fonctionnelle, l'analyse de zone s'intéresse à la répartition spatiale de ses composants. Elle étudie plus particulièrement comment la proximité physique de certains composants peut aboutir à des défaillances, même s'ils sont fonctionnellement indépendants. Elle a été développée dans l'industrie aéronautique pour prendre en compte les interactions de systèmes présents

dans une même zone physique de l'appareil (ailes, soute) [200]. Par exemple, la ZHA prendra ainsi en compte la proximité de systèmes de nature hydraulique avec d'autres, *a priori* indépendants, mais de nature électrique. On retrouve ces considérations, et des approches méthodologiques comparables, dans d'autres industries à risques dont le nucléaire.

2.2.2 Approche par conformité à des référentiels

Ces approches sont très courantes dans le domaine du logiciel de sûreté. La norme IEC 61508 [110], qui se veut générique et centrale, joue effectivement un rôle clé dans ce paysage : on peut en fait y distinguer les normes sectorielles qui ne se sont pas, ou peu, rattachées à l'IEC 61508, pour des raisons industrielles ou historiques, et celles toujours plus nombreuses qui en constituent des déclinaisons. Dans la première catégorie, on citera notamment la DO-178B [76] pour l'avionique. Dans la seconde se trouvent par exemple les références [82, 83] pour le domaine ferroviaire. De façon intermédiaire, on peut citer l'IEC 61513 [58] pour le nucléaire qui sur la forme décline l'IEC 61508, mais dont les normes filles, issues d'un long passé industriel, prennent en fait de grandes libertés par rapport à celle-ci. Si toutes reposent sur une classification en niveaux de criticité, les normes ayant résisté au rattachement à l'IEC 61508 proposent généralement un cadre méthodologique n'exigeant pas la mise en œuvre de techniques spécifiques de sûreté de fonctionnement, alors que l'IEC 61508 s'avère plus prescriptive.

L'IEC 61508. L'IEC 61508 [110] couvre tous les aspects du cycle de vie des systèmes en charge de fonctions de sûreté, qui permettent de prévenir et réduire les risques d'accidents sur les utilisateurs et l'environnement. En particulier, des cibles quantifiées sont précisées quant au niveau de défaillance admissible pour des fonctions de sûreté selon la criticité de leur tâche, aussi bien pour les équipements E/E/EP (Électriques/Électroniques/Électroniques Programmables) que pour le logiciel. Quatre niveaux d'intégrité de sécurité (ou SIL, pour *Safety Integrity Levels*) [11] les graduent. Différentes techniques et méthodologies, couvrant de la spécification jusqu'aux tests, sont prescrites selon le niveau à atteindre. La documentation exigée à chaque étape du cycle de développement est également précisée.

La DO-178B. La DO-178B [76] est à l'origine une recommandation produite et éditée conjointement par les organismes RTCA et EuroCAE en 1992. Elle a aujourd'hui un statut de norme internationale *de facto*. Elle précise les exigences relatives aux processus de développement des logiciels critiques pour la sûreté (en fait, la « navigabilité », ou *airworthiness* en anglais) en aéronautique, en vue de leur certification. Cinq niveaux d'assurance de développement croissants (DAL, pour *Development Assurance Level*) y sont distingués, attribués selon l'impact le plus sévère d'une défaillance du logiciel considéré. À chaque DAL correspond des exigences organisées sur un mode cumulatif : les exigences d'un niveau s'ajoutent à celles du niveau inférieur. Au niveau le plus bas, le développement logiciel n'est soumis à aucune contrainte. Dès le second niveau, tous les processus liés au développement doivent s'accompagner d'une documentation. Cela concerne ainsi la phase de planification du développement, le processus de développement proprement dit, la vérification, la gestion de configuration et le processus d'assurance qualité. Pour chaque processus, les DAL définissent les objectifs et les données à fournir pour démontrer leur atteinte, sans pour autant imposer une méthode spécifique. Les preuves nécessaires à la certification des logiciels sont à la charge des avionneurs. Notons pour finir qu'après plus de vingt-cinq ans de service, une nouvelle version de la DO-178/ED-12 est prévue pour 2012.

2.2.3 Approches quantitatives par modèles probabilistes statiques

Diagrammes de fiabilité. Les diagrammes de fiabilité (ou RBD pour *Reliability Block Diagrams*) [201] découpent le système analysé en blocs fonctionnels, et les connectent en circuits séries ou parallèles selon leur nécessité au fonctionnement global du système. Le chemin formé est généralement proche de la configuration physique, facilitant la modélisation. Les blocs ont un comportement binaire fonctionnement/panne, modélisé par des processus stochastiques. Le système est considéré comme opérationnel tant que le « signal » de fonctionnement peut traverser le diagramme de gauche à droite (de I à O sur la Figure II.2). On peut alors rechercher les combinaisons de défaillances menant à la panne, et quantifier disponibilité et fiabilité du système. La Figure II.2 montre un système comportant un composant A non

11. Nous retrouvons ici les ambiguïtés de traduction discutées dans le Chapitre I, *safety* étant traduit par l'IEC en « sécurité » dans les documents en français.

redondant, et dont la défaillance provoque la défaillance de tout le système ; les autres composants B et C sont redondés. Les RBD ne sont utilisés que pour des cas très simples. Modèle statique, une version dynamique a été développée mais reste essentiellement académique [202].

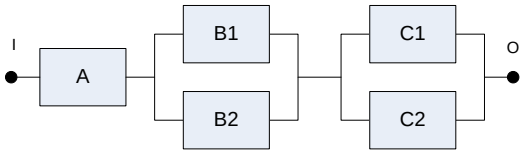
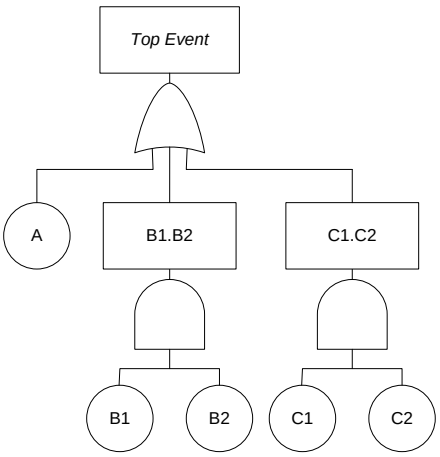


FIGURE II.2 – Exemple de RBD « mixte avec redondance »

Arbres de défaillances (*Fault Trees*). Les arbres de défaillances ont été développés au début des années 60 par les laboratoires Bell pour l'évaluation de la fiabilité du lancement de missiles intercontinentaux. Ils ont rapidement été adoptés par l'industrie aéronautique, avant de se généraliser à de nombreux autres domaines [203]. Avec leur capacité à intégrer les défaillances multiples, les arbres de défaillances complètent souvent l'AMDE(C) dans les industries à risques, comme dans le nucléaire [204] ou le spatial [205]. Notons qu'ils ont également été adaptés au domaine logiciel, notamment pour les logiciels critiques en termes de sûreté [206]. Le principe est simple et puissant : dans une approche déductive, on y articule graphiquement les causes d'un événement redouté dans un arbre logique. L'événement redouté constitue le « sommet » (*top event*), les causes élémentaires sont les feuilles, qui modélisent des événements binaires de type panne/fonctionnement et qui sont reliées par des portes booléennes. Initialement simples OU et ET logiques, elles se sont progressivement diversifiées, incluant portes k/n (vérifiée si et seulement si au moins k entrées sur les n de la porte sont vraies), THEN (imposant un ordre d'événement), etc. (cf. [207]). Il est alors possible d'identifier les coupes minimales, plus petits ensembles d'événements devant se produire pour causer l'événement redouté, et de quantifier les probabilités d'occurrence de celui-ci en associant des probabilités aux feuilles. Les contributions de chaque feuille peuvent également être évaluées. La Figure II.3 donne un exemple simple d'arbre de défaillances (pour le système précédemment modélisé en RBD) et décrit les composants graphiques principalement employés dans ce formalisme. Les grands principes de quantification peuvent être trouvés par exemple dans [152] tandis que des explications plus détaillées sont disponibles notamment dans [188, 207, 204, 205].



Symbole	Nom et description
	Description d'un événement intermédiaire ou de l'événement redouté
	Événement de base (généralement paramétré)
	Événement non développé (qui reste à définir)
	Porte de transfert (renvoie vers un autre arbre)
	Porte ET (vérifiée si tous les événements d'entrée sont vérifiés)
	Porte OU (vérifiée si au moins un des événements d'entrée est vérifié)

FIGURE II.3 – Exemple simple d'arbre de défaillances et symboles classiquement associés

La technique d'origine est purement statique, mais une version dynamique a été développée dès le début des années 90. Nous lui consacrons un paragraphe de la Section 2.2.4. La technique des arbres de défaillances est sans doute une des techniques les plus utilisées industriellement en sûreté de fonctionnement ; elle est largement outillée par des logiciels spécialisés. On peut par exemple citer des outils

développés par les grands instituts de recherche du domaine nucléaire comme *SAPHIRE* de l'Idaho National Lab aux États-Unis, ou *ASTRA* du *Joint Research Center* (JRC) de la Commission européenne [208] ; de nombreuses suites commerciales sont aussi disponibles.

Arbres d'événements (*Event Trees*). Les arbres d'événements ont été introduits en 1975 pour évaluer le risque de fusion du cœur des centrales nucléaires états-uniennes [209]. Contrairement aux arbres de défaillances, ils reposent sur une approche inductive : ils modélisent sous la forme d'arbres les séquences accidentelles pouvant découler d'événements dits « initiateurs » préalablement identifiés. Un embranchement probabilisé de l'arbre représente la réussite ou l'échec d'une contre-mesure visant à limiter les conséquences de l'événement initiateur. La Figure II.4 donne un exemple élémentaire d'arbre d'événements modélisant les conséquences d'une pression trop élevée dans une canalisation, et quantifiant le risque de rupture malgré deux soupapes. Contrairement à leur apparence, les arbres d'événements ne modélisent pas l'aspect séquentiel et relèvent d'une approche purement statique et combinatoire : une telle ambiguïté sémantique est bien soulignée dans [176]. Ils sont souvent employés de façon combinée avec les arbres de défaillances, ces derniers pouvant servir à caractériser les probabilités des embranchements (technique dite de *fault-tree linking* [210]). Ils sont surtout utilisés dans l'industrie chimique [211] et l'industrie nucléaire. Dans cette dernière, ils servent de base aux EPS (Études Probabilistes de Sûreté) appuyant les dossiers de sûreté des centrales nucléaires examinés par les autorités de régulation [151].

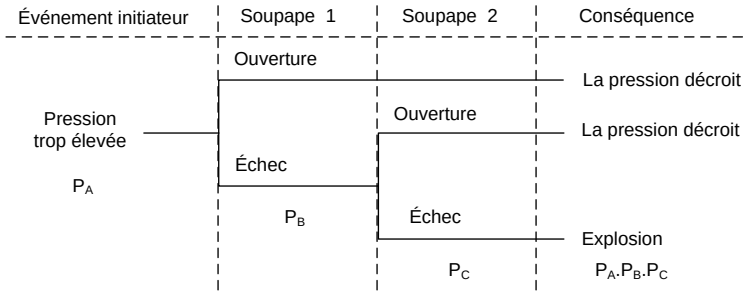


FIGURE II.4 – Exemple simplifié d'arbre d'événements

Diagrammes Cause-Conséquence. Inventés par les laboratoires Risø au Danemark [212] pour les études de sûreté des centrales nucléaires, les diagrammes Cause-Conséquence intègrent dans un même formalisme graphique approche déductive de type arbre de défaillances et approche inductive de type arbre d'événements. Elles sont combinées par des portes OUI/NON, modélisant la réalisation de conditions particulières ou de défaillances de dispositions conçues pour limiter les conséquences d'un événement initiateur. Les causes des conditions ou des défaillances sont analysées par des arbres de défaillances. Un exemple de principe est donné en Figure II.5 : l'issue des deux portes OUI/NON est conditionnée par deux arbres de défaillances. Malgré une définition bien formalisée et l'avantage d'intégrer approches déductive et inductive, les diagrammes Cause-Conséquence ne semblent avoir été employés de façon limitée que dans l'industrie nucléaire [213, 214], et de façon encore plus marginale dans le domaine logiciel [215, 19] et l'industrie ferroviaire [216]. La réf. [217] détaille comment obtenir des quantifications aux performances comparables avec celles des arbres de défaillances et des arbres d'événements.

Réseaux bayésiens. Les réseaux bayésiens constituent un formalisme mathématique aux nombreux débouchés (intelligence artificielle, médical,...) et trouvent une place grandissante dans les modèles de sûreté et de fiabilité depuis une quinzaine d'années [218]. Issus des travaux de Pearl dans les années 80 [219], ils représentent un ensemble de variables aléatoires, sous forme de nœuds, et leurs relations probabilistes, sous forme de flèches orientées selon le sens de l'influence, formant un graphe orienté sans circuit. Chaque variable aléatoire est définie par une table de probabilités conditionnelles relatives aux autres variables dont elle dépend. La conception sous-jacente est celle initiée par Bayes au XVIII^{e} siècle [220], fondamentalement subjectiviste au sens précisé dans la Section 1.2.4, et qui formalise le fait que la probabilité d'un événement puisse être progressivement ajustée par des événements subséquents [179]. La

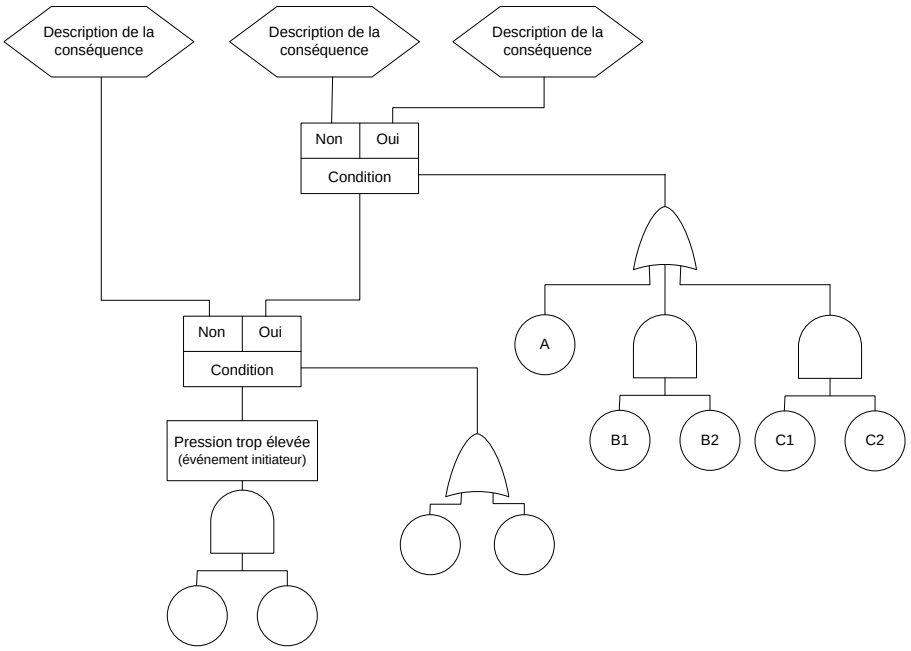


FIGURE II.5 – Exemple simplifié de diagramme Cause-Conséquence

formalisation mathématique repose sur l'expression de probabilités conjointes par des probabilités conditionnelles par l'utilisation du théorème de Bayes. Une introduction de leurs principes et leur utilisation en sûreté et fiabilité est donnée par Bouissou dans [176], un panorama plus complet et un historique sont donnés par Langseth et Portinale dans [218]. Une présentation complète peut être trouvée dans [221]. L'intérêt porté aux réseaux bayésiens est notamment lié à leur extrême polyvalence : ils permettent d'aborder des aspects aussi variés que l'analyse de retour d'expérience, le diagnostic, la prévision, l'optimisation, la détection d'écarts ou encore l'actualisation de modèle. On peut aussi noter que les arbres de défaillances peuvent être considérés comme un cas particulier de réseaux bayésiens [222]. Ceci-dit, malgré leurs avantages incontestables, leur emploi en sûreté et fiabilité reste encore très académique, alors qu'il est déjà bien établi dans d'autres domaines (*e.g.* intelligence artificielle). De nombreux outils logiciels sont disponibles pour leur traitement (*e.g.* ceux de la société Bayesia, ou encore Netica).

2.2.4 Approches quantitatives par modèles probabilistes dynamiques

Les modèles dynamiques, dit aussi comportementaux, offrent des capacités de modélisation plus puissantes que celles des modèles statiques, mais sont aussi plus difficiles à construire et analyser. Il s'agit donc de trouver un juste compromis. Dans son livre [176], Bouissou résume les arguments en faveur de chaque catégorie, et rappelle ainsi qu'à EDF, après avoir recouru à des approches dynamiques, les méthodes booléennes statiques ont finalement été retenues pour les études probabilistes de sûreté des centrales nucléaires, apportant un meilleur compromis entre effort, qualité des résultats et facilité d'échange avec les parties-prenantes.

Modèles et graphes de Markov. Les techniques de modélisation dites markoviennes sont couramment employées pour modéliser des systèmes avec dépendances, redondances, ou comportements séquentiels. Souples et polyvalentes, elles permettent de prendre en compte les réparations et des stratégies de maintenance élaborées [188, 223]. En général, les techniques de Markov s'appuient sur un graphe à états discrets représentant le comportement du système dans le temps. Chaque état du graphe représente un état du système défini par la combinaison de l'état de fonctionnement de ses composants. Lors de la défaillance ou du rétablissement d'un composant, le système passe d'un état à un autre. Les transitions sont caractérisées par des lois de probabilités exponentielles paramétrées par des taux de défaillance (λ) et des taux de réparation (μ). Dans tous les cas, la propriété de Markov est vérifiée, en d'autres termes, les transitions sont sans mémoire : tout le passé du système est résumé par son état présent.

Il existe différentes configurations : les taux de transitions peuvent être constants (processus de Markov homogène) ou variables (non homogène), le temps peut être considéré comme discret ou continu. Notons qu'en français, l'expression chaîne de Markov s'emploie, en principe, uniquement dans les cas d'espaces discrets définis en temps discret, alors qu'en anglais, *Markov chain* est employée en temps discret ou continu. L'utilisation la plus courante des modèles de Markov dans les études de sûreté de fonctionnement reste celle d'un espace à états discrets en temps continu [176]. Leur étude nécessite la résolution d'un système d'équations différentielles linéaires du premier ordre et à coefficients constants, dites de Chapman-Kolmogorov [188]. De nombreuses méthodes, comme l'exponentiation de matrices et les transformées de Laplace, permettent de résoudre ce type d'équations et d'obtenir fiabilité, disponibilité et maintenabilité en fonction du temps. Le calcul des valeurs moyennes type MTTF (*Mean Time To Failure*), MTBF (*Mean Time Between Failures*), etc. est également possible. Des ouvrages comme [188] ou [224] exposent de façon détaillée les techniques de l'analyse markovienne pour la sûreté. Leur emploi direct reste cependant rare car rapidement soumis au risque d'explosion combinatoire : on recourt souvent à des modèles de plus haut niveau [176] comme les réseaux de Petri ou les BDMP (cf. pages suivantes).

Nous présentons ici la démarche de modélisation par graphe de Markov pour l'exemple simple d'un système à deux éléments redondés. Les formules et quantifications associées peuvent être trouvées dans [223]. Un composant n'ayant que deux états, 0 pour fonctionnel et 1 pour défaillant, les états possibles du système sont (0, 0), (1, 0), (0, 1) et (1, 1). Si l'on considère un système en série, (0, 0) est le seul état de fonctionnement, les trois autres correspondant à des états de défaillance. Mis en redondance, (0, 0), (0, 1), (1, 0) correspondent alors aux états de fonctionnement. Le graphe de Markov d'un système redondant « 1 sur 2 » sans réparation possible est donné en Figure II.6a). Dans celle-ci, les numéros d'états 0, 1, 2 et 3 correspondent aux états (0, 0), (0, 1), (1, 0) et (1, 1). Les arcs correspondent aux transitions d'un état à l'autre, selon les taux de défaillance λ de chaque composant. Si le système est réparable, des arcs retours sont ajoutés, comme dans la Figure II.6b) avec les taux de réparation μ correspondants.

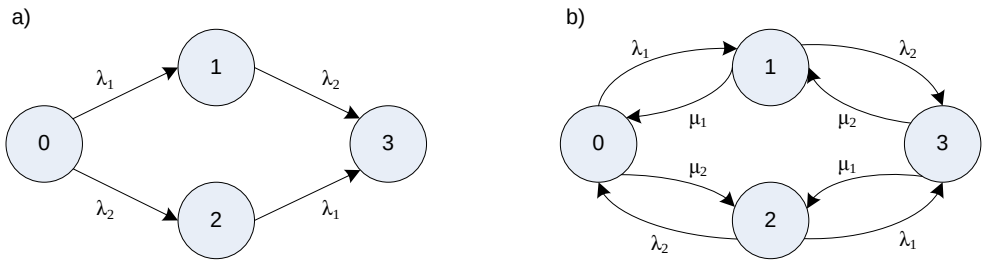


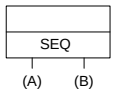
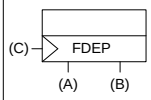
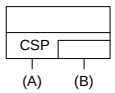
FIGURE II.6 – Graphes de Markov d'un système à deux composants

Notons pour finir que certaines études font appel à des processus plus généraux, dit semi-markoviens, pour lesquels les probabilités de transition d'un état vers un autre ne dépendent que du temps écoulé depuis l'arrivée dans l'état considéré [188]. Ce type de processus conduit à un système d'équations pouvant aussi être résolues analytiquement.

Arbres de défaillances dynamiques (ou DFT, pour *Dynamic Fault Trees*). Les arbres de défaillances dynamiques ont été introduits par Dugan aux États-Unis au début des années 90 [225, 226] pour dépasser les limites dues à la nature statique des arbres de défaillances classiques. Ils visent à associer les capacités de modélisation dynamique propres aux modèles markoviens à un formalisme proche des arbres de défaillances. Pour cela, plusieurs portes dynamiques, décrites dans la Table II.3, sont introduites ; elles peuvent être combinées avec des portes statiques [227]. Les quantifications se font sur la base des techniques d'analyse markovienne décrites précédemment.

Les premiers cas d'utilisation se sont concentrés sur des systèmes informatiques [225, 226]. Une littérature abondante, essentiellement issue de l'université de Virginie, caractérise les DFT. Plusieurs outils informatiques permettent la saisie et la quantification des modèles (*e.g. Galileo* [228] de l'université de Virginie, ou la solution commerciale de Relex). L'utilisation industrielle principale des DFT semble être l'aérospatiale [205, 229] : les difficultés des DFT à modéliser des systèmes réparables, spécifiquement discutées dans le Chapitre III Section 4.4, y ont en effet moins d'importance que dans d'autres contextes.

TABLE II.3 – Présentation des portes spécifiques aux DFT

Symbole	Nom et description
	Porte Séquence (B ne peut se réaliser qu'après A, la porte est vérifiée si A et B sont vérifiés)
	Porte de Dépendance Fonctionnelle (la réalisation de C déclenche la réalisation de A et B)
	Porte de Redondance à froid (*Cold Spare*) (A est l'événement de base principal, B l'événement de base redondant)

Réseaux de Petri (*Petri nets*). Les réseaux de Petri ont été inventés au début des années 60 par Carl Adam Petri [230] et ont fait l'objet depuis d'une abondante littérature et d'applications extrêmement variées, dépassant largement le cadre de la sûreté. Très puissants, ils permettent la spécification graphique d'interactions complexes entre les composants d'un système, telles que concurrences, conflits, synchronisations, boucles, exclusions mutuelles ou encore partages de ressources. En fait, il s'agit plus aujourd'hui d'une famille de formalismes ayant un certain nombre de caractéristiques communes que d'un formalisme unique précisément défini. Le nombre de déclinaisons et d'extensions est en effet très important (nous en avons dénombré plus d'une trentaine, nous n'en citerons ici que quelques unes). Ceci-dit, toutes s'appuient sur les notions de places, de transitions et de jetons dont la Figure II.7 donne les représentations graphiques. Les places, représentées par des cercles, modélisent au gré de l'analyste différentes évolutions ou conditions du système. Les transitions, représentées par des barres, correspondent à des événements susceptibles de faire évoluer l'état du système. Des arcs relient les places et les transitions. L'évolution de l'état du système est modélisée par la circulation de jetons entre les différentes places du réseau, au gré des transitions et des arcs. L'introduction de la notion de temps dans les réseaux de Petri est faite dès les années 70 [231], mais de façon déterministe. Les réseaux de Petri stochastiques (ou SPN, pour *Stochastic Petri Nets*), introduits au début des années 80 dans les travaux de thèse de Natkin [232] en France, et ceux de Molloy [233] aux États-Unis, associent les changements d'état du système à des durées aléatoires, modélisées par des processus stochastiques. Dans les SPN, seules des distributions exponentielles sont considérées. Le SPN est alors équivalent à un graphe de Markov en temps continu, et toutes les évaluations classiquement offertes par les modèles markoviens sont permises. Les GSPN (*Generalized SPN*) [234] ajoutent aux transitions exponentielles, dites temporisées, des transitions probabilisées de durée nulle, dites immédiates. On les différencie graphiquement par un trait plein au lieu du trait « creux » alors utilisé pour les transitions temporisées, comme illustré dans la Figure II.8. Le processus global défini par le réseau de Petri reste markovien, et là encore, toutes les techniques classiques s'appliquent. SPN et GSPN définissent de façon compacte des graphes de Markov potentiellement très grands : revers de la médaille, leur quantification est rapidement confrontée à des problèmes d'explosion combinatoire [176]. Citons enfin les SAN (*Stochastic Activity Networks*) [235], qui peuvent être considérés comme une forme étendue des réseaux de Petri, substituant les transitions par des composants nommés activités, portes d'entrée et portes de sortie, largement paramétrables et élargissant encore les capacités de modélisation.

Une des premières utilisations des réseaux de Petri en sûreté est proposée par Leveson et Stolzy en 1987 [236]. Ils sont depuis employés dans ce cadre à divers titres, notamment pour la modélisation de systèmes de contrôle-commande [237]. Leur puissance et leur polyvalence ont stimulé le développement de nombreux supports logiciels pour leur saisie et leur traitement (on citera dans le domaine universitaire l'outil *SPNP* de l'université de Duke, *GreatSPN* du Politecnico de Turin, *TimeNet* de l'université de Hambourg, ou encore pour les SAN, *Möbius* de l'université d'Illinois). Notons enfin que l'université de Hambourg maintient une page *web* extrêmement complète sur tous les sujets attenant aux réseaux de Petri [12], et une bibliographie d'environ 8500 références sur le thème.

12. http://www.informatik.uni-hamburg.de/TGI/PetriNets/

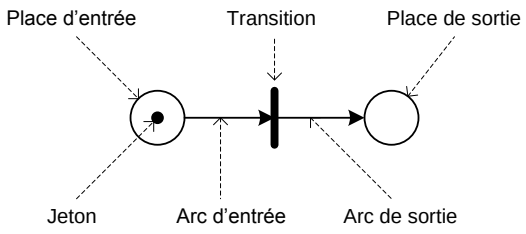


FIGURE II.7 – Composants d'un réseau de Petri

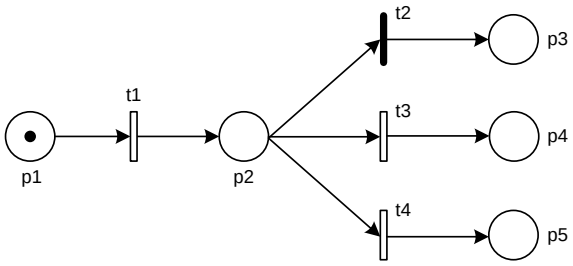


FIGURE II.8 – Exemple de réseau de Petri

Les BDMP (*Boolean logic Driven Markov Processes*). Les BDMP ont été introduits par Bouissou à EDF au début des années 2000 [238]. Ils combinent l'aspect visuel et la lisibilité des arbres de défaillances avec des capacités de modélisation propres aux chaînes de Markov. Ils se différencient des DFT par une sémantique plus lisible et une formalisation mathématique plus rigoureuse. Une présentation complète des BDMP est donnée dans le Chapitre III (ainsi qu'une comparaison argumentée avec les DFT et les réseaux de Petri). Leur support logiciel y est également évoqué.

2.2.5 Les langages de modélisation probabiliste

Les modèles jusqu'ici présentés sont construits « à la main » et saisis pour quantification dans des outils logiciels adaptés : d'une part, cette construction demande beaucoup de temps et d'expertise, d'autre part elle est peu réutilisable. Pour lever ces limites, des langages de modélisation probabiliste ont été spécialement développés, permettant de spécifier un système de telle sorte à pouvoir instancier automatiquement des modèles fiabilistes comme ceux précédemment décrits (arbres de défaillances, réseaux de Petri, graphes de Markov, etc.) et passer selon les cas d'un formalisme à l'autre par simple transformation. Les deux principaux représentants en France sont *Figaro* [239] et *AltaRica* [240]. Figaro, créé au début des années 90 à EDF est plus général qu'AltaRica [241], créé plus tardivement au Labri [13]. Les deux sont utilisés industriellement, essentiellement dans le domaine électrique (production et réseaux) pour le premier et dans l'aéronautique pour le second, et sont outillés de façon complète et performante [242, 243, 244]. La réf. [241] donne une comparaison détaillée des deux langages.

Notons qu'il existe aussi des langages de modélisation plus généraux, qui peuvent être déclinés à des fins de modélisation de sûreté de fonctionnement. Dans le domaine logiciel, c'est par exemple le cas du langage d'analyse et de conception d'architectures AADL [14] [245] employé principalement dans l'avionique et l'industrie automobile pour la conception de systèmes embarqués. Rugina donne dans sa thèse une bonne introduction à AADL et présente comment il peut être utilisé en sûreté de fonctionnement, en le couplant avec des réseaux de Petri [246]. Cependant, si l'utilisation de langages de modélisation plus génériques permet d'élargir la communauté utilisatrice, elle se fait souvent au détriment de la souplesse et de la facilité d'utilisation des langages plus spécialisés.

2.2.6 Tableau récapitulatif

La Table II.4 donne dans les pages suivantes une vision synthétique des différents outils et techniques d'évaluation présentés pour le domaine de la sûreté.

13. Laboratoire Bordelais de Recherche en Informatique (Labri).
14. *Architecture Analysis & Design Language* (AADL, `http://www.aadl.info`).

TABLE II.4: Synthèse des techniques d'évaluation de sûreté présentées

Technique	**Utilisation**	**Description**	**Résultats qualitatifs**	**Résultats quantitatifs**	**Limites**	**Réf.**	**Origines**
Analyse Préliminaire de Danger	Généralisée. Stade préliminaire de conception.	Méthode d'identification simple de type inductif à base de tableaux et de *check-lists*.	Liste partielle de dangers et de risques, recommandations.	Non systématique (selon type d'analyse).	Nécessite d'être approfondie par d'autres méthodes (AMDEC, HAZOP, etc.). Pas de défaillances multiples.	[188]	Années 60, sûreté des missiles aux États-Unis.
AMDE(C)	Généralisée (et plus particulièrement automobile, aéronautique, ferroviaire, logiciel).	Utilisation de tableaux sur un mode qualitatif et inductif, déclinés selon le secteur.	Liste des défaillances. Caractérisation des conséquences (et de leur gravité pour l'AMDEC).	Non systématique (calcul de taux de défaillance et de gravité possible).	Pas de défaillances multiples. Fastidieux pour les systèmes comportant de nombreux éléments.	IEC 60812 [189]	Années 60 aux États-Unis dans le domaine de l'aéronautique.
HAZOP	Analyse de systèmes thermo-hydrauliques (*e.g.*pétrochimique). Déclinée pour le logiciel, mais moins répandue. Orientée conception, utilisable aussi sur existant.	Méthode systématique et structurée de type inductif. Utilise des mots-guides codifiés et des tableaux.	Identification de dangers par écart vis-à-vis d'un comportement attendu. Conséquences et recommandations aussi abordées.	Non systématique (dépend de la version et de l'utilisation).	Pas de défaillances multiples. Fastidieux.	IEC 61882 [192] [191, 194]	Années 60, industrie chimique.
Analyse des Conditions Insidieuses Analyse transitoire (*Sneak analysis*)	Secteurs spatial, militaire et nucléaire, surtout aux États-Unis. Conception de systèmes électriques, mais généralisée à d'autres applications.	Modélisation des relations entre entrées et sorties du système par réseaux arborescents et forêts. Intégrative (s'articule avec HAZOP, AMDEC, etc.).	Identification de chemins imprévus pouvant provoquer un comportement indésirable.	Non.	Expertise nécessaire. Forme très dépendante du domaine d'application.	[196, 197] (Spatial) [198] (Militaire) [199] (Nucléaire)	Années 60, aéronautique et spatial (programme *Apollo*) aux États-Unis.
Zonal Hazard Analysis (ZHA) / Zonal Safety Analysis (ZSA)	Aéronautique, nucléaire, chimie.	Analyse de la répartition physique des composants d'un système.	Identification de dépendances entre systèmes jugés *a priori* indépendants.	Non.	Approche peu documentée et changeante selon le domaine.	[200]	Industrie aéronautique.

Conformité à des référentiels (notamment IEC 61508, DO-178B)	Logiciel embarqué critique (transverse).	Exigences sur les différentes étapes du cycle de vie logiciel. Plus ou moins prescriptives. La norme clé du domaine, l'IEC 61508 est très prescriptive. D'autres normes sectorielles ont des approches plus souples.	Certifications. Plusieurs niveaux sont généralement distingués.	Selon les référentiels et les niveaux de certification.	Domaine en cours de consolidation, la 61508 jouant un rôle pivot.	IEC 61508 [110] (générique) et déclinaisons DO-178B [76] (avionique)	61508 issue de la convergence de l'ISA 84.01 (USA, 1996), du Def-Stan 00-55 (UK, 1991) et de la DIN 19250 (Allemagne, 1994).
Diagrammes de fiabilité (*Reliability Block Diagrams*)	Tout domaine. Réservés aux systèmes simples.	Blocs fonctionnels connectés en série ou en parallèle selon leur nécessité pour le système.	Combinaisons de pannes.	Calcul de fiabilité, de disponibilité. Analyse de sensibilité (contribution des composants).	Systèmes simples. Modèles statiques. Événements indépendants et binaires.	IEC 61078 [201]	
Arbres de défaillance	Emploi généralisé. Très implantés dans le nucléaire et le spatial. Adaptation au logiciel (plutôt académique).	Approche déductive, sous forme d'un arbre booléen.	Combinaisons de pannes (coupes minimales)	Calcul de fiabilité, de disponibilité. Analyse de sensibilité (contribution des composants).	Événements indépendants. Modèles statiques. Fonctionnement des composants « binaire ».	IEC 61025 [207] NRC Hdbk [204] NASA Hdbk [205]	Années 60, sûreté des missiles aux États-Unis
Arbres d'événements	Surtout nucléaire et industrie chimique.	Approche inductive. Arbre avec embranchements probabilisés.	Identification des configurations accidentelles menant à des conséquences redoutées.	Quantification des configurations accidentelles. Probabilité de conséquences.	Evénements indépendants. Modèles statiques. ambiguïté sémantique.	Guide AIChe [211]	Années 70, rapport WASH 1400/ Rasmussen [209]
Diagrammes Cause-Conséquence	Surtout nucléaire (marginalement logiciel et ferroviaire).	Approche hybride entre arbres de défaillances et arbres d'événements.	Combinaisons de pannes et configurations accidentelles.	Cf. arbres de défaillances et arbres d'événements.	Cf. arbres de défaillances et arbres d'événements.	[212]	Années 70, Risø Labs [212] (sûreté nucléaire).
Réseaux bayésiens	Encore académiques en fiabilité (développement industriel lancé). Très polyvalents : analyse de rex, prévision, détection d'écarts, diagnostic, optimisation...	Graphes orientés sans circuit de relations de probabilités conditionnelles entre variables aléatoires ; tables de probabilités conditionnelles associées.	Peu adaptés aux considérations qualitatives : on perd notamment les notions de coupes minimales propres aux arbres de défaillances.	Calcul de loi marginale d'une variable, de sa loi conditionnelle vis-à-vis d'un ensemble d'observations. Force des liens, importance des nœuds.	Modèles statiques. Lisibilité limitée (beaucoup d'information non représentée). Diversité des traitements réduite.	[221]	Travaux de Thomas Bayes (XVIII^{e} s.) Premières utilisations en fiabilité années 90 [218].

Graphes et techniques de Markov	Tout domaine. Utilisés pour modéliser des systèmes avec dépendances et caractéristiques dynamiques.	Graphe d'états. Modélisation des transitions par lois exponentielles (sans-mémoire).	Séquences de pannes.	Calcul de fiabilité et de disponibilité. Caractérisation des séquences menant à la panne.	Explosion combinatoire. Expertise forte nécessaire.	IEC 61165 [223] [188]	Travaux d'A. Markov (début XX^{e} s.)
Dynamic Fault Trees (DFT)	En principe génériques. Encore très académiques ; utilisation plus industrielle dans l'aérospatiale.	Arbres de défaillances avec portes spécifiques (FDED, WSP, PDEP). Spécification de haut-niveau d'un modèle de Markov.	Séquences de pannes.	Calcul de fiabilité et de disponibilité.	Fortes limites avec systèmes réparables. Formalisation mathématique incomplète. Lisibilité des portes.	NASA Hdbk [205] [226]	Début 90. Modélisation de systèmes informatiques embarqués.
Réseaux de Petri	Génériques. Les plus utilisés en fiabilité sont les GSPN, mais il existe de nombreuses variantes (dont SAN, réseaux de Petri colorés, etc.)	Modélisation graphique d'interactions complexes (concurrences, conflits, synchronisations, boucles, partages...) à l'aide de places, arcs, transitions et jetons. Spécification de haut-niveau de modèles de Markov.	Séquences de pannes.	Calcul de fiabilité et de disponibilité.	Diversité des formalismes. Explosion combinatoire. Expertise forte nécessaire.	[247]	Thèse de C. A. Petri en 1962 [230]
BDMP	Industrie électrique, mais potentiellement génériques.	Formalisme proche des arbres de défaillances, avec capacités de modélisation dynamique.	Séquences de pannes.	Calcul de fiabilité et de disponibilité.	Communauté utilisatrice encore restreinte. Difficulté à modéliser des comportements cycliques.	[238]	EDF, début des années 2000.
Langages de modélisation fiabiliste	Génériques.	Langages dédiés à la spécification de modèle.	Transformations entre différents formalismes.	Large éventail de possibilités.	Paysage fragmenté, mais passerelles entre les langages.	–	Précurseur : Figaro [239]

2.3 Quelques techniques pour l'évaluation en sécurité informatique

La « boîte à outils » de l'évaluation de sécurité informatique est plus jeune et moins fournie que celle de la sûreté. Plusieurs raisons peuvent l'expliquer. Tout d'abord, la problématique de la sécurité informatique n'est identifiée qu'à partir de la fin des années 60, quelques temps après les premières expérimentations de systèmes multi-utilisateurs et de réseaux informatiques, amenant des problèmes de confidentialité pour le traitement de données sensibles [248]. Elle ne prend cependant toute son importance qu'avec le développement d'Internet, révolutionné par le *Web* au début des années 90. Les premières approches méthodologiques structurées en sûreté peuvent quant à elles être retracées dès l'entre deux guerres, notamment dans le domaine de l'aéronautique (*e.g.* étude comparée de la sûreté des avions mono, bi ou quadrimoteurs [188]). Alors que la sécurité informatique n'en est qu'à ses balbutiements dans les années 80, les études de sûreté sont déjà bien structurées, outillées (AMDE, arbres de défaillances, HAZOP, etc.) et largement pratiquées dans les principales industries à risques. Autre aspect expliquant la moindre maturité de la boîte à outil sécurité informatique : les systèmes à évaluer mais aussi les menaces à considérer évoluent à une vitesse incomparablement plus rapide qu'en sûreté. La tâche des techniques d'évaluation est plus délicate, leur « *turn-over* » est aussi plus important. La sélection proposée ici, et synthétisée par la Table II.8 p. 56, donne un aperçu représentatif des techniques actuelles, mais également de leurs limites et difficultés associées.

En particulier, l'évaluation quantitative de la sécurité est unanimement reconnue comme une problématique ardue et encore largement ouverte [249]. La question de la pertinence des approches probabilistes, déjà abordée en Section 1.2.4, est débattue plus largement au Chapitre III Section 6.1. D'une façon plus générale, les problématiques d'évaluation quantitatives de la sécurité se rattachent directement aux multiples débats et recherches portant sur la notion de mesure ou métrique [15] de sécurité. De nombreuses initiatives et approches jalonnent cette thématique d'investigation, que nous ne pourrons qu'effleurer au travers de certains passages de la présente section. Jaquith donne un panorama bien plus complet dans [250] ; des vues synthétiques et complémentaires peuvent de plus être trouvées dans [251, 252, 253]. Enfin, le NIST a récemment identifié les grandes directions de recherche en la matière dans [254], alors que les premiers standards reflétant des consensus nationaux ou internationaux font leur apparition (*e.g.* [255, 256]).

Comme pour le précédent panorama, les techniques d'évaluation ici présentées ont été retenues sur la base de leur visibilité dans la littérature normative, scientifique et technique, mais aussi pour servir la Section 3 et les chapitres suivants du mémoire.

2.3.1 Référentiels de conformité et certifications

Également regroupées dans la littérature en « approches par critères » [257], ces techniques s'appuient sur une description précise d'exigences définies *a priori*, dont des critères de satisfaction sont explicitement décrits et généralement regroupés selon le niveau de sécurité visé par l'évaluation. Quand les critères sont évalués par des tierces parties spécialisées, la notion de certification y est fréquemment associée. L'approche par critère a un riche historique en matière de sécurité informatique. Elle est dominée aujourd'hui par les Critères Communs (CC) [258], norme mondiale de référence pour l'évaluation des produits de sécurité. Ils permettent de valider de façon modulaire des fonctions de sécurité d'un produit selon différents degrés d'assurance, les EAL (*Evaluation Assurance Levels*), échelonnés de 1 à 7. L'Annexe A donne un historique et une description rapide de cette norme.

En parallèle, d'autres référentiels plus sectoriels ont vu le jour : on peut citer les certifications VISA ou EMV bien établies dans le domaine bancaire. Plus récemment, des schémas de certification pour la sécurité des systèmes d'informatique industrielle ont fait leur apparition (*e.g.* ISA Secure [259], décrit en Annexe A ou Achille [16]). Encore balbutiantes, il est trop tôt pour juger de leur pertinence [260].

Enfin, on ne peut clôturer cette section sans évoquer les normes ISO/IEC 27001 [115] et 27002 [117], fréquemment citées dans la littérature comme référentiels de sécurité informatique. Le périmètre n'est plus un produit ou une fonction de sécurité, mais une entreprise, une administration ou tout autre type d'organisation. En fait seule l'ISO/IEC 27001, définissant des exigences pour la gestion de la sécurité de l'information, est à proprement parler « certifiable ». L'IEC/ISO 27002 constitue quant à elle un catalogue structuré de contrôles de sécurité que les organisations peuvent sélectionner et mettre en œuvre d'un point de vue technique et organisationnel, articulés par le jeu de politiques de sécurité et de procédures

15. Nous emploierons ces termes indistinctement.
16. http://www.wurldtech.com/cyber-security/achilles-certification/achilles-certification.aspx

appropriées. L'ISO/IEC 27002 n'est donc pas un référentiel d'évaluation, bien qu'il soit fréquent de la voir présentée comme tel. Toutes deux s'insèrent dans une série de standards ISO/IEC dite 27k (toutes les normes de la série sont numérotées en 27xxx) dont la plupart complètent ou précisent les notions et mécanismes évoqués dans l'ISO/IEC 27001.

2.3.2 Modélisation et évaluation formelle de politiques de sécurité

Un modèle formel de politique sécurité donne un cadre dans lequel on peut exprimer de façon formelle et vérifiable les exigences définies dans leurs principes par une politique de sécurité : dans le cadre du modèle formel considéré, les exigences de la politique correspondent à de propriétés mathématiquement prouvables. Les modélisations formelles de politique de contrôle d'accès interviennent dans les évaluations de sécurité de systèmes très sensibles, et ont connu de nombreux développements. Elles constituent la partie la plus visible et la plus mise en œuvre des travaux de modélisation formelle de politique sécurité.

Nous donnons en Annexe B un aperçu des principaux jalons en la matière, dont les modèles classiques tels que Bell-Lapadula ou Biba, les modèles à base de matrice de contrôle d'accès ou de rôles. La même annexe évoque aussi les modèles d'autorisation à base de flux ou de traces d'exécution (notamment la *non-interference*), et les modèles utilisés en vérification de primitives cryptographiques.

Par extension, les techniques de vérification formelle de protocoles de sécurité, et plus largement de *model-checking* à des fins de sécurité, peuvent être rattachées à cette section. La thèse de Foster [261] et la référence [262] donnent de bons panoramas du premier aspect, tandis qu'un état de l'art pourra être par exemple trouvé dans la thèse de Sheyner [263].

2.3.3 Méthodes qualitatives structurées d'analyse et de gestion de risque

Il existe de très nombreuses méthodes d'analyse de risques en sécurité informatique, ayant chacune leurs spécificités en termes d'historique, de domaine d'application, d'approche et de périmètre. On citera :

- au niveau français, EBIOS [17] et MÉHARI [18]. La première, développée et promue par l'ANSSI [19], a vu le jour en 1995. Elle est la méthode privilégiée dans les administrations et ministères français, mais est également très employée dans les grandes entreprises. Elle fait l'objet d'un club utilisateur vivace [20]. Elle est complètement documentée et outillée [264, 265], et couvre tous les aspects de la gestion de risques. La seconde, MEHARI, est développée et promue depuis 1995 par le CLUSIF [21]. Elle est employée essentiellement dans les grandes entreprises françaises, où elle côtoie EBIOS. Cette dernière a un développement international plus poussé ;

- au Royaume-Uni, CRAMM [22]. Créée en 1985 par le gouvernement britannique, la méthode est maintenant développée et promue par le secteur privé (Siemens notamment). Très intégrée avec l'outil logiciel éponyme, elle reste la méthode d'analyse et de gestion de risques la plus répandue dans les administrations et ministères britanniques, mais est également employée dans des organisations internationales (*e.g.* OTAN) et par les entreprises du secteur privé dans plusieurs pays européens ;

- aux États-Unis, la SP800-53 [111], MORDA [23] [266], OCTAVE [24] ou encore SQUARE [25] [267]. Les organisations fédérales s'appuient sur la première, développée par le NIST. La seconde est une des méthodes développées par la NSA. La troisième et la quatrième sont plus ciblées vers le secteur privé : OCTAVE, issue des travaux de l'université Carnegie Mellon, est très portée sur les aspects organisationnels et déclinée selon la taille de l'organisation analysée ; SQUARE est plus orientée processus de conception d'architecture sécurisée.

- Au niveau international enfin, l'ISO/IEC 27005 [118] a été publiée en juin 2008 et a déjà acquis une considérable popularité de par ses principes simples et flexibles. Contrairement aux autres méthodes citées, elle reste à un niveau très générique et n'est donc pas spécifiquement outillée.

17. Expression des Besoins et Identification des Objectifs de Sécurité (EBIOS).
18. MÉthode Harmonisée d'Analyse de Risques (MEHARI).
19. Agence Nationale de la Sécurité des Systèmes d'Information (ANSSI, http://www.ssi.gouv.fr).
20. http://www.club-ebios.org/
21. CLUb de la Sécurité des systèmes d'Information Français (CLUSIF, http://www.clusif.asso.fr).
22. Ccta (*central communication and telecommunication agency) Risk Analysis and Management Method* (CRAMM).
23. *Mission Oriented Risk and Design Analysis* (MORDA).
24. *Operationally Critical Threat, Asset, and Vulnerability Evaluation* (OCTAVE, http://www.cert.org/octave).
25. *Security QUality Requirements Engineering* (SQUARE, http://www.cert.org/sse/square.html).

Le lecteur soucieux d'avoir une vue plus complète pourra consulter la page *web* dédiée au sujet [26] maintenue à jour par l'Agence européenne de la sécurité des réseaux et de l'information (ENISA [27]). Elle présente de façon structurée les spécificités de 13 méthodes de gestion et d'analyse de risques, ainsi que de 22 outils informatiques pouvant les appuyer de façon dédiée ou générique. Pour chaque méthode, 21 attributs sont renseignés, précisant notamment le type d'organisation visée, le domaine de la gestion de risques couvert (cf. Fig. II.1), le niveau de détail et de maturité de la méthode, etc. Cette page permet également de comparer outils et méthodes les uns par rapport aux autres sur la base des attributs les décrivant.

2.3.4 Evaluation de la vulnérabilité et tests de sécurité

Tests d'intrusion. Si les évaluations de la sécurité en conception permettent une caractérisation « théorique » du niveau de sécurité, elles peuvent être complétées par des tests de nature opérationnelle, type tests de pénétration et d'intrusion. Ces tests peuvent se faire à différents périmètres : sur un composant spécifique, à l'échelle d'une architecture voire d'une organisation. Ils nécessitent l'intervention encadrée de spécialistes de ces évaluations, qui opèrent selon différentes modalités. L'OSSTMM [28] [268] distingue six types d'approche, résumés par la Table II.5 et illustrés par la Figure II.9.

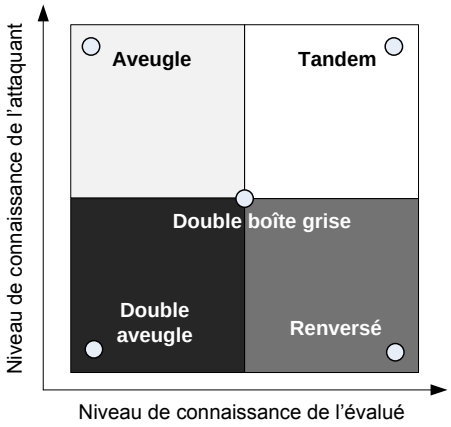


FIGURE II.9 – Positionnement respectif des six types d'approches de test d'intrusion selon [268]

Les tentatives d'intrusion peuvent exploiter différents « canaux » : l'OSSTMM différencie ainsi canal physique, canal organisationnel et humain, ondes radio, réseaux de données et réseaux de télécommunications. Il existe une offre importante de sociétés de services spécialisées dans les tests d'intrusion, et une littérature abondante sur le sujet (cf. par exemple le guide méthodologique de l'OWASP [29] [269] pour les applications *Web*). Notons enfin que les tests de pénétration ne doivent être considérés que comme un moyen parmi d'autres dans une démarche d'audit de sécurité plus complète [115].

Scanners automatisés de vulnérabilités. Les vulnérabilités des composants ou architectures informatiques peuvent être partiellement évaluées de façon automatisée. Depuis leur introduction au début des années 90 [270, 271], l'utilisation de « scanners » de vulnérabilités est une pratique répandue, avec des suites logicielles *open source* et commerciales aujourd'hui performantes (*e.g. Nessus*, *CORE Impact* ; *Achille* pour un dispositif spécifique à l'informatique industrielle). Elles testent la présence de vulnérabilités existantes. Pour trouver de nouvelles vulnérabilités logicielles, les techniques dites de *fuzzing* ont connu un essor important ces dernières années, bien qu'elles demandent une expertise extrêmement pointue. Plus globalement, le document du NIST [272] donne une liste d'outils de test sécurité pour les réseaux.

26. http://www.enisa.europa.eu/act/rm/cr/risk-management-inventory
27. *European Network and Information Security Agency* (ENISA, http://www.enisa.europa.eu).
28. *Open Source Security Testing Methodology Manual* (OSSTMM, http://www.isecom.org/osstmm).
29. *Open Web Application Security Project* (OWASP, http://www.owasp.org).

TABLE II.5 – Description des six approches de test d'intrusion de l'OSSTMM [268]

Désignation	Description
Aveugle	L'auditeur n'a aucune connaissance sur le système évalué et ses protections. Le responsable du système connaît les détails de l'audit, cadre, délais, y compris modes opératoires (canaux exploités, techniques d'attaque, etc.).
Double aveugle (boîte noire)	L'auditeur n'a aucune connaissance sur le système évalué. Le responsable du système ne connaît pas le périmètre et les modalités précises du test de pénétration (canaux exploités, techniques d'attaque, etc.).
Boîte grise (test de vulnérabilité)	L'auditeur a une connaissance limitée du système évalué et de ses protections. Le responsable du système est au courant des détails de l'audit, y compris de ses modes opératoires (canaux exploités, techniques d'attaque, etc.).
Double boîte grise	L'auditeur a une connaissance limitée du système évalué et de ses protections. Le responsable du système est notifié du périmètre de l'audit, mais en ignore les modalités pratiques (canaux exploités, techniques d'attaque, etc.).
Tandem	Auditeur et responsable du système évalué ont une connaissance complète.
Renversé	L'auditeur a une connaissance complète du système évalué et de ses protections. Le responsable du système ne connaît pas le périmètre et les modalités précises du test (canaux exploités, techniques d'attaque, etc.).

Métriques de la vulnérabilité. Les remarques faites en introduction de la Section 2.3 quant à la notion de métrique de sécurité restent valables pour le cas particulier de la quantification du niveau de vulnérabilité : le domaine est à la fois ardu et foisonnant. Nous ne présenterons ici que deux approches, de natures différentes, mais bénéficiant toutes deux d'une large visibilité industrielle. Les lecteurs intéressés pourront consulter les références indiquées dans l'introduction susmentionnée.

Tout d'abord la notion de surface d'attaque (*attack surface*) a été introduite par Microsoft [273] et développée par l'université Carnegie Mellon [274] pour mesurer la sécurité d'un produit logiciel donné (*e.g.* un type de système d'exploitation, un progiciel, un système de gestion de base de données, etc.). Plutôt que de s'intéresser au nombre de bugs trouvés dans un code source, ou au nombre d'avis de sécurité émis pour le système considéré, la notion de surface d'attaque vise à refléter l'expertise requise et l'opportunité pour un attaquant de compromettre avec succès un système en prenant en compte la présence et la facilité d'exploitation de vecteurs d'attaque types. Elle s'applique sur un mode comparatif, sur des systèmes de même genre.

Dans un autre registre, le CVSS (*Common Vulnerability Scoring System*) [275] est un système de notation, non plus de la vulnérabilité d'un système au sens général, mais d'une vulnérabilité logicielle en particulier, *i.e.* d'une faille caractérisée et exploitable par un attaquant. Le CVSS est développé et promu par le NIST, il est mis en œuvre par de nombreuses organisations au niveau mondial [30]. Il note une vulnérabilité selon ses caractéristiques intrinsèques (*e.g.* possibilité d'exploitation à distance, type d'impact), temporelles (*e.g.* temps nécessaire à son exploitation) et environnementales (*e.g.* nombre de systèmes concernés, possibilité de dommages collatéraux en cas d'exploitation).

2.3.5 Modélisation graphique d'attaque

Les techniques de modélisation graphique d'attaque permettent de représenter graphiquement et d'appuyer l'analyse des différentes possibilités et scénarios envisageables par un attaquant pour atteindre un ou plusieurs objectifs. Elles peuvent appuyer la préparation de tests d'intrusion et différentes étapes d'une analyse de risques en sécurité. Ces modèles graphiques permettent de formaliser et d'améliorer la couverture de l'analyse, partager et communiquer sur les scénarios considérés, mais également dans certains cas, fournir des éléments quantitatifs. Ils sont souvent employés pour étayer l'évaluation des probabilités d'occurrence dans la phase d'estimation des risques (cf. Section 1.1.1). Ils peuvent aussi prendre en compte des aspects défensifs, dans le but d'évaluer l'efficacité de contre-mesures. Étant donné la pertinence de cette catégorie d'outils pour la suite de nos développements (notamment ceux du Chapitre III), nous proposons ici un état de l'art plus approfondi que pour les autres techniques. La Table II.7 p. 54 en donne une vue synthétique.

30. http://www.first.org/cvss/

2.3.5.1 Deux approches dominantes : arbres d'attaque et réseaux de Petri. Les premières représentations graphiques d'attaques informatiques peuvent être identifiées dès les années 80 [276, 277], avec les approches à base d'arbres de menaces, puis sont rejointes par des modélisations à base de réseaux de Petri, au début des années 90 [278, 279]. Ces deux types d'approche sont encore à ce jour les modèles dominants du domaine, le plus répandu étant celui des arbres d'attaque.

Arbres d'attaque (*attack trees*) et arbres de menaces (*threat trees*). Inspirés des arbres de défaillances du domaine de la sûreté de fonctionnement dans les années 80, ils sont popularisés par Schneier à la fin des années 90 [22]. Les premières utilisations d'arbres pour modéliser les menaces peuvent être situées dans les années 80. Le logiciel Lava développé dès 1983 par le laboratoire de Los Alamos aux États-Unis y recourt de façon sous-jacente pour modéliser les menaces pesant sur les installations nucléaires [280]. Par ailleurs, Weiss [276] et Amoroso [277] signalent tous deux le requis d'analyse par arbres de menaces (*threat trees*) du standard MIL-Std-1785 du DoD [281] [31]. Weiss, mentionnant explicitement l'influence des arbres de défaillances, décrit dans [276] l'élaboration et l'utilisation d'un arbre de menaces, décomposant dans un arbre booléen une menace macroscopique en menaces élémentaires. Il explique aussi comment calculer avec une telle structure des niveaux de risque et prendre des décisions de sécurité en attribuant aux feuilles de l'arbre des impacts et des efforts pour l'attaquant. En 1994, Amoroso expose un formalisme très similaire [277], reconnaissant l'influence de Weiss et du standard du DoD. Y sont représentées dans un arbre logique les menaces groupées en termes d'atteintes à la confidentialité, à la disponibilité et à l'intégrité, permettant leur énumération structurée, mais également comme chez Weiss, l'évaluation et la hiérarchisation des risques. L'ensemble est là aussi désigné sous le nom d'arbre de menaces ; la Figure II.10 donne un exemple repris de Amoroso, qui décrit les menaces pesant sur des données médicales dans un système informatique hospitalier. Cette technique a depuis été exposée dans différents ouvrages de référence en sécurité [158, 157, 282].

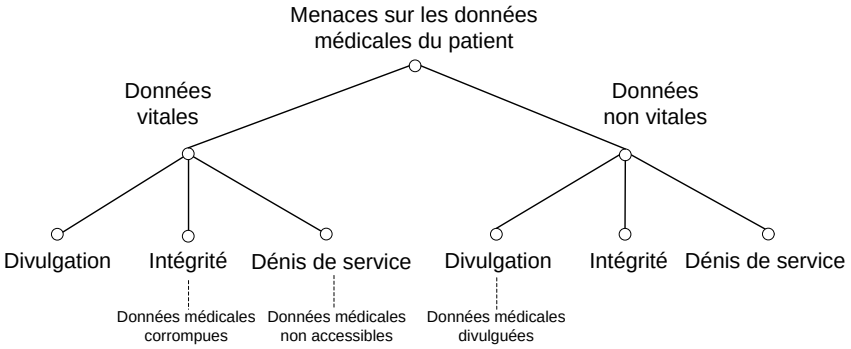


FIGURE II.10 – Exemples d'arbre de menaces, d'après Amoroso [277]

La dénomination « arbres d'attaque » et le modèle associé sont dus à Schneier ; ils correspondent à la forme dérivée sans doute la plus connue des arbres de défaillances en sécurité, et ont été popularisés par Schneier essentiellement en trois temps : d'abord en tant que co-auteur d'un article de conférence synthétisant les réflexions d'un groupe de travail de la NSA [283] (1998) ; un an après, par un article de revue reprenant et développant la méthode (et souvent cité à tort comme l'article fondateur) [22] ; enfin par un chapitre dédié dans un livre publié en 2000 [156]. Schneier modélise sous forme d'arbre booléen les différentes techniques et moyens envisageables par un attaquant pour atteindre son objectif, sommet de l'arbre (dans la terminologie consacrée). Certains moyens doivent être employés conjointement et sont groupés par un ET logique, d'autres sont des alternatives et sont groupés par un OU. Des objectifs intermédiaires jalonnent la progression vers l'objectif final. On le voit, la différence entre les arbres d'attaque et les arbres de menaces est mince, au point de pouvoir être confondus. Elle tient, en principe, à la nature différente des éléments modélisés par les feuilles et les nœuds de l'arbre dans chaque cas ; en pratique, la frontière s'estompe car les éléments modélisés sont souvent très hétérogènes dans un même modèle et sans cohérence particulière avec l'appellation.

31. Le standard est indiqué comme étant de 1988 : nos recherches bibliographiques n'ont pu aboutir qu'à un document du 1er janvier 1989, de même référence et de même titre, mais ne mentionnant pas les arbres de menaces.

À l'instar de Weiss et Amoroso, Schneier propose d'associer aux feuilles différents attributs. L'attribution de coûts permet de trier les scénarios sur ce critère, et de raisonner selon le profil estimé de l'attaquant. Des seuils et des indicateurs de compétence ou de moyens nécessaires (*e.g.* outillage spécifique) peuvent aussi être définis à cette fin. La thèse d'Edge donne une description complète de l'utilisation de ce genre de paramètres [284]. L'attribution de probabilités de succès, estimées par l'analyste, permet de faire des quantifications de coupes minimales comme sur les arbres de défaillances. Enfin, un formalisme alternatif sous forme de texte est proposé. La Figure II.11 reprend un des exemples, désormais canonique, décrit par Schneier dans [22]. Il représente de façon simplifiée l'attaque d'un coffre-fort, soulignant au passage l'application naturelle du formalisme au domaine de la sécurité physique.

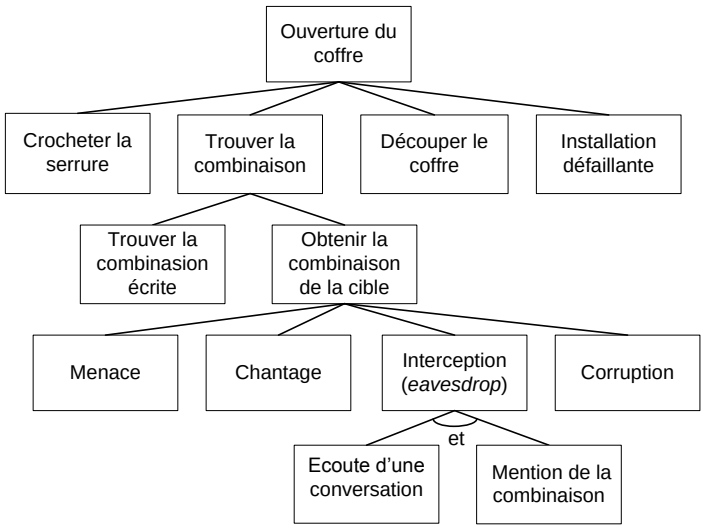


FIGURE II.11 – Exemple canonique d'arbre d'attaque, traduit de [22]

En 2001, dans le sillage de Schneier, Moore *et al.* adoptent un formalisme légèrement différent et développent encore la méthode [285] : ils mettent notamment en avant la possibilité d'utiliser les arbres d'attaque pour modéliser des motifs (*patterns*) élémentaires d'attaque réutilisables. Ces motifs peuvent constituer une bibliothèque à la disposition de l'analyste ; leur sélection peut être automatisée selon le système considéré. D'autres reprendront l'idée et proposeront diverses représentations structurées (*e.g.* en grammaire BNF (Backus-Naur Form) [286], ou en XML [287]). Une fois structurés, les arbres d'attaque peuvent alors aussi appuyer des systèmes de détection d'intrusions en formalisant les schémas d'attaque à détecter [286, 288, 289, 290]. En 2006, Fovino *et al.* proposent une notation semi-formelle permettant d'améliorer la sémantique des arbres d'attaque, leur composition, et leur intégration dans les méthodes d'analyse de risques en distinguant trois type de nœuds : opérations, vulnérabilités et assertions [291].

Les arbres d'attaque ont été utilisés dans de très nombreux secteurs et applications. On peut citer de façon non-exhaustive l'analyse de la sécurité des protocoles (*e.g.* Modbus [292] ou BGP [293] ; cf. la thèse de Foster pour une description plus poussée et une application plus générale [261]), mais aussi les systèmes de banque en ligne [294], de vote [295], de tatouage électronique [296], de drone [297] ou encore le domaine des réseaux *ad-hoc* [298]. Ils ont aussi été employés dans la modélisation d'attaques et l'évaluation de risques pour des systèmes en lien direct avec le contexte de cette thèse : des systèmes de sûreté de centrales nucléaires [299, 300], des systèmes SCADA [32] [301, 302], des systèmes de compteurs électriques intelligents [303] et plus largement des systèmes de métrologie [304].

Plusieurs méthodes génériques d'analyse de risques ou de spécification de systèmes suggèrent d'y faire appel dans les phases d'identification des menaces et de modélisation de scénarios. C'est le cas notamment de SQUARE et MORDA [305], évoquées en Section 2.3.3, ou encore de Strata-Gem, présentée en 2008 par Clark *et al.* [306]. Le récent guide de la NRC sur la sécurité informatique des centrales nucléaires [60] y fait aussi une rapide allusion. À côté de leur utilisation directe dans des analyses de risques, ils ont également été considérés comme modèles sous-jacents à des applications comme la détection

32. *Supervisory Control And Data Acquisition* (SCADA).

d'intrusions [288], la détection d'attaques d'origine interne (*internal threat*) en particulier [307], l'étude *post-mortem* (*forensics*)[308], ou encore l'identification et l'évaluation des problèmes de protection de données personnelles (*privacy)* [309].

Les arbres d'attaque sont parfois désignés par d'autres appellations selon les aspects de la modélisation mis en avant : en fait, on peut considérer que, à quelques nuances près, *attack trees*, *threat trees*, comme déjà signalé, mais aussi *vulnerability trees*, *security trees*, ou encore *protection trees* et *defense trees*, évoqués ultérieurement, correspondent tous à la famille des arbres d'attaque. À l'instar des arbres de défaillances, leur facilité d'appréhension, leur lisibilité et leur nature hiérarchique sont autant d'atouts qui expliquent leur popularité. Bien sûr, ils partagent aussi leurs limites intrinsèques, à savoir leur caractère statique, restreignant les capacités de modélisation, et une dépendance forte aux compétences et au point de vue de l'analyste, impliquant un travail d'équipe pour éviter une subjectivité trop prononcée.

Une formalisation mathématique complète et rigoureuse est développée dans [310]. De nombreuses extensions et méthodes de quantification ont été proposées dans le domaine académique. Buldas *et al.* donnent ainsi une méthode pour rationaliser les choix de l'attaquant en fonction des risques et des profits qu'il perçoit [311] ; Jürgenson et Willemson exposent dans [312] une approche alternative, permettant des quantifications plus précises. Les mêmes auteurs ont aussi élargi le modèle de Buldas *et al.* pour prendre en compte des incertitudes dans les paramètres des arbres quantifiés [313]. Bistarelli *et al.* associent aux feuilles d'un *defense tree* des valeurs de retours sur investissement comme fonctions d'utilité dans un jeu à deux joueurs entre attaquant et défenseur [314] ; les équilibres de Nash sont calculés pour identifier les stratégies optimales. Yager généralise les portes OU et ET par l'utilisation de portes k/n dont le nombre de fils devant être vérifié pour leur réalisation est modélisé en logique floue [315] ; les formules de quantification de l'arbre sont précisées dans ce contexte.

Au-delà de l'amélioration des capacités de quantification, le modèle standard d'arbre d'attaque tel qu'incarné par les écrits de Schneier a été étendu selon de nombreuses autres directions. Par exemple, Daley *et al.* proposent un découpage en strates [316] alors que Grunze décrit une version modulaire, reposant sur des notions de profils [317]. Dans [282], Howard et Leblanc augmentent le formalisme des contre-mesures déployées pour limiter les risques associés aux menaces identifiées. Bistarelli *et al.* développeront cette idée sous forme d'arbres de défense (*defense trees)* précédemment évoqués [318, 314]. Edge fera de même dans sa thèse avec les arbres de protection (*protection trees)* [284].

Notons enfin que plusieurs travaux académiques ont ajouté une dimension dynamique aux arbres d'attaque. Citons à nouveau Jürgenson et Willemson qui modulent l'évaluation des scénarios selon l'ordre des étapes considérées [319] ; Camtepe *et al.* augmentent aussi les arbres d'attaque d'une logique temporelle dans [290] ; Khand suggère quant à lui de reprendre le formalisme des DFT présenté en Section 2.2.4 pour modéliser les attaques, mais n'évoque pas les évaluations quantitatives possibles [320].

Les arbres d'attaque « classiques » (statiques) sont spécifiquement ciblés par deux outils de modélisation informatique : *SecureITree* de la société Amenaza [33], et *Attack Tree+* de la société Isograph Software [34], spécialisée dans les logiciels de sûreté de fonctionnement et d'analyse de risques. Ceci-dit, il est possible, pour une utilisation basique, de s'appuyer de façon détournée sur tout logiciel de modélisation par arbres de défaillances (cf. Section 2.2.3).

Utilisation des réseaux de Petri en sécurité. Si les arbres de menaces sont les premières formes de modélisation graphique d'attaque, elles sont rejointes par des modélisations à base de réseaux de Petri dès la première moitié des années 90. En 1994, Dacier les emploie dans sa thèse comme modèle de quantification sous-jacent aux graphes de privilèges (cf. Section 2.3.5.2), modélisant la progression d'un attaquant dans la prise de possession d'un système [257]. La même année, Kumar et Spafford se servent des réseaux de Petri pour modéliser des scénarios de référence pour un dispositif de détection d'intrusions [278]. En 1997, Ho *et al.* décrivent une architecture mettant en œuvre ce principe, et améliorent le processus en élargissant la détection à un ordre seulement partiel dans les étapes des attaques décrites par les modèles en réseaux de Petri [279]. McDermott met en avant leur dimension graphique en sécurité sous forme d'*attack nets*, représentant des scénarios d'attaque de manière flexible [321]. Le grand intérêt des réseaux de Petri réside d'une part dans leur puissance de modélisation, soulignée en Section 2.2.4, et qui permet de prendre en compte notamment l'aspect séquentiel des attaques, la modélisation d'actions concurrentes et celle de diverses formes de dépendance. Elle tient d'autre part dans leurs capacités de

33. http://www.amenaza.com
34. http://www.isograph-software.com/atpover.html

quantification, notamment dans leur forme généralisée stochastique, qui rend opérationnels les outils classiques de l'analyse markovienne (cf. Section 2.2.4).

Les Figures II.12a) et II.12b) reprennent l'illustration choisie par McDermott, l'attaque dite de Mitnick, du nom du célèbre *hacker* l'ayant mise en œuvre pour la première fois. Elle mélange dénis de service par paquet SYN (*SYN flooding*), détournement de sessions TCP (*TCP hijacking*) et une usurpation de relation de confiance par le mécanisme Unix `rhost`. Dans la Figure II.12a), la place `p0` représente le point de départ de l'attaquant, qui a les droits administrateur sur une machine TCP/IP. Un jeton y figure initialement. La transition `t0` correspond à l'étape de reconnaissance de la cible par l'attaquant, atteignable par celui-ci *via* TCP/IP (utilisation de `finger`, `rpcinfo`, etc.). Une reconnaissance réussie réplique le jeton dans les places `p1`, `p2` et `p3`. La place `p1` représente l'identification d'une adresse routable mais inutilisée sur le réseau de la cible. La place `p2` correspond à l'identification d'une machine « usurpable » considérée comme de confiance par la cible. La place `p3` modélise l'identification de la machine cible, ayant une relation de confiance avec la machine identifiée en `p2`. La transition `t1` représente la construction de paquets SYN avec une fausse adresse source, visant à saturer la cible. La transition `t3` correspond à la recherche des numéros de séquences TCP nécessaires à l'usurpation d'identité. La place `p5` représente la capacité de l'attaquant à prévoir le numéro de séquence TCP de la machine de confiance. La transition `t2` correspond au début du déni de service par avalanche de paquets SYN sur cette machine. La place `p6` représente le système quand les *buffers* de la machine sont effectivement saturés. Arrivé à ce point, l'état de l'*attack net*, avec ses jetons, est donné par la Figure II.12b). La transition `t4` modélise l'usurpation de la session TCP avec la machine de confiance. La place `p7` modélise l'établissement d'une connexion par la cible avec la machine de l'attaquant qu'elle croit de confiance. La transition `t5` représente la modification du fichier `rhost` de la cible, permettant à l'attaquant d'obtenir un accès administrateur, et boucler sur une nouvelle attaque du même genre.

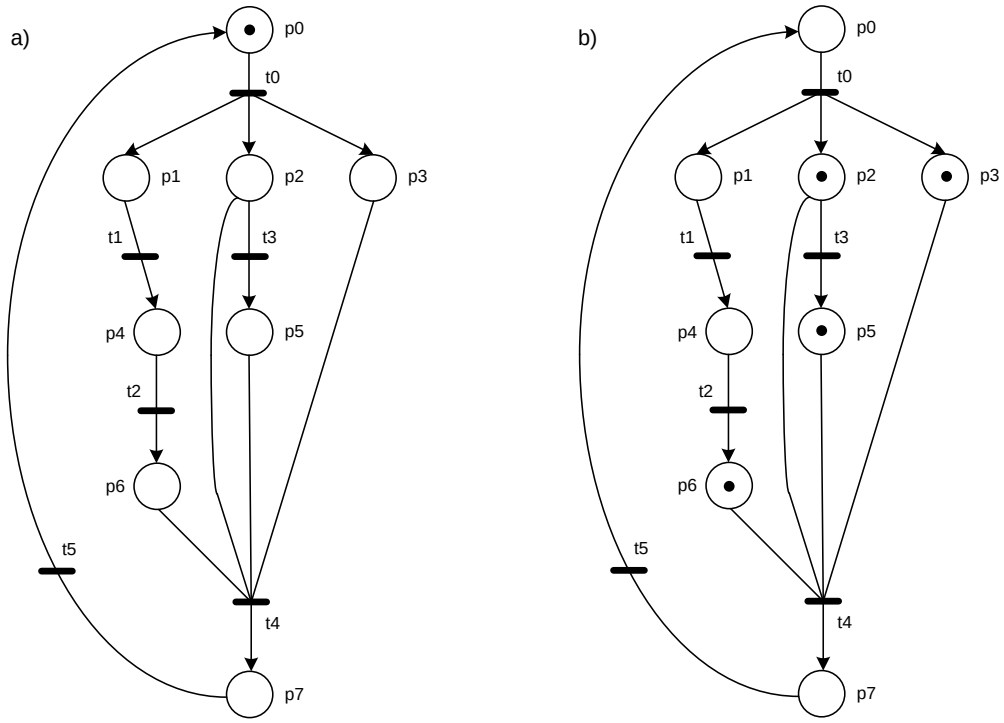


FIGURE II.12 – Attaque de Mitnick modélisée en *attack net* (d'après [321])

Depuis ces travaux pionniers, et malgré une lisibilité moindre que celles des arbres d'attaque, cette approche souple et puissante a fait des émules. Sans pouvoir être exhaustif, on citera notamment les développements de Horvath et Dörges combinant les réseaux de Petri avec le concept de motifs de sécurité (*security patterns*) [322], la gestion et le partage de ces motifs par un système de type Wiki proposée par Steffan et Schumacher [323] ou encore les travaux de Ten *et al.* qui mettent à profit les

capacités de modélisation des réseaux de Petri pour évaluer la sécurité d'un système SCADA [324]. Propre des réseaux de Petri, on trouve en fait une grande diversité de formalismes, rendant l'appropriation et l'industrialisation de l'approche délicate : ainsi, uniquement parmi les travaux cités, Kumar et Spaford se basent sur des réseaux de Petri colorés (*coloured Petri nets*), Dacier sur des réseaux de Petri stochastiques classiques (SPN), McDermott sur des *disjunctive Petri nets* (sans référence précise), Horvath et Dörges sur des *reference nets* [322], Ten *et al.* sur des GSPN [324]. On peut aussi par exemple ajouter l'utilisation d'*aspect-oriented Petri nets* dans [325]. Par ailleurs, toute une série d'articles s'appuient sur les SAN pour modéliser différentes problématiques de sécurité : on mentionnera particulièrement ceux de l'université de l'Illinois Urbana-Champaign sur les architectures tolérantes aux intrusions (*e.g.* [326, 327]), et ceux de Rrushi sur la détection d'intrusions dans les centrales nucléaires [328, 329].

Enfin, plusieurs articles établissent des correspondances formelles entre modèles à base de réseaux de Petri et arbres d'attaque. C'est le cas de Dalton *et al.* dans [330], et d'une façon plus convaincante de Pudar *et al.* dans [331]. Nous reviendrons sur certaines approximations de ces correspondances dans le Chapitre III Section 4.5. Notons seulement pour l'instant que là encore, de nouvelles variantes de réseaux de Petri sont employées (*deterministic time transitions Petri nets* dans [331]).

2.3.5.2 Modèles alternatifs. À côté de ces deux approches à forte visibilité, on trouve un grand nombre de propositions alternatives. Plus éparses, elles visent à appuyer les analyses de risques à diverses étapes et avec une rigueur et des capacités de modélisation variées. Certaines ont fait l'objet de développements restés isolés, d'autres, tels les graphes d'attaque ou les approches basées sur les réseaux bayésiens, ont une visibilité plus significative.

Quelques modèles isolés. En 1992, Caelli *et al.*[332] proposent par exemple une méthodologie de description graphique et d'évaluation des architectures de sécurité informatique et physique sous forme de réseau liant menaces et contre-mesures. Ces dernières y transforment les premières jusqu'à l'acceptation, au gré des liens constituant le réseau. La nature des éventuelles boucles formées permet d'identifier des situations favorables, ou au contraire problématiques pour la sécurité de l'ensemble. Un système d'indice de menace complète le formalisme pour opérer des quantifications.

En 1994, Dacier embarque les réseaux de Petri dans un formalisme de plus haut niveau, les graphes de privilèges, qui modélisent la progression de l'attaquant dans l'obtention de droits d'accès le menant à son objectif [257]. Dans un tel graphe, un nœud représente un ensemble de privilèges, un arc une méthode pour transférer ces privilèges à l'attaquant, correspondant à l'exploitation d'une vulnérabilité. Le temps et les efforts nécessaires à chaque étape sont modélisés par des processus stochastiques de loi exponentielle ; le modèle intègre cependant la « mémoire » de l'attaquant qui ne peut repasser par des états de privilèges déjà acquis, et son « bon sens », qui l'empêche de régresser. Dans ces hypothèses, les chemins critiques sont identifiés dans le graphe de Markov sans circuit formé.

En 1996, Meadows représente graphiquement dans une approche beaucoup plus informelle des attaques sur les protocoles cryptographiques, en articulant les étapes à franchir dans un graphe orienté [333]. La difficulté de chaque étape est indiquée par un code couleur ; le formalisme est essentiellement visuel et n'a pas de capacités de quantification. Notons que Meadows proposera un modèle graphique beaucoup plus rigoureux ultérieurement, en faisant correspondre un langage de spécification formel (NPATRL) à une représentation d'arbre de défaillances [334].

En 1997, Moskowitz et Kang décrivent un modèle proche des diagrammes de fiabilité (cf. Section 2.2.3), sans pour autant indiquer une telle filiation [335]. Les mesures de sécurité sont connectées en série ou en parallèle dans un graphe orienté partant d'une source, origine de la menace, à un puits, objectif de l'attaque. Le graphe permet d'identifier les *insecurity flows* et de les quantifier par des calculs de probabilités.

Plus près de nous, McQueen *et al.* introduisent les graphes de compromission (*compromise graph*) en 2006, dont ils illustrent l'emploi pour l'attaque d'un système SCADA [336]. Les nœuds du graphe représentent les étapes d'une attaque et les arêtes les temps nécessaires à leur réalisation (TTC pour *Time To Compromise*), définis en fonction du nombre de vulnérabilités connues et du niveau de l'attaquant. Cinq types de compromission sont considérés : reconnaissance, brèche du périmètre, pénétration, escalade (de privilège), dommage. Le modèle permet de comparer l'efficacité de différentes mesures de sécurité en comparant le TTC du chemin le plus court dans chaque cas. Byres et Leversage reprennent une approche similaire dans [337, 338], en la couplant à une décomposition de l'attaque selon les zones d'architecture concernées du système visé et quelques adaptations quant au mode de calcul des TTC.

Graphes d'attaque. En 1998, Phillips et Swiler sont les premiers à introduire la dénomination de graphe d'attaque (*attack graph*) [339, 340], par la suite largement reprise. Constatant les limites des approches à base d'arbres inspirées des modèles de sûreté, ils suggèrent d'utiliser des graphes dans lesquels un nœud représente un état possible du système durant le déroulement de l'attaque (niveaux d'accès logique de l'attaquant et effets de l'attaque sur plusieurs machines à ce point du déroulement), et une arête représente un changement d'état provoqué par une action de l'attaquant. Le graphe est généré automatiquement à partir de trois types d'entrée : des *attack templates* (représentations génériques d'attaques incluant les conditions nécessaires à leur réalisation), une description détaillée du système attaqué (topologie, configurations des composants, etc.), et une description du profil de l'attaquant (capacités, outils, etc.). Des quantifications sont réalisables en attribuant des poids aux arêtes, probabilités ou temps moyens de succès, et en cherchant les plus courts chemins dans le graphe.

À partir de 2002, Sheyner contribue largement à populariser le terme en associant les graphes d'attaque aux techniques de *model-checking* [341, 263]. Pour limiter les risques d'explosion combinatoire, de multiples méthodes sont alors développées : Amman *et al.* [342] contraignent le graphe par une propriété de monotonie, éliminant les retours en arrière en termes d'obtention de privilèges ; Noel et Jajodia prennent en compte des aspects de configuration [343, 344]. En 2006, Ou *et al.* optimisent la représentation et la génération des graphes d'attaque dans [345], les transformant en graphes d'attaque logiques (*logical attack graphs*) de taille polynomiale avec le nombre de composants du réseau informatique analysé. Au même moment, Lippman *et al.* proposent des *multiple-prerequisite graphs* (graphes à pré-requis multiples) qui limitent aussi très fortement la complexité des graphes (dans certains cas, linéairement avec la taille du réseau) [346]. Ces mêmes auteurs ont rédigé un état de l'art particulièrement complet sur les contributions du domaine entre 2002 et 2005 [347]. En 2006, Mehta *et al.* donnent un algorithme de classement des chemins du graphe dans [348]. En 2008, Malhotra *et al.* font de même [349] en s'appuyant sur la notion de surface d'attaque, présentée en Section 2.3.4. Wang *et al.* introduisent une métrique intégrant les probabilités d'existence des vulnérabilités considérées dans le graphe [350]. La grande majorité des auteurs cités ont également travaillé sur les aspects de visualisation (*e.g.* [351, 352, 353, 354]). Kotenko et Stephashkin décrivent quant à eux dans [355] une plate-forme logicielle complète de mise en œuvre des concepts et métriques des graphes d'attaque. Sur un plan plus théorique, Braynov et Jadliwala étendent leur utilisation à la participation de plusieurs attaquants [356].

Dawkins et Hale développent un concept voisin des graphes d'attaque nommé chaînes d'attaque (*attack chains*) [357]. Elles reposent sur une approche déductive arborescente, mais permettent également des raisonnements inductifs par le biais de *goal-inducing attack chains*, pour extraire les scénarios menant à un but donné. Les modèles sont aussi capables de générer des arbres d'attaque, quantifiables par les méthodes classiques. Les aspects de mise en œuvre logicielle sont précisés dans [358].

Approches basées sur les réseaux bayésiens. En 2005, Liu et Man utilisent pour la première fois les réseaux bayésiens en sécurité informatique [359], dans une approche inspirée des graphes d'attaque précédemment discutés. Leur graphe d'attaque bayésien est constitué de nœuds représentant des variables aléatoires binaires modélisant l'état de compromission d'une machine, et d'arêtes représentant les relations de dépendance probabiliste entre ces nœuds. Il est construit automatiquement par mise en correspondance des vulnérabilités des machines et de *templates* d'attaques, à la façon de Phillips et Swiler. L'originalité tient dans le paramétrage et les quantifications possibles, exploitant les capacités des réseaux bayésiens (cf. Section 2.2.3). Dans la suite des travaux de Wang *et al.* évoqués précédemment, Frigault *et al.* emploient en 2008 des réseaux bayésiens pour modéliser des dépendances entre vulnérabilités : l'exploitation réussie d'une vulnérabilité peut le cas échéant faciliter l'exploitation d'une autre dans la progression de l'attaquant [360]. Les probabilités individuelles des vulnérabilités sont estimées selon leur note CVSS[35], puis combinées selon les relations causales modélisées par le réseau bayésien. Wang *et al.* augmentent ensuite leur modèle de la dimension temporelle en s'appuyant sur des réseaux bayésiens dynamiques dans [360]. Ils s'inspirent d'An *et al.*, qui les avaient utilisés dès 2005 pour modéliser les risques d'atteinte à la vie privée dans une base de données [362].

Sommestad *et al.* s'appuient aussi sur les techniques bayésiennes dans une série de papiers initiée en 2008 [363, 364, 365, 366, 367]. Leur approche est plus analogue à la construction d'arbres d'attaque que celle de Wang *et al.*. Ils emploient un formalisme évolué de réseaux bayésiens : les diagrammes d'influence étendus [368] (*extended influence diagrams*, EID). Les variables aléatoires y sont représentées

35. *Common Vulnerability Scoring System* (CVSS, présenté en Section 2.3.4). Houmb *et al.* combineront aussi CVSS et réseaux bayésiens dans le cadre d'analyses de risques [361].

graphiquement par des nœuds de chance (*chance nodes*). Ils prennent des valeurs sur une échelle discrète et sont connectés entre eux par des arcs de cause (*causal arcs*), qui modélisent des liens de cause à effet, ou par des arcs de définition (*definitional arcs*) qui modélisent des relations plus arbitraires. Le formalisme s'appuie aussi sur les notions de nœuds de décision (*decision nodes*) et de nœuds d'utilité (*utility nodes*) (cf. les articles précités pour plus de précisions). La Table II.6 présente les principaux composants des diagrammes d'influence étendus. Les EID généralisent les arbres d'attaque classiques mais également les arbres de défense (cf. Section 2.3.5.1). Ils modélisent en plus les facteurs d'influence des valeurs qui correspondraient aux feuilles de ces structures équivalentes. La Figure II.13 représente un exemple tiré de [365]. La partie supérieure (nœuds en gris foncé) est un équivalent d'arbre d'attaque à trois feuilles, dont l'objectif est de réussir une authentification par mot de passe. La partie médiane (nœuds en gris clair) correspond aux contre-mesures qui seraient ajoutées dans un formalisme type *defense tree*. Enfin, la partie inférieure (nœuds en blanc) correspond à la modélisation des éléments influençant l'efficacité des contre-mesures ; elle est spécifique aux formalismes bayésiens. Notons que la réf. [367] applique cette approche à la modélisation sécurité de systèmes de communication du réseau électrique.

TABLE II.6 – Syntaxe des diagrammes d'influence étendus

Type de nœud	Nœud d'utilité	Nœud de décision	Nœud de chance
Type de relation	Relation causale	Relation informationelle	Relation de définition

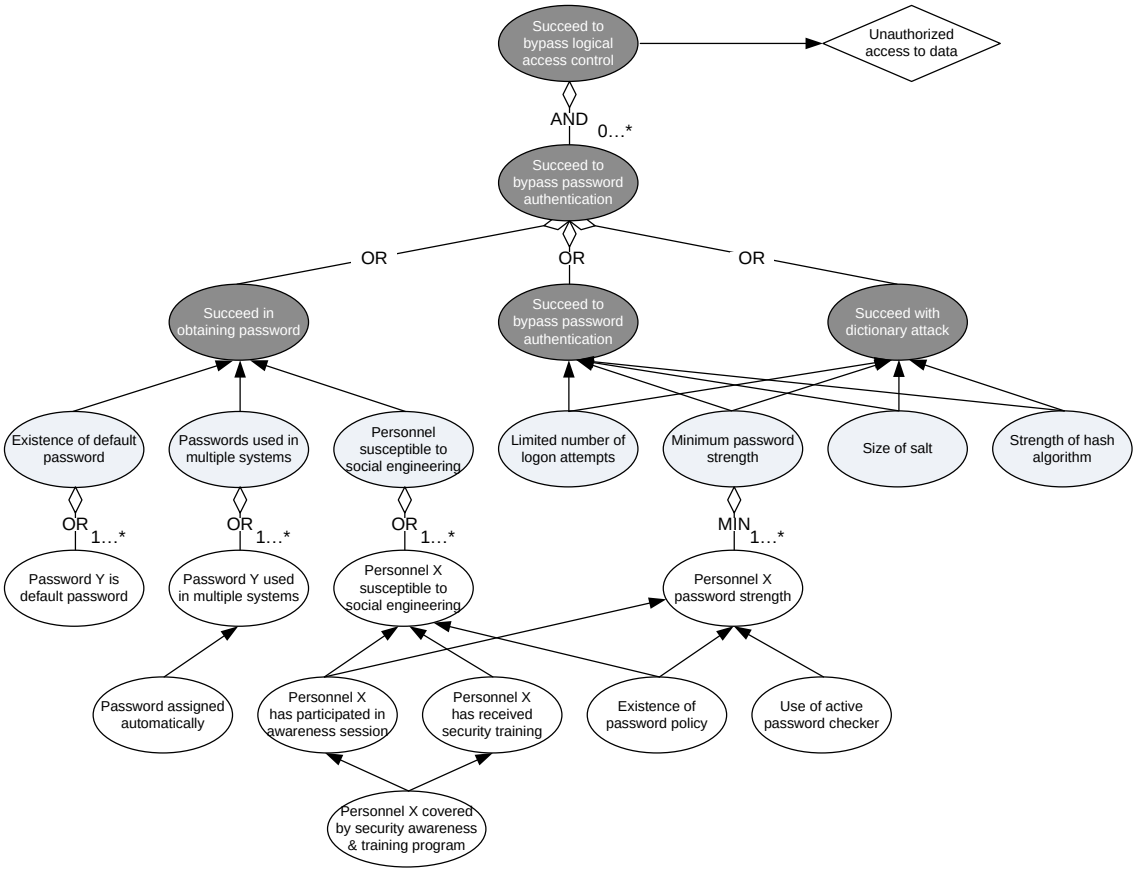


FIGURE II.13 – Modélisation en diagramme d'influence étendu de l'attaque d'un mot de passe [365]

Approches dérivées des diagrammes de cas d'utilisation UML [36]. Parmi ces approches, on citera notamment les *abuse cases* de McDermott et Fox [369], les *security use cases* de Firesmith [370] ou encore les *misuse cases* de Sindre et Opdahl [371, 372, 373, 374, 375] (et augmentés par Røstad dans [376]). Ces derniers sont les plus répandus. Ces techniques ne sont pas spécifiquement dédiées à la modélisation du déroulement des attaques, mais visent plus généralement à saisir les comportements malveillants à prendre en compte dans l'élaboration d'exigences de sécurité et pour leur explication. Leur souplesse permet cependant de modéliser graphiquement des scénarios d'attaques de façon expressive, mais sans formalisme mathématique supportant les quantifications. Elles peuvent ceci-dit être couplées avec d'autres types de modèles permettant ce type d'analyse, dont notamment les arbres d'attaque. Opdahl et Sindre comparent ces deux techniques et leur appropriation respective par les utilisateurs dans [377]. Leur intégration semble être non seulement simple, mais également utile et pertinente [378, 379]. À noter que Sindre a aussi adapté un autre type de diagramme UML, les diagrammes d'activité (*activity diagrams*), à la sécurité : les *mal-activity diagrams* [380] sont présentés comme une alternative aux *misuse case* pour les situations où ces derniers sont considérés par l'auteur comme mal-adaptés. C'est notamment le cas des situations comprenant des nombreuses interactions à l'intérieur du système, ou en dehors du système à spécifier. Les cas d'étude portent principalement sur des attaques type *social engineering*. Ce formalisme n'a pas eu le succès de *misuse case*, nous ne le signalons qu'à des fins de couverture thématique.

2.3.6 Vue macroscopique des techniques de modélisation graphique d'attaque

Nous donnons page suivante dans la Table II.7 une vue macroscopique des familles de modélisation graphique d'attaque sur la base des descriptions précédentes. Les formalismes n'ayant fait l'objet que de travaux isolés ne sont pas mentionnés (*e.g.* le modèle de Caelli *et al.* [332] ou les *insecurity flow diagrams* [335] de Moskowitz et Kang). De plus, les formalismes ont été regroupés par famille de modèles, selon la technique principale sous-jacente. Ces familles ont elles-mêmes été divisées en deux catégories : modèles statiques et modèles dynamiques [37]. Pour chaque famille, une rapide description de la technique et des types de résultats produits est donnée. Elle s'accompagne d'une synthèse des points forts, limites et hypothèses nécessaires. Ces derniers éléments s'appuient principalement sur l'analyse donnée par les promoteurs des différents formalismes dans les articles cités, sur des articles comparant explicitement certains d'entre eux (*e.g.* [381, 323, 377]), et dans certains cas, sur notre propre appréciation.

2.3.7 Tableau récapitulatif des techniques d'évaluation en sécurité

La Table II.8 donne dans les pages suivantes une vision synthétique des différents outils et techniques d'évaluation présentés pour le domaine de la sécurité.

36. *Unified Modeling Language* (UML, `http://www.uml.org/`).
37. À noter que les réseaux bayésiens ont été classés dans les modèles statiques : l'emploi des réseaux bayésiens dynamiques en sécurité, cité en Section 2.3.5.2, est jugé trop marginal pour être ici considéré.

TABLE II.7 – Vue macroscopique des principales familles de modélisations graphiques d'attaque

	Famille et réf.	Type/Description	Résultats qualitatifs	Résultats quantitatifs	Points forts	Hypothèses & limites
Modèles statiques	Arbres d'attaque (*attack trees, threat trees, protection trees, vulnerability trees*,...) [277, 22, 285, 310, 318, 291]	Arbres logiques articulant actions et étapes à franchir (feuilles et portes OU et ET) vers un objectif d'attaque (sommet). Les contre-mesures sont parfois représentées.	Identification des scénarios d'attaque. Identification des points faibles.	Probabilité de l'événement redouté, Contribution des feuilles (selon nature). Effet des contre-mesures.	Faciles à appréhender. Hiérarchiques. Utilisation répandue. Extensions du formalisme incrémentales.	Quantifications conditionnées par l'indépendance des feuilles. Modèles statiques. Pas de boucles/cycles.
	Modèles à base de réseaux bayésiens / diagrammes d'influence [359, 365, 367]	Graphes de liens conditionnels entre variables aléatoires ; tables de probabilités conditionnelles associées. Plutôt utilisés en sécurité sous des formes évoluées type diagrammes d'influence.	Peu adaptés aux considérations qualitatives. On perd notamment les notions de coupes minimales propres aux arbres d'attaque.	Très variés. En outre, calcul de distributions marginales en fonction de valeurs observées ou estimées. Force des liens, importance des nœuds.	Généralisent les arbres d'attaque (pouvoir de modélisation supérieur). Prise en compte des incertitudes, intégration de sources multiples pour la définition des paramètres.	Modèles statiques. Lisibilité limitée (beaucoup d'information non représentée).
	Diagrammes UML adaptés à la sécurité (*abuse case, misuse case, security use case*) [369, 370, 371, 375, 376]	Extensions des diagrammes de cas d'utilisation UML pour la sécurité. Utilisés plutôt pour formaliser des exigences de conception.	Spécification visuelle de cas problématiques en termes de sécurité (violation de politique), et interactions avec les cas d'usage normaux.	Pas directement.	Facilité de construction. Capacité d'intégration et d'assemblage avec d'autres formalismes.	Non adaptés à certains types d'attaques interactives. Très macroscopiques. Descriptions textuelles et certaines informations importantes non représentées.
Modèles dynamiques	Modèles à base de réseaux de Petri [278, 321, 324, 331]	Modélisation d'interactions complexes (concurrences, synchronisations, boucles, partages...) à l'aide de places, arcs, transitions et jetons. En GSPN, spécification compacte de graphes de Markov.	Identification des scénarios d'attaque.	Probabilité de l'événement redouté. Contribution des feuilles (selon nature). Effets de contre-mesures.	Grand pouvoir de modélisation. Communauté scientifique et utilisatrice importante. Nombreux outils.	Diversité des formalismes. Explosion combinatoire pour les traitements. Expertise nécessaire.
	DFT (*Dynamic Fault Trees*) en sécurité [320]	Arbres d'attaque avec portes spécifiques modélisant les aspects temporels et les dépendances.	Identification des scénarios d'attaques.	En théorie, temps moyen de succès (non documenté en sécurité).	Conservent le caractère hiérarchique et une partie de la sémantique des arbres d'attaque.	Peu documentés et utilisés en sécurité (un seul papier recensé). Lisibilité des portes discutable.
	Graphes d'attaque [339, 341, 342, 347, 350]	Graphes générés par *model-checking* ou construction automatisée. Nombreuses variantes.	Identification de scénarios d'attaques.	Selon les variantes.	Automatisation. Nombre de situations couvertes.	Explosion combinatoire. Problèmes de lisibilité des résultats.

TABLE II.8: Synthèse des techniques d'évaluation de sécurité présentées

Technique	**Utilisation**	**Description**	**Résultats qualitatifs**	**Résultats quantitatifs**	**Limites**	**Réf.**	**Origines**
Critères Communs	Très utilisés dans les administrations. Produits informatiques de sécurité. Internationalement reconnus.	Référentiel d'évaluation normalisé extrêmement modulaire. Distingue notamment le périmètre, les fonctions de sécurité et le niveau d'évaluation.	Certification validant l'évaluation de fonctions de sécurité d'un produit selon différents degrés d'assurance EAL 1 à 7.	Dépend du niveau.	Processus complexe, long et coûteux.	ISO 15408 [258]	Filiation avec l'*Orange Book* (TCSEC) puis les ITSEC.
Certification ISA Secure	Informatique industrielle.	Référentiel d'évaluation.	Certification EDSA (3 niveaux).	Dépend du niveau.	Schéma émergeant, en évolution.	EDSA [259]	ISA 99
IEC/ISO 27001	Transverse.	Schéma de gestion de la sécurité de l'information : procédures, documentation, PDCA (*Plan-Do-Check-Act*).	Certification de l'organisation évaluée.	Non.	Essentiellement procédural. Certification peu répandue en France.	IEC/ISO 27001 [115]	Adapté du standard britannique BS 7799.
Modèles formels de politique d'accès	Conception de produits de sécurité pour secteurs sensibles (militaire, banque, infrastructures critiques...).	Cadre d'expression et de vérification formelle (mathématique) d'exigences de sécurité. Surtout contrôle d'accès et autorisation.	Validation formelle.	Dépend des modèles.	Expertise. Attaques « en dehors » du modèle.	[382, 383, 384]	Bell-Lapadula, Biba, *Orange Book*
Méthodes qualitatives d'analyse et de gestion de risques	Génériques. Chaque secteur privilégie une approche sur des bases historiques. Utiles en conception comme en exploitation.	Nombreuses méthodes, mais suivant toutes une approche assez similaire incluant la caractérisation des menaces, des vulnérabilités et de la criticité des conséquences.	Variable. En général, identification, hiérarchisation, traitement et communication du risque.	Éventuellement (la méthode peut faire appel à certains stades à des outils quantitatifs).	Compromis profondeur de l'analyse et facilité de maintenance / suivi de l'évolution du risque délicat à trouver.	IEC/ISO 27005 [118] NIST SP800-53 [111] etc.	Finance.

Tests d'intrusion	Généralisée.	Plusieurs modalités possibles selon le niveau d'information accordé aux auditeurs et aux audités.	Identification de vulnérabilités. Éventuellement, préconisations d'améliorations.	Éventuellement.	Très dépendant de l'expertise des auditeurs. Sensibles sur systèmes en production, notamment en info. industrielle.	OSSTMM [268]	Militaire. « *Red team* » de Los Alamos et Argonne Labs.
Scanners automatisés de vulnérabilités	Tout secteur. Emploi généralisé.	Série de tests techniques automatisés par logiciel.	Identification de vulnérabilités.	% de vulnérabilités sur une base donnée.	Vision très incomplète des vulnérabilités.	–	Premiers outils début des années 90 (*e.g.* COPS [270] en 1990, SATAN [271] en 1993.
Mesures / métriques de la vulnérabilité	Général.	Très nombreuses propositions (*e.g. attack surface*, CVSS)	–	Par définition.	Domaine foisonnant et sans consensus.	[250] [251, 252]	–
Modélisations graphique d'attaque	Essentiellement académiques, sauf pour les arbres d'attaque.	Modèle dominant = arbres d'attaque, mais nombreux autres modèles, dont ceux à base de réseaux de Petri.	Identification de scénarios d'attaques. Evaluation de stratégies de défense.	Selon modèles, calcul probabilités et/ou temps moyen de succès. Parfois autres métriques.	Dépend du modèle. Les arbres d'attaque sont statiques.	Cf. Table II.7.	Années 80, Militaire.

3 Inspirations réciproques et fertilisations croisées

Sûreté et sécurité ont longtemps évolué comme deux disciplines distinctes, animées par des communautés cloisonnées développant chacune de leur côté des outils et méthodes adaptés à leur domaine [20, 21]. Pourtant, comme nous l'avons montré en Section 1.1, les similarités sont nombreuses, et les outils de l'un peuvent bien souvent trouver une utilisation dans l'autre, *modulo* certaines adaptations. Ce constat a été établi dès le début des années 90, en outre par Jonsson [385] ou Brewer [386], mais les ponts sont longs à bâtir entre les communautés, encore aujourd'hui très segmentées. Nous dressons dans cette section un panorama des fertilisations croisées entre sûreté et sécurité recensées dans la littérature.

3.1 De la sûreté à la sécurité

3.1.1 Concepts architecturaux

Transposition de techniques de tolérance aux fautes à la sécurité (tolérance aux intrusions). La tolérance aux fautes est une des grandes catégories de techniques de sûreté de fonctionnement [16]. Elle vise à permettre au système de remplir sa fonction et assurer le service voulu en dépit de fautes. Les techniques classiques de tolérance aux fautes incluent la redondance, la diversification ou encore la séparation [41]. Elles sont mises en œuvre de longue date en sûreté de fonctionnement dans les secteurs sensibles tels que le spatial ou le nucléaire, notamment pour les systèmes assurant des fonctions de sûreté. Le principe de tolérance aux fautes n'a été considéré, puis transposé, au domaine de la sécurité que tardivement : celui-ci n'a longtemps considéré que les aspects de prévention et de détection d'attaques (*i.e.* fautes de nature malveillante, provoquées par des intrusions ou des codes hostiles) [386].

Les premières transpositions sont menées au milieu des années 80, où le LAAS introduit le concept de tolérance aux intrusions (*intrusion tolerance*) [387, 388], et propose une application de ce concept à travers une technique dite de « fragmentation-réplication ». Inspirée de [389], elle sera améliorée dans de nombreux travaux (*e.g.* [390]) sous l'appellation fragmentation-redondance-dissémination (*Fragmentation-Redundancy-Scattering*, ou FRS). Les principes de redondance y sont mis à profit pour protéger l'information en confidentialité, en intégrité et en disponibilité par un découpage spécifique en fragments redondés. A la même période, Dobson et Randell de l'université de Newcastle upon Tyne au Royaume-Uni défendent une approche similaire, désignée par *security fault tolerance* [391]. Ils soulignent plus largement la pertinence des outils et des approches de la sûreté de fonctionnement en sécurité informatique, et militent pour leur utilisation. Les équipes du LAAS comme celles de Newcastle partagent le même constat quant aux limites inhérentes aux approches alors « classiques » de la sécurité, centrées sur la prévention et la détection de fautes. La complexité des systèmes, leur nature de plus en plus distribuée, l'imprévisibilité des attaquants rendent hélas les intrusions et les attaques réussies inévitables : les systèmes doivent être aussi conçus pour les tolérer.

L'idée fait florès, le concept de tolérance aux intrusions traverse l'Atlantique [392], et au gré de la multiplication des travaux, va progressivement donner naissance à une discipline aujourd'hui encore exploratoire mais bien structurée. Verissimo en donne une excellente synthèse dans [393]. Les approches de tolérance aux intrusions ont notamment bénéficié de grands projets de recherche collaborative, notamment aux États-Unis sous l'impulsion du DoD (programmes ITS et OASIS notamment [38]) mais également en Europe dans le cadre du Programme Cadre de Recherche et Développement (PCRD) européen. On citera en particulier le projet MAFTIA [39] [394], mené de 2000 à 2003, et ses résultats scientifiques dans le domaine des protocoles et intergiciels pour les applications Internet tolérantes aux intrusions [395]. Par ailleurs, la tolérance aux intrusions est de plus en plus considérée comme une composante de notions plus larges comme la *survivability*, dans le sillage des travaux de l'université Carnegie Mellon sur ce concept [396], et la « résilience »[397], autour de laquelle une grande partie de la recherche européenne du domaine semble aujourd'hui s'articuler. Les résultats du projet CRUTIAL [40] [398, 399], mais également ceux du projet AMBER [41] et du réseau d'excellence ReSIST [42], dédiés à l'étude de la résilience pour les grands systèmes informatiques complexes, s'inscrivent dans cette dynamique.

38. Le site http://www.tolerantsystems.org donne un accès aux différents projets de ces programmes.
39. *Malicious-and Accidental-Fault Tolerance for Internet Applications* (MAFTIA, http://www.laas.fr/TSF/cabernet/maftia/results/index.htm).
40. *Critical UTility InfrastructurAL resilience* (CRUTIAL, http://crutial.cesiricerca.it).
41. *Assessing, Measuring, and BEnchmarking Resilience* (AMBER, http://www.amber-project.eu).
42. *Resilience for Survivability in IST* (ReSIST, http://www.resist-noe.org).

Enfin, d'autres travaux se sont intéressés à des aspects spécifiques des techniques de tolérance aux fautes pour la sécurité, sans explicitement les intégrer dans un cadre plus global d'architecture tolérante aux intrusions. Ainsi, Littlewood et Strigini ont étudié les liens entre redondance, diversité et sécurité dans [31] ; Totel *et al.* mettent à profit des approches à base de diversification pour améliorer les performances de systèmes de détection d'intrusions [400].

Défense en profondeur. L'approche dite de « défense en profondeur » (*defense-in-depth*) est aujourd'hui considérée comme un principe fondamental en sécurité informatique. D'origine militaire, elle a été popularisée et institutionnalisée pour la sûreté des centrales nucléaires. Une première déclinaison se concrétise dans les réacteurs modernes par trois barrières successives indépendantes permettant de confiner la matière radioactive (gaine combustible, cuve réacteur et enceinte réacteur) [401]. Plus généralement, l'idée est que chaque dispositif doit *a priori* être considéré comme vulnérable ; chaque barrière doit être indépendante des autres et auto-suffisante pour assurer la protection de l'environnement. La défense en profondeur se manifeste aussi par le cumul d'une conception spécifiquement tournée vers la sûreté, avec des procédures et un contrôle opérationnel également adaptés à cette fin. Elle conduit en outre à la diversification et la redondance des dispositifs de sûreté dans la centrale [402]. Repris ensuite en sécurité informatique (*e.g.* [403, 404, 113]), le concept de défense en profondeur y est cependant utilisé de façon assez floue et variée selon les contextes. Il se rapproche en fait bien souvent de la métaphore dite du « fromage suisse » de Reason [405], elle aussi issue du monde de la sûreté et illustrée par la Figure II.14. L'événement redouté y survient à condition de plusieurs barrières défaillantes et conditions accidentelles réunies. La référence [23] propose une analyse et une rationalisation de la démarche de défense en profondeur pour l'informatique. Notons enfin que l'on rencontre aussi l'expression sécurité en profondeur (*security in depth*) pour exprimer la même notion en sécurité physique.

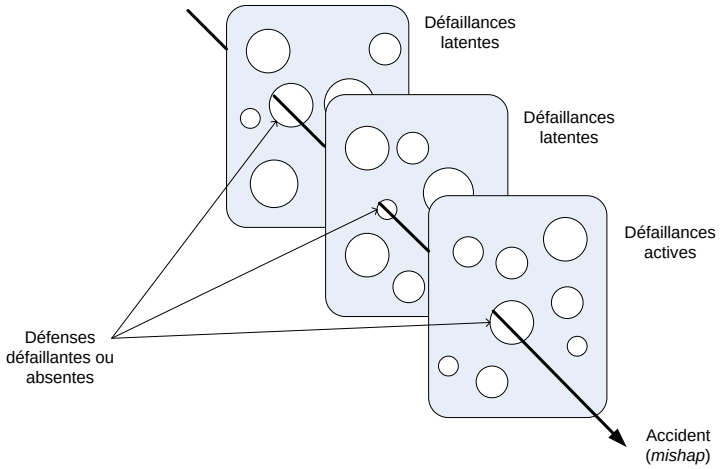


FIGURE II.14 – Illustration du modèle dit du « Fromage Suisse » de Reason [405]

3.1.2 Formalismes graphiques

Des arbres de défaillances aux arbres d'attaque. Nous n'avons identifié qu'une seule adaptation significative d'un formalisme graphique employé en sûreté pour le domaine de la sécurité informatique : les arbres d'attaque (et arbres de menaces). Inspirés des arbres de défaillances, ils constituent plus globalement la seule adaptation d'un outil sûreté à avoir trouvé une place dans les pratiques d'analyse de risques en sécurité en dehors du milieu académique, notamment aux États-Unis. La Section 2.3.5.1 en donne un historique et une description complète. En synthèse, les travaux de Schneier [22] ont largement popularisé ce type de modélisation, employé dans des contextes très variés (*e.g.* [292, 293, 261, 294, 295, 298, 301, 302, 303, 304]) et intégré à différentes méthodes d'analyse de risques (*e.g.* [267, 266]). De nombreuses extensions ont été apportées, dont l'intégration des contre-mesures (*defense trees* notamment [318]), l'interfaçage avec la logique floue [315], la théorie des jeux [314, 311, 312] ou encore la prise en compte d'aspects temporels et dynamiques (avec entre autres approches la réutilisation des DFT [320]).

3.1.3 Analyses de risques structurées

HAZOP en sécurité. Winther *et al.* sont les premiers à proposer l'utilisation de HAZOP en sécurité informatique. Ils définissent en outre des mots-guides plus adaptés que ceux employés pour la sûreté des systèmes programmés [406]. Dans sa thèse, Foster adapte quant à elle HAZOP pour améliorer la collecte et l'analyse des besoins et des exigences des protocoles de sécurité avant leur conception effective, dans une démarche dénommée *Vulnerability Identification and Analysis* (VIA) [407, 261]. Srivatanakul [408] met à profit la formalisation par *use cases* UML des comportements attendus du système pour alimenter une version adaptée de HAZOP dont il décrit précisément la démarche d'application. Les mots-guides là aussi sont spécifiquement choisis pour une utilisation sécurité. Plusieurs exemples d'application détaillés sont fournis.

Globalement, l'utilisation de HAZOP en sécurité permet d'identifier de nouveaux risques qu'une analyse macroscopique moins systématique aurait pu laisser échapper ; elle permet en outre de structurer le questionnement et force l'analyste à considérer des scénarios inhabituels. Cependant, comme en sûreté, le travail en équipe reste nécessaire pour assurer une largeur de couverture suffisante. De plus, le niveau d'abstraction, lié à la liste de mots-guides restreinte, peut cacher des attaques et des risques qui ne seront pas considérés : c'est par exemple le cas des aspects communications dans l'approche de Srivatanakul [408]. Le choix des mots-guides est dans tous les cas critique et requiert la plus grande attention. Enfin, la méthode est assez fastidieuse à mettre en œuvre et intrinsèquement répétitive.

Analyse des conditions insidieuses et Analyse transitoire (*sneak analysis*) en sécurité. En 2004, Baybutt propose une adaptation de l'analyse des conditions insidieuses (cf. Section 2.2.1) au domaine de la sécurité informatique, appliquée aux systèmes industriels. Dans [409, 410], il expose une approche en neuf étapes, qui s'appuie sur une analyse de la topologie réseau de l'architecture à étudier pour identifier les chemins d'attaque potentiels. Ces chemins relient des « sources » (attaquants internes comme externes) vers des cibles, exploitant des vulnérabilités ou contournant des systèmes de sécurité informatique. Les résultats sont consignés sous forme de tableaux, permettant également d'élaborer de façon systématique des recommandations. Évidemment, l'analyse dépend grandement de l'exactitude et de la complétude de la description de l'architecture en donnée d'entrée [408].

Analyse par zone (ZSA, *Zonal Safety Analysis*). Srivatanakul présente dans sa thèse une adaptation des concepts de l'analyse par zone, décrite en Section 2.2.1, à la sécurité [408]. Il y intègre les dimensions physique et logique (informatique), mais également comportementale et temporelle. Il analyse différents types de canaux pouvant exister entre ces zones : proximité physique, dépendance fonctionnelle, dépendance environnementale, adjacence procédurale et relation temporelle. La méthode s'appuie sur l'utilisation de mots-guides de type HAZOP et formalise là aussi les résultats de l'analyse sous forme de tableaux. Son intérêt principal réside dans la prise en compte de dimensions transverses, souvent ignorées dans des analyses plus verticales.

De FMEA à IMEA (*Intrusion Modes and Effects Analysis*). Gorbenko *et al.* adaptent l'approche FMEA, décrite en Section 2.2.1, à la sécurité en la rebaptisant IMEA (*Intrusion Modes and Effects Analysis*). Les entrées des tables ne correspondent plus à des modes de défaillance accidentelle mais à des techniques d'intrusion. L'approche est décrite et employée sur une architecture de *Web Services* dans [411] ; une analyse par IMEA d'un système SCADA est donnée dans [412].

Des *safety cases* aux *security assurance cases*. La notion de *safety case* est utilisée dans les industries à risques essentiellement au Royaume-Uni. Inge en donne un historique dans [413]. Un *safety case* peut être défini comme une argumentation structurée, appuyée par un faisceau d'éléments probants, permettant d'établir de façon convaincante, compréhensible et valide qu'un système est jugé suffisamment sûr dans un environnement opérationnel donné [43]. La motivation sous-jacente des *safety cases* est simple : plutôt que d'avoir à mettre à jour à un rythme soutenu les réglementations en fonction de l'évolution des pratiques et techniques, la charge est renversée sur les opérateurs, dont les *safety cases* sont régulièrement

43. Nous traduisons ici au mieux les définitions données en anglais dans [106] (« A structured argument, supported by a body of evidence that provides a compelling, comprehensible and valid case that a system is safe for a given application in a given operating environment ») et dans [414] (« A documented body of evidence that provides a convincing and valid argument that a system is adequately safe for a given application in a given environment. »).

évalués par les régulateurs. Elle est donc sur le fond proche de la notion de démonstration de sûreté utilisée en France dans le nucléaire, mais reste assez éloignée de la conception états-unienne [415], souvent plus prescriptive. Un *safety case* couvre typiquement [413] :

- la définition du périmètre du système ou de l'activité concerné, ainsi qu'une description détaillée de son contexte ou son environnement ;
- le système de gestion mis en œuvre pour assurer la sûreté ;
- la mention des exigences réglementaires et légales, les normes et référentiels applicables associés aux preuves de leur respect ;
- la « preuve » (*evidence*) que les risques sont bien identifiés et contrôlés de façon appropriée, et que le niveau de risque résiduel est acceptable ;
- des garanties quant à l'objectivité de l'argumentation et des preuves.

La notion de *safety case* a été transposée et généralisée dans le domaine logiciel en *assurance case* [416]. Bien sûr, les *assurance cases* peuvent être utilisés eux-mêmes à des fins de sûreté [417, 416], mais également associés à d'autres propriétés. En outre, entre 2004 et 2007, un groupe de travail international s'est régulièrement réuni pour travailler sur une adaptation du concept au domaine de la sécurité informatique (*assurance case for security* ou *security assurance case*). Il incluait en outre des représentants du JRC de l'Union européenne et de l'université Carnegie Mellon. La référence [415] donne une synthèse de ses travaux, qui ont alimenté de nombreuses publications académiques et ouvert des pistes à ce jour encore pertinentes. En outre, l'approche semble avoir pris une dynamique industrielle aux États-Unis, notamment *via* l'initiative *Build Security In* du US CERT [44] qui promeut les travaux de Carnegie Mellon effectués par la suite [418, 419].

Enfin, Kelly décrit dans sa thèse [420] une notation dénommée GSN (*Goal Structured Notation*) articulant graphiquement les arguments, preuves et justifications constituant un *safety case.* Elle améliore sa lisibilité, mais aussi sa communication et sa maintenance. Une telle notation reste pertinente dans le cadre de *security assurance cases* précédemment discutés. Moleyar et Miller s'en sont aussi servi pour augmenter le formalisme des arbres d'attaque et associer des contre-mesures aux vulnérabilités [421].

GEMS (*General Error Modeling System*). Dans [422], Brostoff et Sasse militent pour prendre en compte dès la conception les aspects non-techniques de la sécurité, et en particulier le facteur humain et la dimension sociotechnique des systèmes considérés. Ils proposent d'appliquer pour cela le modèle GEMS (*General Error Modelling System)* de Reason [405], issu de la sûreté. GEMS distingue notamment les fautes latentes des fautes actives en prenant en compte le comportement humain, dépassant le cadre des analyses habituellement menées en sécurité.

Des niveaux SIL aux niveaux SAL. Le concept de SIL (*Safety Integrity Level*) a, depuis son introduction au début des années 90, pris une place de plus en plus importante dans le domaine de la sûreté des systèmes E/E/EP. Initialement introduit de façon sectorielle dans des normes anglo-saxonnes (cf. la première version du standard militaire Def-Stan-00-55 [423] dès 1991 au Royaume-Uni et l'ISA 84.01 en 1996 aux États-Unis [424]), il a par la suite été généralisé et internationalisé par l'IEC 61508 [110] (cf. Section 2.2.2). Kube et Singer proposent dans [425] des *Security Assurance Levels*, niveaux d'assurance en sécurité inspirés des niveaux SIL de la sûreté ; un groupe de travail commun entre l'ISA 99, comité technique de l'ISA sur la sécurité des systèmes d'informatique industrielle, et l'ISA 84, en charge des aspects sûreté, continue depuis ces travaux [426]. Ils doivent être prochainement intégrés à un document de la série normative ISA 99 [427]. De plus, la certification EDSA évoquée en Section 2.3.1 fait directement appel à ces notions de SAL et revendique explicitement la filiation avec l'IEC 61508 [259].

3.1.4 Techniques de test

Injection de fautes. Les techniques d'injection de fautes, au niveau matériel (altérations électriques ou radiatives) ou logiciel (corruption de variables, de registres ou de mémoires), font partie de l'arsenal

44. https://buildsecurityin.us-cert.gov/bsi/articles/knowledge/assurance.html

de la sûreté de fonctionnement depuis plusieurs décennies pour l'élimination et la prévision de fautes [41, 428]. Les techniques s'appliquent à des systèmes déjà réalisés, éventuellement prototypes ou dès les premières étapes de la conception, en s'appuyant sur des modèles de simulation. En 1996, Voas suggère d'adapter ces techniques pour la sécurité informatique, en injectant de façon adaptative des entrées erronées dans un code applicatif pour provoquer des comportements anormaux du logiciel à des fins malveillantes [429]. Depuis, le principe a été largement repris et les techniques améliorées, aussi bien d'un point de vue défensif, pour la découverte de vulnérabilités en amont de la mise en production du logiciel, que d'un point de vue offensif pour attaquer des applications existantes [430].

Modèles de croissance de fiabilité (*Reliability growth*) pour la sécurité. En 2005, Ozment [431] applique les techniques de prédiction de fautes et d'évaluation de la fiabilité logicielle dites de « *reliability growth* » à la sécurité, modélisant selon différentes approches le rythme de découverte de vulnérabilités dans un logiciel. Rescorla avait suggéré l'approche dans [432], mais Ozment approfondit considérablement la démarche, en s'intéressant plus particulièrement aux vulnérabilités d'*Open BSD*. Les estimations issues des modèles doivent servir de métrique de sécurité, notamment pour comparer différents produits. Ozment précise les hypothèses nécessaires et les difficultés d'une telle adaptation, il compare la qualité des prédictions effectuées par les différents modèles vis-à-vis de l'historique des vulnérabilités de son cas d'étude. Les résultats obtenus sont encourageants, mais les données d'entrée ne sont, de l'aveu même de l'auteur, pas suffisantes pour asseoir la crédibilité de ses déductions.

3.2 De la sécurité à la sûreté

3.2.1 Du noyau de sécurité (*security kernel*) au noyau de sûreté (*safety kernel*)

La concentration des fonctions critiques de sécurité pour un système dans un noyau distinct du reste du système, est une approche courante en sécurité informatique, mise en œuvre dès les années 70 [433, 434]. À cette époque, le noyau regroupe en fait les fonctions de contrôle nécessaires à la mise en œuvre des politiques multi-niveaux évoquées en Section 2.3.2 et dans l'Annexe B. En 1986, Rushby soutient que cette approche architecturale peut être utile à d'autres propriétés associées à la sûreté [435]. Il donne des pistes de formalisation et discute des implications pour les architectures logicielles. Quelques années auparavant, Leveson avait également souligné l'intérêt des approches architecturales à base de noyau pour les logiciels de sûreté [436]. Elle mettait notamment en avant les avantages d'un noyau de taille et de complexité réduites, limitant les erreurs, et offrant une meilleur maintenabilité. Plusieurs architectures logicielles de sûreté ont depuis été proposées sur ces principes (*e.g.* [437, 438]).

3.2.2 Modèles formels pour les propriétés de sûreté

Nous avons évoqué en Section 2.3.2 des modèles formels de sécurité, spécifiquement conçus pour caractériser et contrôler rigoureusement les propriétés des politiques de contrôle d'accès. Ces modèles formels de sécurité ont pour certains été considérés dans une problématique sûreté. En 1998, Simpson *et al.* [439] s'inspirent de la propriété de *non-interference*, introduite en sécurité par Goguen et Meseguer en 1982 [440], pour modéliser rigoureusement un comportement *fail-safe* (faute sans impact sur la sûreté), *fail-stop* (faute et réparations associées sans impact sur la sûreté) et *fail-operational* (faute et réparations associées sans impact sur la sûreté, ni sur les aspects fonctionnels du système). La formalisation est faite en CSP (*Communicating Sequential Processes*) [441]. Peu de temps après, Stavridou et Dutertre [442] reprennent plus globalement l'idée d'adapter les modèles formels de sécurité en sûreté, et élargissent notamment la réflexion à d'autres propriétés de sécurité. Enfin, Totel *et al.* proposent dans [443] un modèle de contrôle d'intégrité permettant l'intégration de composants de niveaux de criticité hétérogènes (dont des COTS) dans des architectures à forts enjeux de sûreté. Des communications bidirectionnelles entre objets de criticités différentes peuvent y être autorisées sous certaines conditions et validations. Le modèle théorique, orienté objet, a donné lieu à des réalisations expérimentales dans le cadre du projet européen GUARD [45] il y a une dizaine d'années [444], et continue à susciter l'intérêt dans le domaine de l'avionique (cf. les travaux récents du LAAS à ce sujet [42, 445]).

45. *Generic Upgradable Architecture for Real-time Dependable Systems* (GUARDS).

3.2.3 Vers plus de prévention de fautes en sûreté

En 1993, Brewer constate dans [386] que la communauté sécurité privilégie les approches de prévention de fautes, tandis que la communauté sûreté a tendance à privilégier les approches de tolérance aux fautes. Si la sécurité commence alors effectivement à s'inspirer de ces dernières (cf. Section 3.1.1), il suggère que la sûreté devrait faire de même pour les techniques préventives, et souligne notamment la pertinence du concept de *reference-monitor*, concept d'architecture proche du noyau de sécurité, permettant de mettre en œuvre des politiques de sécurité rigoureuses par conception en contrôlant toutes les demandes d'accès aux ressources.

3.2.4 Utilisation des *misuse cases* en sûreté

Dans [446], Sindre examine l'intérêt du formalisme sécurité des diagrammes de *misuse cases*, qu'il a inventés, pour la sûreté. Il les compare notamment avec quatre approches classiques du domaine, à savoir les arbres de défaillances, HAZOP, les diagrammes CCA et l'AMDE (cf. Sections 2.2.3 et 2.2.1). Alexander a lui aussi considéré une telle adaptation dans [447, 374]. La réf. [448] rapporte une étude comparée de l'utilisation des *misuse cases* et de l'AMDE par un groupe d'étudiants sur un cas d'école, permettant de mieux situer avantages et limites des deux approches. Les *misuse cases* ne sont pas adaptés à toutes les situations, notamment pour les systèmes avec dominante de processus continus et physiques ; de plus, ils ne se substituent pas aux méthodes traditionnelles de la sûreté telles que celles citées en Section 2.2, mais s'inscrivent plutôt de façon complémentaire, en amont de celles-ci [446].

3.3 Un exemple d'influence mutuelle : niveaux SIL et niveaux EAL

Les sections précédentes ont présenté des influences méthodologiques entre sûreté et sécurité « à sens unique » : elles correspondaient soit à des adaptations de techniques ou d'approches du domaine de la sûreté vers celui de la sécurité (Section 3.1), soit de la sécurité vers la sûreté (Section 3.2). Ceci-dit, les inspirations et influences entre ces deux communautés ne sont pas systématiquement monodirectionnelles, et s'inscrivent parfois dans une dialectique plus subtile. Nous l'illustrons ici à travers l'exemple des niveaux SIL et des niveaux EAL (*Evaluation Assurance Levels*), évoqués respectivement en Section 2.2.2 et Section 2.3.1. Contrairement à un niveau SIL dans son domaine, un niveau EAL ne donne pas d'indication sur le niveau de sécurité dans l'absolu du produit évalué, mais sur la qualité de l'évaluation de fonctions de sécurité, définies différemment selon chaque évaluation. La complémentarité des deux approches, SIL et EAL, est remarquable [175, 21]. Elle a été soulignée à de nombreuses reprises aussi bien par la communauté sécurité [425] que par la communauté sûreté [449, 20]. Novak *et al.* s'inspirent des deux approches pour définir leur modèle de développement et d'exploitation visant à intégrer sûreté et sécurité [450, 451, 29] (nous y reviendrons au Chapitre IV).

Plusieurs projets européens ont travaillé au rapprochement des concepts de SIL et d'EAL : dès le second PCRD, le projet *Drive Safely* s'intéresse à la complémentarité des niveaux ITSEC, ancêtres des Critères Communs, et des SIL dans ce qui était encore à l'époque une version préliminaire de l'IEC 61508 [449] ; cette démarche est reprise par le projet ACRUDA [46] du 4e PCRD. Fin des années 90, le projet SQUALE [47][452] propose quant à lui d'associer à chaque grand attribut de la taxonomie de Laprie et Avizienis [16] un niveau de confiance entre 1 et 4 (cf. Table II.9), formant un *dependability profile*. S'y ajoutent des niveaux, gradués de 1 à 3, caractérisant la rigueur, le détail et l'indépendance des activités d'évaluation associées, dans l'esprit des EAL. Le livrable [453] détaille les correspondances entre les critères SQUALE et les notions de SIL telles que définies dans différents standards, ainsi qu'avec d'autres référentiels comme la DO-178B en sûreté, les ITSEC et les Critères Communs en sécurité.

À la suite du projet *Drive Safely*, l'industrie automobile britannique a également poursuivi l'intégration d'approches types SIL et EAL, pour aboutir à des résultats intégrés dans les recommandations MISRA [48][454]. La Section 3.1.3 évoque les travaux de Kube et Singer pour adapter les concepts de niveaux SIL en sécurité dans le domaine de l'informatique industrielle [425]. On notera également différentes passerelles établies entre les niveaux EAL et les niveaux de la norme DO-178B du domaine de l'avionique (cf. Section 2.2.2), comme celle proposée par Alves-Foss dans [455].

46. *Assessment Criteria and RUles for Digital Architectures* (ACRUDA).
47. *Security, Safety and Quality Evaluation for Dependable Systems* (http://research.cs.ncl.ac.uk/cabernet/www.laas.research.ec.org/squale/).
48. *The Motor Industry Software Reliability Association* (MISRA, http://www.misra.org.uk).

TABLE II.9 – Niveaux SQUALE

Attribut	Niveau de confiance
Availability (Disponibilité)	A1-A4
Confidentiality (Confidentialité)	C1-C4
Reliability (Fiabilité)	R1-R4
Integrity (Intégrité)	I1-I4
Safety (Sûreté-Innocuité)	S1-S4
Maintenability (Maintenabilité)	M1-M4

3.4 Synthèse des fertilisations croisées inventoriées

Les inspirations réciproques précédemment inventoriées ont été opérées sur des nombreux aspects, allant de concepts architecturaux (noyau, défense en profondeur, utilisation de la diversité) aux méthodes formelles, en passant par les méthodologies de test aux analyses de risques. La Table II.10 p. 65 regroupe les différentes adaptations identifiées dans les pages précédentes. On peut noter une large prédominance des inspirations allant de la sûreté vers le domaine de la sécurité. Cela peut s'expliquer notamment par la maturité plus avancée de la discipline, les premières méthodes d'analyse étant apparues dès la première moitié du XX^{e} siècle, alors que la problématique même de sécurité informatique n'avait alors pas lieu d'être [20]. Ceci-dit, la dynamique et l'exposition grandissante des problématiques de sécurité ont conduit à la constitution d'une communauté scientifique et technique aujourd'hui vivace, structurant et outillant tous les jours un peu plus le domaine par des approches rigoureuses et originales. Il ne fait pas de doutes que les adaptations de techniques de sécurité aux problématiques spécifiques de la sûreté se multiplieront à l'avenir. Nous présentons quelques pistes dans la section suivante.

3.5 Éléments de prospective

Malgré le nombre d'inspirations réciproques ayant déjà abouti à des adaptations fructueuses, la richesse des boîtes à outils des domaines sûreté et sécurité, et les barrières encore fortes entre les deux communautés, laissent entrevoir encore un large potentiel de fertilisation croisée. Nous proposons pour clore ce chapitre quelques éléments de prospective en la matière.

3.5.1 De la sécurité à la sûreté

Modélisation formelle de comportements sûrs. La Section 3.2.2 a présenté des travaux d'adaptation de modélisations formelles de politique sécurité en sûreté, menés dans les années 90. Si les résultats alors obtenus sont restés très théoriques, ils semblaient prometteurs et ouvraient des perspectives intéressantes en termes de composition de propriétés pour l'analyse de systèmes complexes. Ceci-dit, depuis, peu d'efforts semblent avoir été investis dans ce type d'exploration théorique, alors que les modèles de sécurité ont continué à évoluer. L'adaptation à la sûreté de modèles formels de sécurité plus récents que ceux considérés jusque là constitue à notre sens une piste de recherche d'intérêt.

Analyse de risques. Si certaines méthodes d'analyse de risques sûreté ont été adaptées à la sécurité (*e.g.* HAZOP, cf. Section 3.1.3), nous n'avons pas identifié d'adaptations menées dans l'autre sens. Pourtant, même si la formalisation des méthodes d'analyse de risques est plus récente en sécurité (cf. Section 2.3.3 pour des exemples), leur diversité et leur utilisation toujours plus systématique dans ce domaine nous amène à les considérer comme une réserve d'idées et d'approches méthodologiques remarquable, qui pourrait sans nul doute inspirer la communauté sûreté si elle s'y penchait avec plus d'attention.

3.5.2 De la sûreté à la sécurité

Prise en compte du facteur humain. Comme déjà affirmé en Section 1.1.4, l'importance du facteur humain est un point commun fort entre sûreté et sécurité. Nous partageons le constat de Brostoff et Sasse [422] : ces aspects bénéficient de longue date d'approches méthodologiques structurées en sûreté, dont pourrait largement bénéficier la sécurité, où leur prise en compte est plus récente. Or mis à part le

travail de ces auteurs (cf. le paragraphe sur l'adaption de GEMS Section 3.1.3), nous n'avons pas identifié de mise à profit explicite du savoir-faire acquis dans le domaine de la sûreté en sécurité informatique. Sur la base de nos recherches bibliographiques, ce champ d'investigation ne semble que trop peu exploré en comparaison de son potentiel.

Modélisations graphiques. L'adaptation de modèles issus de la sûreté en sécurité a déjà mené à des résultats significatifs, notamment au travers des arbres d'attaque présentés en Section 2.3.5.1. Ceci-dit, la richesse des modèles graphiques de la sûreté laissent entrevoir la possibilité d'adaptations encore non considérées, et aux perspectives tout aussi engageantes.

En outre, alors que l'adaptation des approches déductives a mené aux arbres de menaces et aux arbres d'attaque, les approches inductives, incarnées par la méthode des arbres d'événements, sont encore rarement considérées en sécurité informatique. À l'exception notable de Smith et Lim dans les années 80 [280], nous n'avons en effet pas pu identifier de telles utilisations. Notons qu'elles ont reçu plus d'attention pour le risque terroriste [456, 457, 458, 459], ou en sécurité physique comme dans les travaux de Cojazzi *et al.*, en association avec des arbres d'attaque [460]. À notre connaissance, aucune des approches intégrant raisonnements inductifs et déductifs, comme l'analyse par diagramme de Cause-Conséquence (cf. Section 2.2.3), n'a été adaptée au risque malveillant.

Mais c'est sans doute dans les formalismes graphiques à base de modèles dynamiques, plus adaptés aux problématiques de sécurité, que le potentiel parait le plus prometteur. Le Chapitre III est consacré à l'adaptation de l'un d'entre eux.

4 Conclusion

Dans ce chapitre, nous avons commencé par identifier et caractériser les grandes similitudes et les différences d'ordre général entre sûreté et sécurité. En outre, si la notion de risque joue dans les deux cas un rôle fondamental, la nature différente du risque considéré et la séparation historique des communautés de chaque domaine ont conduit, malgré leurs similitudes, à l'établissement de pratiques et d'approches méthodologiques distinctes. La deuxième partie a dressé un panorama des « boîtes à outils » de chaque domaine en termes de techniques et méthodes d'évaluation, au travers desquelles les ressemblances et les différences discutées précédemment transparaissent. Un rapprochement récent, et encore timide, des communautés sécurité et sûreté a déjà donné lieu à diverses adaptations d'un domaine à l'autre. Elles tirent bénéfice du cousinage précédemment caractérisé, tout en prenant en compte les spécificités du domaine cible. Certaines, comme les arbres d'attaque, ont déjà acquis une relative reconnaissance. D'autres ne font seulement qu'émerger ou n'ont pas encore été envisagées. Le potentiel pour de futures inspirations réciproques fructueuses reste important. Plusieurs pistes ont été identifiées. Parmi elles, l'adaptation de modèles graphiques dynamiques issus de la sûreté nous parait particulièrement prometteuse. Le chapitre suivant décline une telle adaptation.

TABLE II.10 – Table récapitulative des inspirations réciproques entre sûreté et sécurité

De la sûreté à la sécurité			
Type	**Outil/Méthode sûreté**	**Adaptation au domaine de la sécurité**	**Références**[49]
Concept d'architecture	Architectures de tolérance de fautes	Architectures et techniques de tolérance aux intrusions Technique FRS ; *survivable networks* Détection d'intrusions améliorée par diversification	[388, 391, 393, 395] [388, 390] [396] [400]
	Défense en profondeur	Défense en profondeur	[403, 23]
Modélisation graphique	Arbres de défaillances	Arbres de menaces, arbres d'attaque	[276, 22]
	Arbres de défaillances dynamiques	Arbres d'attaque dynamiques	[320]
Analyse de risques	HAZOP	HAZOP pour la sécurité *Vulnerability Identification and Analysis* (VIA) HAZOPs	[406] [407, 261] [408]
	Sneak Path analysis	*Sneak path security analysis*	[409, 410]
	Zonal analysis	*Security zonal analysis*	[408]
	Safety cases, *Goal Structured Notation* (GSN)	*Security Assurance case*	[415, 418, 419]
	FMEA	*IMEA (Intrusion Modes and Effects Analysis)*	[411, 412]
	GEMS	GEMS pour la sécurité	[422]
Tests	Niveaux SIL	Niveaux SAL (*Security Assurance Levels)*	[425, 461]
	Injection de fautes	Injection de fautes, *Fuzzing*	[429]
	Software Reliability Growth modeling	*Software Security Growth modeling*	[431]
De la sécurité à la sûreté			
Type	**Outil/Méthode sécurité**	**Adaptation au domaine de la sûreté**	**Références**[49]
Architecture	Noyau de sécurité (*security kernel*)	Noyau de sûreté (*safety kernel*)	[435, 436]
Modélisation graphique	*Misuse case*	*Misuse case* pour la sûreté	[446, 447, 374, 448]
Modèle formel	Propriété de *Non-interference*... ...et propriétés de *Non deducibility* et de *Causality*	Formalisation de comportements sûrs (*fail-safe*, *fail-stop*,...)	[439] [442]
	Modèles formels de contrôle d'accès orientés intégrité (notamment Biba)	Politique de contrôle d'intégrité avec niveaux multiples de criticité (modèle Totel)	[443]

49. Les références indiquées correspondent aux premiers travaux initiant l'adaptation de la technique considérée ; il ne s'agit pas d'une bibliographie exhaustive.

Chapitre III

Adaptation du formalisme BDMP au domaine de la sécurité

Tous les modèles sont faux, mais certains sont plus utiles que d'autres.
(P. McCullagh et J.A. Nelder, *Generalized Linear Models*, 1989.)

Le précédent chapitre s'est achevé sur un panorama des inspirations réciproques entre les méthodes et outils de la sécurité et ceux de la sûreté. Si ce type de démarche a déjà conduit à de nombreuses adaptions fructueuses, il est encore riche de potentiel, notamment dans le domaine des modèles graphiques. Les BDMP (*Boolean logic Driven Markov Processes*) désignent un formalisme de modélisation graphique souple et puissant issu du domaine des études de fiabilité et de sûreté. Dans ce chapitre, nous proposons de mettre à profit ses capacités dans le domaine de la sécurité. Une telle adaptation permet de dépasser bien des limites inhérentes aux techniques traditionnellement utilisées pour modéliser des scénarios d'attaque.

La Section 1 est une revue critique des grandes familles de techniques de modélisation graphique d'attaque. En Section 2, nous développons les principes et la théorie fondant l'adaptation des BDMP à la modélisation d'attaque. Les extensions spécifiquement conçues pour la prise en compte des aspects de détection et de réaction sont exposées en Section 3. La Section 4 explique comment se situe ce nouveau formalisme par rapport à l'état de l'art. La Section 5 présente les développements logiciels ayant abouti à un outil opérationnel mettant en œuvre les travaux théoriques précédemment exposés. Enfin, la Section 6 regroupe les considérations associées aux hypothèses du modèle et à leurs limites ; y sont aussi décrites les perspectives ouvertes et les pistes de développements futurs.

1 Revue critique des formalismes de modélisation graphique d'attaque

1.1 Considérations préliminaires

1.1.1 Périmètre et objectif

Les formalismes de modélisation graphique d'attaque ont pour but de permettre la représentation graphique et l'analyse des différents scénarios envisageables par un attaquant pour atteindre un ou plusieurs objectifs. Le point de vue peut être celui de l'attaquant, celui du système attaqué, ou encore un point de vue neutre. Ces modélisations peuvent également prendre en compte des aspects défensifs, dans le but d'évaluer l'efficacité de contre-mesures. Le domaine de la modélisation graphique d'attaque est foisonnant. Comme déjà mentionné, cette richesse s'explique par l'utilité manifeste de ces approches dans la conduite des évaluations de sécurité et des analyses de risques : elles facilitent notamment la diversité et la mise en commun nécessaires des expertises dans la recherche de vulnérabilités de systèmes, et jouent également un rôle clé dans la communication des risques et la sensibilisation relative à ceux-ci [323]. Un état de l'art est donné dans le Chapitre II, en Section 2.3.5.

Dans le cadre de la présente section, nous nous intéressons plus particulièrement aux formalismes qui permettent la construction et la modification des modèles de façon graphique, que ce soit à la main ou par l'intermédiaire d'un outil de saisie informatisé. Cette précision a son importance car certains types de techniques mentionnées en II-2.3.5 conduisent à des représentations graphiques, mais ne répondent pas à ce critère. C'est notamment le cas des approches à base de *model-checking* et de génération automatisée de graphes d'attaque, que nous ne prendrons pas en compte dans ce chapitre.

Notre objectif est ici de dépasser les considérations historiques et les descriptions du Chapitre II pour analyser de façon plus critique l'état de l'art du domaine. Nous nous appuyons pour cela sur les regroupements de la Table II.7 (p. 54) et approfondissons les premiers éléments critiques qui y figuraient.

1.1.2 Méthodologie et cas d'application

Afin de compléter les appréciations de la Table II.7 et d'illustrer de façon visuelle les différentes familles de formalismes mentionnées, nous avons modélisé une même situation avec des approches représentatives de ces familles. Le cas d'étude choisi, encore communément rencontré dans le domaine de l'informatique industrielle [462, 463] et introduit dans [464], peut être décrit informellement de la façon suivante : l'objectif de l'attaquant est de prendre possession à distance d'un serveur d'accès distant accessible seulement par modem. Pour cela, on suppose qu'il ne peut trouver ce modem que par *wardialing* [1]. De plus, étant donné la configuration du système et les ressources de l'attaquant, celui-ci a pour seules alternatives d'attaquer le mot de passe protégeant l'accès au serveur ou de trouver et exploiter une vulnérabilité logicielle. Pour la première alternative, les seules techniques d'attaque considérées sont l'attaque dite par force brute (recherche exhaustive) et par *social engineering* ; pour la seconde alternative, aucune distinction n'est faite quant à la nature de la vulnérabilité recherchée et exploitée ; elle peut indistinctement concerner le modem, le serveur ou les protocoles de communication et d'authentification. La dimension défensive n'est pas précisée.

NB : Cette description ne prend en compte qu'un nombre limité d'alternatives et reste très générale afin d'une part d'obtenir des modèles de petite taille facilitant la comparaison, et de rester d'autre part à la portée des approches présentant les capacités de modélisation les plus limitées. Le paramétrage des modèles n'est pas abordé pour les mêmes raisons.

1.1.3 Critères d'évaluation

Dans une approche librement inspirée du livre de Bouissou sur la modélisation en sûreté de fonctionnement [176], nous caractérisons chaque formalisme selon cinq critères :

- **facilité d'apprentissage** : sans doute le plus subjectif des cinq critères, elle exprime l'effort d'appropriation impliqué par une mise en œuvre basique du formalisme. Elle est en outre liée aux pré-requis théoriques nécessaires pour l'utiliser et au caractère plus ou moins intuitif de ses composants graphiques et de sa logique de fonctionnement ;

1. Désigne une technique de recherche de modem par balayage automatisé de plages de numéros téléphoniques.

- **lisibilité** : nous entendons par lisibilité la capacité du formalisme à produire des modèles faciles à appréhender par l'analyste et supportant ses raisonnements. Elle est entre autres choses corrélée à l'expressivité, mais aussi au nombre des éléments graphiques nécessaires à la saisie d'un modèle ;
- **puissance de modélisation** : ce critère reflète à la fois la largeur du spectre de scénarios d'attaques que peut couvrir le formalisme, et sa capacité à prendre en compte leurs aspects les plus significatifs. Les capacités à modéliser séquences, réactions ou cycles sont particulièrement considérées ;
- **capacité de quantification** : il s'agit d'évaluer ici d'une part la diversité et l'adéquation des quantifications vis-à-vis des besoins d'une analyse de risques en sécurité, et d'autre part, l'existence et l'efficacité des techniques et outils permettant ces quantifications ;
- **capacité de passage à l'échelle** : ce critère correspond à la notion anglo-saxonne de *scalability* et d'aptitude à la modélisation de systèmes complexes par leur taille, le nombre de leurs éléments et la diversité de leurs composants. Elle dépend, en outre, de la capacité du formalisme à être factorisé et composé.

Une note entre 1 (la plus faible) et 4 (la plus forte) est attribuée pour chacun des cinq critères, sur la base des éléments de l'état de l'art donné en II-2.3.5 ; elle est également fondée sur la déclinaison du cas d'étude dans chaque formalisme. Malgré nos efforts bibliographiques, une telle évaluation garde bien sûr une part non négligeable de subjectivité. Son but est avant tout d'aider à mieux situer les formalismes précédemment évoqués les uns par rapport aux autres, mais également de caractériser le contexte et les carences ayant mené au développement de la contribution associée à ce chapitre, exposée dans les Sections 2 et suivantes.

1.2 Éléments de comparaison

1.2.1 Arbres d'attaque

La Figure III.1 correspond à la modélisation du cas d'étude choisi pour cette section par arbre d'attaque classique (cf. II-2.3.5.1). Le modèle étant statique, les notions de séquence ne sont pas prises en compte : par exemple, dans le cas d'étude modélisé, la recherche d'une vulnérabilité semble être simultanée avec son exploitation. Le formalisme est clair ; sa nature hiérarchique permettrait d'affiner l'analyse de façon ciblée, en décomposant une ou plusieurs feuilles de l'arbre. Diverses techniques d'analyse et de quantification inspirées de la sûreté de fonctionnement sont envisageables (cf. II-2.3.5.1). La Table III.1 synthétise notre évaluation.

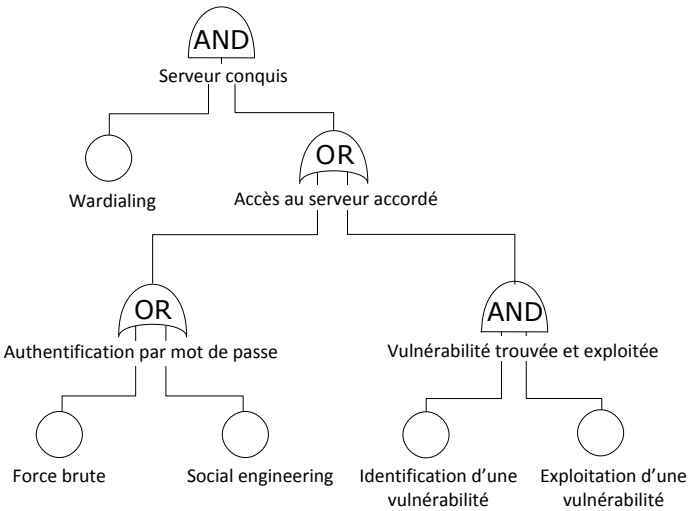


FIGURE III.1 – Modélisation du cas d'étude par arbre d'attaque

TABLE III.1 – Appréciation des arbres d'attaque pour la modélisation graphique d'attaque

Critère	Note	Commentaires
Facilité d'apprentissage	4	L'appropriation du formalisme est très rapide, aussi bien sur le fond que sur la forme.
Lisibilité	4	La sémantique est simple et intuitive, avec un nombre d'éléments limité et stable.
Puissance de modélisation	2	Les comportements séquentiels et les dépendances ne peuvent pas être pris en compte. Impossible aussi de prendre en compte des boucles.
Capacité de quantification	4	Toutes les techniques associées aux arbres de défaillances sont applicables. Quelques quantifications spécifiques aux arbres d'attaque (*e.g.* calcul de coût, etc.).
Capacité de passage à l'échelle	3	La nature hiérarchique du modèle et la possibilité de raisonner en *pattern* permettent de traiter des scénarios d'attaques complexes sur des systèmes réalistes.

1.2.2 Réseaux bayésiens

Nous adoptons dans la Figure III.2 le formalisme utilisé par Sommestad *et al.* dans [363, 364, 365, 366, 367], à savoir les diagrammes d'influence étendus. Les relations entre les différents nœuds correspondent à des opérateurs AND et OR, à l'instar des arbres d'attaque, et qui sont ici traduits par des arcs de définition (nous renvoyons le lecteur aux papiers de Sommestad *et al.* pour les tables de probabilités conditionnelles correspondantes). Notons que dans ce cas de figure, il serait trivial de revenir à une modélisation en réseau bayésien classique, les arcs de définition correspondant à des arcs déterministes et le nœud d'utilité Serveur conquis n'étant pas nécessaire. Le modèle résultant est visuellement très proche de l'arbre d'attaque équivalent. Par rapport à celui-ci, il ne présente pas d'intérêt particulier, ni graphiquement, ni sur le plan des quantifications. Ceci-dit, il serait simple d'ajouter des variables influençant le paramétrage de réussite des actions dans des modalités qu'un arbre d'attaque ne pourrait pas exprimer. Les réseaux bayésiens ont un pouvoir de modélisation supérieur. La Table III.2 résume notre appréciation.

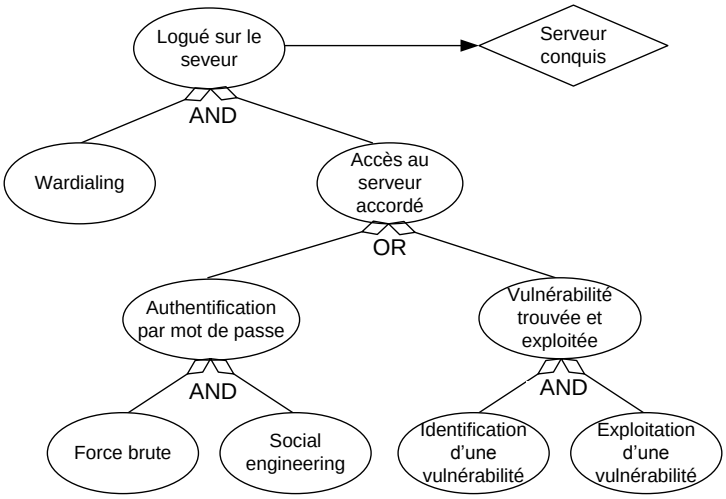


FIGURE III.2 – Modélisation du cas d'étude par diagramme d'influence étendu

1.2.3 Diagrammes de *misuse case*

La Figure III.3 correspond à la modélisation en diagramme de *misuse case*[2] du cas d'étude. Les ovales en noir représentent à proprement parler les *misuse cases*, qui correspondent ici aux techniques d'attaque, les ovales en blanc correspondent aux utilisations normales du système (*use cases*), qui vont

2. Ce terme fait écho aux diagrammes de *use cases* (cas d'usage), le préfixe *-mis* indiquant une mauvaise utilisation. Il pourrait être littéralement traduit par « cas de mal-usage », mais nous préférons garder la dénomination d'origine. Par ailleurs, dans la littérature, le terme *misuse case* désigne par synecdoque à la fois le formalisme de façon générale, un diagramme particulier, ou un type spécifique d'éléments composant ces diagrammes.

TABLE III.2 – Appréciation des réseaux bayésiens pour la modélisation graphique d'attaque

Critère	Note	Commentaires
Facilité d'apprentissage	2	Les bases théoriques (probabilités conditionnelles et théorème de Bayes), bien que simples, doivent être assimilées même pour une utilisation basique. La prise en main est moins accessible que pour les arbres d'attaque.
Lisibilité	2	Les réseaux bayésiens bruts comportent très peu d'informations visuelles. Des formes évoluées comme les diagrammes d'influence étendus améliorent un peu la situation.
Puissance de modélisation	3	Plus généraux que les arbres d'attaque, les modèles à base de réseaux bayésiens restent des modèles statiques, modélisant difficilement les aspects séquentiels.
Capacité de quantification	3	Il existe de nombreux outils de calcul, mais la diversité des traitements est réduite.
Capacité de passage à l'échelle	3	Une partie de la complexité des modèles peut être traitée dans les tables de probabilités conditionnelles.

être « abusées » par l'attaquant. La présence sur un même diagramme des *misuse cases* et de *use cases* est en soi une spécificité du formalisme, permettant de mieux appréhender les interactions et d'avoir une vue globale du système. Certaines relations entre les éléments ne sont pas représentées mais peuvent être spécifiées en attributs des éléments graphiques, pour par exemple prendre en compte d'éventuelles dépendances (pré-conditions, post-conditions, notion de déclenchement). Dans le présent cas d'étude, ceci permet de préciser, mais de façon non graphique, que la ligne de communication doit d'abord être trouvée par *wardialing* avant de considérer les autres *misuse cases*. De plus, comme le souligne un de ses créateurs [375], une des faiblesses du formalisme tient dans sa flexibilité même et la grande liberté d'utilisation associée. Ainsi, une même situation peut être modélisée de très nombreuses façons, avec des descriptions et attributs associés aux composants tout aussi variés.

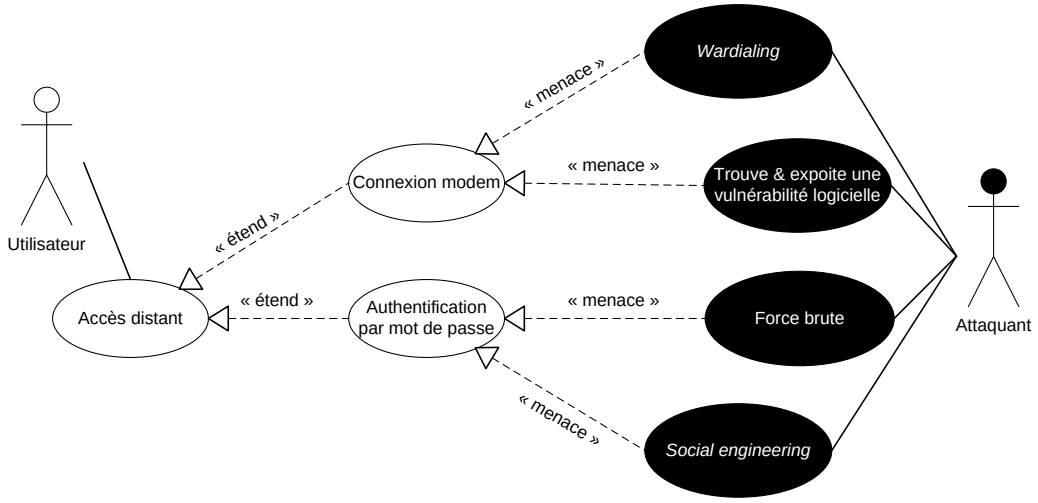


FIGURE III.3 – Modélisation du cas d'étude par *misuse case*

Dans la Figure III.4, nous avons modélisé le cas d'étude avec la notation augmentée introduite par Røstad dans [376]. Le diagramme donne là encore une bonne vue d'ensemble, et permet de dissocier les attaques (ici les *misuse cases* en noir) des vulnérabilités exploitées (indiquées en gris). La modélisation par arbre d'attaque tend à mélanger ces deux aspects, que ne distingue pas non plus le formalisme original des *misuse cases*.

La Table III.3 synthétise notre appréciation des diagrammes de *misuse case* selon les cinq critères d'évaluation convenus en début de section.

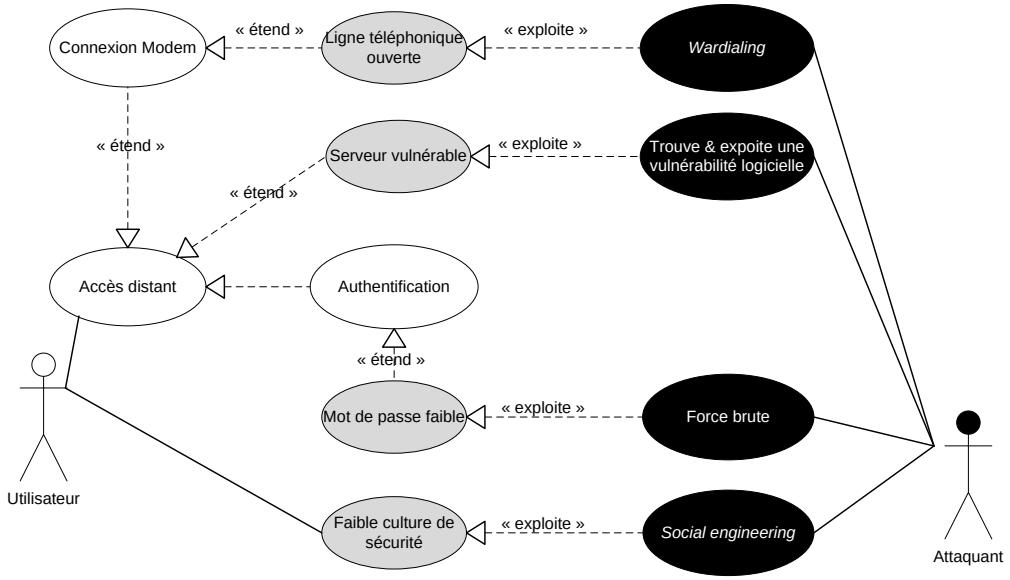


FIGURE III.4 – Modélisation du cas d'étude par *misuse case* étendu (notation de Røstad [376])

TABLE III.3 – Appréciation des *misuse cases* pour la modélisation graphique d'attaque

Critère	**Note**	**Commentaires**
Facilité d'apprentissage	3	La sémantique est explicite, le formalisme très souple dans son utilisation.
Lisibilité	2	Les diagrammes deviennent rapidement confus dès que l'on quitte les cas d'école.
Puissance de modélisation	1	Orientés spécification dans une approche formalisée graphiquement, ils restent très macroscopiques. Non adaptés à certains types d'attaques [375].
Capacité de quantification	1	Aucune.
Capacité de passage à l'échelle	2	Mal adaptés aux scénarios complexes. Leur caractère composable permet de limiter ce défaut : les notions de *templates* et de *patterns* ont été formalisées [372, 373, 465].

1.2.4 Les *Dynamic Fault Trees* comme variantes dynamiques des arbres d'attaque

La Figure III.5 représente le cas d'usage en *Dynamic Fault Tree* (DFT). La différence principale avec le modèle en arbre d'attaque tient dans l'utilisation de portes SEQ, dont les fils ne peuvent se réaliser que dans une séquence bien déterminée (graphiquement de gauche à droite). La notation ici adoptée reprend la notation habituelle des DFT tels qu'utilisés en sûreté de fonctionnement (cf. Section II.3), et non celle de Khand, seul auteur à notre connaissance à avoir proposé l'utilisation de DFT en sécurité [320]. Les portes SEQ permettent de prendre en compte la dimension séquentielle, significative dans les scénarios d'attaques, qui échappe aux arbres d'attaque. Le cas d'étude ne nécessite pas l'emploi des autres portes dynamiques (*e.g.* FDEP) qui élargissent les capacités de modélisation, hélas au prix de modèles moins lisibles. Comme l'illustre la porte SEQ employée dans la Figure III.5, ces portes dynamiques spécifient des comportements prenant en compte la position des éléments fils (le fils de droite ne peut se réaliser qu'une fois le fils de gauche réalisé).

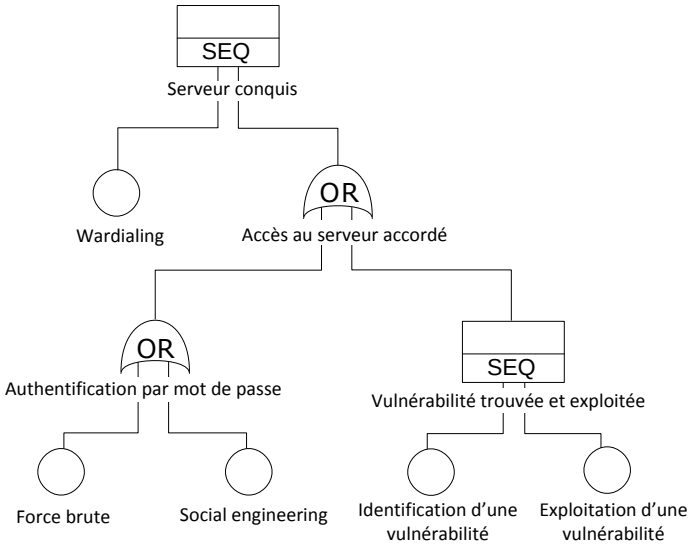


FIGURE III.5 – Modélisation du cas d'étude par *Dynamic Fault Tree*

TABLE III.4 – Appréciation des DFT pour la modélisation graphique d'attaque

Critère	**Note**	**Commentaires**
Facilité d'apprentissage	3	Intégration de nouvelles portes dans un formalisme proche des arbres d'attaque.
Lisibilité	3	Le comportement des nouvelles portes n'est pas explicite ; la lecture et la compréhension sont moins intuitives qu'avec les arbres classiques, mais restent faciles.
Puissance de modélisation	3	Les nouvelles portes augmentent les capacités de modélisation des arbres d'attaque en intégrant des formes de dépendance et de séquence, mais les cycles restent proscrits.
Capacité de quantification	2	L'article de Khand [320] n'aborde pas les aspects de quantification de son adaptation. Les DFT utilisés en sûreté de fonctionnement peuvent bénéficier des outils d'analyse markovienne, avec certaines limites. De plus, la modélisation stochastique est alors limitée à des lois exponentielles.
Capacité de passage à l'échelle	3	La nature hiérarchique du modèle facilite le passage à l'échelle. Les approches de type *patterns* utilisées avec les arbres d'attaque classiques sont ici aussi pertinentes.

1.2.5 Modèles basés sur les réseaux de Petri, cas d'étude en GSPN

Lors de la présentation des réseaux de Petri en II-2.3.5.1, nous avons souligné la grande diversité des formalismes de cette famille. Elle se retrouve dans leur emploi en matière de modélisation graphique d'attaque. Nous avons ici choisi d'utiliser les GSPN (*Generalized Stochastic Petri Nets*), forme bien connue dans le domaine de la sûreté de fonctionnement et que l'on trouve également bien représentée dans l'emploi des réseaux de Petri en sécurité. Notre choix aurait aussi pu se porter sur les SAN (*Stochastic Activity Networks*) ou sur une des variantes des CP-nets (*Coloured Petri Nets*), ces formalismes occupant également une place significative dans l'état de l'art en sécurité. Ceci-dit, les premiers ont été jugés moins représentatifs des réseaux de Petri, introduisant des portes absentes de la grande majorité des autres variantes et substituant les transitions par des notions d'activités. Les seconds ajoutent le concept de couleur qui n'est pas nécessaire à la modélisation du cas d'étude.

La Figure III.6 représente la modélisation du cas d'étude en GSPN, effectuée avec l'aide de Marc Bouissou [464]. Les transitions « pleines », en noir, correspondent à des transitions temporisées alors que les transitions « creuses », en blanc sont des transitions instantanées. Les arcs en pointillé fin sont des arcs inhibiteurs : ils sont nécessaires si l'on souhaite modéliser le fait qu'une étape franchie dans le parcours de l'attaquant reste « réalisée » quand celui-ci progresse dans l'attaque et passe à une autre étape (comportement équivalent aux fonctions de structure d'un arbre d'attaque). La Section 4.5 développe plus précisément ces aspects de modélisation.

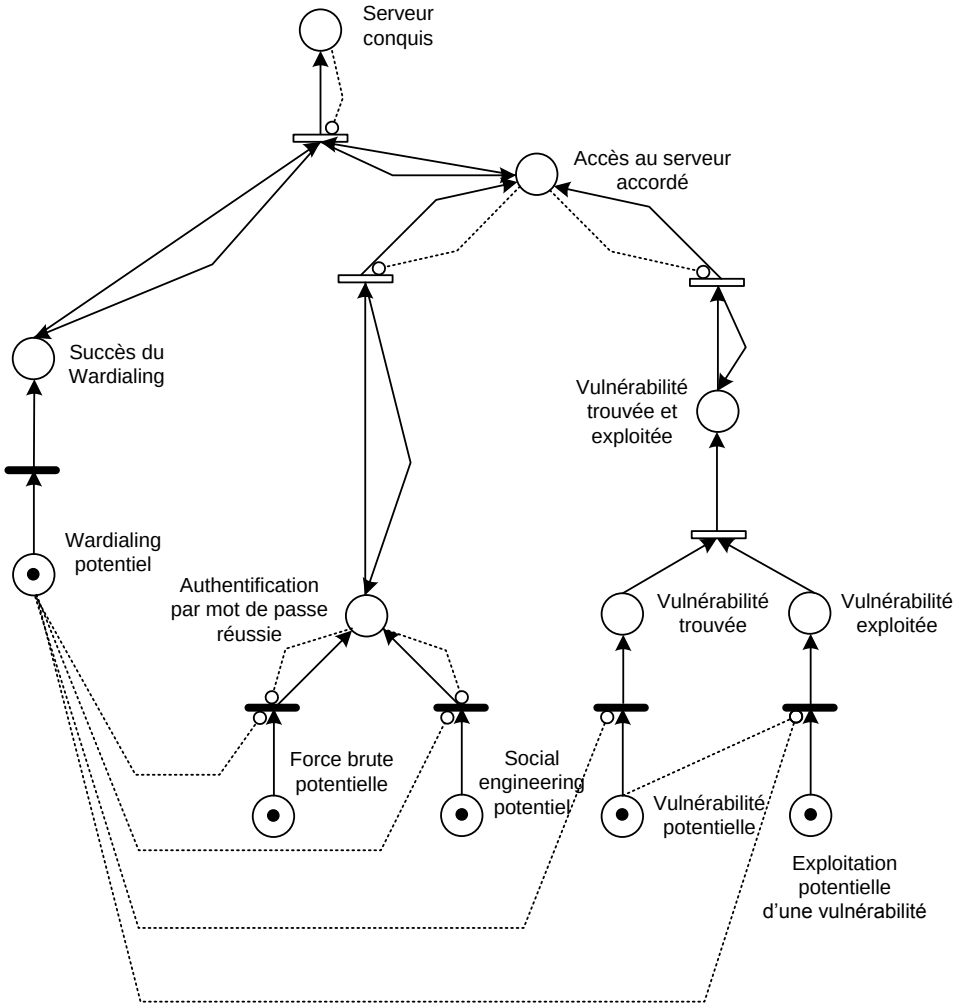


FIGURE III.6 – Modélisation du cas d'étude en GSPN

La Table III.5 propose une appréciation de l'utilisation des réseaux de Petri pour la modélisation graphique d'attaque. Leur grande diversité rend l'évaluation encore plus délicate que pour les autres formalismes. Nous avons essayé de rester le plus générique et objectif possible.

TABLE III.5 – Appréciation des réseaux de Petri pour la modélisation graphique d'attaque

Critère	**Note**	**Commentaires**
Facilité d'apprentissage	2	Les réseaux de Petri sont moins accessibles que les autres formalismes présentés. En outre, le fonctionnement n'est pas aussi intuitif qu'il y parait (*e.g.* les jetons ne « traversent » pas les transitions mais sont créés et détruits).
Lisibilité	2	Dépendante de la variante considérée. D'un point de vue général, la grande diversité des réseaux de Petri dessert leur lisibilité : un même élément graphique a différentes significations selon la variante, voire dans une même variante (*e.g.* SAN).
Puissance de modélisation	4	Tout type de situation peut être représenté, y compris synchronisations, boucles, partages, concurrences, conflits...
Capacité de quantification	4	De nombreux outils, techniques et algorithmes efficaces sont disponibles.
Capacité de passage à l'échelle	2	Les réseaux deviennent rapidement complexes. Leur composition est délicate ; à part pour quelques variantes très spécifiques, il est difficile de réutiliser des sous-modèles.

1.3 Synthèse

Le Tableau III.6 regroupe les notes attribuées aux cinq familles de formalismes évaluées, sans reprendre les justifications (se référer aux paragraphes dédiés pour cela).

TABLE III.6 – Évaluation comparée de cinq formalismes de modélisation graphique d'attaque

Formalisme **Critère**	Arbres d'attaque	Modèles à base de réseaux bayésiens	*Misuse cases*	DFT	Modèles à base de réseaux de Petri
Facilité d'apprentissage	4	2	3	3	2
Lisibilité	4	2	2	3	2
Puissance de modélisation	2	3	1	3	4
Capacité de quantification	4	3	1	2	4
Capacité de passage à l'échelle	3	3	2	3	2

La popularité et le succès des arbres d'attaque sont cohérents avec cette évaluation comparée. Ils ont de bonnes capacités à tous les plans sauf en termes de puissance de modélisation, intrinsèquement limitée par leur caractère statique. L'utilisation de DFT améliore la situation, mais au détriment des autres critères. Ceci-dit, les DFT nous paraissent offrir un formalisme aux capacités équilibrées, qui méritent sans doute plus que la seule publication trouvée au sujet de leur utilisation en sécurité [320]. Les modèles à base de réseaux bayésiens ont une bonne capacité de modélisation et offrent des possibilités de quantification avancées. Leur emploi en sécurité est encore récent, mais les travaux de Sommestad *et al.*, et l'utilisation du formalisme de diagramme d'influence étendu en font déjà un outil opérationnel. Les *misuse cases* sont très macroscopiques et présentent peu d'avantages considérés sous le seul angle de la modélisation d'attaque. Ils s'inscrivent dans une démarche plus large de spécification de systèmes, non prise en compte ici. Enfin, les réseaux de Petri sont sans doute le formalisme aux capacités de modélisation et de quantification les plus avancées, mais ils sont moins accessibles et moins lisibles que la plupart des autres formalismes.

En conclusion, notre état de l'art suggère qu'un formalisme qui conserverait les qualités des arbres d'attaque en améliorant leur puissance de modélisation trouverait une place d'intérêt dans le domaine des formalismes de modélisation graphique d'attaque.

2 Modélisation d'attaques par les BDMP

2.1 De la sûreté de fonctionnement à la sécurité

2.1.1 Origine et présentation générale des BDMP

Les BDMP (*Boolean logic Driven Markov Processes*) désignent un formalisme graphique issu du domaine de la sûreté de fonctionnement. Inventés par Bouissou au début des années 2000, ils ont depuis été employés à EDF dans des études de sûreté de systèmes élémentaires de centrales nucléaires, de systèmes d'évacuation des crues de barrages hydrauliques, dans des analyses de disponibilité de postes électriques [466] ou d'alimentation électrique d'installations diverses (*e.g. data centers* [467], usines, aéroports). Ils combinent l'aspect visuel des arbres de défaillances, héritant de leur lisibilité et de leur facilité d'appropriation (cf. Chap. II Section 2.3.5.1), avec la puissance de modélisation des modèles de Markov (cf. Chap. II Section 2.2.4). En première approche, on peut présenter les BDMP comme modifiant la sémantique classique des arbres de défaillances selon deux modalités principales :

- ils associent aux feuilles de l'arbre [3] des processus de Markov qui modélisent le comportement des composants selon plusieurs modes. Plus explicitement, chaque feuille peut être considérée dans un mode « sollicité », correspondant à un état du système où le composant modélisé par la feuille contribue au fonctionnement global du système, ou dans un mode « non-sollicité », qui signifie que le composant correspondant n'est pas requis (il est par exemple au repos car en redondance à froid). Un processus de Markov simple modélise pour chaque mode le comportement du composant en termes de défaillance et de réparation. La Figure III.7 correspond aux processus d'une feuille permettant de modéliser les redondances. La valeur des taux de défaillance (λ) diffère selon le mode (celle du taux de réparation μ est par contre supposée identique) ; la feuille correspond à une redondance à froid quand $\lambda_0 = 0$.
- ils introduisent un nouveau type de lien, nommé gâchette, représenté par une flèche rouge en pointillé, qui permet de sélectionner le mode des feuilles en fonction de l'état d'autres feuilles. En d'autres termes, ce lien spécifie graphiquement de quels autres composants le mode de sollicitation d'un composant dépend.

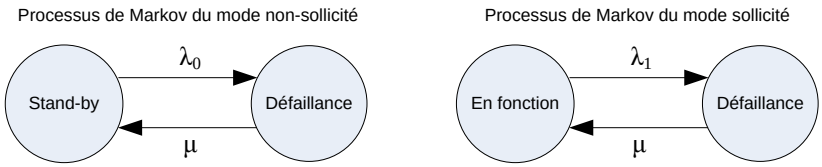


FIGURE III.7 – Exemples de processus de Markov associés aux modes des feuilles

Pour une feuille donnée, les processus correspondant à chaque mode et les fonctions spécifiant le passage d'un mode à l'autre, déclenché notamment par les gâchettes, forment un « processus de Markov piloté ». La Figure III.8 représente un BDMP [4] élémentaire, avec ses composants de base : feuilles (f1 à f4), portes logiques (G1, G2 et G3), événement redouté (r), et une gâchette entre les portes G1 et G2. Par l'entremise de la gâchette, le mode de f3 et f4 dépend de la réalisation de la porte G1, et donc *a fortiori* des feuilles f1 et f2.

Ce formalisme offre trois avantages essentiels par rapport aux autres modèles dynamiques en sûreté de fonctionnement :

- d'une part, il permet la définition de modèles dynamiques complexes tout en restant presque aussi lisible et facile à construire qu'un arbre de défaillances. En particulier, les BDMP peuvent être utilisés pour construire simplement et rapidement des modèles correspondant à de nombreuses situations courantes dans les études de sûreté, telles que redondances passives, simples ou en cascade,

3. Nous utiliserons le terme « arbre » pour les BDMP, en référence aux arbres d'attaque et aux arbres de défaillances, bien qu'il s'agisse formellement d'un graphe orienté sans circuit (cf. Section 2.2).

4. Par abus de langage, nous désignerons aussi par BDMP une instance particulière du modèle général.

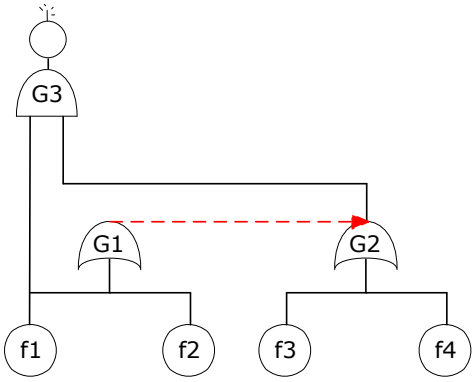


FIGURE III.8 – Exemple d'un BDMP simple [238]

défaillances de cause commune, reports de charge, fonctionnements différenciés selon les séquences d'événements, etc. ;

- d'autre part, leurs propriétés mathématiques autorisent le traitement efficace de BDMP équivalents à des processus de Markov avec un espace d'états extrêmement grand. Un mécanisme d'élagage, dit de « filtrage des événements pertinents », permet en effet de réduire considérablement la combinatoire dans l'exploration des chemins menant à l'événement redouté, effectuée lors du traitement du modèle ;
- enfin, en plus des calculs classiques de disponibilité et de fiabilité, ils permettent d'obtenir des informations qualitatives d'intérêt sous la forme de listes des séquences menant à l'événement redouté, caractérisées quantitativement et ordonnées selon leur contribution à la probabilité d'occurrence de l'événement redouté dans le temps de mission considéré pour le système.

Une définition complète de la formalisation mathématique des BDMP et de leurs propriétés est donnée dans [238]. Nous la reprenons largement dans la section suivante pour l'adapter à la modélisation sécurité.

2.1.2 Les BDMP appliqués à la sécurité

Tout comme les BDMP ont à l'origine assigné une nouvelle sémantique aux arbres de défaillances, il est possible de tirer bénéfice de ce formalisme en sécurité, en modifiant la sémantique des arbres d'attaque. Dans ce cas, les feuilles sont également associées à des processus de Markov pilotés, mais qui représentent non plus le comportement de panne des composants d'un système, mais des actions ou des événements élémentaires dont la réalisation peuvent servir à un objectif d'attaque. Pour cela, les processus de Markov pilotés ont dans un cadre sécurité les caractéristiques suivantes :

- ils ont deux modes, *Actif* et *Inactif*, correspondant respectivement au fait que l'action ou l'événement qu'ils modélisent peut ou ne peut pas encore être réalisé, étant donné la progression de l'attaque. Dans le cadre théorique présenté dans la section suivante, nous les désignerons aussi par mode 1 (Actif) et mode 0 (Inactif) ;
- à tout moment, le choix du mode d'un processus de Markov piloté dépend d'une fonction booléenne de l'état des autres processus du BDMP, spécifiée notamment par les gâchettes. Plus concrètement, celles-ci vont permettre de modéliser les séquences d'activation des différentes feuilles du BDMP, caractérisant la progression dans les scénarios d'attaque possibles capturés par l'ensemble de l'arbre. Des exemples ultérieurs illustrent ce principe.

2.2 Définition formelle

2.2.1 Les composants d'un BDMP

Notons P_i les processus de Markov pilotés associés aux feuilles i d'un arbre d'attaque $\mathcal{A}$. Formellement, un BDMP utilisé pour la sécurité est un ensemble $\{\mathcal{A}, r, T, P\}$ composé de :

- un arbre d'attaque $\mathcal{A} = \{E, L, g\}$ où :
  - $E = G \cup B$, est un ensemble d'éléments avec G, l'ensemble des portes logiques, et B, l'ensemble des événements de base jouant un rôle dans la progression de l'attaque (par exemple, des actions de l'attaquant). Les événements de base constituent les feuilles du BDMP,
  - $L \subset G \times E$, est un ensemble d'arêtes orientées, telles que (E, L) soit un graphe orienté sans circuit avec $\forall i \in G, fils(i) \neq \emptyset$ et $\forall j \in B, fils(j) = \emptyset$ (on note $E \xrightarrow{fils} 2^E$, $fils(i) = \{j \in E/(i,j) \in L\}$),
  - $g : G \to \mathbb{N}^*$ est une fonction définissant le paramètre k des portes logiques, qui sont toutes considérées comme des portes k/n (avec $k = 1$ pour les portes OU et $k = n$ pour les portes ET, n étant le nombre de fils de la porte [5]).
- r, l'objectif final de l'attaquant, équivalent de l'événement redouté en sûreté de fonctionnement (sommet ou *top event*). Formellement, il correspond à une des racines de (E, L) ;
- un ensemble de gâchettes T, défini comme un sous ensemble de $(E - \{r\}) \times (E - \{r\})$ tel que $\forall (i,j) \in T, i \neq j$ et $\forall (i,j) \in T, \forall (k,l) \in T, i \neq k \Rightarrow j \neq l$. Si i est appelé origine, et j cible, cela signifie que l'origine et la cible d'une gâchette doivent être différentes, et que deux gâchettes ne peuvent pas avoir la même cible. Les gâchettes sont représentées par des flèches rouges en pointillé ;
- un ensemble P de processus de Markov pilotés $\{P_i\}_{i \in B}$. Un P_i se définit comme un ensemble $\left\{Z_0^i(t), Z_1^i(t), f_{0\to1}^i, f_{1\to0}^i\right\}$ où :
  - $Z_0^i(t)$ et $Z_1^i(t)$ sont deux processus de Markov homogènes à états discrets modélisant le comportement de la feuille i selon son mode. Pour k dans $\{0, 1\}$ représentant le mode de la feuille i considérée, l'espace des états de $Z_k^i(t)$ est A_k^i. Chaque A_k^i contient un sous-ensemble S_k^i correspondant aux états de succès ou de réalisation de l'événement de base modélisé par le processus P_i ;
  - $f_{0\to1}^i$ et $f_{1\to0}^i$ sont deux « fonctions de transfert de probabilités ». Elles décrivent en termes de probabilités les transferts entre processus de chaque mode au moment du basculement d'un mode à l'autre. La probabilité d'arriver dans un état donné du processus du mode cible dépend de l'état du processus de la feuille dans le mode de départ, au moment du basculement. Par exemple, une telle fonction peut spécifier que pour une feuille en mode Inactif dans un état de non-réalisation, un changement de mode l'amènera dans un état de réalisation avec une probabilité γ et dans un état de non-réalisation avec une probabilité 1 - γ. Ces fonctions sont définies formellement comme suit :
    - pour tout $x \in A_0^i$, $f_{0\to1}^i(x)$ est une distribution de probabilités sur A_1^i telle que si $x \in S_0^i$, alors $\sum_{j \in S_1^i} (f_{0\to1}^i(x))(j) = 1$,
    - pour tout $x \in A_1^i$, $f_{1\to0}^i(x)$ est une distribution de probabilités sur A_0^i telle que si $x \in S_1^i$, alors $\sum_{j \in S_0^i} (f_{1\to0}^i(x))(j) = 1$.

    (Ces conditions signifient simplement qu'une feuille réalisée juste avant un changement de mode ne peut devenir non-réalisée à cause du changement de mode.)

Gâchettes et processus de Markov pilotés P_i sont intimement liés : les P_i basculent automatiquement d'un mode à l'autre, *via* les fonctions de transfert de probabilités appropriées, selon l'état de variables booléennes externes au processus, appelées sélecteurs de mode (cf. Section 2.2.2). La valeur des sélecteurs de mode est définie par l'entremise des gâchettes. Une gâchette modifie le mode des P_i associés aux feuilles du sous-arbre vers lequel elle pointe. Dans le cas simple où une seule gâchette est présente dans le modèle, quand la feuille ou la porte à l'origine d'une gâchette passe de FAUX à VRAI, le mode des feuilles du sous-arbre pointé par la gâchette change d'Inactif à Actif. Dans le BDMP de la Figure III.8, la réalisation de f1 ou f2 fait passer les feuilles f3 et f4 du mode Inactif au mode Actif par l'entremise de la gâchette. Les processus de Markov associés évoluent en conséquence, et les feuilles changent d'état selon les modalités définies par les fonctions de transfert. Quand plusieurs gâchettes sont présentes, leurs actions se combinent de la manière décrite formellement dans la section suivante.

Ces mécanismes modélisent le progrès de l'attaquant dans les scénarios capturés par le BDMP global. Les sections suivantes formalisent plus rigoureusement et illustrent cette première description.

5. Comme rappelé en II-2.2.3, une porte k/n est vérifiée si et seulement si au moins k de ses fils sont vérifiés.

2.2.2 Les trois familles de fonctions booléennes du temps

Un BDMP définit un processus stochastique global, modélisant la progression de l'attaque et le comportement dynamique de l'attaquant. Pour cela, chaque élément i de $\mathcal{A}$ est associé à trois fonctions booléennes : une fonction de structure $S_i(t)$, un sélecteur de mode $X_i(t)$ et un indicateur de pertinence $Y_i(t)$. Au niveau du BDMP, les trois familles de fonctions sont définies ci-dessous. Pour simplifier les notations, le temps t n'est pas indiqué mais devrait apparaître partout :

- $(S_i)_{i\in E}$ est la famille des fonctions de structure. Elles obéissent à la relation suivante :

$$\forall i \in G, S_i \equiv \Big(\sum_{j\in fils(i)} S_j \geq g(i)\Big) \text{ et } \forall j \in B, S_j \equiv (Z^j_{X_j} \in S^j_{X_j})$$

  avec X_j indiquant le mode de P_j au temps t. $S_j = 1$ correspond à la réalisation d'un événement de base (comme la réussite d'une action de l'attaquant) ;

- $(X_i)_{i\in E}$ correspond aux sélecteurs de mode, indiquant quel mode est choisi pour chaque processus. Si i est un sommet de $\mathcal{A}$, alors $X_i = 1$, sinon :

$$X_i \equiv \neg\left[(\forall x \in E, (x,i) \in L \Rightarrow X_x = 0) \vee (\exists x \in E/(x,i) \in T \wedge S_x = 0)\right].$$

  Cela signifie que $X_i = 1$ sauf si l'origine d'une gâchette pointant sur i a sa fonction de structure égale à 0, ou si i a au moins un père, et que tous ses pères sont inactifs (*i.e.* ont un sélecteur de mode à 0) ;

- $(Y_i)_{i\in E}$ correspond à la famille des indicateurs de pertinence. Ils servent à marquer les processus qui peuvent être « élagués » pendant le traitement du processus de Markov global sous-jacent lors de l'exploration des séquences. Le filtrage des événements pertinents réduit considérablement les risques d'explosion combinatoire, sans introduire d'approximation dans les cas de quantification qui nous intéressent (nous développons ces aspects dans la section suivante). Si $i = r$ (objectif final), alors $Y_i = 1$ (l'événement est pertinent), sinon :

$$Y_i \equiv (\exists x \in E/(x,i) \in L \wedge Y_x = 1 \wedge S_x = 0) \vee (\exists y \in E/(i,y) \in T \wedge S_y = 0).$$

  En d'autres termes, $Y_i = 1$ si et seulement si :

  - $i = r$,
  - ou i a au moins un père « pertinent » j qui n'est pas encore vérifié ($S_j = 0$),
  - ou i correspond à l'origine d'au moins une gâchette pointant un élément j non vérifié ($S_j = 0$).

Le Tableau III.7, repris de [238], indique les expressions des fonctions booléennes S_i, X_i, Y_i dans le cas du BDMP précédemment employé pour introduire le formalisme et représenté sur la Figure III.8.

TABLE III.7 – Expression des fonctions booléennes S_i, X_i, Y_i pour le cas de la Fig. III.8

Fonctions de structure S_i	**Sélecteurs de mode** X_i	**Indicateurs de pertinence** Y_i
$S_r = S_{G3} = S_{f1} \wedge S_{G2}$	$X_r = X_{G3} = 1$	$Y_r = Y_{G3} = 1$
$S_{G2} = S_{f3} \vee S_{f4}$	$X_{G2} = S_{G1}$	$Y_{G2} = \neg S_r$
$S_{G1} = S_{f1} \vee S_{f2}$	$X_{G1} = 1$	$Y_{G1} = 1$
$S_{f1} = 1 \Leftrightarrow P_{f1}$ dans un état de succès/réalisation	$X_{f1} = X_{G1} \vee X_r = 1$	$Y_{f1} = \neg S_{G1} \wedge Y_{G1}$
$S_{f2} = 1 \Leftrightarrow P_{f2}$ dans un état de succès/réalisation	$X_{f2} = X_{G1} = 1$	$Y_{f2} = \neg S_{G1} \wedge Y_{G1}$
$S_{f3} = 1 \Leftrightarrow P_{f3}$ dans un état de succès/réalisation	$X_{f3} = X_{G2} = S_{G1}$	$Y_{f3} = Y_{G2} \wedge \neg S_{G2}$
$S_{f4} = 1 \Leftrightarrow P_{f4}$ dans un état de succès/réalisation	$X_{f4} = X_{G2} = S_{G1}$	$Y_{f4} = Y_{G2} \wedge \neg S_{G2}$

Nous donnons un premier aperçu du fonctionnement des trois familles de fonctions booléennes en nous appuyant sur ce même modèle dans la Figure III.9. Les valeurs des S_i, X_i, et Y_i sont indiquées pour chaque élément par un triplet évoluant au gré du déroulement d'un scénario simple. La Figure III.9a) correspond à l'état du BDMP une fois initialisé alors qu'aucune feuille n'a encore été réalisée. Seuls r, G3, G2, f1 et f2 sont en mode Actif, et tous les événements sont pertinents. La Figure III.9b) illustre la

situation au bout d'un temps t, au moment où la feuille f1 se réalise. La porte G1 est alors également réalisée : S_{f1} et S_{G1} prennent donc la valeur 1 ; l'origine de la gâchette étant vérifiée, elle active la porte G2 et les feuilles f3 et f4 dont les sélecteurs de mode X_{G2}, X_{f3} et X_{f4} deviennent alors égaux à 1. La feuille f2 n'est alors plus pertinente, son indicateur de pertinence Y_{f2} passe à 0 (nous revenons dans la section suivante sur l'intérêt de ces indicateurs). La feuille f3, dans son mode Actif, se réalise quelque temps après comme représenté dans la Figure III.9c) : G2, G3 se vérifient, l'événement redouté r aussi.

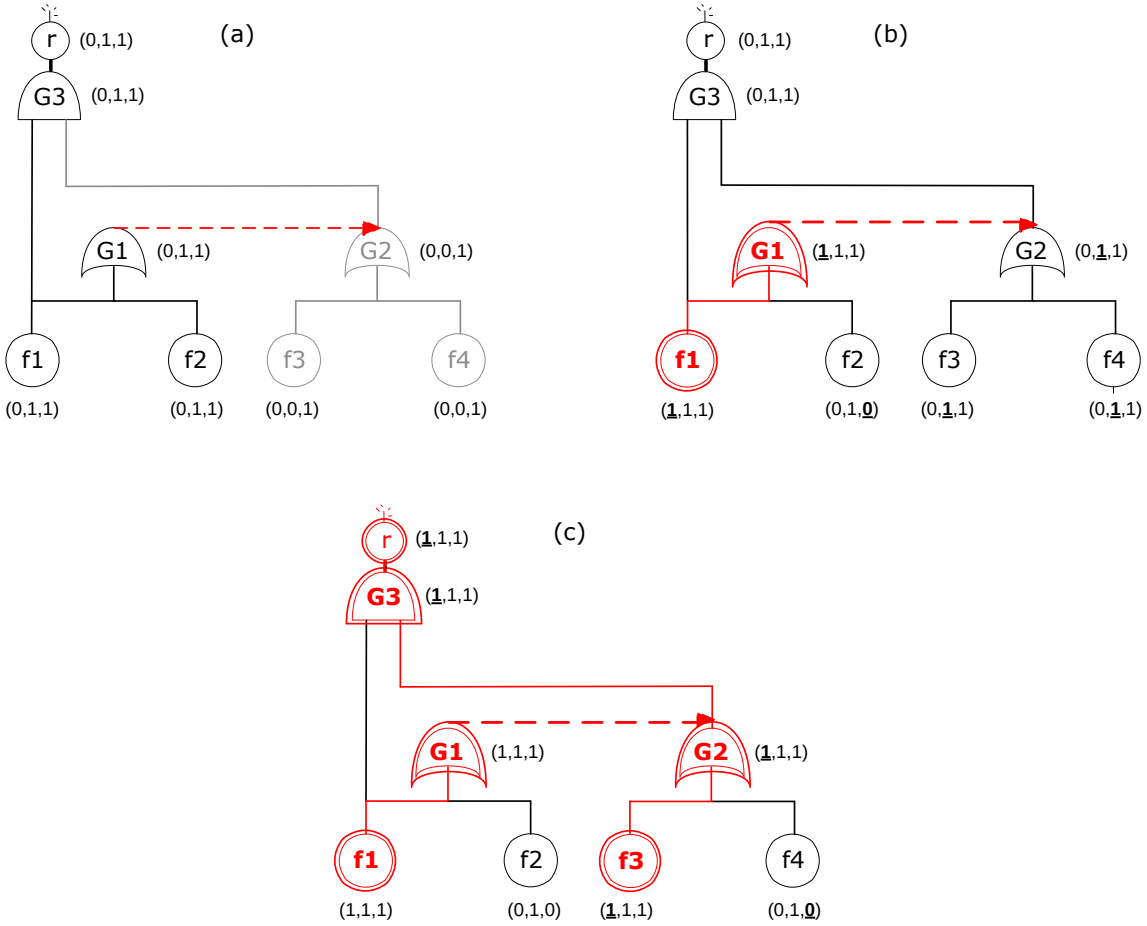


FIGURE III.9 – Valeurs des fonctions booléennes S_i, X_i, Y_i pour un scénario du cas d'illustration

2.2.3 Propriétés mathématiques

Les BDMP constituent un formalisme mathématique robuste dans le sens des deux théorèmes suivants :

Théorème 1. *Les fonctions (S_i), (X_i), (Y_i) sont définies pour tout $i \in E$ quelle que soit la structure du BDMP.*

Théorème 2. *Toute structure de BDMP associée à un état initial, défini par les modes et les états des P_i, spécifie de manière unique un processus de Markov homogène valide.*

La preuve de ces théorèmes peut être trouvée dans [238].

En plus de leur robustesse, les BDMP permettent une réduction combinatoire remarquable par le mécanisme de filtrage des événements pertinents, qui correspond à l' « élagage » de certains processus du graphe de Markov sur la base de leur Y_i. Ce mécanisme peut être illustré de la façon suivante : dans la Figure III.10, après la réalisation d'un événement de base P_i, les autres événements de base $P_{j\neq i}$ ne

sont plus pertinents : rien ne change par rapport à la réalisation de r si nous les inhibons. Le nombre des séquences menant à l'objectif est n si les événements pertinents sont filtrés ($\{P1, Q\}$, $\{P2, Q\}$, etc.), il est exponentiel sinon ($\{P1, Q\}$, $\{P1, P2, Q\}$, $\{P1, P3, Q\}$, etc.).

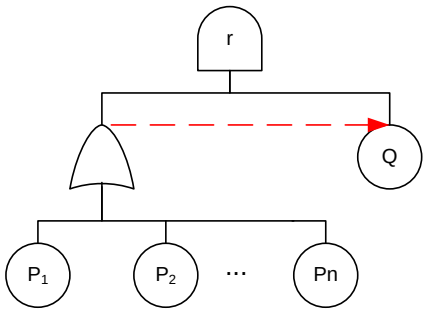


FIGURE III.10 – Cas pour lequel le filtrage des événements pertinents est particulièrement efficace

Théorème 3. *Si les P_i sont tels que $\forall i \in B, \forall t, \forall t' \geq t,\ S_i(t) = 1 \Rightarrow S_i(t') = 1$ (ce qui est toujours vérifié dans notre cas), alors $Pr(S_r(t) = 1)$ reste inchangée que l'on filtre les événements pertinents ou pas (autrement dit, que l'on élimine les événements avec $Y_i = 0$ ou pas).*

La preuve de ce théorème est donnée dans [238]. Il implique qu'un filtrage sur la base des Y_i ne change pas les valeurs quantitatives d'intérêt pour l'analyste (cf. Section 2.4.1). En fait, le filtrage des événements pertinents correspond au comportement même de l'attaquant : en effet, si l'on se reporte à la Figure III.10 avec les P_i représentant des techniques différentes pour atteindre l'objectif intermédiaire représenté par la porte OU, alors une fois un des P_i réalisé, l'attaquant ne poursuivra pas ses efforts sur les $P_{j\neq i}$, et se concentrera sur la réalisation de Q. Les $P_{j\neq i}$ sont donc bien à élaguer pour ne considérer que les séquences d'attaque « rationnelles ».

2.2.4 Les feuilles de base et leurs processus de Markov pilotés

La définition de trois types de feuilles est suffisante pour couvrir un large spectre de modélisation d'attaques. Leurs représentations graphiques et la définition de leurs processus de Markov pilotés sont données dans la Table III.8. Leurs descriptions génériques sont données ci-dessous. Les deux premières sont adaptées des feuilles utilisées en sûreté de fonctionnement, simplifiées car sans transition de retour type réparation ; la troisième a été créée pour les besoins propres à cette adaptation.

- Les feuilles de type *Attacker Action* (AA) (action de l'attaquant [6]) modélisent les pas de l'attaquant vers l'accomplissement de son objectif. Le mode Inactif correspond à des actions qui n'ont pas encore été tentées par l'attaquant. Le mode Actif correspond à des tentatives de réalisation en cours (ou une action déjà intentée), dont le temps nécessaire au succès suit une loi de probabilité exponentielle, de paramètre λ différencié selon les feuilles. Quand la valeur de X_i passe de 0 (Inactif) à 1 (Actif), l'état de la feuille i passe de Potentiel (P) à En-cours (E) ; quand X_i revient de 1 à 0, si l'attaquant n'a pas réussi, la feuille i revient systématiquement dans l'état Potentiel (P), si l'attaquant a réussi, la feuille i revient systématiquement dans l'état Succès (S). Formellement, les fonctions de transfert de probabilités peuvent s'écrire comme suit :

$$
\begin{aligned}
f_{0\to1}(P) &= \{\Pr(E) = 1, \Pr(S) = 0\}, \\
f_{0\to1}(S) &= \{\Pr(E) = 0, \Pr(S) = 1\}, \\
f_{1\to0}(E) &= \{\Pr(P) = 1, \Pr(S) = 0\}, \\
f_{1\to0}(S) &= \{\Pr(P) = 0, \Pr(S) = 1\}.
\end{aligned}
$$

6. Comme pour les deux autres types de feuille, nous garderons les dénominations en anglais, car elles correspondent, avec leurs sigles associés, à celles utilisées dans la mise en œuvre logicielle présentée en Section 5, et employées dans les modèles de ce mémoire.

- Les feuilles de type *Instantaneous Security Event* (ISE) (événement de sécurité instantané) modélisent des événements dont la réalisation est instantanée, et s'opère avec une probabilité γ au moment où la feuille bascule du mode Inactif au mode Actif. Dans le mode Inactif, l'événement ne peut pas se réaliser et la feuille reste dans l'état Potentiel (P). Quand la feuille passe en mode Actif, l'événement est alors soit Réalisé (R), soit Non-Réalisé (NR), selon le tirage de la probabilité γ. Formellement, les fonctions de transfert de probabilités peuvent s'écrire comme suit :

$$\begin{aligned} f_{0\to1}(P) &= \{\Pr(NR) = 1-\gamma, \Pr(R) = \gamma\}, \\ f_{0\to1}(R) &= \{\Pr(NR) = 0, \Pr(R) = 1\}, \\ f_{1\to0}(R) &= \{\Pr(NR) = 0, \Pr(R) = 1\}, \\ f_{1\to0}(NR) &= \{\Pr(P) = 1, \Pr(R) = 0\}. \end{aligned}$$

- Les feuilles de type *Timed Security Event* (TSE) (événement de sécurité temporisé) modélisent des événements de base temporisés, dont la réalisation contribue au progrès de l'attaquant vers son objectif, mais qui ne sont pas sous son contrôle direct. Les temps nécessaires à la réalisation de ces feuilles une fois activées suivent des lois de probabilité exponentielles de paramètres λ, différenciables par feuille. Quand la feuille revient en mode Inactif, l'événement peut être soit Réalisé (R), soit Non-Réalisé (NR), selon qu'il est déjà réalisé en mode Actif ou pas. S'il ne s'est pas réalisé, l'analyste peut décider de rendre possible sa réalisation en mode Inactif en attribuant une valeur non nulle à λ'. Ceci peut s'avérer utile dans les approches par phase décrites en Section 2.3.2. Formellement, les fonctions de transfert de probabilités peuvent s'écrire comme suit :

$$\begin{aligned} f_{0\to1}(P) &= \{\Pr(NR) = 1, \Pr(R) = 0\}, \\ f_{0\to1}(NR) &= \{\Pr(NR) = 1, \Pr(R) = 0\}, \\ f_{0\to1}(R) &= \{\Pr(NR) = 0, \Pr(R) = 1\}, \\ f_{1\to0}(NR) &= \{\Pr(NR) = 1, \Pr(R) = 0\}, \\ f_{1\to0}(R) &= \{\Pr(NR) = 0, \Pr(R) = 1\}. \end{aligned}$$

2.3 Modélisations élémentaires

2.3.1 Modélisation de séquences

Comme déjà souligné en Section 1, la capacité à modéliser des séquences joue un rôle important dans la modélisation de scénarios d'attaque : dans la majorité des cas, un certain nombre d'actions ou d'événements doivent en effet avoir lieu avant que d'autres étapes puissent être franchies par l'attaquant. Les gâchettes permettent une modélisation facile et lisible de tels aspects. La Figure III.11 représente de façon simplifiée l'attaque d'un système d'exploitation en trois actions, liées par une telle contrainte de séquence : l'attaquant procède d'abord à la caractérisation du système ciblé (*fingerprinting*), il peut ensuite identifier une vulnérabilité typique de ce logiciel, avant de passer à son exploitation. Des exemples plus élaborés peuvent être trouvés en Sections 2.6.1 et 2.6.2.

2.3.2 Modélisation d'alternatives concurrentes ou exclusives

Pour un objectif intermédiaire donné, un attaquant peut avoir différentes alternatives. La modélisation naturelle de cette situation avec les BDMP et les arbres d'attaque classiques consiste à utiliser une porte OU. Ceci-dit, plusieurs cas de figure peuvent être distingués. La Figure III.12 représente deux approches distinctes pour modéliser une telle situation, à l'aide d'un exemple modélisant une caractérisation de système d'exploitation (*OS fingerprinting*). Dans la Figure III.12a), où seule une porte OU est utilisée, techniques passives et techniques actives sont essayées simultanément, ce qui peut ne pas correspondre à un comportement réaliste de l'attaquant. Les techniques passives étant plus discrètes, elles devraient en principe être tentées en premier, et abandonnées si infructueuses au bout d'un certain temps pour des techniques actives, plus efficaces, mais moins discrètes [468]. Les gâchettes ne peuvent modéliser un

TABLE III.8 – Description des trois types de feuilles et de leurs processus de Markov pilotés

Types de feuilles et représentations	Mode Inactif (X_i=0)	Transfert entre les modes	Mode Actif (X_i=1)
Attacker Action (AA)	**P**otentiel — **S**uccès	P ⇄ E (avec Pr = 1) S ⇄ S (avec Pr = 1)	**E**n-cours —λ→ **S**uccès ; $S_i \leftarrow 1$
ISE! *Instantaneous Security Event (ISE)*	**P**otentiel — **R**éalisé	P → NR (avec Pr = 1-γ) P → R (avec Pr = γ) R ⇄ R (avec Pr = 1) P ← NR (avec Pr = 1)	**N**on **R**éalisé —λ→ **R**éalisé ; $S_i \leftarrow 1$
TSE *Timed Security Event (TSE)*	**P**otentiel **N**on **R**éalisé —λ'→ **R**éalisé ; $S_i \leftarrow 1$	P → NR (avec Pr = 1) NR ⇄ NR (avec Pr = 1) R ⇄ R (avec Pr = 1)	**N**on **R**éalisé —λ→ **R**éalisé ; $S_i \leftarrow 1$

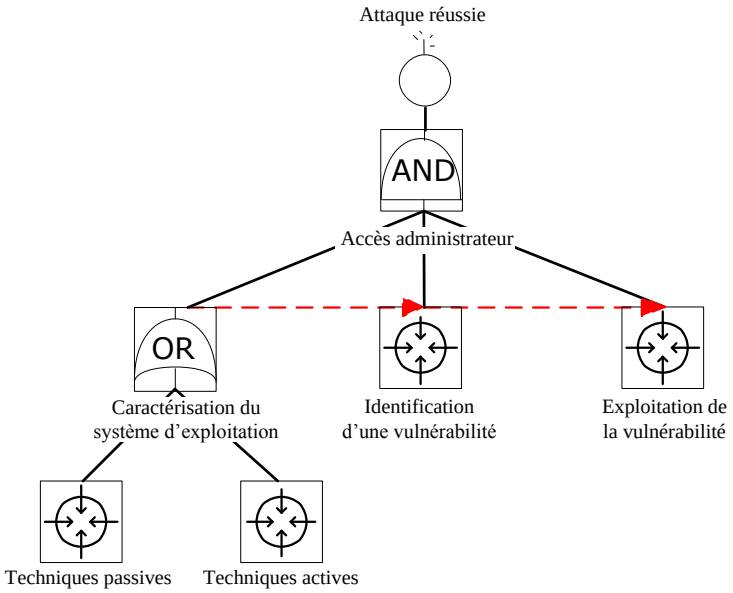


FIGURE III.11 – Séquences pour l'attaque d'un système d'exploitation

tel comportement. Des feuilles « phases », représentées par des horloges, sont alors introduites dans la Figure III.12b). Elles permettent de modéliser le comportement souhaité en activant et désactivant des feuilles ou des sous-arbres entiers après une durée suivant une loi exponentielle. Leur définition est donnée dans [469] : si aucune gâchette ne pointe vers une feuille de phase, celle-ci est initialisée à VRAI et devient fausse au bout d'un temps distribué exponentiellement. Si *a contrario*, elle est pointée par une gâchette, elle est initialisée à FAUX, et quand l'origine de la gâchette passe de VRAI à FAUX, la feuille passe instantanément à VRAI. Elle revient à FAUX après un temps distribué exponentiellement. Un tel comportement permet de lier un nombre arbitraire de feuilles de phase, éventuellement en circuit, et reste cohérent avec le cadre théorique des BDMP. Dans la Figure III.12b), deux feuilles de phase suffisent à modéliser le comportement voulu : seules les techniques passives sont essayées initialement, elles sont abandonnées si elles n'aboutissent pas au bout d'un temps distribué exponentiellement au profit des techniques actives.

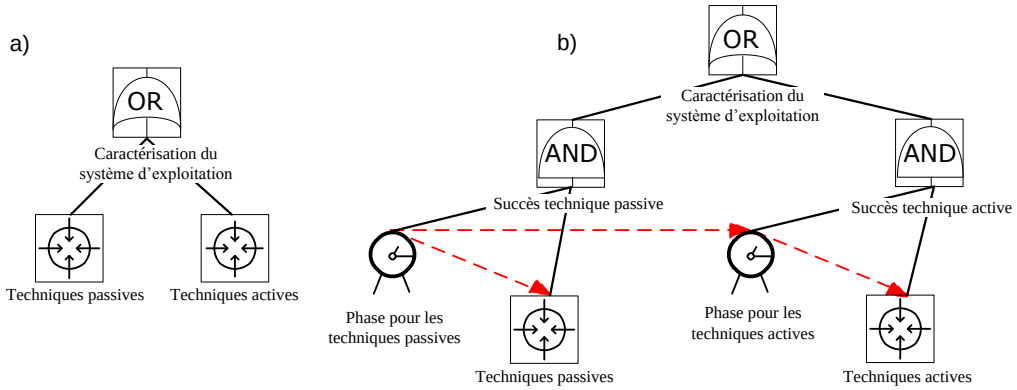


FIGURE III.12 – Modélisation d'alternatives concurrentes ou exclusives

2.4 Quantifications

2.4.1 Quantifications temporelles

L'intérêt des BDMP ne réside pas seulement dans la simple capacité de représentation des séquences. Ils permettent aussi diverses quantifications d'ordre temporel, dont notamment le calcul de la probabilité pour un attaquant d'atteindre son objectif dans un temps donné ou encore du temps moyen global pour la réalisation de l'attaque. En plus de ces quantifications générales, le traitement des BDMP donne également l'énumération de tous les chemins (ou séquences) d'attaque menant à l'objectif final, ordonnés par leur probabilité d'occurrence dans le temps d'observation choisi (dit temps de mission). Ces résultats peuvent être efficacement obtenus grâce à une méthode de calcul développée pour les grands modèles markoviens, et donc applicable au traitement des BDMP [470].

2.4.2 Calculs par exploration de chemins

Comme indiqué dans la Section 2.2.3, les BDMP sont une représentation de haut niveau de processus de Markov à espace d'états potentiellement gigantesque. Ce type de modèle pose habituellement des problèmes d'explosion combinatoire, limitant l'efficacité des solutions analytiques classiques [176]. Confronté à de telles situations dans ses études de sûreté de fonctionnement, EDF a développé une approche originale basée sur l'exploration des chemins menant à la panne.

Une telle approche permet de calculer la probabilité de l'événement redouté, mais donne aussi nombre d'informations inaccessibles par les méthodes classiques comme la liste des séquences y menant, ordonnées par probabilité d'occurrence dans le temps de mission, ou l'importance relative de ces séquences. Elle fournit des résultats exacts pour les petits modèles en procédant à une exploration du graphe de façon exhaustive ; elle offre des approximations maîtrisées pour les grands modèles en limitant le nombre de séquences explorées à celles ayant une probabilité dépassant un certain seuil. La probabilité de l'événement redouté à l'instant t correspond à la somme des probabilités de réalisation des séquences explorées y menant

avant l'instant t, les séquences étant mutuellement exclusives. Par exemple, dans le graphe de Markov de la Figure III.13, si l'état initial est numéroté 1, si ceux correspondant à l'événement redouté sont numérotés 4 et 7, et si l'exploration est exhaustive, les séquences menant de l'état initial à l'événement redouté sont (1,2,3,4), (1,2,3,7), (1,5,3,4), (1,5,3,7) et (1,5,6,7).

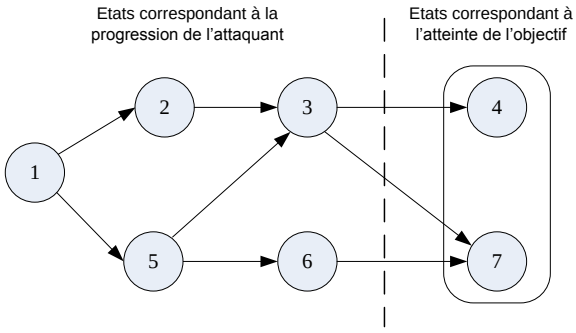


FIGURE III.13 – Exemple de graphe de Markov pour l'identification des séquences

Le principe du calcul de la probabilité d'une séquence décrit dans [471] est redonné ici sommairement. Si l'on nomme S l'ensemble des n séquences considérées menant à l'événement redouté r, et $P_s(t)$ la probabilité qu'une séquence $s \in S$ se produise avant un instant t, alors par définition des probabilités conditionnelles :

$$P_s(t) = Pr(s \text{ parcourue jusqu'au bout}) \times Pr(\text{réalisation de } r \text{ avant } t \ / \ s \text{ parcourue jusqu'au bout}).$$

Avec $t \to \infty$, on trouve que $P_s(\infty) = Pr(s \text{ parcourue jusqu'au bout})$, on peut donc écrire :

$$P_s(t) = P_s(\infty) \times Pr(\text{réalisation de } r \text{ avant } t \ / \ s \text{ parcourue jusqu'au bout}).$$

Le premier terme peut être calculé facilement : il correspond au produit des probabilités d'aller, à chaque état de la séquence, vers l'état suivant de la séquence plutôt que vers un autre état du graphe. Quand les taux de transition sont constants, ces probabilités sont obtenues par des formules analytiques simples (cf. [471]).

Le second terme est plus problématique. On numérote de 1 à n les états d'une séquence s menant à l'événement redouté r, $n+1^{e}$ état de la séquence considérée. La Figure III.14 représente l'extrait pertinent du graphe de Markov correspondant, où $\lambda_{i,j}$ est le taux de transition de l'état i vers l'état j.

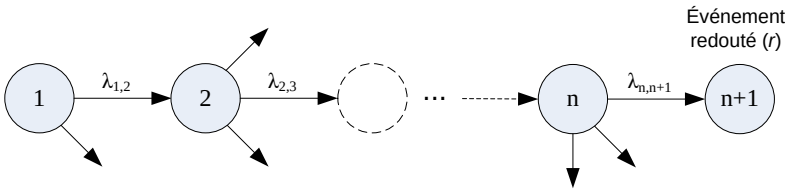


FIGURE III.14 – Une séquence s extraite d'un graphe de Markov plus complet

La durée nécessaire à la réalisation de r sachant que la séquence s a été parcourue jusqu'au bout est égale à la somme des temps de séjour T_i dans les états 1 à n. D'après une propriété des processus markoviens, les durées de séjour dans les états sont indépendants des transitions empruntées pour sortir des états. Avec les notations de la Figure III.14, chaque durée T_i suit une loi exponentielle de paramètre $\Lambda_i = \sum_{i \neq k} \lambda_{i,k}$.

Au final, on a donc $P_s(t) = P_s(\infty) \times Pr((\sum_{i=1}^{n} T_i) < t)$, le premier terme étant facile à calculer, le second étant la fonction de répartition d'une somme de variables de lois exponentielles indépendantes. La réf. [471] développe la solution applicable aux cas où les temps moyens passés dans chaque état sont tous

différents (la formule classique du produit de convolution de lois exponentielles peut alors s'appliquer) ; il a fallu attendre les travaux de Harrison en 1990 [472] pour pouvoir généraliser la solution aux cas où certaines des durées sont identiques [473] (en se basant sur les transformées de Laplace).

Bien au-delà de cette description très générale, la thèse de Lefebvre [474] donne une description détaillée des différents cas de figure rencontrés, et décrit de nombreuses optimisations et heuristiques pour ce genre de calcul.

Soulignons pour finir que l'approche par exploration de chemins bénéficie pleinement du mécanisme de filtrage des événements pertinents des BDMP décrit en Section 2.2.3, qui réduit considérablement la combinatoire et le nombre des chemins à explorer.

2.4.3 Paramétrage des feuilles

Plus concrètement, les quantifications nécessitent le paramétrage des différentes feuilles composant le BDMP. L'analyste doit ainsi attribuer des valeurs aux paramètres λ des lois exponentielles caractérisant les feuilles AA et TSE, ainsi qu'aux paramètres γ des feuilles ISE :

- la définition des taux de succès ou de réalisation λ se fait en raisonnant en termes de temps moyen de succès, ou MTTS pour *Mean Time To Success*, pour les feuilles AA, et en temps moyen de réalisation, ou MTTR pour *Mean Time To Realization*, pour les feuilles TSE, en analogie au MTTF utilisé en sûreté de fonctionnement. Dans le cas d'une loi exponentielle de paramètre λ, un tel temps moyen correspond à la valeur $1/\lambda$. Une approche similaire a été adoptée dans de nombreux autres travaux de modélisation de sécurité (*e.g.* [177, 475, 476]). La Section 6.1.1 revient sur les hypothèses et les limites d'un tel choix ;
- la définition des probabilités de réalisation à l'activation γ se fait par rapport à des retours d'expérience de situations similaires, à défaut d'avoir des statistiques pertinentes, ou par avis d'expert plus informels.

D'une façon générale, l'attribution de ces valeurs correspond à une vision subjectiviste des probabilités (cf. Chap. II Section 1.2.4) ; elles doivent être estimées en tenant compte, en outre, de la difficulté intrinsèque des actions modélisées, des conditions favorisant ou éloignant l'occurrence d'un événement donné, des ressources et compétences estimées de l'attaquant et du niveau de protection des systèmes attaqués. Dans ces conditions, les valeurs choisies ne correspondent qu'à une formalisation numérique des avis et croyances des analystes sécurité, nécessairement subjectifs, et seulement traduits ici sous un aspect plus formel que dans les analyses de risques classiques telles que décrites en II-2.3.3. Malgré la difficulté de la tâche, l'estimation de probabilités (au sens large) reste une étape incontournable à toute démarche d'analyse de risques ; la forme ici numérique ne doit pas faire oublier les limites d'un tel exercice, évoquées en II-1.2.3.

2.4.4 Quantifications non-temporelles

En plus des quantifications temporelles, les BDMP permettent également des quantifications indépendantes du temps, à l'instar de celles effectuées avec les arbres d'attaque classiques. Le principe général consiste à attribuer différentes valeurs statiques aux feuilles de l'arbre, et soit de les « propager » jusqu'au sommet en respectant des règles variant selon la nature des valeurs attribuées et des portes rencontrées pour obtenir des valeurs globales au modèle [284], soit les considérer sur des chemins spécifiques dans l'arbre, pour caractériser et comparer des scénarios particuliers [477].

Un premier type de valeurs attribuées aux arbres d'attaque classiques correspond à des probabilités de réussite, ici indépendantes du temps : une fois les coupes minimales identifiées par les techniques classiquement employées avec les arbres de défaillances [188], elles permettent une hiérarchisation de scénarios, en supposant les événements de base de l'arbre d'attaque indépendants. Nous ne considérons pas ce type de traitement, ces aspects étant déjà couverts dans les BDMP par le paramétrage des processus stochastiques associés aux feuilles. Ceci-dit, de nombreux autres paramètres ont été proposés dans la littérature attenante aux arbres d'attaque. Les notions de coûts et d'indicateurs booléens sont certainement parmi les plus employées (et ce, dès le papier de Schneier [22]). Dans le premier cas, l'approche consiste à associer un coût, en général monétaire, pour l'attaquant à la réalisation de chaque feuille, permettant de calculer et de comparer le coût global de scénarios d'attaque donnés, ou de calculer un coût moyen associé à la réalisation de l'objectif, tous scénarios pris en compte. Dans le second cas, l'idée est de marquer

chaque feuille d'un indicateur booléen modélisant une propriété ou une exigence spécifique nécessaire à sa réalisation. Une telle exigence peut par exemple correspondre au besoin d'un appui interne à l'organisation attaquée, à la nécessité de posséder un équipement ou un outil particulier, ou encore, la nécessité d'avoir connaissance d'une information particulière. Une fois les scénarios d'intérêt retenus sur d'autres critères (*e.g.* par calcul de coupes minimales dans le cas d'arbres d'attaque classiques, par exploration et classification des séquences pour un BDMP), il est alors possible de savoir s'ils requièrent la propriété ou l'exigence spécifique, et faire le cas échéant, de nouvelles sélections. Enfin, un troisième type de paramètre peut être mentionné : il s'agit du niveau de compétence requis pour la réalisation d'une action [22, 284]. Généralement défini selon une gradation simple (*e.g.* simple / intermédiaire / difficile / expert), ce paramétrage permet de fixer un seuil de compétence nécessaire à la réalisation de chaque feuille. Les séquences sont alors caractérisées par le niveau de compétences maximum des actions qui la composent, et peuvent être filtrées sur ce critère, selon le profil estimé de l'attaquant. Notons que si une telle estimation contribue aussi à la détermination des taux λ de succès pour un BDMP, ces approches peuvent être combinées, le paramétrage des taux permettant de faire des quantifications temporelles probabilistes, la définition de seuils permettant d'éliminer de façon déterministe un certain nombre de scénarios.

En fait, plus globalement, les différents types de paramètres jusqu'ici exposés (coûts, indicateurs booléens, seuils de compétence) peuvent éventuellement être utilisés conjointement. De plus, il est possible de généraliser les exemples donnés en trois catégories de paramètres statiques, qui correspondent, dans l'ordre des exemples donnés, aux indicateurs de type entier, de type booléen et de type gradué. En effet, au-delà des paramètres précédemment exposés, de nombreuses autres instances peuvent être imaginées dans les trois catégories identifiées, du moment que leur comportement mathématique est clairement spécifié pour les quantifications (*e.g.* par l'utilisation d'opérateurs de somme, de *min*, de *max*, de moyenne arithmétique, etc.). Edge propose ainsi l'utilisation d'indicateurs de coût pour l'attaquant, mais aussi de gêne pour les utilisateurs, d'impact financier pour l'organisation et plus largement de paramètres permettant de faire des calculs de risque à partir du modèle [284]. Dans la même philosophie, Patel *et al.* associent aux feuilles de leurs arbres dans [301] un *threat impact index* (indice d'incidence de la menace) et un *cyber-vulnerability index* (indice de cyber-vulnérabilité). À chaque fois, les paramètres statiques permettent à l'analyste de définir précisément des critères et seuils pour mieux caractériser et sélectionner les scénarios d'attaque les plus pertinents vis-à-vis du contexte de l'analyse. Par exemple, on peut ainsi choisir de ne considérer que les scénarios ne dépassant pas un coût donné de mise en œuvre, en rapport avec le profil d'attaquant, ou d'exclure ceux nécessitant une collaboration interne.

L'approche d'exploration de chemins adoptée pour la quantification des BDMP rend la prise en compte des paramètres statiques simple et directe. Les critères de sélection de scénarios peuvent en effet être appliqués *a posteriori* des traitements purement probabilistes. Notons qu'ils pourraient aussi être embarqués dans les algorithmes d'exploration de chemins, accélérant les calculs par arrêt d'exploration des séquences ne respectant pas les critères. Nous n'avons pas eu l'occasion de mettre en œuvre ces optimisations et les retenons comme perspectives de développement. Dans le cas où plusieurs paramètres statiques interviennent, la hiérarchisation des séquences peut se faire de multiples manières, allant de la simple pondération de facteurs à des approches d'optimisation multi-facteurs plus élaborées (*e.g.* l'approche de Dewri *et al.* dans [478]).

Note : Les quantifications du domaine temporel constituant une spécificité et un avantage conséquent des BDMP sur les arbres d'attaque, nous nous concentrerons sur celles-ci et n'illustrerons pas dans les pages suivantes l'utilisation de paramètres statiques tels que décrits dans cette section, bien qu'ils aient été mis en œuvre dans le logiciel support des modélisations de ce chapitre (Cf. Section 5).

2.5 Analyse hiérarchique et passage à l'échelle

De façon semblable aux arbres de défaillances et aux arbres d'attaque, les BDMP offrent de par leur nature hiérarchique une grande flexibilité quant à la granularité d'analyse et à la profondeur de décomposition des attaques. Par exemple, chacune des trois feuilles de la Figure III.11 décrivant l'attaque d'un système d'exploitation aurait pu être détaillée dans des sous-arbres précisant les modalités et alternatives associées à chacune de ces étapes. En fait, il est possible de décomposer très finement les attaques, tout en conservant une certaine lisibilité grâce au caractère hiérarchique des BDMP. Des modèles de plus de cent feuilles sont ainsi couramment traités pour les études de sûreté de fonctionnement [243].

2.6 Exemples

2.6.1 Cas n° 1 : attaque d'un serveur d'accès distant connecté par modem

Ce premier cas d'étude représenté dans la Figure III.15, reprend la situation considérée en Section 1.1.2. pour comparer différents formalismes de modélisation graphique d'attaque. Le BDMP associé modélise une attaque très simplifiée d'un serveur d'accès distant accessible par modem. Son but est avant tout d'illustrer de façon progressive les mécanismes précédemment décrits. À cette fin, le niveau de décomposition des attaques a été gardé au plus simple, et seules des feuilles de type AA ont été utilisées.

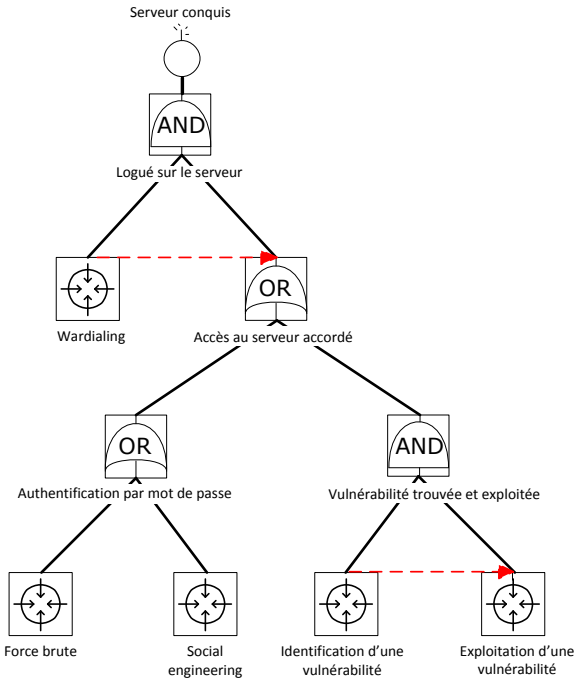


FIGURE III.15 – BDMP de l'attaque d'un serveur d'accès distant

Globalement, les deux gâchettes permettent de prendre en compte la dimension séquentielle de l'attaque. Tout d'abord, avant de tenter une quelconque technique offensive sur le système, une communication doit être établie. La première gâchette permet d'imposer la réalisation de la feuille Wardialing, correspondant à la recherche du numéro de la ligne sur laquelle est connecté le modem, comme préalable à l'activation des autres feuilles. L'attaquant pourra alors ensuite tenter de casser la protection par mot de passe, ou exploiter une vulnérabilité logicielle dans le système (modem, serveur ou protocole). Dans le premier cas, deux techniques sont modélisées, l'attaque par force brute (recherche exhaustive) et le *social engineering*, représentées par les feuilles éponymes ; comme déjà signalé, elles pourraient facilement faire l'objet d'une décomposition plus fine, au même titre que les autres feuilles du BDMP. Dans le second cas, une gâchette permet d'indiquer qu'avant d'être exploitée, une vulnérabilité doit être identifiée par l'attaquant.

La Figure III.16 illustre l'activation progressive des éléments du BDMP, au gré de la réalisation des feuilles, selon un scénario d'attaque particulier parmi ceux modélisés par le BDMP. L'activation suit les mécanismes décrits en Section 2.2, et s'appuie sur les changements de valeur des booléens (S_i) (fonctions de structure) et (X_i) (sélecteurs de mode) au gré de la progression de l'attaque. Les composants en noir correspondent à des composants en mode actif $(X_i = 1)$, ceux en gris clair sont en mode inactif $(X_i = 0)$. Les éléments dont les contours sont doublés et les noms en gras sont dans un état de réalisation $(S_i = 1)$. La couleur des liens, qui n'ont pas de variables booléennes attribuées, évolue seulement pour des raisons de lisibilité.

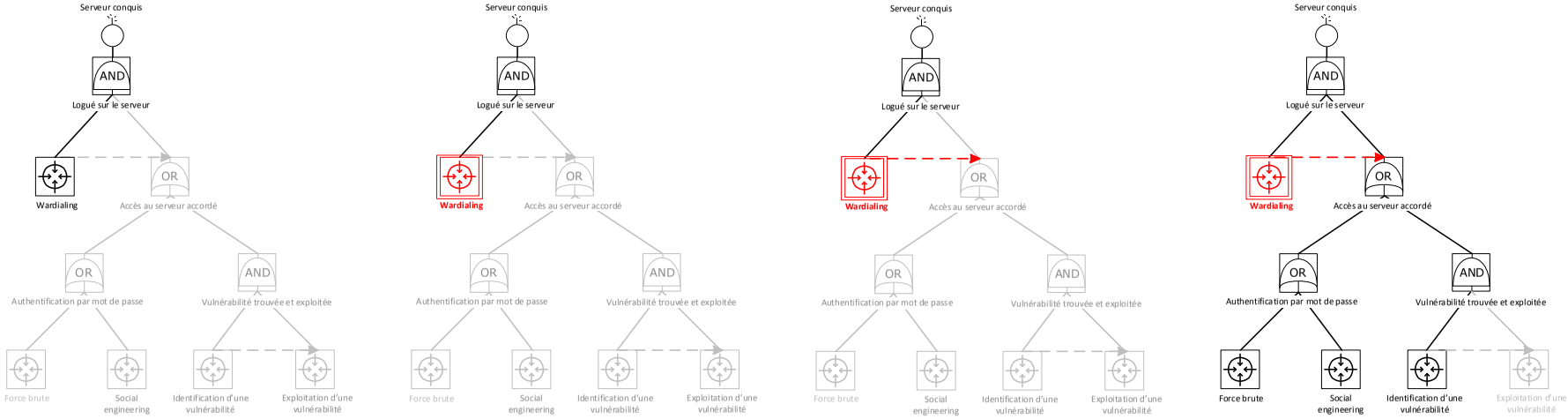


FIGURE III.16 – Activation des éléments du BDMP au gré de la progression de l'attaque

Bien entendu, le processus d'activation des feuilles présenté dans la Figure III.16 s'accomplit de façon temporisée, le petit modèle considéré n'impliquant que des feuilles de type AA. Nous pouvons d'ailleurs nous intéresser maintenant aux aspects de quantification temporelle, décrits en Section 2.4.1. Pour cela, les paramètres suivants ont été arbitrairement choisis, caractérisant le temps moyen nécessaire à la réalisation de chaque action :

- Pour la feuille Wardialing : $\lambda = 10^{-5}\ s^{-1}$, *i.e.* MTTS ≈ 28 h ;
- Pour les feuilles Force brute, Identification d'une vulnérabilité, Exploitation d'une vulnérabilité : $\lambda = 10^{-4}\ s^{-1}$, ce qui correspond à un MTTS $\approx 2,8$ h ;
- Pour l'attaque de type Social engineering sur le mot de passe : $\lambda = 5 \times 10^{-6}\ s^{-1}$, *i.e.* MTTS ≈ 55 h.

Avec ces valeurs, le MTTS global est de $1,07 \times 10^5$ s, soit environ 30 heures, alors que la probabilité pour l'attaquant d'atteindre son objectif en un jour est de 0,55 (mais cette probabilité peut être calculée pour un temps arbitraire). La Table III.9 donne la liste de toutes les séquences d'actions menant à la réalisation de l'objectif de l'attaquant, quantifiées pour un temps de mission d'une journée (86 400 s). Les séquences sont ordonnées selon leur contribution respective, indiquée dans la colonne de droite, à la probabilité globale de réussite de l'attaque. La durée moyenne de chaque séquence est aussi indiquée.

TABLE III.9 – Liste et quantification des séquences

Sequences	**Probabilité pendant le temps de mission**	**Durée moyenne (s)**	**Contribution**
Wardialing, Force brute.	$0,2717$	$4,878 \times 10^3$	$0,498$
Wardialing, Identification vulnérabilité, Force brute.	$0,1272$	$9,756 \times 10^3$	$0,233$
Wardialing, Identification vulnérabilité, Exploitation vulnérabilité.	$0,1272$	$9,756 \times 10^3$	$0,233$
Wardialing, Social engineering.	$0,0136$	$4,878 \times 10^3$	$0,025$
Wardialing, Identification vulnérabilité, Social engineering.	$0,0064$	$9,756 \times 10^3$	$0,011$

Enfin, ce petit modèle nous permet d'illustrer à nouveau les deux approches de modélisation d'alternatives pour l'attaquant, présentées en Section 2.3.2. Dans le modèle de la Figure III.15 jusque là considéré, une fois la ligne de modem trouvée, l'attaquant mène de front une attaque de type Force brute, une action de Social engineering et l'Identification d'une vulnérabilité. Il est possible de modéliser un comportement plus séquentiel en considérant que l'attaquant tentera d'abord les techniques d'attaque de mot de passe, avant d'essayer, si elles s'avèrent infructueuses au bout d'un temps donné, l'attaque par exploitation de vulnérabilité. La Figure III.17 introduit des feuilles phases pour ce faire.

2.6.2 Cas n° 2 : attaque hors-ligne d'un fichier protégé par mot de passe

Ce deuxième cas de figure illustre une utilisation plus complète des BDMP, puisqu'on y retrouve l'utilisation de phases, mais surtout des trois types de feuilles AA, TSE et ISE, éclairant leur emploi respectif. Le BDMP de la Figure III.18 représente l'attaque d'un fichier protégé par mot de passe, dont une copie a été dérobée par l'attaquant. On suppose que la saisie du mot de passe est la seule façon d'accéder aux informations qu'il contient, cible finale de l'attaquant (on ne considère ainsi pas les failles éventuelles du système de protection du fichier, par exemple dans les techniques cryptographiques employées ou l'interface du conteneur logiciel ; ces attaques peuvent faire l'objet de modèles distincts). L'attaquant a besoin de ces informations sous une semaine, temps de mission considéré pour les études quantitatives du modèle. Une telle situation peut avoir lieu par exemple dans le cadre d'un appel d'offre se déroulant dans un environnement hautement compétitif.

La partie haute de la Figure III.18 donne la structure macroscopique de l'attaque :

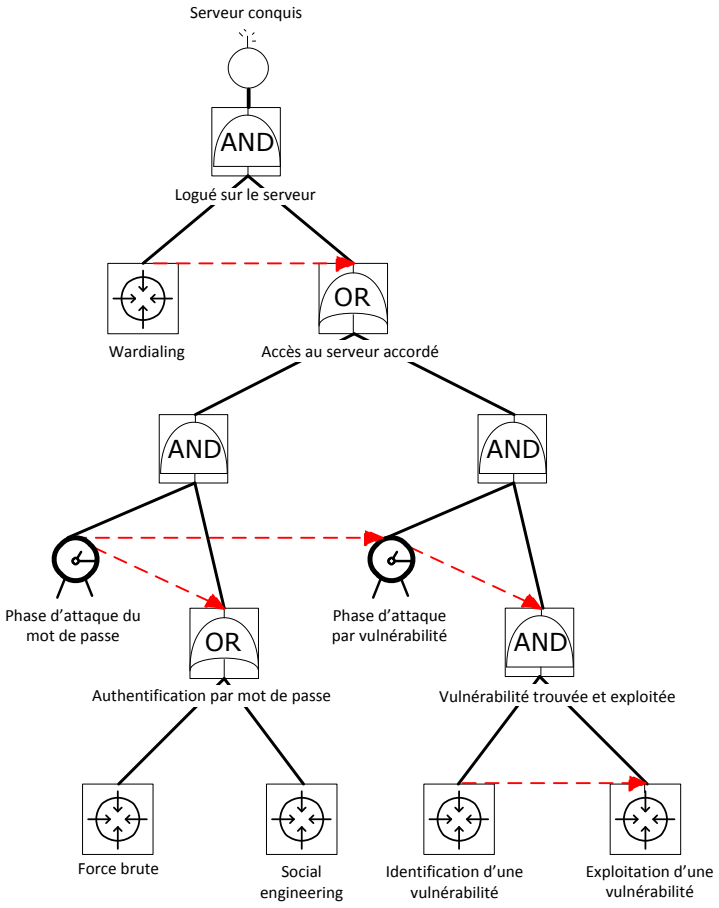


FIGURE III.17 – Ajout de phases dans le cas d'étude n° 1

- l'attaquant ayant le fichier sous son contrôle, il tente de casser le mot de passe durant toute la semaine par différentes techniques (porte Alternatives de cassage de mot passe) ;
- en parallèle de ces activités, il essaye d'autres approches en deux temps :
  - dans un premier temps, il se concentre sur les attaques de type *social engineering*, auquel il consacre un temps moyen de 2 jours,
  - dans un second temps, si les techniques de *social engineering* n'ont pas abouti, l'attaquant bascule vers des tentatives d'installation de *keylogger* [7] sur la machine d'un utilisateur susceptible de saisir le mot de passe recherché.

Les parties médiane et basse de la Figure III.18 détaillent ces deux approches :

- pour la phase de *social engineering*, nous avons réduit le modèle au simple enchaînement d'une phase de reconnaissance, qui sera éventuellement utile à l'attaquant par ailleurs, et des techniques d'approche par *email* et par téléphone, représentées par les feuilles AA Exécution du piège par email et Exécution du piège téléphonique. La feuille instantanée (ISE) Utilisateur piégé modélise la probabilité estimée que la ou les personnes ciblées se fassent piéger ;
- la phase d'installation du *keylogger* est elle-même décomposée en deux temps :
  - dans un premier temps, une installation distante, moins risquée pour l'attaquant, est recherchée. L'attaque ici modélisée repose sur le piégeage d'un *email* et de sa pièce jointe. On peut noter dans le sous-arbre correspondant l'emploi d'une feuille TSE, qui modélise le temps probable avant que l'utilisateur ciblé n'ouvre l'*email* piégé. Ce type de feuille est employé pour modéliser un événement temporisé hors de contrôle de l'attaquant, ce qui est bien le cas ici. La feuille ISE modélise la probabilité que la charge malveillante s'exécute correctement dans l'environnement logiciel de l'utilisateur,
  - dans un second temps, si l'installation distante du *keylogger* est infructueuse, l'attaquant se résout à une installation physique du *keylogger*, nécessitant une phase de reconnaissance spécifique en plus de celle effectuée précédemment, et l'accès physique à la machine.

Afin d'illustrer les capacités de quantification temporelle des BDMP, nous avons paramétré le modèle de la Figure III.18 avec les valeurs indiquées dans la Table III.10.

TABLE III.10 – Paramètres du cas d'étude

Nom de la feuille	Type	Paramètre	Remarque
Déduction, Dictionnaire	AA	$\lambda = 0$	Succès considéré comme impossible (mdp aléatoire)
Force brute	AA	$\lambda = 3,802 \times 10^{-7} s^{-1}$	MTTS $(1/\lambda) \approx$ un mois
Phase de social engineering	Phase	172,800 s	Durée moyenne = 2 jours
Reconnaissance générique	AA	$\lambda = 1,157 \times 10^{-5} s^{-1}$	MTTS $(1/\lambda) \approx 1$ jour
Exécution du piège par email	AA	$\lambda = 1,157 \times 10^{-5} s^{-1}$	MTTS $(1/\lambda) \approx 1$ jour (accès distant régulier)
Exécution du piège tél.	AA	$\lambda = 5,787 \times 10^{-6} s^{-1}$	MTTS $(1/\lambda) \approx 2$ jours
Utilisateur piégé	ISE	$\gamma = 0,33$	1 chance sur 3 (ciblé mais utilisateur sensibilisé)
Phase keylogger	Phase	432 000 s	Durée moyenne de la phase = 5 jours
Phase distante	Phase	172 800 s	Durée moyenne de la phase = 2 jours
Préparation de la charge	AA	$\lambda = 5,787 \times 10^{-6} s^{-1}$	MTTS $(1/\lambda) \approx 2$ jours
Ouverture de la P.J. piégée	TSE	$\lambda = 1,157 \times 10^{-5} s^{-1}$	MTTS $(1/\lambda) \approx 1$ jour
Bonne exécution de la charge	ISE	$\gamma = 0,1$	1 chance sur 10 (nombreux facteurs inconnus)
Phase physique	Phase	259 200 s	Durée moyenne = 3 jours
Reconnaissance physique	AA	$\lambda = 5,787 \times 10^{-6} s^{-1}$	MTTS $(1/\lambda) \approx 2$ jours
Installation locale du keylogger	AA	$\lambda = 1,157 \times 10^{-5} s^{-1}$	MTTS $(1/\lambda) \approx 1$ jour
Mot de passe intercepté	TSE	$\lambda = 1,157 \times 10^{-5} s^{-1}$	MTTS $(1/\lambda) \approx 1$ jour

7. Parfois traduit par « enregistreur de frappes », nous emploierons le terme anglais beaucoup plus courant pour désigner un logiciel transmettant vers l'attaquant à l'insu de l'utilisateur les touches frappées sur son clavier.

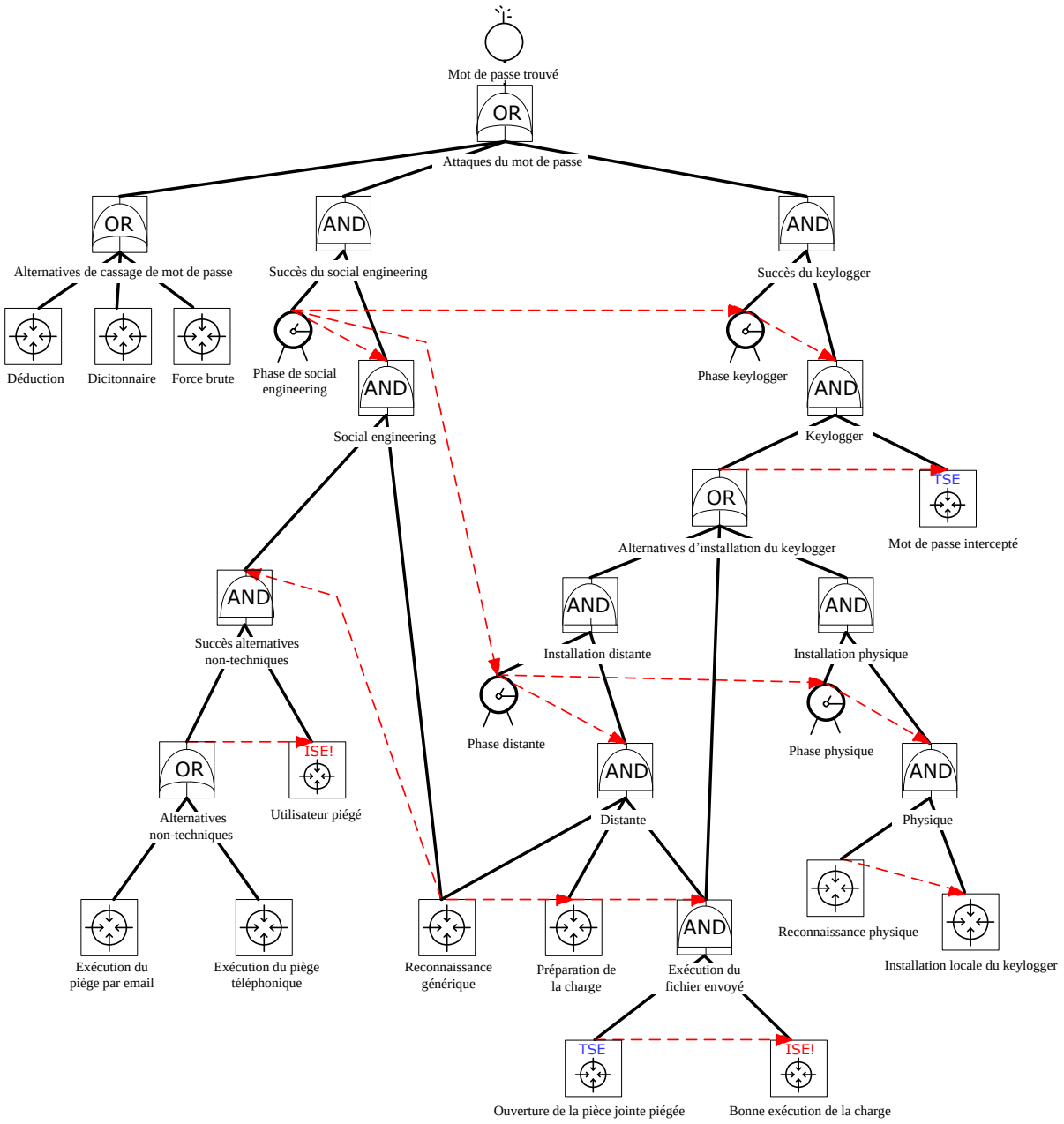


FIGURE III.18 – Attaque d'un fichier protégé par mot de passe

Analyse générale. Les paramètres de la Table III.10 conduisent à une probabilité de succès en une semaine de $0,422$ avec un MTTS global de 22 jours environ. Une exploration exhaustive du modèle aboutit à 654 séquences d'attaque. La Table III.11 en montre une sélection représentative, dans laquelle les séquences sont ordonnées selon leur contribution, *i.e.* le rapport entre leur probabilité d'occurrence dans le temps de mission considéré et celle de la réalisation de l'attaque dans ce même temps toutes séquences confondues.

Dans cette table, le début d'une phase est marqué par une balise <phase> et se termine par </phase> (si l'objectif n'a pas été atteint entre temps). Même si les phases ne sont pas en soi des événements de base, elles font partie intégrante de la description des séquences, structurant leur chronologie. Il en est de même pour les feuilles réalisées inutilement, repérées en italiques dans la description des séquences. En fait, la plupart des séquences incluent au moins un événement de base réalisé mais dont la réalisation n'a finalement pas contribué à l'atteinte de l'objectif final de l'attaque : ces séquences sont dites non minimales. Si l'on met de côté ces événements inutiles, on obtient les sous-séquences minimales (sous-entendu, de succès).

Les séquences n° 1 à 4 sont déjà minimales, et représentent d'un point de vue probabilité d'occurrence 47 % de toutes les séquences. Les séquences n° 5 et n° 6 sont de bons exemples de séquences non minimales. Force brute est une feuille spécifique, car sa réalisation constitue aussi la seule séquence minimale à un élément ; elle apparaît directement comme séquence minimale en ligne n° 3, mais ponctue également un nombre important de séquences non minimales. En fait, la contribution consolidée de toutes les séquences terminées par le succès de l'attaque par force brute pèse 40 % de toutes les séquences. Un tel poids, malgré le MTTS important de cette feuille, peut être expliqué par l'absence d'autres étapes à réaliser. Ce constat nous amène à une considération plus générale : une analyse poussée ne devrait pas considérer la seule liste des séquences mais également d'autres vues dont les contributions consolidées des séquences minimales. Nous y revenons en Section 6.3.7. Les séquences n° 3 à 19 n'impliquent que deux séquences minimales ; la séquence n° 20 amène une nouvelle sous-séquence minimale, puis il faut attendre la séquence n° 34 pour en trouver encore une nouvelle. Cette dernière illustre une spécificité des feuilles TSE, capables de se réaliser en mode Inactif si elles ont déjà été activées.

TABLE III.11 – Sélection de séquences avec quantification

	Séquence	**Probabilité (1 semaine)**	**Durée moyenne (s)**	**Contrib.**
1	<Phase Social Eng.>Reconnaissance générique, Exécution du piège par email, Utilisateur piégé	$1,059 \times 10^{-1}$	$9,889 \times 10^{4}$	25,1 %
2	<Phase Social Eng.>Reconnaissance générique, Exécution piège tél., Utilisateur piégé	$5,295 \times 10^{-2}$	$9,889 \times 10^{4}$	12,5 %
3	Force brute	$2,144 \times 10^{-2}$	$5,638 \times 10^{4}$	5,1 %
4	<Phase Social Eng.></Phase Social Eng.><Phase keylogger><Phase distante></Phase distante><Phase physique>Reconnaissance physique, Installation locale du keylogger, Mot de passe intercepté	$1,749 \times 10^{-2}$	$2,976 \times 10^{5}$	4,1 %
5	<Phase Social Eng.></Phase Social Eng.><Phase keylogger><Phase distante>*Reconnaissance générique* </Phase distante><Phase physique>Reconnaissance physique, Installation locale du keylogger, Mot de passe intercepté	$1,350 \times 10^{-2}$	$3,677 \times 10^{5}$	3,2 %
6	<Phase Social Eng.>*Reconnaissance générique, Exécution du piège par email, Utilisateur piégé(échec),* Force brute	$1,259 \times 10^{-2}$	$2,610 \times 10^{5}$	3,0 %
...				
20	<Phase Social Eng.></Phase Social Eng.><Phase keylogger><Phase distante>Reconnaissance gén., Préparation de la charge, Bonne exéc. de la charge, Mot de passe intercepté	$2,500 \times 10^{-3}$	$2,761 \times 10^{5}$	0,6 %
...				
34	<Phase Social Eng.></Phase Social Eng.><Phase keylogger><Phase distante>Reconnaissance gén., *Préparation de la charge* </Phase distante><Phase physique>*Ouverture de la P.J. piégée, Bonne exéc. de la charge*, Reconnaissance physique, Installation locale keylogger, Mot de passe intercepté	$1,506 \times 10^{-3}$	$4,594 \times 10^{5}$	0,4 %

Étude de sensibilité. Afin de compléter l'analyse donnée dans la section précédente, et de fournir des éléments pour faciliter des décisions de sécurité, nous étudions maintenant l'incidence de changements de valeur de différents paramètres du modèle sur la probabilité de réussite globale de l'attaque dans le temps de mission (P_s). Nous considérons dans un premier temps ces variations pour les feuilles de type AA, d'abord individuellement, puis en comparant plus formellement leur incidence respective sur P_s [8]. Nous étudions ensuite l'incidence des paramètres des feuilles de type ISE.

La Figure III.19 donne l'évolution de P_s lorsque le temps moyen nécessaire au succès (MTTS) de l'attaque par force brute varie autour de sa valeur initialement choisie (1 mois, repérée par un losange blanc). La baisse du MTTS correspond à un mot de passe plus simple et donc plus rapide à trouver par recherche exhaustive automatisée, tandis ce que l'augmentation du MTTS correspond à un mot de passe plus difficile à attaquer par cette technique. Les valeurs représentées vont de 2 à 60 jours, avec un pas de 4 jours. L'incidence sur P_s est nette, s'étalant entre $0,353$ et $0,978$ (soit une multiplication d'environ 2,8). Elle est particulièrement marquée pour des MTTS inférieurs à 1 mois. Comme déjà souligné dans la section précédente, l'attaque par force brute constitue une séquence minimale d'ordre 1 : la réalisation de cette feuille seule suffit pour que l'attaque soit réussie, ce qui explique l'influence de son paramétrage.

La Figure III.20 correspond à l'évolution de P_s lorsque le MTTS de préparation de la charge varie autour de sa valeur initialement choisie (2 jours) par tranche de 6 h entre une valeur minimum de 6 h et un maximum de 96 h (*i.e.* 4 jours). Une diminution correspond à un système informatique dont les vulnérabilités sont plus nombreuses, plus connues ou plus faciles à exploiter ; une durée moyenne plus importante pour l'attaquant reflétera *a contrario* un système moins vulnérable (par une politique de mise à jour améliorée par exemple). L'effet de ces variations sur P_s est minime (de $0,418$ à $0,436$, *i.e.* une augmentation d'environ 4 %), notamment vis-à-vis des variations observées pour la feuille Force brute.

La Figure III.21 montre l'effet des changements de valeur du temps moyen nécessaire à l'étape de reconnaissance générique (initialement de 24 h) sur P_s. Cette durée dépend de différents facteurs techniques (*e.g.* configuration et « durcissement » [9] du système , accessibilité réseau) et organisationnels (*e.g.* publication d'informations utiles à l'attaquant). Une variation entre 1 h et 48 h montre une incidence plus significative que pour le MTTS de préparation de la charge malveillante, P_s variant de $0,391$ à $0,477$ (soit environ 22 %).

La Figure III.22 illustre l'effet de variation du MTTS de l'installation locale du *keylogger* sur la probabilité de réussite de l'attaque globale P_s. Cette durée est directement liée à la surveillance notamment physique des locaux et à la vigilance des personnels. Là aussi, initialement paramétré avec une valeur de 24 h, l'analyse d'incidence sur P_s est menée pour un MTTS allant de 1 h à 48 h. P_s varie dans des proportions similaires à celles de la variation associée à la feuille Reconnaissance générique, allant de $0,401$ à $0,467$ (soit environ 17 %).

Bien que les courbes des Figures III.19 à III.22 nous ont déjà permis de comparer informellement les incidences respectives des paramètres λ des feuilles de type AA sur P_s, les Figures III.23 et III.24 constituent des bases plus facilement exploitables dans cette optique. La Figure III.23 représente l'évolution de P_s selon la variation des paramètres considérés, normalisés par rapport à leurs valeurs initiales. Elle confirme le rôle prépondérant de la feuille d'attaque par force brute par rapport aux trois autres feuilles dans la réussite de l'attaquant, soulignant l'importance de la qualité du mot de passe pour l'empêcher d'arriver à ses fins. La Figure III.24 « zoome » sur les trois autres feuilles, permettant de constater la moindre incidence du MTTS de préparation de la charge par rapport à celui de l'installation du *keylogger*, lui-même moins influant que le MTTS de la reconnaissance générique. À ressources de sécurité contraintes, une fois la qualité du mot de passe assurée, il sera donc plus fructueux de concentrer les efforts de sécurité sur ce dernier aspect.

Nous nous intéressons enfin aux effets des paramètres des feuilles de type ISE, à savoir Utilisateur piégé et Bonne exécution de la charge. La Figure III.25 montre l'incidence des probabilités de réalisation correspondantes sur P_s. Les deux courbes peuvent être directement superposées sans normalisation particulière, les probabilités s'échelonnant naturellement entre 0 et 1. Cette figure appuie le constat suivant : il est plus intéressant pour le cas d'étude considéré de limiter la probabilité qu'un utilisateur se fasse piéger par des attaques de type *social engineering* plutôt que de tenter de renforcer les systèmes pour réduire les chances de bonne exécution d'un *malware*.

8. L'ensemble de données numériques représentées par les courbes de cette section est consigné dans l'Annexe C.

9. Le durcissement d'un système (*system hardening*) correspond au processus de réduction de fonctionnalités et de services offerts par un système au minimum requis afin de limiter la surface d'attaque. En particulier, les informations permettant d'identifier facilement les services et logiciels exécutés sur la machine sont éliminées.

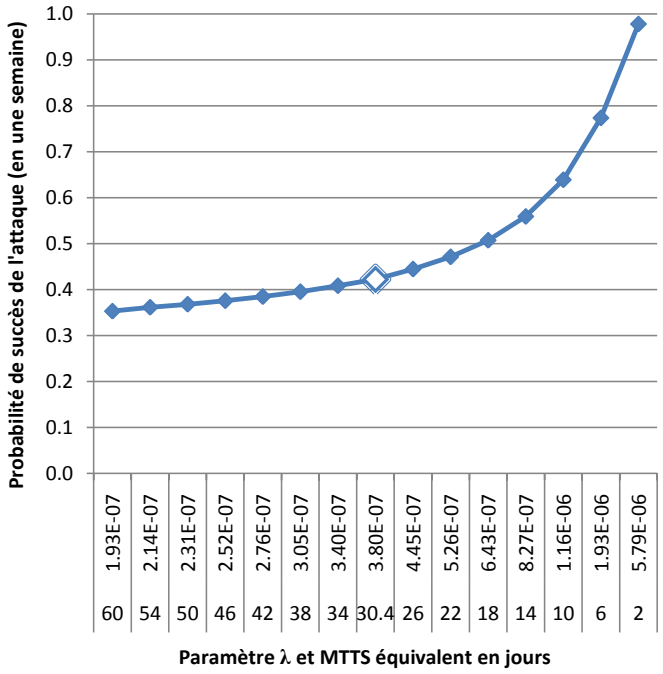


FIGURE III.19 – Incidence du paramétrage de la feuille Force brute sur P_s

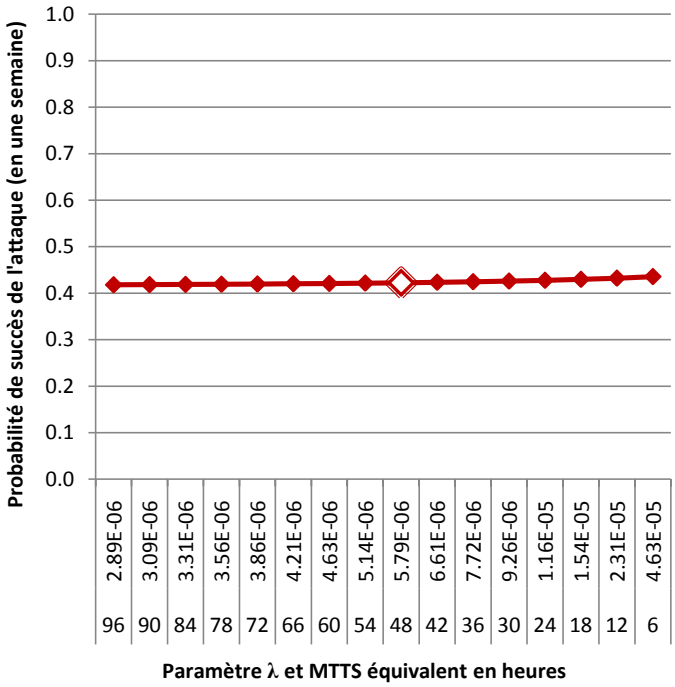


FIGURE III.20 – Incidence du paramétrage de la feuille Préparation de la charge sur P_s

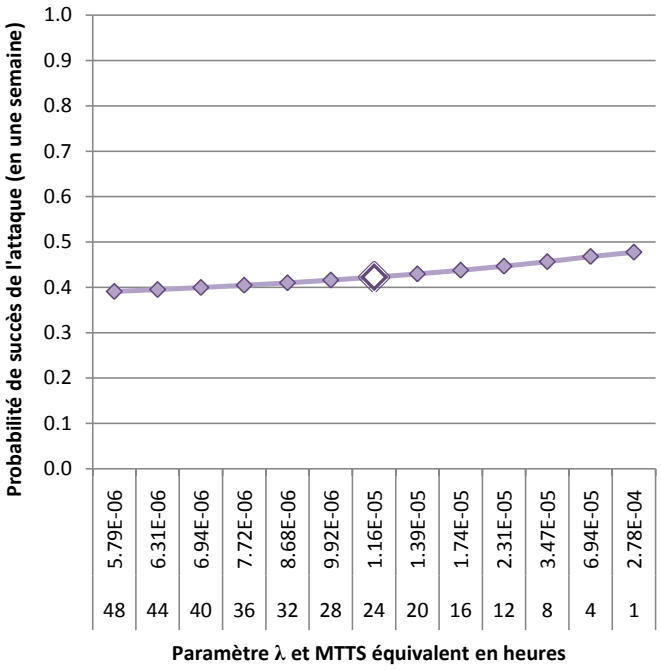


FIGURE III.21 – Incidence du paramétrage de la feuille Reconnaissance générique sur P_s

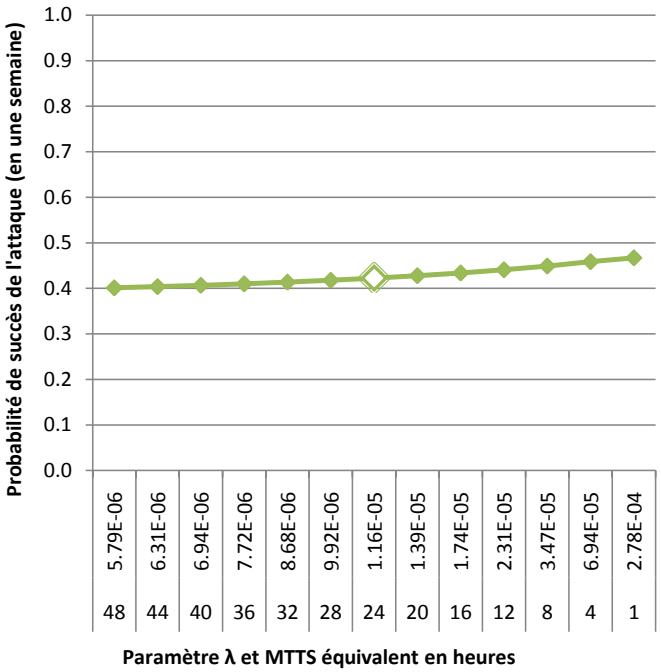


FIGURE III.22 – Incidence du paramétrage de la feuille Installation locale du keylogger sur P_s

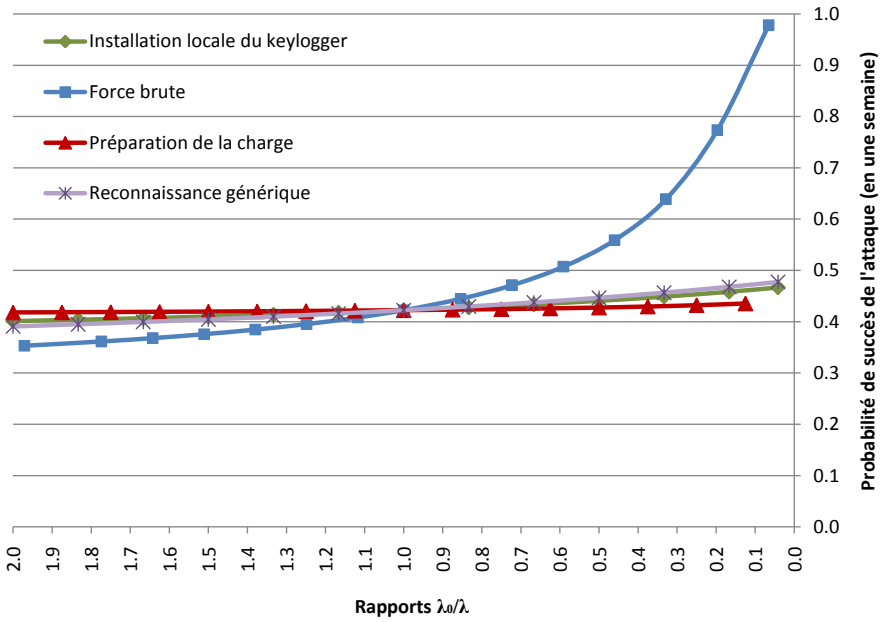


FIGURE III.23 – Comparaison de l'incidence du paramétrage des quatre feuilles de type AA sur P_s

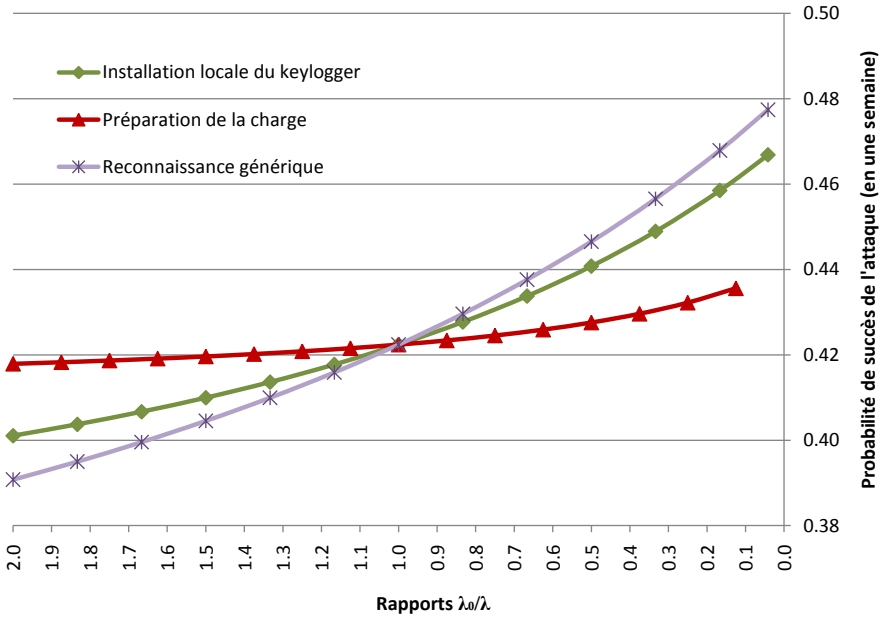


FIGURE III.24 – Zoom sur l'influence de trois des quatre feuilles AA sur P_s

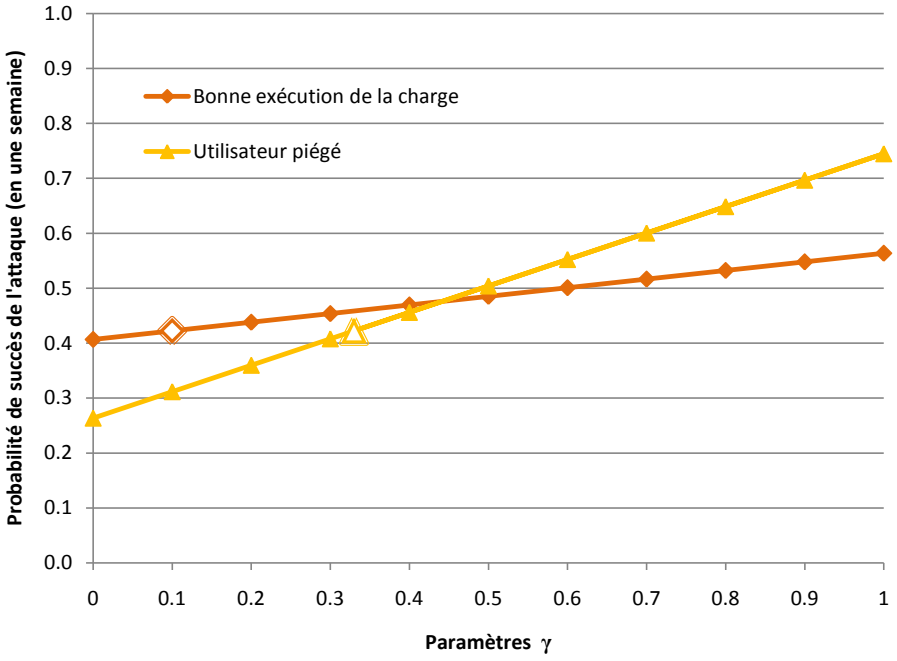


FIGURE III.25 – Incidence du paramétrage des feuilles ISE sur P_s

Globalement, il convient cependant de souligner que les comparaisons précédentes s'appuient sur des études paramétriques *ceteris paribus*, or des variations combinées mineures pourraient influencer P_s de façon plus marquée que certaines variations plus fortes mais limitées à un paramètre. L'analyse doit alors faire appel à des techniques d'optimisation et de décision à paramètres multiples, qui feront l'objet de travaux futurs (cf. Section 6.3.7).

3 Intégration des aspects défensifs : détections et réactions

Les approches classiques de la sécurité couvrent généralement les aspects de protection, de détection et de réaction [113, 23]. Dans l'adaptation des BDMP à la sécurité présentée en Section 2, le niveau de protection peut être considéré comme tacitement pris en compte, d'une part *via* la structure du BDMP, modélisant les seuls chemins considérés comme possibles pour l'attaquant, et d'autre part *via* les valeurs des paramètres (λ et γ), reflétant la compétence de l'attaquant mais aussi ses difficultés face à un niveau de protection donné. Dans cette section, nous nous intéressons à la modélisation des aspects de détection et de réaction, et présentons comment le cadre théorique des BDMP a été spécifiquement étendu pour prendre en compte ces dimensions.

3.1 La décomposition IEFA

L'intégration de la détection dans une modélisation dynamique nous a conduit à considérer quatre types de détection distincts pour les feuilles temporisées AA et TSE. Ces types se différencient par l'instant où la détection peut avoir lieu :

- une détection de type I (pour Initial) peut avoir lieu au moment où l'attaquant commence son action (cas des feuilles AA), ou au moment où l'événement modélisé est activé (cas des feuilles TSE) ;
- une détection de type E (pour En-cours) peut avoir lieu durant les tentatives de l'attaquant (cas des feuilles AA), ou pendant que l'événement modélisé est activé mais pas encore réalisé (cas des feuilles TSE) ;
- une détection de type F (pour Finale) peut avoir lieu au moment où l'attaquant réussit son action (cas des feuilles AA), ou au moment où l'événement modélisé est réalisé (cas des feuilles TSE) ;
- une détection de type A (pour *A posteriori*) peut avoir lieu après que l'action ou l'événement a été réalisé, sur la base des traces laissées par l'action ou l'événement en question.

Chacun de ces types a sa pertinence propre, dépendant de l'action ou de l'événement considéré. Une telle distinction permet une modélisation nuancée de la détection ; nous la désignons par l'acronyme IEFA [10], formé à partir des dénominations des quatre types de détection distingués.

Les feuilles ISE quant à elles, de par leur caractère instantané, ont été traitées quelque peu différemment. Nous avons distingué deux types de détection : la détection en cas de réalisation de l'événement (tirage favorable) et la détection en cas de non réalisation de l'événement (tirage défavorable).

3.2 Extension du cadre théorique

La modélisation des aspects de détection et de réaction nous ont amené à étendre le cadre théorique présenté en Section 2.2 des façons suivantes :

- en associant à chaque élément du BDMP une nouvelle fonction booléenne du temps [11] D_i appelée « indicateur de statut de détection » ;
- en remplaçant le mode Actif par deux modes : Actif Non-détecté (AN), et Actif Détecté (AD) ;
- en sélectionnant le mode d'un élément non plus en fonction de X_i uniquement, mais en fonction de la valeur booléenne X_iD_i, comme décrit par la Table III.12. Soulignons que dans les notations formelles utilisées dans les pages qui suivent, un 0 en indice correspond au mode Inactif et couvre les valeurs $X_iD_i = 00$ ou 01 ;
- en étendant les processus de Markov pilotés associés aux feuilles, avec de nouveaux états, transitions et fonctions de transfert permettant de modéliser les détections et les réactions.

10. L'acronyme utilisé dans la publication formalisant cette approche [479] est IOFA, pour *Initial*, *On-going*, *Final*, *A posteriori*. Il est changé ici uniquement pour des raisons de traduction.

11. Comme en Section 2.2, nous n'indiquerons pas le temps dans les descriptions formelles pour alléger les notations, mais il devrait apparaître partout.

TABLE III.12 – Le nouveau sélecteur de modeX_iD_i et les modes correspondants

X_iD_i	00	01	10	11
Mode	Inactif		Actif Non-détecté (AN)	Actif Détecté (AD)

3.2.1 Intégration des détections et des réactions dans les processus de Markov pilotés

Afin d'intégrer détections et réactions, nous redéfinissons les processus de Markov pilotés tels que présentés en Section 2.2.1. Dans ce cadre étendu, un processus de Markov piloté P_i est un ensemble $\left\{Z^i_0(t), Z^i_{10}(t), Z^i_{11}(t), f^i_{0\to10}, f^i_{0\to11}, f^i_{10\to11}, f^i_{10\to0}, f^i_{11\to0}\right\}$, où :

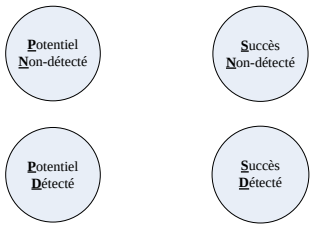
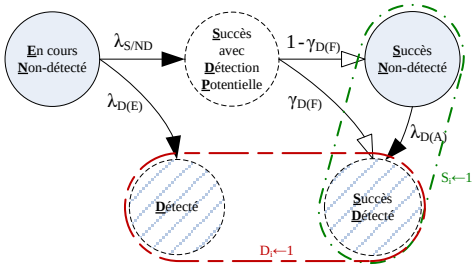
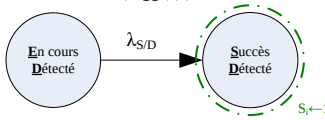
- $Z^i_0(t), Z^i_{10}(t), Z^i_{11}(t)$ sont trois processus de Markov homogènes à espaces d'états discrets. Pour k dans {0,10,11}, l'espace d'états de $Z^i_k(t)$ est A^i_k. Chaque A^i_k contient un sous-ensemble S^i_k qui correspond aux états de succès ou de réalisation de l'événement de base modélisé par P_i, et un sous-ensemble D^i_k qui correspond aux états de détection ;
- $f^i_{0\to10}, f^i_{0\to11}, f^i_{10\to11}, f^i_{10\to0}, f^i_{11\to0}$ sont cinq fonctions de transfert de probabilités telles que :
  - pour tout $x \in A^i_0$, $f^i_{0\to10}(x)$ est une distribution de probabilités sur A^i_{10}, telle que si $x \in S^i_0$, alors $\sum_{j\in S^i_{10}}(f^i_{0\to10}(x))(j) = 1$, et si $x \in D^i_0$, alors $\sum_{j\in D^i_{10}}(f^i_{0\to10}(x))(j) = 1$,
  - pour tout $x \in A^i_0$, $f^i_{0\to11}(x)$ est une distribution de probabilités sur A^i_{11}, telle que si $x \in S^i_0$, alors $\sum_{j\in S^i_{11}}(f^i_{0\to11}(x))(j) = 1$, et si $x \in D^i_0$, alors $\sum_{j\in D^i_{11}}(f^i_{0\to11}(x))(j) = 1$,
  - pour tout $x \in A^i_{10}$, $f^i_{10\to11}(x)$ est une distribution de probabilités sur A^i_{11}, telle que si $x \in S^i_{10}$, alors $\sum_{j\in S^i_{11}}(f^i_{10\to11}(x))(j) = 1$, et si $x \in D^i_{10}$, alors $\sum_{j\in D^i_{11}}(f^i_{10\to11}(x))(j) = 1$,
  - pour tout $x \in A^i_{10}$, $f^i_{10\to0}(x)$ est une distribution de probabilité sur A^i_0, telle que si $x \in S^i_{10}$ alors $\sum_{j\in S^i_0}(f^i_{10\to0}(x))(j) = 1$, et si $x \in D^i_{10}$, alors $\sum_{j\in D^i_0}(f^i_{10\to0}(x))(j) = 1$,
  - pour tout $x \in A^i_{11}$, $f^i_{11\to0}(x)$ est une distribution de probabilité sur A^i_0, telle que si $x \in S^i_{11}$ alors $\sum_{j\in S^i_0}(f^i_{11\to0}(x))(j) = 1$, et si $x \in D^i_{11}$, alors $\sum_{j\in D^i_0}(f^i_{11\to0}(x))(j) = 1$.

Notons que $f^i_{11\to10}$ n'a pas à être définie : dans notre cadre, un attaquant une fois détecté ne peut plus redevenir non détecté.

Les processus de Markov pilotés présentés en Section 2.2.4 sont réaménagés pour intégrer les aspects de détection et de réaction, comme décrit dans les Tables III.13, III.14 et III.15. Ils intègrent la décomposition IEFA : la détection est possible pour une action ou un événement temporisé à son tout début, durant sa réalisation, à son issue ou même *a posteriori*. Les paramètres de transition associés aux détections sont marqués par un D en indice. Dans le cas des feuilles AA et TSE, le D est suivi entre parenthèses du type de détection caractérisé (c'est-à-dire d'un I, d'un E, d'un F ou d'un A). Dans le cas des feuilles ISE, il est suivi par l'issue caractérisée : «/R » dans le cas d'une réalisation favorable à l'attaquant, «/NR » si l'issue est défavorable. Les paramètres de succès et de réalisation sont liés au statut de détection de la feuille : un «/D » en indice signifie « sachant que l'attaquant est détecté », un «/ND » signifie « sachant que l'attaquant n'a pas été détecté ». Les disques dont la circonférence est en pointillé représentent des états instantanés, les autres sont des états temporisés. Par états instantanés, nous désignons :

- des états introduits artificiellement à des fins de lisibilité. C'est par exemple le cas des états SDP dans la Table III.13. Ils pourraient être facilement éliminés en fusionnant les transitions temporisées rentrantes vers ces états avec les transitions instantanées sortantes, mais l'intelligibilité des graphes en pâtirait ;
- des états spéciaux « de déclenchement », qui ont été introduits pour changer les valeurs des D_i, et déclencher les changements idoines de modes sur la base de l'évolution interne aux feuilles modélisées. C'est par exemple le cas dans la Table III.13, dans le mode AN : une arrivée dans l'état « Détecté » ou dans l'état « Succès Détecté » déclenche un changement de mode instantané vers le mode AD. Les deux arrivées mettent le statut de détection D_i à 1, faisant passer la valeur booléenne X_iD_i utilisée comme sélecteur de mode, de 10 à 11. Ces états spéciaux de déclenchement sont représentés par des disques hachurés.

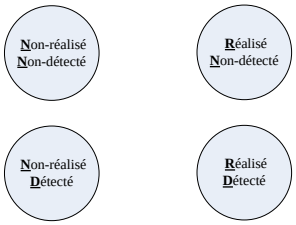
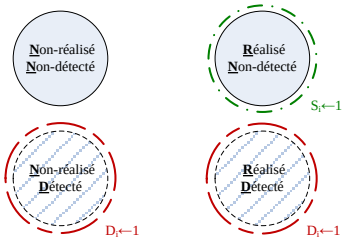
TABLE III.13 – Processus de Markov piloté de la feuille *Attacker Action* (AA)

Processus de Markov	**Fonctions de transfert de probabilités**
Inactif $(Z_0^i(t))$  	$f^i_{0\to10}(PN)$= {*Pr(EN)= 1 – $\gamma_{D(I)}$, Pr(D)= $\gamma_{D(I)}$, Pr(SD)= 0, Pr(SN)= 0*} *(PD)= {Pr(EN)= 0, Pr(D)= 1, Pr(SD)= 0, Pr(SN)= 0}* *(SN)= {Pr(EN)= 0, Pr(D)= 0, Pr(SD)= 0, Pr(SN)= 1}* *(SD)= {Pr(EN)= 0, Pr(D)= 0, Pr(SD)= 1, Pr(SN)= 0}*
Actif Non-détecté $(Z_{10}^i(t))$  	$f^i_{0\to11}(PN)$= {*Pr(ED)= 1, Pr(SD)= 0*}* *(PD)= {Pr(ED)= 1, Pr(SD)= 0}* *(SN)= {Pr(ED)= 0, Pr(SD)= 1}** *(SD)= {Pr(ED)= 0, Pr(SD)= 1}* $f^i_{10\to11}(EN)$= {*Pr(ED)= 1, Pr(SD)= 0*}* *(D)= {Pr(ED)= 1, Pr(SD)= 0}*** *(SD)= {Pr(ED)= 0, Pr(SD)= 1}*** *(SN)= {Pr(ED)= 0, Pr(SD)= 1}** $f^i_{11\to0}$ *(ED)= {Pr(PN)= 0, Pr(PD)= 1, Pr(SD)= 0, Pr(SN)= 0}* *(SD)= {Pr(PN)= 0, Pr(PD)= 0, Pr(SD)= 1, Pr(SN)= 0}* $f^i_{10\to0}$ *(EN)= {Pr(PN)= 1, Pr(PD)= 0, Pr(SD)= 0, Pr(SN)= 0}* *(SN)= {Pr(PN)= 0, Pr(PD)= 0, Pr(SD)= 0, Pr(SN)= 1}*
Actif Détecté $(Z_{11}^i(t))$  	

** La détection a eu lieu au niveau d'une autre feuille.*

*** Bien que D et SD soient de durée nulle, ces lignes sont nécessaires pour spécifier la fonction de transfert, le transfert pouvant être déclenché par la feuille elle-même.*

TABLE III.14 – Processus de Markov piloté de la feuille *Instantaneous Security Event* (ISE)

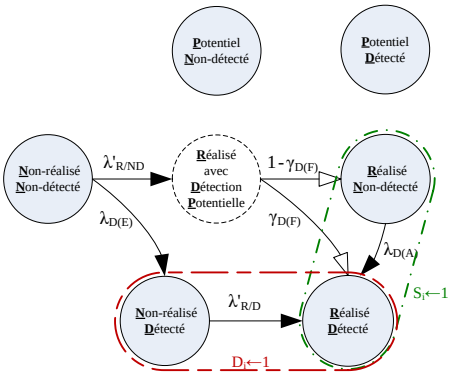
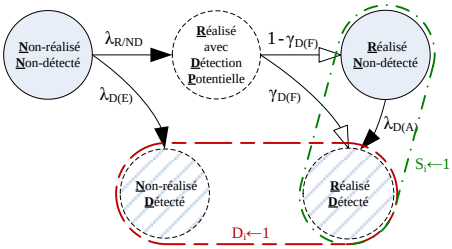
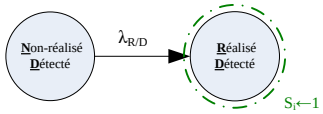
Processus de Markov	**Fonctions de transfert de probabilités**
Inactif $(Z_0^i(t))$ 	$f^i_{0\to10}$ *(NN)={Pr(NN)= (1–$\gamma_{S/ND}$)(1–$\gamma_{D/NR}$), Pr(RN)= $\gamma_{S/ND}$(1–$\gamma_{D/R}$), Pr(ND)= (1–$\gamma_{S/ND}$)$\gamma_{D/NR}$, Pr(RD)= $\gamma_{S/ND}\gamma_{D/R}$}* *(RN)={Pr(NN)= 0, Pr(RN)= 1, Pr(ND)= 0, Pr(RD) = 0}**** *(ND)={Pr(NN)= 0, Pr (RN)= 0, Pr(ND)= 1–$\gamma_{S/D}$, Pr(RD) = $\gamma_{S/D}$}* *(RD)={Pr(NN)= 0, Pr (RN)= 0, Pr(ND)= 0, Pr(RD) = 1}*
Actif Non-détecté $(Z_{10}^i(t))$ 	$f^i_{0\to11}$ *(NN)={Pr(ND)= (1 – $\gamma_{S/ND}$), Pr(RD)= $\gamma_{S/ND}$}** *(RN)={Pr(ND)= 0, Pr(RD)= 1}* *(ND)={Pr(ND)= (1 – $\gamma_{S/D}$), Pr (RD)= $\gamma_{S/D}$}** *(RD)={Pr(ND)=0, Pr (RD)=1}* $f^i_{10\to11}$ *(NN)={Pr(ND)= 1, Pr(RD)= 0}** *(RN)={Pr(ND)= 0, Pr(RD)= 1}** *(ND)={Pr(ND)= 1, Pr(RD)= 0}***** *(RD)={Pr(ND)= 0, Pr(RD)= 1}*****
Actif Détecté $(Z_{11}^i(t))$ 	$f^i_{11\to0}$ *(ND)={Pr(NN)= 0, Pr(RN)= 0, Pr(ND)= 1, Pr(RD)= 0}* *(RD)={Pr(NN)= 0, Pr(RN)= 0, Pr(ND)= 0, Pr(RD)= 1}* $f^i_{10\to0}$ *(NN)={Pr(NN)= 1, Pr(RN)= 0, Pr(ND)= 0, Pr(RD)= 0}* *(RN)={Pr(NN)= 0, Pr(RN)= 1, Pr(ND)= 0, Pr(RD)= 0}*

** La détection a eu lieu au niveau d'une autre feuille.*

*** Bien que D et SD soient de durée nulle, ces lignes sont nécessaires pour spécifier la fonction de transfert, le transfert pouvant être déclenché par la feuille elle-même.*

**** On suppose qu'une fois la feuille réalisée, les éventuelles réactivations ne peuvent plus provoquer de détection.*

TABLE III.15 – Processus de Markov piloté de la feuille *Timed Security Event* (TSE)

Processus de Markov	**Fonctions de transfert de probabilités**
Inactif $(Z_0^i(t))$ 	$f^i_{0\to10}$(PN)={Pr(NN)= 1 – $\gamma_{D(I)}$, Pr(ND)= $\gamma_{D(I)}$, Pr(RD)= 0, Pr(RN)= 0} (PD)={Pr(NN)= 0, Pr(ND)= 1, Pr(RD)= 0, Pr(RN)= 0} (NN)={Pr(NN)= 1, Pr(ND)= 0, Pr(RD)= 0, Pr(RN)= 0} (RN)={Pr(NN)= 0, Pr(ND)= 0, Pr(RD)= 0, Pr(RN)= 1} (ND)={Pr(NN)= 0, Pr(ND)= 1, Pr(RD)= 0, Pr(RN)= 0} (RD)={Pr(NN)= 0, Pr(ND)= 0, Pr(RD)= 1, Pr(RN)= 0} $f^i_{0\to11}$(PN)={Pr(ND)= 1, Pr(RD)= 0}* (PD)={Pr(ND)= 1, Pr(RD)= 0} (NN)= {Pr(ND)= 1, Pr(RD)= 0}* (ND)= {Pr(ND)= 1, Pr(RD)= 0} (RD) = {Pr(ND)= 0, Pr(RD)= 1} (RN) = {Pr(ND)= 0, Pr(RD)= 1}*
Actif Non-détecté $(Z_{10}^i(t))$ 	$f^i_{10\to11}$(NN)= {Pr(ND)= 1, Pr(RD)= 0}* (ND)= {Pr(ND)= 1, Pr(RD)= 0}** (RD) = {Pr(ND)= 0, Pr(RD)= 1}** (RN) = {Pr(ND)= 0, Pr(RD)= 1} * $f^i_{11\to0}$(ND)={Pr(PN)=0, Pr(PD)=0, Pr(NN)=0, Pr(ND)=1, Pr(RD)=0, Pr(RN)=0} (RD)={Pr(PN)=0, Pr(PD)=0, Pr(NN)=0, Pr(ND)=0, Pr(RD)=1, Pr(RN)=0} $f^i_{10\to0}$(NN)={Pr(PN)=0, Pr(PD)=0, Pr(NN)=1, Pr(ND)=0, Pr(RD)=0, Pr(RN)=0} (ND)={Pr(PN)=0, Pr(PD)=0, Pr(NN)=0, Pr(ND)=1, Pr(RD)=0, Pr(RN)=0} (RD)={Pr(PN)=0, Pr(PD)=0, Pr(NN)=0, Pr(ND)=0, Pr(RD)=1, Pr(RN)=0} (RN)={Pr(PN)=0, Pr(PD)=0, Pr(NN)=0, Pr(ND)=0, Pr(RD)=0, Pr(RN)=1}
Actif Détecté $(Z_{11}^i(t))$ 	* *La détection a eu lieu au niveau d'une autre feuille.* ** *Bien que D et SD soient de durée nulle, ces lignes sont nécessaires pour spécifier la fonction de transfert, le transfert pouvant être déclenché par la feuille elle-même.*

3.2.2 Propagation des réactions

Le modèle de Markov étendu de la feuille AA en mode Actif Non-détecté (cf. Table III.13) est une bonne illustration de la façon dont une détection est prise en compte en interne d'une feuille et peut provoquer un changement local de mode, du mode Actif Non-détecté au mode Actif Détecté. Ce basculement s'accompagne d'un changement de valeur du taux de succès, qui passe d'une valeur $\lambda_{S/ND}$ à une nouvelle valeur $\lambda_{S/D}$, reflétant une réalisation plus difficile, ou même impossible si $\lambda_{S/D} = 0$, pour l'attaquant une fois détecté. Le même genre de mécanisme est utilisé pour les autres feuilles. Ceci-dit, de tels changements de mode peuvent aussi être provoqués de façon externe, c'est-à-dire par une détection ayant eu lieu au niveau d'une autre feuille. En fait, on peut distinguer les possibilités suivantes :

- la détection a une incidence strictement locale ; seule l'action ou l'événement détecté est affecté, le reste du BDMP est inchangé, en particulier, les autres feuilles gardent leurs paramètres inchangés ;
- la détection a une incidence étendue, changeant non seulement la feuille correspondant à l'action ou l'événement détecté, mais également d'autres feuilles du BDMP, de manière sélective ;
- la détection a une incidence globale ; en cas de détection, tous les D_i sont positionnés à 1, ce qui implique que tous les événements et actions futurs seront considérés en mode Détecté, avec les paramètres modifiés associés.

Nous adoptons cette dernière approche dans le reste de ce chapitre : elle est à la fois pertinente en termes de modélisation sécurité, et directe en termes de formalisation et de mise en œuvre.

3.3 Cas d'application

Nous reprenons le cas d'application de la Section 2.6.2 en le complétant par des possibilités de détection et de réaction. Deux actions et deux événements de sécurité instantanés sont susceptibles d'être détectés, tandis que les changements provoqués par une détection sont reflétés par des changements de paramètre dans ces quatre feuilles, ainsi que pour deux autres directement liées. La Table III.16 présente ces différents éléments.

TABLE III.16 – Paramètres utilisés pour la modélisation des détections et réactions

Feuille	**Type**	**Paramètres quand attaquant non-détecté**	**Paramètres de détection**	**Paramètres quand détecté (mode AD)**
Utilisateur piégé	ISE	$\gamma_{S/ND} = 0,33$	$\gamma_{D/R} = 0$ $\gamma_{D/NR} = 0,5$	Non pertinent
Bonne exécution de la charge	ISE	$\gamma_{S/ND} = 0,1$	$\gamma_{D/R} = 0,1$ $\gamma_{D/NR} = 0,33$	$\gamma_{S/D} = 0,1$ (inchangé)
Ouverture P.J. piégée	TSE	$\lambda_{R/ND} = 1,157 \times 10^{-5} s^{-1}$ (MTTS ≈ 1 jour)	Détection impossible	$\lambda_{R/D} = 5,787 \times 10^{-6} s^{-1}$ (MTTS ×2 ≈ 2 jours)
Mot de passe intercepté	TSE	$\lambda_{R/ND} = 1,157 \times 10^{-5} s^{-1}$ (MTTS ≈ 1 jour)	Détection impossible	$\lambda_{R/D} = 5,787 \times 10^{-6} s^{-1}$ (MTTS ×2 ≈ 2 jours)
Reconnaissance physique	AA	$\lambda_{S/ND} = 5,787 \times 10^{-6} s^{-1}$ (MTTS ≈ 2 jours)	$\lambda_{D(O)} = 3,858 \times 10^{-6} s^{-1}$ (MTTS ≈ 3 jours), $\gamma_{D(I)}, \gamma_{D(F)}, \lambda_{D(A)} = 0$	$\lambda_{S/D} = 2,893 \times 10^{-6} s^{-1}$ (MTTS ×2 ≈ 4 jours)
Installation locale du keylogger	AA	$\lambda_{S/ND} = 1,157 \times 10^{-5} s^{-1}$ (MTTS ≈ 1 jour)	$\lambda_{D(O)} = 3,472 \times 10^{-5} s^{-1}$ (MTTS ≈ 8 heures), $\gamma_{D(I)}, \lambda_{D(A)} = 0$ $\gamma_{D(F)} = 0,1$	$\lambda_{S/D} = 5,787 \times 10^{-6} s^{-1}$ (MTTS ×2 ≈ 2 jours)

3.3.1 Analyse générale

Globalement, la prise en compte des aspects de détection et de réaction, tels que paramétrés selon la Table III.16, réduit la probabilité de succès pour l'attaquant en une semaine d'environ 14 %, passant de 0,423 à 0,364. Cette réduction somme toute modeste peut être expliquée par le fait que

la séquence prépondérante, la simple attaque hors-ligne par force brute, n'est pas sujette dans notre modèle au risque de détection. En fait, même en considérant les détections systématiques et les réactions parfaites (réalisation de l'action ou de l'événement rendue impossible), l'attaquant continuerait à avoir une probabilité de $0,2$ de succès sur les seules chances de réussite de l'attaque par force brute.

Du point de vue de l'analyse des séquences d'attaque, leur nombre est bien plus élevé que dans le cas sans détection (4321 vs 654 en Section 2.6.2) : cela s'explique par la prise en compte dans la description des séquences des événements de détection et de réaction. La Table III.17 donne une sélection représentative ordonnée par contribution. Elle reprend les conventions de notation introduites précédemment pour la Table III.11 en Section 2.6.2, augmentées des indications de détection mises entre parenthèses. Là encore, les deux premières séquences correspondent au succès direct de techniques de *social engineering*, suivies par celui de l'attaque par force brute. Dans le cas présent, ce trio de tête est suivi par plusieurs séquences non minimales terminées sur le succès d'une attaque par force brute, avant que les premières séquences correspondant au succès d'un courriel piégé d'une charge malveillante apparaissent (séquences n° 14 et n° 17). Ceci diffère de la Table III.11 où les séquences basées sur des approches physiques apparaissent en premier, alors qu'elles sont ici reléguées à la séquence n° 20 et plus. Ceci est dû aux possibilités de détection et de réaction associées à de telles séquences. Dans la séquence n° 20, l'attaquant échoue dans sa tentative de *social engineering* par courriel forgé et se fait détecter : les valeurs des paramètres utilisées dans les feuilles des étapes subséquentes sont donc celles des modes détectés.

Table III.17 – Sélection de séquences avec quantification

	Séquence	**Probabilité (1 semaine)**	**Durée moyenne (s)**	**Contrib.**
1	<Phase Social Eng.>Reconnaissance générique, Exécution du piège par email, Utilisateur piégé	$1,091 \times 10^{-1}$	$9,889 \times 10^{4}$	30,0 %
2	<Phase Social Eng.>Reconnaissance générique, Exécution piège tél., Utilisateur piégé	$5,456 \times 10^{-2}$	$9,889 \times 10^{4}$	15,0 %
3	Force brute	$2,144 \times 10^{-2}$	$5,638 \times 10^{4}$	5,9 %
4	<Phase Social Eng.>*Reconnaissance générique*, Force brute	$1,055 \times 10^{-2}$	$9,889 \times 10^{4}$	2,9 %
...	([...], Force brute) × 9			
14	<Phase Social Eng.></Phase Social Eng.><Phase keylogger><Phase distante>Reconnaissance générique, Préparation de la charge, Bonne exécution de la charge (non détecté), Mot de passe intercepté	$2,250 \times 10^{-3}$	$2,761 \times 10^{5}$	0,6 %
...	([...], Force brute) × 2			
17	<Phase Social Eng.>Reconnaissance générique </Phase Social Eng.><Phase keylogger><Phase distante>Préparation de la charge, Bonne exécution de la charge (non détecté), Mot de passe intercepté	$1,923 \times 10^{-3}$	$2,688 \times 10^{5}$	0,5 %
...	([...], Force brute) × 2			
20	<Phase Social Eng.>*Reconnaissance générique, Exéc. du piège par email, Utilisateur piégé (échec et détection)* </Phase Social Eng.><Phase keylogger><Phase distante></Phase distante><Phase physique>Reconn. physique, Installation locale keylogger, Mot de passe intercepté	$1,549 \times 10^{-3}$	$5,991 \times 10^{5}$	0,4 %

3.3.2 Étude de sensibilité

Comme en Section 2.6.2, l'analyse peut être complétée par l'étude de l'incidence des différents paramètres du modèle sur la probabilité de réussite globale de l'attaque dans le temps de mission (notée P_s). En plus des paramètres de réussite λ_S et γ_S des feuilles AA et ISE, ou de durée moyenne de réalisation des feuilles TSE, l'incidence des paramètres de détection sur P_s peut aussi être observée.

À titre d'exemple, nous représentons en Figure III.26 l'évolution de P_s selon différentes valeurs pour les paramètres de détection de type I considérés dans le modèle (cf. Table III.16). Les valeurs initialement choisies sont indiquées par des marqueurs blancs. Comme dans la Section 2.6.2, les calculs sont faits paramètre par paramètre *ceteris paribus*. Une approche permettant l'étude des effets de l'évolution de plusieurs paramètres à la fois constitue un axe d'amélioration pour des travaux futurs. Globalement, l'incidence des trois paramètres sur P_s est modeste, caractérisée par la pente faible des courbes de la Figure III.26. La Figure III.27 représente les même données sur un axe des ordonnées plus resserré.

Elle confirme une incidence plus marquée de la détection des tentatives de *social engineering* (piège téléphonique ou par *email*) sur P_s par rapport aux deux autres détections. Une amélioration de ce paramètre correspondrait par exemple à une formation, une sensibilisation et une vigilance accrues des personnels de l'organisation attaquée. Elle est préférable à une amélioration de même ordre de grandeur du temps de détection de l'exécution d'une charge malveillante. D'autres études paramétriques peuvent être menées pour appuyer des décisions de sécurité à ressources contraintes, ceci-dit, aucun ajustement ne remet en cause la prépondérance de l'attaque par force brute, évoquée en Section 3.3.1, dans la réussite globale de l'attaque.

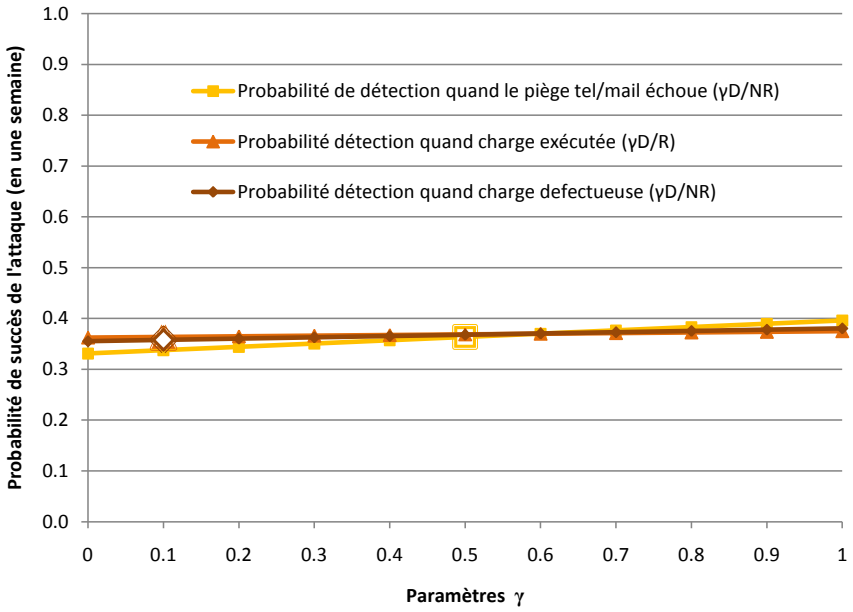


FIGURE III.26 – Une incidence modeste pour les paramètres de détection de type I sur P_s

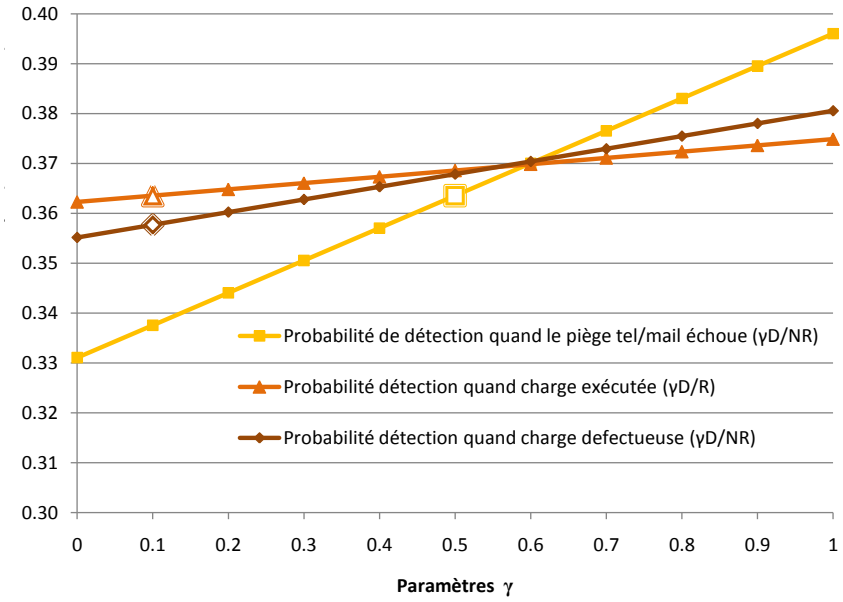


FIGURE III.27 – Comparaison de l'incidence des paramètres de détection de type I sur P_s

4 Éléments de comparaison avec les autres formalismes

Afin de mieux cerner l'intérêt des BDMP pour la modélisation d'attaques, nous revenons dans cette section aux cinq familles de formalismes analysées en début de chapitre, et discutons de leurs avantages et inconvénients respectifs par rapport aux BDMP, ainsi que de leur éventuelle complémentarité. Sur la base de cette comparaison, nous évaluons ensuite les BDMP selon les critères de la Section 1, synthétisant leurs apports et limites vis-à-vis des formalismes existants.

4.1 BDMP et arbres d'attaque

Malgré leur ressemblance visuelle et leur structure commune d'arbre logique, les BDMP et les arbres d'attaque diffèrent assez fondamentalement : les premiers sont un modèle dynamique alors que les seconds sont statiques. Si l'on considère le caractère éminemment dynamique de la sécurité (discuté en outre en II-1.2.3), ceci constitue un avantage majeur pour les BDMP. En particulier, les notions de phases et de séquences ne peuvent pas être prises en compte par les arbres d'attaque classiques, alors qu'elles font partie intégrante des scénarios d'attaque types à modéliser. De plus, bien que certaines variantes d'arbres d'attaque prennent en compte explicitement les aspects défensifs (*e.g.* [318, 284]), elles le font là aussi de façon statique et ne restituent pas la dimension temporelle des détections et réactions telles que discutées dans la Section 3. Enfin, si les BDMP permettent les traitements classiquement associés aux arbres d'attaque (calcul de coûts d'attaque, sélection des scénarios par profil d'attaquant, cf. Section 2.4.4), les quantifications d'ordre temporel des BDMP ne peuvent pas être effectuées sur des arbres d'attaque.

4.2 BDMP et *misuse cases*

Comme déjà mentionné, les *misuse cases* correspondent à un formalisme très macroscopique qui n'est pas adapté à la modélisation détaillée de scénarios d'attaque. Les arbres d'attaque et *a fortiori* les BDMP s'avèrent plus appropriés pour cela. Les diagrammes de *misuse cases* permettent par contre de mieux comprendre les interactions entre utilisateurs légitimes, systèmes et attaquants, et s'inscrivent dans une démarche plus large de spécification de système. En fait, au même titre que Tøndel *et al.* suggèrent dans [378] une utilisation combinée des arbres d'attaque avec les *misuse cases*, ces derniers pourraient également s'articuler de façon intelligible et pertinente avec des modèles BDMP. La réciproque est valable : les BDMP constituent un formalisme *a priori* adapté pour approfondir certaines parties d'un diagramme de *misuse cases*.

4.3 BDMP et réseaux bayésiens

Les réseaux bayésiens étant un formalisme statique, les précédentes considérations sur l'importance de la prise en compte de la dimension dynamique sont valables ici également. Il serait par contre réducteur de limiter à ces seuls aspects la comparaison entre BDMP et réseaux bayésiens. Ces derniers permettent en effet une plus grande souplesse dans la modélisation que les simples relations booléennes offertes par les arbres d'attaque, qu'ils généralisent. Aussi, bien que les réseaux bayésiens soient de nature statique, nous attribuerons aux BDMP une note équivalente sur ce plan. Notons que nous ne considérons pas ici la variante dynamique des réseaux bayésiens, son utilisation étant encore très marginale en sécurité et sa lisibilité encore plus limitée que celle du formalisme statique. Sur ce plan, les réseaux bayésiens comportent une quantité importante d'informations non représentées graphiquement, mais indispensables à la compréhension du modèle (tables de probabilités conditionnelles) et sont, à ce titre, moins directement lisibles que les BDMP.

Ceci-dit, au-delà de la simple comparaison, la collaboration des deux formalismes nous parait constituer une piste prometteuse. Il serait en effet avantageux de définir les paramètres des feuilles des BDMP par l'entremise de réseaux bayésiens, permettant de collecter différentes sources d'informations susceptibles de les influencer. En outre, ces influences pourraient inclure des considérations d'efficacité de politiques sécurité ou de dispositifs de protection mis en place comme cela est fait dans les modèles de Sommestad *et al.* (Cf. II-2.3.5.2).

4.4 BDMP et DFT (*Dynamic Fault Trees*)

DFT et BDMP ont été conçus sur une idée générale similaire : allier un formalisme proche des arbres de défaillances avec des capacités de modélisation dynamique. Alors que nous proposions d'adapter les BDMP à la sécurité [480], Khand introduisait l'emploi des DFT en sécurité [320]. Si les deux formalismes se ressemblent, ils présentent aussi plusieurs différences notables, sur le fond comme sur la forme.

Sur la forme tout d'abord, la prise en compte de la dimension dynamique par les BDMP ne se manifeste principalement que par l'ajout d'un composant à ceux employés dans les arbres statiques, la gâchette, alors que les DFT introduisent plusieurs portes spécifiques à cette fin (Cf. Chapitre II Section 2.2.4). De plus, la sémantique des gâchettes peut être considérée comme plus intuitive que celles de ces portes supplémentaires ; elle est en tout cas objectivement moins sujette à erreur, les portes des DFT rendant primordiales les positions respectives de leurs fils.

Sur le fond et d'un point de vue plus théorique, la formalisation mathématique des BDMP est plus complète et plus robuste (au sens donné et justifié par les définitions et propriétés mathématiques de la Section 2.2.3) que celle des DFT. En effet, comme souligné par Coppit *et al.* dans [481], la description des DFT est incomplète théoriquement, et amène à des représentations ambigües ; de plus, la correspondance avec les processus de Markov n'est pas définie pour le cas général (ces auteurs proposent d'ailleurs de remédier partiellement à ces carences par une notation formelle en langage Z). De plus, la puissance de modélisation des BDMP est supérieure aux DFT : dans leur domaine d'origine, *i.e.* la sûreté de fonctionnement, les BDMP modélisent sans difficultés systèmes non-réparables et réparables, alors que les DFT sont très limités pour ce dernier type de systèmes. La modélisation des défaillances de cause commune ou de certaines configurations de redondance constitue un autre point faible des DFT [482], auquel ne sont pas sujets les BDMP. En fait, comme montré dans [483], les BDMP peuvent être considérés comme une généralisation des DFT. Enfin d'un point de vue traitement et quantification, les DFT ne bénéficient pas du mécanisme de filtrage des événements pertinents et souffrent de leurs carences théoriques, rendant certaines configurations difficilement analysables.

4.5 BDMP et réseaux de Petri

Les réseaux de Petri constituent une famille de formalismes aux capacités de modélisation extrêmement puissantes. Bien que les BDMP puissent couvrir un nombre important de situations, ils n'égalent pas les réseaux de Petri à cet égard. Les situations les plus caractéristiques sont celles impliquant des boucles : les BDMP y sont très mal adaptés alors que les formalismes Petri les gèrent facilement. Ceci-dit, cette puissance de modélisation va souvent de pair avec les difficultés d'apprentissage et de lisibilité évoquées en Section 1.2.5. La nature de ces dernières est très dépendante du formalisme considéré. Ainsi, en GSPN, la modélisation de fonctions de structure équivalentes aux simples portes ET et OU des arbres d'attaque et des BDMP implique l'emploi d'arcs inhibiteurs et de multiples composants graphiques. Ces difficultés sont notamment mentionnées dans [244] et illustrées pour une porte ET dans la Figure III.28. La partie b) de la figure représente en GSPN la porte logique ET de la partie a) [12].

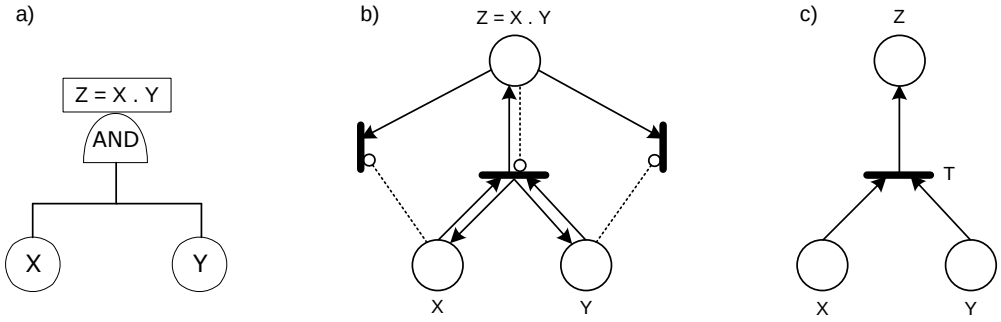


FIGURE III.28 – Modélisation comparée d'un ET logique en BDMP et en réseau de Petri

12. À noter que la comparaison est faite dans un cadre de sûreté de fonctionnement où les systèmes sont considérés comme réparables, ce qui nécessite les transitions latérales de la partie b) de la figure.

Il est surprenant de constater que de nombreux papiers comparant arbres d'attaque et modélisations par réseaux de Petri [330, 331, 321, 323, 289] font un raccourci pour le moins hâtif en postulant l'équivalence de la porte ET avec la structure représentée dans la partie c) de la Figure III.28. En effet, le comportement modélisé n'est pas strictement équivalent : une fois la transition T tirée, les places X et Y sont vides, comme si les « étapes » précédentes n'étaient plus réalisées. Ce n'est pas le cas d'une porte logique ET, pour laquelle les fils doivent rester dans un statut réalisé pour que la sortie reste vraie. Une telle différence pourrait paraître accessoire si elle n'avait pas des conséquences significatives en termes de modélisation, notamment pour des aspects de détection. En outre, la détection de type A (*a posteriori*) de l'extension des BDMP présentée en Section 2 nécessite un tel comportement pour être modélisable.

A contrario, un formalisme comme les SAN modéliserait sans difficultés particulières, et de façon très compacte, les équivalents de portes ET et OU. En fait, et plus généralement, les SAN peuvent modéliser de la sorte tous les composants des BDMP. Le problème posé est en fait à l'opposé des GSPN : les représentations graphiques en SAN intègrent un grand nombre d'information dans les descriptions, non représentées graphiquement, des activités, portes d'entrée et portes de sortie, éléments caractéristiques de ce formalisme. Dans les modèles SAN, des composants graphiquement identiques peuvent avoir des comportements très divers, leurs spécifications étant effectuées composant par composant. En fait, quels que soient les formalismes Petri considérés, la lisibilité des modélisations alors nécessaires pour obtenir des équivalents aux modèles BDMP reste très limitée.

Enfin, d'un point de vue traitement, si les réseaux de Petri bénéficient de nombreux outils et techniques, ils ne peuvent s'appuyer de façon native sur les mécanismes de filtrage des événements pertinents des BDMP, qui limitent le risque d'explosion combinatoire lors des traitements.

4.6 Synthèse sur les formalismes de modélisation graphique d'attaque

Les commentaires précédents nous permettent de mieux évaluer l'apport des BDMP par rapport à l'état de l'art, et de les caractériser vis-à-vis des cinq critères proposés en Section 1. La Table III.18 présente cette évaluation.

Table III.18 – Appréciation des BDMP pour la modélisation graphique d'attaque

Critère	Note	Commentaires
Facilité d'apprentissage	3	Le visuel des BDMP est proche de celui des arbres d'attaque, mais l'intégration de la dimension dynamique demande quelques efforts. Le mécanisme des gâchettes reste cependant intuitif. La compréhension en profondeur de la théorie sous jacente n'est pas nécessaire à la construction de petits modèles. La prise en compte des détections et des réactions peut être faite progressivement.
Lisibilité	4	Les BDMP héritent de la lisibilité des arbres d'attaque. L'ajout des gâchettes constitue le changement graphique principal.
Puissance de modélisation	3	Dépendances, séquences et réactions sont facilement modélisables. La principale limite tient dans la difficulté à modéliser des comportements en boucle.
Capacité de quantification	4	De par leur caractère hybride, les BDMP permettent de cumuler les techniques de quantification booléennes propres aux arbres d'attaque avec les techniques et outils de l'analyse markovienne pour les quantifications temporelles (cf. Section 6.3.1 si l'on sort du cadre markovien). Le mécanisme de filtrage des événements pertinents réduit les éventuels problèmes d'explosion combinatoire.
Capacité de passage à l'échelle	3	Comme pour les arbres d'attaque, la nature hiérarchique du modèle, mais également la possibilité de raisonner en *patterns* (cf. Section 6.3.3), permettent de traiter des scénarios d'attaque complexes sur des systèmes réalistes.

Nous rappelons dans la Table III.19 les évaluations effectuées en Section 1 et y intégrons l'appréciation des BDMP, permettant ainsi de mieux appréhender la teneur de notre contribution. Soulignons que ces formalismes ne sont pas nécessairement en compétition. D'une part, certains peuvent faire directement appel à d'autres (par exemple, les *misuse cases* peuvent s'appuyer sur des arbres d'attaque ; les BDMP peuvent intégrer des réseaux de Petri), d'autre part, une même analyse de risques peut employer différents formalismes selon le système ou la phase de l'analyse considérés.

En fait, les trois formalismes les plus proches, et *a fortiori*, les plus rivaux sont les arbres d'attaque, les DFT et les BDMP. Les Sections 4.1 et 4.4 regroupent des arguments tendant à démontrer que les BDMP constituent l'approche la plus satisfaisante. De telles conclusions ne peuvent être faites aussi

distinctement vis-à-vis des autres formalismes considérés, pour les raisons de complémentarité et de dépendance du contexte précédemment évoquées.

TABLE III.19 – Evaluation comparée des formalismes de modélisation graphique d'attaque

Formalisme **Critère**	Arbres d'attaque	Modèles à base de réseaux bayésiens	*Misuse cases*	DFT	Modèles à base de réseaux de Petri	BDMP pour la sécurité
Facilité d'apprentissage	4	2	3	3	2	3
Lisibilité	4	2	2	3	2	4
Puissance de modélisation	2	3	1	3	4	3
Capacité de quantification	4	3	1	2	4	4
Capacité de passage à l'échelle	3	2	2	3	2	3

5 Mise en œuvre logicielle

5.1 Support : la plate-forme logicielle *KB3*

5.1.1 *KB3*, bases de connaissances *Figaro* et outils associés

KB3 désigne la plate-forme logicielle utilisée et développée par EDF depuis une vingtaine d'années pour ses études de sûreté de fonctionnement [243]. Elle permet la construction de modèles statiques ou dynamiques variés par assemblage graphique de composants élémentaires décrits dans des bases de connaissances [484]. Ces composants peuvent être abstraits (portes logiques, liens, etc.), pour la saisie de modèles conceptuels, ou représenter des éléments plus concrets (vannes, moteurs, interrupteurs, etc.) pour saisir des schémas plus directement associés aux systèmes étudiés. L'objectif est de permettre la modularité, mais aussi la capitalisation et la réutilisation des connaissances des experts. Les descriptions sont écrites en *Figaro* [239], un langage objet spécifiquement conçu pour construire des modèles stochastiques par saisie graphique (cf. Chap. II Section 2.2.5). Les BDMP, mais également une grande partie des modèles graphiques présentés au Chapitre II Section 2.2, ont fait l'objet du développement de bases de connaissances correspondantes [485]. En plus d'une construction facile par l'interface graphique de *KB3* [486], elles permettent un traitement homogène par les mêmes codes de calcul, analysant les modèles générés en langage *Figaro*. En effet, une fois les modèles construits par *KB3*, ils peuvent être quantifiés selon différentes méthodes et leurs outils associés : *YAMS* permet de les traiter par simulation de Monte-Carlo [487] et *FIGMAT-SF* s'appuie sur une résolution matricielle [488], mais *Figseq* reste l'outil privilégié. Il met en œuvre la méthode d'exploration de chemins présentée dans la Section 2.4.2, et sur laquelle nous revenons d'un point de vue algorithmique dans le paragraphe suivant. La Figure III.29 donne une représentation macroscopique de la plate-forme *KB3*.

5.1.2 *Figseq* et les algorithmes d'exploration de chemins

Figseq met en œuvre les techniques de quantification de modèles markoviens par exploration de chemins, développées et largement utilisées à EDF dans les études de sûreté de fonctionnement [176]. Le principe général consiste à ne pas construire le graphe de Markov du modèle étudié dans son ensemble, pour éviter les problèmes d'explosion combinatoire, mais de l'explorer de proche en proche. Un parcours exhaustif reste possible, mais les grands modèles peuvent bénéficier d'approximations maîtrisées, en limitant l'exploration du graphe sur des critères de profondeur ou de seuil de probabilité.

L'application de ce principe peut se faire selon deux algorithmes, améliorés par diverses optimisations et heuristiques au fil de leur utilisation industrielle. Le premier, dit SN pour Séquences Normales, est réservé aux systèmes non-réparables ou dont le temps de mission est au plus de l'ordre de grandeur du temps de séjour dans l'état initial. La quantification de nos modèles en sécurité a été faite par cette approche. La probabilité de l'événement redouté à l'instant t correspond à la somme des probabilités de réalisation des séquences explorées y menant avant l'instant t. Le principe du calcul de la probabilité d'une séquence a été donné en Section 2.4.2.

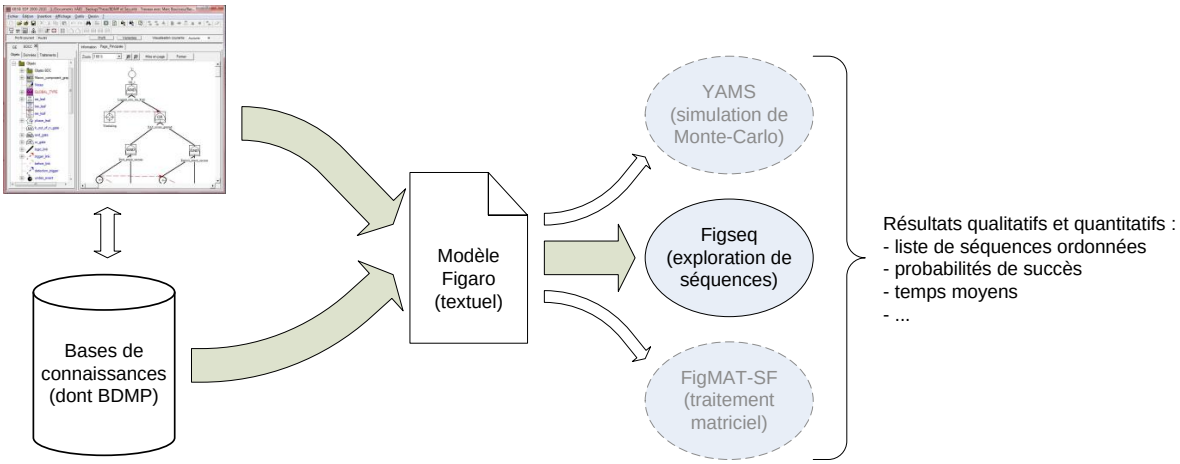


FIGURE III.29 – La plate-forme *KB3* et ses codes de calcul

Le second algorithme mis en œuvre par *Figseq* se nomme SRI, pour Sans Retour à l'état Initial ; il est réservé aux systèmes complètement et rapidement réparables. Il ne correspondait pas aux caractéristiques de nos modèles et n'a donc pas été utilisé dans le cadre de nos travaux. Le lecteur pourra se référer à l'ouvrage de Bouissou pour une explication didactique de son principe [176], ou aux articles de référence ayant jalonné son évolution [489, 490, 470].

5.2 Réalisations

5.2.1 Une base de connaissances sécurité

Avec l'aide de Yann Deflesselle, nous avons spécifié et développé une base de connaissances mettant en œuvre les principes exposés en Sections 2 et 3. En outre, les feuilles décrites par les modèles de Markov des Tables III.13, III.14 et III.15 ont pu être réalisées en *Figaro*, et intégrées avec les portes et liens nécessaires à la construction de modèles complets. En plus des feuilles spécifiques à la modélisation sécurité (AA, TSE et ISE), tous les éléments de la base de connaissances des BDMP classiques restent utilisables dans la construction des modèles sécurité : en outre, il est possible de spécifier des feuilles Petri par GSPN, et d'utiliser, en plus des portes AND et OR, des portes k/n et des portes PAND [13] (appelées alors portes *THEN*). La documentation de la base de connaissances historique décrit précisément tous ces composants [485]. Pour le développement de la base de connaissances sécurité, nous nous sommes appuyés sur l'outil *Visual Figaro*, logiciel dédié à l'aide au développement de base de connaissances [491]. Le résultat est complètement opérationnel et bénéficie de toutes les facilités et outils de traitement de la plate-forme *KB3*.

5.2.2 Outils d'aide à l'analyse

Les études de sensibilité des Sections 2.6.2 et 3.3.2 ont été réalisées grâce à un script écrit en *Python* itérant à volonté des traitements du modèle considéré par *Figseq*. Également développé avec l'appui de Yann Deflesselle, il permet de saisir facilement les paramètres à étudier, les valeurs minimales et maximales, et les pas d'incrémentation à appliquer aux paramètres choisis pour l'étude. Il constitue une base qui a encore vocation à s'étoffer en fonctionnalité et en profondeur d'analyse *via* des développements ultérieurs (cf. Section 6.3).

13. *Priority AND* (PAND), porte vérifiée si et seulement si tous ses fils se réalisent dans un ordre déterminé.

6 Discussion, limites et perspectives

6.1 Modélisation stochastique des attaques

6.1.1 Pertinence du cadre markovien et des lois exponentielles

Les BDMP, comme leur désignation complète l'indique, sont à l'origine basés sur l'emploi de processus de Markov et les distributions de probabilités exponentielles associées. Ce cadre markovien est nécessaire pour la démonstration des propriétés mathématiques exposées en Section 2.2.3, permettant en outre d'assurer qu'un BDMP décrit dans tous les cas un processus de Markov homogène. En particulier, l'utilisation des techniques de quantification basées sur l'exploration de chemins présentées en Section 2.4.2 ne peut se faire qu'à cette condition. Étant donné les propriétés à la fois simples et puissantes des processus de Markov, ils sont en fait communément adoptés en sûreté de fonctionnement (cf. Chapitre II Section 2.2.4), où les événements étudiés se prêtent, sous certaines conditions bien identifiées, à ce type de modélisation [188]. Aussi, ce choix n'avait rien de surprenant lors de la conception des BDMP, issus de cette communauté, et correspond toujours à l'état de l'art dans le domaine. Ceci-dit, il semble légitime de s'interroger quant à la pertinence de ces hypothèses dans une utilisation en sécurité.

Premier constat, l'utilisation de lois exponentielles pour modéliser les temps ou les efforts nécessaires aux attaquants pour réaliser leurs actions est relativement commune dans la littérature académique dédiée à la modélisation des processus d'attaque. L'idée d'utiliser des distributions exponentielles en sécurité semble avoir été introduite par Littlewood *et al.* en 1993 [177]. En 1994, le travail de Dacier sur les graphes de privilèges [257], évoqués en début de chapitre, repose également sur les techniques de l'analyse markovienne, et sur la modélisation par lois exponentielles des étapes d'une attaque, associées aux transitions d'un réseau de Petri stochastique sous-jacent. Le tout permet de calculer MTTF et METF (*Mean Effort To Failure*) globaux. Il faut attendre Jonsson et Olovsson en 1997 [475] pour voir une étude expérimentale conforter, sous certaines conditions, la modélisation par loi exponentielle d'un processus d'attaque. Celui-ci y est décomposé en trois phases : apprentissage, plus ou moins long selon la compétence de l'attaquant, phase standard, où les techniques classiques et les vulnérabilités connues sont explorées, et phase d'innovation, où l'attaquant, ayant fait le tour des approches standard, doit innover pour réussir l'attaque. Les résultats expérimentaux montrent que le temps moyen d'intrusion, appelé *Mean Time To Breach*, ou MTTB en référence au MTTF fiabiliste, peut être fidèlement approché par une loi exponentielle pendant la phase standard. Depuis, une telle approximation a été adoptée à de nombreuses reprises. On peut citer Ortalo *et al.* [492], qui reprennent en 1999 le modèle de Dacier dans une expérimentation confirmant certaines hypothèses du modèle, mais aussi Gupta *et al.* dans leur comparaison d'architectures tolérantes aux intrusions basée sur des SAN à transitions exponentielles [326] (2003), ou encore plus récemment, les travaux de thèse de Sallhammar, qui modélise la progression de l'attaquant par un graphe de Markov dont les taux de transition sont pondérés par des calculs typiques de la théorie de jeux [476]. De plus, tous les modèles de sécurité à base de GSPN impliquent des transitions instantanées et exponentielles : Ten *et al.* modélisent ainsi l'attaque de mots de passe et le passage d'un coupe-feu avec ce type de lois dans [324].

Si les travaux de Jonsson et Olovsson ont certainement contribué à la popularité des lois exponentielles pour la modélisation des efforts de l'attaquant, il faut aussi y associer un intérêt strictement pratique et mathématique. En effet, dans ce cadre théorique, tous les outils et techniques développés en sûreté de fonctionnement pour l'analyse markovienne peuvent s'appliquer (cf. Chap. II, Section 2.2.4). De plus, comme souligné par Dacier [257], les lois exponentielles sont très avantageuses pour formaliser l'hypothèse sous-jacente à nos raisonnements, à savoir que l'on peut estimer un temps moyen nécessaire à l'attaquant pour réaliser une action donnée. En effet, modéliser une telle durée par une loi exponentielle revient à définir la probabilité de succès avant un temps t par $Pr(t) = 1 - e^{-\lambda t}$. D'une part, le paramètre λ suffit à définir complètement la distribution de probabilité quel que soit t, d'autre part, sa valeur inverse $1/\lambda$ correspond au temps moyen de réussite, valeur estimée par l'analyste. On remarquera qu'une telle définition implique d'une part que la probabilité de réussite est nulle à $t = 0$, et qu'elle tend vers un à l'infini : on considère que l'attaquant peut toujours réussir une action modélisée par une feuille AA, mais que cette réussite nécessite un temps plus ou moins long. Elle peut le cas échéant tendre vers l'infini (c'est le cas quand on indique, de façon abusive, $\lambda = 0$).

De plus, une des propriétés principales des distributions exponentielles est leur comportement « sans-mémoire », fondement même des techniques d'analyse de Markov. Elle signifie dans notre cadre que les paramètres λ, *i.e.* les taux de succès et de détection, restent constants dans le temps, ou plus concrètement,

qu'ils n'évoluent pas en fonction du moment où l'action est considérée. S'il s'agit bien évidemment d'une simplification de la réalité, une telle hypothèse peut se justifier quand les techniques employées et/ou les composants attaqués diffèrent selon les feuilles considérées. Cela est globalement le cas des exemples proposés en Sections 2.6.1 et 2.6.2.

6.1.2 Autres lois et modélisations du comportement de l'attaquant

D'autres modélisations stochastiques des attaques ont été proposées dans la littérature, se débarrassant de la propriété sans mémoire des lois exponentielles, mais introduisant de la complexité dans le formalisme et dans le traitement des modèles. Ainsi, Madan *et al.* développent un cadre semi-markovien, offrant divers types de distribution pour modéliser l'effort de l'attaquant [493]. Ils suggèrent de considérer des lois hypo-exponentielles pour modéliser des taux de succès croissant de façon monotone avec le temps, des lois hyper-exponentielles pour des taux décroissant de façon monotone, ou des lois de Weibull permettant selon le choix de leurs paramètres de modéliser des taux constants, croissant, ou décroissant au fil du temps. Les distributions gamma et log-logistiques sont également mentionnées. McQueen *et al.* modélisent quant à eux le temps nécessaire à l'attaquant pour gagner de nouveaux privilèges par la combinaison de trois processus stochastiques distincts [494]. Ces trois processus sont paramétrés selon la connaissance estimée des vulnérabilités et des techniques d'exploitation par l'attaquant, et sont basés respectivement sur une loi gamma, une loi béta, et une loi exponentielle. Ce modèle est repris et légèrement modifié par Byres et Leversage dans [337, 338].

Alternativement à l'utilisation de lois de probabilités issues des approches classiques de sûreté de fonctionnement, il existe de nombreux travaux également basés sur l'emploi de probabilités, mais déterminées par le biais de la théorie des jeux. Utilisée dès sa naissance après la seconde guerre mondiale dans le domaine de la stratégie militaire [495], la théorie des jeux a rapidement trouvé sa place dans la modélisation de la malveillance en général, et appliquée aux problématiques de terrorisme en particulier (cf. Sandler *et al.* pour un panorama [496]). La protection des infrastructures critiques, dont les réseaux électriques, a ainsi été modélisée sous cet angle dans de nombreux travaux (*e.g.* [497]). Roy *et al.* donnent une taxonomie et un état de l'art de l'utilisation de la théorie des jeux en sécurité informatique dans [498]. Anderson *et al.* la situent dans le cadre plus large de l'emploi des approches économiques en sécurité informatique dans [260]. Globalement, nous rejoignons l'avis émis par Bier [499] : la modélisation de la malveillance s'appuiera avantageusement sur une combinaison de techniques issues de la sûreté de fonctionnement, pour leur capacité à modéliser des situations complexes, et d'outils issus de la théorie des jeux, pour prendre en compte l'attitude interactive de l'attaquant. C'est notamment ce que proposent Sallhammar [476], Bistarelli [314] ou encore Buldas *et al.* [311]. L'intégration de techniques de la théorie des jeux est une des pistes privilégiées pour de futurs développements de nos travaux (cf. Section 6.3.5).

Enfin, on peut noter l'émergence d'approches quantitatives de la sécurité appuyées sur la théorie de l'automatisme et du contrôle. On pourra se reporter pour des exemples à [500, 501].

6.1.3 *Time to compromise* et temps du compromis

Malgré les modèles alternatifs précédemment cités, nous avons choisi de conserver dans nos travaux d'adaptation et d'extension des BDMP une modélisation stochastique markovienne. Les raisons principales ont déjà été citées en Section 6.1.1 : mise à profit des propriétés mathématiques du formalisme original, utilisation des outils de traitement des modèles markoviens, mais aussi cohérence avec l'état de l'art et adéquation des lois exponentielles avec la pratique d'estimation de temps moyens nécessaires à la réalisation des actions d'un attaquant. Malgré les hypothèses et limites associées à la propriété sans mémoire, ce choix nous a paru constituer un compromis pertinent. Notons qu'il peut cependant être remis en question avec un impact limité sur le formalisme, mais plus profond sur les propriétés mathématiques et les techniques de quantification comme nous l'évoquerons en Section 6.3.1.

En fait, une fois cette hypothèse assumée, l'essentiel du débat porte ensuite sur la validité du paramétrage des feuilles : en effet, quand les estimations sont faites, les quantifications produites par les BDMP sont « mécaniques » et mathématiquement correctes. Elles portent le sens et la crédibilité que l'on voudra bien accorder aux valeurs choisies pour les paramètres du modèle. Nous rejoignons alors un débat plus large sur la pertinence d'utiliser la notion de probabilité en sécurité, déjà évoqué en outre en II-1.2.4. Dans une vision subjectiviste assumée, les probabilités restent à notre sens un outil adapté pour formaliser l'incertitude, inhérente à la notion de risque et caractéristique de la modélisation du comportement d'un attaquant. D'un point méthodologique, l'évaluation de probabilités d'occurrence, au

sens général, est une étape indispensable d'une analyse de risques (cf. Chap. II Section 1.1.1) ; toutefois, l'emploi d'une modélisation stochastique et d'un modèle graphique formel comme les BDMP doit y être considéré à sa juste place. Cette place est celle d'un outil de formalisation et d'aide à l'analyse, intrinsèquement limité par la nature même de la menace à caractériser, cependant mieux appréhendée de façon méthodique et outillée que par raisonnements informels et intuitions strictement qualitatives.

6.2 Limites intrinsèques des BDMP

Indépendamment des limites et débats associés à la modélisation stochastique de la sécurité, le formalisme des BDMP a, comme tout modèle, ses propres limites. Nous listons ici les principales, bien caractérisées dans leurs applications en sûreté de fonctionnement, et que nous commentons vis-à-vis de leurs implications en modélisation sécurité :

- dans leur utilisation d'origine, les BDMP sont peu adaptés aux systèmes dont les éléments sont en mouvement, créés ou détruits dans le temps de mission considéré. On peut donner comme exemple la modélisation de files d'attente, ou plus concrètement dans le domaine des installations industrielles, d'une chaîne de montage. En d'autres termes, si les BDMP sont un formalisme dynamique, le système à modéliser doit avoir une architecture prédéterminée [176]. En termes de modélisation d'attaques, les alternatives offertes à l'attaquant doivent être fixées à l'avance par l'analyste ;

- les notions de boucles et de cycles ne peuvent être prises en compte que de façon limitée, sous peine de perdre grandement en lisibilité. En particulier, l'outil *KB3* autorise l'intégration de réseaux de Petri (de type GSPN), définissables au gré de l'utilisateur sous forme de feuilles particulières. Cependant, leur emploi exagéré rend le BDMP moins lisible de par les comportements spécifiques, et éventuellement complexes, de chaque feuille. De façon moins pénalisante pour la lisibilité mais plus limitée, les feuilles phases présentées en Section 2.3.2 peuvent être chaînées de façon cyclique ;

- le cadre théorique, et les propriétés mathématiques précisées en Section 2.2, ne sont valables que pour des systèmes cohérents : en d'autres termes, la fonction de structure qui lie l'événement redouté (ou objectif de l'attaquant dans notre contexte) aux booléens S_i doit être monotone. Dans une approche fiabiliste, un système cohérent correspond au fait qu'une défaillance ne peut pas « réparer » le système, et rétablir la fonction qu'il est chargé d'assurer. Dans notre contexte, cette contrainte impose une modélisation des scénarios d'attaque sans portes NON, conséquence directe de l'exigence de monotonie de la fonction de structure. Les différents cas d'application et modélisations effectuées n'ont pas nécessité l'introduction d'une telle porte, l'exclusion des alternatives une fois une action réussie étant assurée par le mécanisme de filtrage des événements pertinents. De plus, il est possible d'aménager le comportement des gâchettes sans sortir du cadre théorique pour prendre en compte certains cas particuliers ;

- malgré le mécanisme de filtrage des événements pertinents et les techniques d'exploration des chemins, la taille des modèles a aussi ses limites. Un modèle comportant un nombre trop important de portes et d'événements de base sera difficilement quantifié par méthode analytique, mais pourra éventuellement être traité par simulation de Monte-Carlo. Il est délicat de caractériser précisément un tel seuil, éminemment dépendant de la structure de l'arbre, mais aussi des valeurs des paramètres. Par exemple, les BDMP comportant de nombreuses portes k/n avec des probabilités équitablement réparties (cas des systèmes très redondants en fiabilité) sont identifiés comme problématiques. En fait, bien que ce chiffre n'ait aucune justification théorique, le retour d'expérience de l'utilisation des BDMP dans les études de sûreté de fonctionnement à EDF indique qu'il n'y a pas ou peu de problèmes à quantifier des BDMP comportant moins d'une cinquantaine de feuilles. Au dessus, la structure, et en particulier les éventuelles symétries, peuvent considérablement influencer les performances de traitement. Le retour d'expérience dans leur utilisation en sécurité n'est à ce jour pas suffisant pour y confirmer la validité de ces constats, mais la nature des adaptations effectuées sur le plan théorique nous conduit à penser qu'ils sont transposables. L'ordre de grandeur indiqué pour la taille des modèles quantifiables sans difficulté donne une marge de manœuvre confortable pour la modélisation de scénarios d'attaque au regard de la pratique courante, et de l'état de l'art des autres formalismes.

6.3 Perspectives

6.3.1 Vers les BDSP ?

Nous avons évoqué le cadre markovien des BDMP et mentionné l'existence d'autres modélisations stochastiques pour les attaques dans la Section 6.1. En fait, une grande partie du formalisme des BDMP pourrait être étendue pour autoriser l'utilisation d'autres types de processus stochastiques que les seuls processus de Markov. Le formalisme graphique d'une telle extension, dont les gâchettes, resterait globalement inchangé ; la notion de processus piloté pourrait facilement être généralisée et l'articulation de la dynamique des modèles par les familles de fonctions booléennes du temps (S_i, X_i, Y_i et D_i) resterait valable. En fait, deux aspects seraient principalement touchés. Pour commencer, il serait délicat de conserver la même désignation... Les BDMP pourraient alors être renommés BDSP, pour *Boolean logic Driven Stochastic Processes*, élargissant explicitement le périmètre de la modélisation. Plus substantiellement, le second aspect concerne la quantification : les techniques de quantification par exploration de chemins, et plus généralement celles employées pour les modèles markoviens, devraient être remplacées par des simulations de Monte-Carlo. Notons que l'outil *KB3* s'interface déjà nativement avec un outil de simulation de Monte-Carlo (cf. Section 5.1). Nous prévoyons de développer et de tester une version BDSP de la base de connaissances évoquée en Section 5.2.1 prochainement, offrant la possibilité d'utiliser notamment la modélisation stochastique de McQueen *et al.* [494] (cf. Section 6.1.2).

6.3.2 Mise à profit du booléen de détection pour les quantifications

L'introduction d'un booléen de détection D_i a permis d'intégrer les aspects de détection et de réaction aux modélisations d'attaques par les BDMP. Cette prise en compte influence directement les valeurs globales comme le MTTS ou la probabilité de succès pour l'attaquant, tel qu'illustré dans le cas d'application de la Section 3.3. Elle modifie également qualitativement et quantitativement la liste de séquences produite par l'analyse d'un modèle. En complément, de nouveaux indicateurs pourraient être définis et calculés facilement sur la base du booléen D_i. En particulier, il serait intéressant d'ajouter aux quantifications globales le temps moyen de détection, *i.e.* le MTTD (*Mean Time To Detect*) et la probabilité d'être détecté dans le temps de mission considéré. Ces valeurs pourraient également être calculées pour chaque séquence identifiée. La hiérarchisation des séquences se ferait ensuite non pas uniquement sur des critères de probabilité de succès, mais également sur les chances de détection, selon un profil d'attaquant à déterminer en fonction de son aversion à la détection. Diverses pondérations sont alors imaginables.

6.3.3 Constitution d'une bibliothèque d'attaques

La construction de modèles au cours de ces travaux de thèse nous a conduit à identifier de façon informelle des motifs récurrents dans les scénarios d'attaque. Une identification plus rigoureuse et une catégorisation de ces motifs pourraient aboutir à l'élaboration d'une bibliothèque de petits BDMP, modélisant des étapes ou des techniques d'attaque récurrentes. Elle éviterait à l'utilisateur de les ressaisir complètement et permettrait leur assemblage pour la construction de modèles complets. L'idée, déjà proposée et mise en œuvre dans le cadre des arbres d'attaque, pourrait bénéficier des formalisations et du retour d'expérience issus de ces modèles.

6.3.4 Intégration d'extensions proposées pour les arbres d'attaque

De nombreuses extensions apportées aux arbres d'attaque, telles que celles discutées au Chapitre II Section 2.3.5.1, pourraient être transposées avantageusement aux BDMP. C'est le cas par exemple de la méthode de Jürgenson et Willemson [312] pour rationaliser les choix de l'attaquant en fonction des risques et des profits qu'il perçoit. La prise en compte des incertitudes sur le paramétrage des feuilles de l'arbre par l'approche proposée par ces mêmes auteurs dans [313] pourrait aussi présenter de l'intérêt. On peut également mentionner l'introduction de portes OWA (*Ordered Weighted Averaging*) décrites par Yager [315] : ces portes généralisent les portes OU et ET par des portes k/n spécifiques dont le nombre de fils devant être vérifié pour leur réalisation est modélisé en logique floue. Plus généralement, l'utilisation de quantificateurs flous pour le paramétrage de feuilles est une piste intéressante : une littérature abondante existe sur l'emploi de nombres flous dans les arbres de défaillances classiques depuis les travaux de Tanaka *et al.* [502], puis de Singer [503]. L'adaptation à un modèle dynamique nécessiterait cependant un travail théorique que nous n'avons pas estimé.

6.3.5 Théorie des jeux

Comme indiqué en Section 6.1.2, Bier souligne la pertinence de combiner méthodes issues de la sûreté de fonctionnement et de la théorie des jeux pour la modélisation sécurité [499]. Sallhammar concrétise cette idée et modélise l'avancée de l'attaque par un processus de Markov dont les transitions sont pondérées par des probabilités calculées par des jeux stochastiques [476]. D'autres contributions combinent arbres d'attaque et théorie des jeux [311, 318]. Ce type d'association pourrait être également mis à profit avec les BDMP, notamment dans la détermination des valeurs des différents paramètres λ et γ.

6.3.6 Intégration avec d'autres formalismes graphiques

Dans la Section 4, nous avons indiqué deux pistes prometteuses d'intégration des BDMP avec d'autres formalismes de modélisation d'attaque. Le première concerne une utilisation conjuguée avec les diagrammes de *misuse case*, la seconde avec les réseaux bayésiens. Nous n'avons pas identifié *a priori* d'obstacles théoriques aux rapprochements de ces approches à forte complémentarité, et projetons de les initier prochainement.

6.3.7 Étude de sensibilité et outils d'aide à l'analyse

Nous avons rapidement présenté un outil d'aide à l'analyse sécurité en Section 5.2.2, employé pour les analyses des Sections 2.6.2 et 3.3.2. Il permet de mener sous forme graphique des études de sensibilité des feuilles du BDMP, afin de faciliter les décisions de sécurité pour améliorer la protection du système attaqué. Cependant, l'approche mise en œuvre correspond à une façon partielle et informelle d'aborder la problématique de l'étude de sensibilité des modèles dynamiques, qui reste un problème de recherche ouvert. En effet, si les modèles statiques bénéficient de la définition de facteurs d'importance [14] bien établis [188, 224], ce n'est pas le cas des modèles dynamiques. Toutefois, l'état de l'art en la matière n'est pas vierge : on peut notamment citer les travaux de Dugan et Ou [504], ceux de Cao *et al.* [505], ceux de Natvig [506] ou ceux plus récents de Do *et al.* [507, 508, 509]. Ces derniers donnent, en français, une bonne introduction au problème et aux grandes familles d'approches associées dans [510]. Il serait intéressant d'évaluer leur pertinence dans le cadre de nos modèles, et de calculer des facteurs d'importance aidant l'analyste dans ses optimisations de sécurité. Par ailleurs, l'approche graphique adoptée dans les Sections 2.6.2 et 3.3.2 ne prend en compte que des variations de valeur paramètre par paramètre. Elle ne permet pas de considérer des optimisations impliquant des ajustements coordonnés de plusieurs valeurs : des techniques d'analyse plus élaborées devront pour cela être mises en œuvre. Nous prévoyons d'aborder ces développements dans de prochains travaux.

En plus des études de sensibilité, une autre piste à poursuivre dans ce type d'aide à l'analyse consiste à améliorer la restitution des séquences d'attaque fournies par *Figseq*. Nous avons déjà mentionné la possibilité de prendre en compte des attributs de nature statique tels que le coût, ou d'éventuelles exigences en termes d'outillage, d'appui interne ou de compétence (cf. Section 2.4.4 pour une discussion complète à ce sujet). La hiérarchisation des séquences d'attaque selon ces critères peut se faire de multiples façons : nous l'avons mise en œuvre par un simple mécanisme de seuil, mais des approches plus avancées de type pondération semblent tout à fait envisageables. Ces réflexions rejoignent ainsi les extensions possibles quant à l'intégration des notions de risque dans les paramètres des feuilles et vis-à-vis des choix rationnels de l'attaquant, évoquées dans les sections précédentes.

Enfin, les analyses de séquences données en Sections 2.6.2 et 3.3 ont déjà souligné l'intérêt de considérer les contributions des séquences dites minimales : l'automatisation du calcul de ces contributions reste à mener. Pour cela, nous comptons nous appuyer sur les travaux de Bouissou sur les notions de contenu minimal des séquences [511].

6.3.8 Application à d'autres domaines

L'adaptation des BDMP présentée dans ce chapitre a été en premier lieu effectuée pour une utilisation en sécurité informatique. Les différents cas d'usage de ce mémoire relèvent ainsi tous de ce contexte. Cependant, au même titre que pour les arbres d'attaque, l'emploi des BDMP en sécurité peut être considéré dans une perspective plus large, et concerner d'autres domaines que la seule sécurité informatique. En

14. Les facteurs d'importance caractérisent la contribution d'un composant à l'occurrence de l'événement redouté.

fait, partout où les arbres d'attaque ont été mis en œuvre, les BDMP peuvent constituer une alternative avantageuse sur les aspects déjà décrits en Section 4 : c'est par exemple le cas de la sécurité et la protection physique [512, 513], des garanties vis-à-vis des inspections nucléaires (*nuclear safeguards*) et le risque de prolifération [514, 515] ou du risque terroriste en général (*e.g.* [456]). Suite à notre travail d'adaptation, les BDMP peuvent désormais être considérés comme un formalisme polyvalent, capable à la fois de modéliser des problématiques de sûreté, mais également de problématiques de sécurité. Le chapitre suivant met à profit cette polyvalence pour étudier les relations de ces deux problématiques dans un usage intégratif des BDMP.

7 Conclusion

L'adaptation et l'extension du formalisme des BDMP à la sécurité offrent un compromis original en termes de lisibilité, capacités de passage à l'échelle, puissance de modélisation et capacités de quantification. Le cadre théorique des BDMP tel que défini en sûreté de fonctionnement a pu être transposé rigoureusement pour la modélisation de scénarios d'attaque, mais également étendu pour intégrer des aspects spécifiques à la sécurité. Plusieurs modalités de détection et de réaction peuvent maintenant être prises en compte dans le paramétrage des modèles. Visuellement, ce formalisme étendu hérite de la structure hiérarchique des arbres d'attaque, autorisant différentes profondeurs d'analyse et restant facile d'appropriation ; il va cependant beaucoup plus loin de par ses capacités de modélisation dynamique et ses possibilités de quantification. Une mise en œuvre logicielle complète et opérationnelle a permis de valider le modèle théorique proposé, et tire profit des développements et outils déjà disponibles dans le cadre des études de sûreté.

Chapitre IV

Modélisation et caractérisation des interdépendances sûreté-sécurité

Le couple, c'est ne faire qu'un. Reste à savoir lequel.
(Oscar Wilde.)

Dans le précédent chapitre, nous avons adapté les BDMP, un formalisme de modélisation dynamique utilisé pour les études de sûreté, à la modélisation graphique de scénarios d'attaque. Une telle adaptation a abouti à une approche convaincante au regard de l'état de l'art du domaine. Elle s'inscrit dans une filiation d'autres adaptations fructueuses entre sûreté et sécurité, évoquées dans le Chapitre II. Toutefois, si l'intérêt des travaux de fertilisation croisée ne fait pas de doutes, il devient également impératif de mieux caractériser et maîtriser leurs interdépendances, les problématiques de sûreté et de sécurité convergeant progressivement vers les mêmes systèmes. À cette fin, nous proposons dans ce chapitre de tirer parti de l'adaptation des BDMP à la sécurité pour en faire un formalisme intégrateur, capable de prendre en compte aspects de sûreté et de sécurité dans un même modèle graphique, pour mieux caractériser leurs interactions.

En Section 1, nous précisons la problématique et les enjeux associés aux interdépendances entre sûreté et sécurité, et en proposons une catégorisation macroscopique. L'état de l'art de cette thématique fait l'objet de la Section 2. La Section 3 correspond à la contribution principale de ce chapitre, et montre comment les BDMP peuvent aider à caractériser les situations où sécurité et sûreté doivent être considérées conjointement. La Section 4 situe cette contribution par rapport aux travaux déjà menés en la matière alors que la Section 5 introduit deux approches visant à en étendre la portée et l'utilité. La Section 6 discute des limites de nos propositions et mentionne plusieurs pistes et perspectives en découlant.

1 Interdépendances entre sûreté et sécurité

1.1 Caractérisation de la problématique

La convergence des risques de sûreté et de sécurité pour les systèmes informatiques du domaine industriel a été évoquée dans l'Introduction Générale de ce mémoire. La numérisation et l'interconnexion des systèmes pilotant ou surveillant des processus industriels conduisent à une situation nouvelle, où des systèmes auparavant isolés et seulement considérés sous l'angle du risque sûreté, doivent maintenant être aussi protégés des risques informatiques de nature malveillante. En d'autres termes, alors que sûreté et sécurité ont longtemps été considérées comme des problématiques cloisonnées, cette situation est révolue. La prise en compte de cette convergence est d'autant plus critique que la superposition d'exigences et de contre-mesures relatives à chacune des deux problématiques correspond à une configuration mal maîtrisée, et qui peut s'avérer dans certains cas contre-productive.

En effet, il semble communément admis que la sûreté des opérations peut être conditionnée par la sécurité [21, 516] : par exemple, la modification malveillante de données de capteurs ou de configurations d'automates peut empêcher des systèmes de sûreté de protéger une installation industrielle en état accidentel, voire même amener cette installation dans celui-ci. Toutefois, il existe des relations plus subtiles et plus variées qui doivent être considérées quand sécurité et sûreté se rencontrent. Un exemple du monde physique, donné dans [26, 34] permet de l'illustrer de façon simple et parlante : considérons un système d'ouverture et de fermeture automatisées de porte, point d'accès unique à un local. D'un point de vue sûreté, un tel système peut nécessiter un comportement de type *fail-safe* : en cas de panne, la porte est laissée ouverte pour rendre possible l'intervention d'équipes d'urgence ou une évacuation en cas de danger pour ses occupants (*e.g.* incendie). D'un point de vue sécurité au contraire, le système peut dans le cas d'un local à accès restreint nécessiter un comportement de type *fail-secure* : en cas de panne, la porte est maintenue fermée pour empêcher d'éventuels intrus de profiter de la panne, qu'ils auraient pu provoquer pour enfreindre l'interdiction d'accès. Bien sûr, une fois cet antagonisme identifié, de nombreuses solutions techniques sont imaginables ; cet exemple a pour seul but d'illustrer la possibilité d'exigences contradictoires portant sur un même système lorsque sûreté et sécurité sont considérées séparément. La Section 1.2.3 en fournit d'autres sur un plan physique mais aussi informatique. De plus, au-delà de ces exemples d'antagonisme, d'autres types d'interdépendance sont susceptibles d'influencer directement le niveau de risque du système ou de l'installation considérée. Nous en proposons une catégorisation générique dans la section suivante, appuyée par différents cas.

1.2 Une catégorisation macroscopique des interdépendances

D'un point de vue général, une lecture critique de la littérature traitant des relations entre sûreté et sécurité (cf. Chap. II et Section 2 de ce chapitre), et notre propre retour d'expérience en analyse de risques pour les systèmes d'informatique industrielle, nous ont amené à distinguer quatre grandes catégories d'interdépendance :

- **dépendance conditionnelle :** le respect d'exigences de sûreté conditionne le niveau de sécurité ou réciproquement ;
- **renforcement :** des mesures prises à des fins de sûreté contribuent également à la sécurité ou réciproquement ;
- **antagonisme :** les exigences de sûreté et de sécurité mènent conjointement à des situations conflictuelles ;
- **indépendance :** pas d'interaction.

Notons que si les relations entre sûreté et sécurité ont été discutées de façon générale par de nombreux auteurs (cf. Chap. II), peu ont explicitement établi ce type de catégorisation. En 1999, Eames et Moffet distinguent deux types d'interaction [26]. La première correspond à l'antagonisme de notre catégorisation. La seconde s'appuie sur une distinction entre exigences primaires (*primary requirements*) et exigences dérivées (*derived requirements*) : les exigences primaires ont leur justification dans leur propre domaine alors que les exigences dérivées, constituant le second type d'interaction en plus de l'antagonisme, ont leur justification dans le domaine dual (*i.e.* exigences sécurité fondées sur des raisons de sûreté ou

réciproquement). Ce second type correspond à la dépendance conditionnelle de notre catégorisation. Le renforcement et l'indépendance identifiés dans cette dernière ne sont pas explicitement évoqués. En 2007, Pan et Liu [517] distinguent l'antinomie, l'homologie et la dépendance : les deux premières distinctions renomment seulement celles de Eames et Moffet (avec un plagiat caractérisé quant à leur description), la troisième est à notre sens introduite artificiellement, car elle ne diffère pas sur le fond de la définition donnée pour la notion de dépendance.

Dans les sections suivantes, nous commentons et illustrons par différents exemples les trois premières catégories identifiées, à savoir la dépendance conditionnelle, le renforcement, et l'antagonisme ; l'indépendance ne présente pas d'intérêt particulier dans le cadre de notre discussion.

1.2.1 Exemples et considérations sur la dépendance conditionnelle

Ce type de relation est généralement connu et identifié, notamment dans le sens de la sécurité conditionnant la sûreté [21, 516, 406]. L'exemple générique mentionné au début de la Section 1.1 en est une directe illustration : la modification malveillante de données de capteurs ou de configurations d'automates peut effectivement empêcher des systèmes de sûreté de protéger une installation industrielle en état accidentel. Cette relation de dépendance conditionnelle se retrouve dans tous les contextes où des systèmes informatiques pilotent ou supervisent de processus industriels présentant un risque de sûreté. Par exemple, les références [518, 519] soulignent les besoins de sécurité des systèmes de signalisation ferroviaire pour garantir la sûreté des trains ; celle du transport aérien est aussi conditionnée par des problématiques de sécurité informatique dans [520, 521] ; on retrouve cette dépendance exprimée dans [522] pour l'industrie pétrolière, ou encore [523] pour l'automobile. Nordland l'illustre pour la signalisation ferroviaire, la télémaintenance des plates-formes pétrolières et la télémédecine dans [516].

La dépendance réciproque est plus rarement soulignée, mais dans certaines situations, la sûreté peut conditionner le niveau de sécurité d'un système ou d'une installation [386, 20]. À titre d'illustration générique, des conditions catastrophiques liées à un incident de sûreté, peuvent, si elles sont mal gérées, affaiblir le niveau de sécurité d'un système ou d'une organisation et provoquer des actes malveillants opportunistes. Ceci est particulièrement le cas lorsque sûreté et sécurité s'appuient sur des ressources partagées [159], qu'elles soient techniques ou humaines. Brewer mentionne une telle dépendance dans le contexte des systèmes informatiques dans [386].

1.2.2 Exemples et considérations sur le renforcement

Dans certains cas, sûreté et sécurité se renforcent mutuellement. Les similarités présentées dans le Chapitre II expliquent ces situations, dont l'identification peut amener des optimisations de ressources, voire éviter certains doublons. À titre d'illustration dans le monde nucléaire, certaines dispositions prises à des fins de sûreté, comme les séparations physiques et la diversification de certaines fonctions, contribuent également à la protection physique de l'installation contre le sabotage [142]. Sur un plan informatique, la plupart des approches de tolérance aux intrusions citées dans le Chapitre II Section 3.1.1 contribuent également à tolérer les fautes accidentelles (*e.g.* FRS [388, 390]). La réciproque n'est cependant pas aussi souvent valable. Ainsi, l'efficacité des techniques de redondance à tolérer les fautes est conditionnée par leur indépendance vis-à-vis du processus de création et d'activation des fautes [41] : cette indépendance peut être mise à mal par une intelligence malveillante alors qu'elle est plus facile à assurer dans des hypothèses purement accidentelles. En particulier, redonder un système ou un traitement à l'identique peut être adapté pour la tolérance aux fautes accidentelles mais s'avérera inefficace face à un attaquant, capable d'exploiter une vulnérabilité reproduite alors à l'identique. Les approches de diversification sont plus susceptibles d'être efficaces en contexte accidentel et malveillant. En fait, certaines approches de tolérance aux fautes initialement proposées dans une problématique accidentelle sont intrinsèquement mixtes : c'est le cas du modèle Totel (cf. Chap. II Section 3.2.2) [443], utilisé par Laarouchi *et al.* pour assurer sûreté et sécurité dans les communications entre systèmes commerciaux et systèmes avioniques supportant des fonctions plus critiques [445, 42]. Autres exemples, les techniques de programmation défensive ou d'analyse statique de code, largement promues par les référentiels de sûreté tels que l'IEC 61508 [110] sont également bénéfiques en termes de sécurité informatique [524]. Plus largement encore, l'identification des comportements anormaux et la journalisation des événements et activités servent aussi sécurité et sûreté, en détection d'attaque et en anticipation de pannes, mais également dans les analyses *a posteriori*.

1.2.3 Exemples et considérations sur l'antagonisme

La Section 1.1 a déjà fourni un exemple type de cette interaction moins connue et plus insidieuse que les deux premières. Dans un même genre d'exemple générique, on peut également mentionner la problématique des évacuations de personnes : celles-ci doivent être faites le plus rapidement et directement possible d'un point de vue sûreté, alors que des considérations de sécurité peuvent les ralentir dans certains cas (*e.g.* installations nucléaires) [28, 142]. Plus trivialement, le nombre de portes d'évacuation d'un bâtiment sera réduit à un minimum dans une optique sécurité, facilitant le contrôle d'accès ; il sera plus grand s'il est défini uniquement d'un point de vue sûreté. Autre illustration, Sun *et al.* rapportent une anecdote caractérisant également bien de telles situations contradictoires [34]. Nous en donnons ici une traduction approximative :

> « Un constructeur européen de voiture de luxe avait remarqué que l'un des ses modèles était bien plus volé que ses autres modèles. Ce mystère fut résolu quand un voleur révéla que le modèle en question pouvait être ouvert facilement, même quand les portes étaient verrouillées. Il suffisait de sauter sur le toit. En fait, une fonction de sûreté déverrouillait la voiture en cas d'accident et de tonneau. Pour détecter cette dernière situation, la voiture surveillait la pression/le poids appliqué sur le toit ».

Jalouneix *et al.* mentionnent un exemple très différent : dans les installations nucléaires, les quantités de matières radioactives doivent être mesurées de façon très précise dans une optique sécurité et de lutte contre le risque de vol ou de détournement ; les principes de sûreté amènent quant à eux à prendre des marges de conservatisme nécessaires vis-à-vis des risques de réaction critique [142].

Pour des cas relevant du domaine informatique, il est possible de transposer le dilemme des accès physiques exposé en Section 1.1 aux architectures réseaux : des exigences liées à la disponibilité des systèmes informatiques en charge de la sûreté ou au diagnostic de situations à risque peuvent amener à multiplier les échanges et connexions réseaux, rendant plus difficile la sécurisation de ces dispositifs. Autre exemple, des exigences de contrôle d'accès logique pour l'opérateur sur des systèmes en salle de commande d'une installation industrielle peuvent rentrer en conflit avec les besoins de réactivité et de gestion de crise propres à la sûreté [11].

1.3 Des points de vue multiples

Les catégories précédemment discutées peuvent être considérées dans une perspective de conception, mais également d'un point de vue opérationnel pour des systèmes existants. Elles restent aussi pertinentes du point de vue d'un attaquant. Par exemple, un attaquant peut opportunément mettre à profit des incidents de sûreté dans la réalisation de son objectif : dans certains cas, une défaillance accidentelle contribuera directement à sa progression, facilitant sa tâche ou amoindrira les risques d'être détecté. Un attaquant peut également analyser les scénarios et démonstrations de sûreté quand elles sont disponibles pour préparer ses offensives, non seulement pour les raisons évoquées précédemment, mais aussi pour éventuellement augmenter les conséquences de son attaque. Dans sa thèse [525], Igure s'appuie sur les analyses de protocoles de communication industriels effectuées à des fins de sûreté pour identifier des vulnérabilités exploitables par un attaquant.

Au-delà des exemples et des descriptions génériques, le défi consiste à identifier et caractériser de telles interdépendances au plus tôt, durant les phases de conception et de spécification, dans le but de gérer leurs conséquences et d'optimiser les ressources et performances du système considéré. Ce défi est encore largement à relever. L'état de l'art en la matière est décrit dans la section suivante.

2 État de l'art

2.1 Contributions générales d'ordre organisationnel

Une première catégorie de contributions traite d'aspects organisationnels et de processus génériques liés à la spécification, à la conception et au développement de systèmes.

En 1999, Eames et Moffet proposent d'adopter une méthodologie intégrée d'analyse de risques et de spécification dans [26]. Leur démarche harmonisée s'appuie sur une comparaison critique des analyses de risques types menées habituellement de façon distincte en sûreté et en sécurité, et s'articule en quatre

phases : modélisation du système, analyse qualitative, analyse quantitative et définition des exigences. Elle est brièvement illustrée par un cas d'étude de système militaire de contrôle aérien.

En 2005, Lautieri *et al.* donnent dans [27, 526, 527] un aperçu de *SafSec*, une méthodologie élaborée sur commandite du ministère de la Défense britannique en vue d'optimiser les efforts de développement associés aux exigences de certification de sûreté et de sécurité pour les systèmes avioniques. Développée et depuis mise en œuvre industriellement par la société Praxis-HIS (devenue Altran-Praxis) [1], elle est structurée autour de 18 objectifs dont l'atteinte permet d'obtenir des éléments pertinents à la fois pour des certifications de sûreté (*e.g.* DO-178B) et de sécurité (*e.g.* Critères Communs). Elle s'appuie en outre sur les notions de *safety cases* et le formalisme GSN évoqués en II-3.1.3.

En 2006, l'EFCOG [2], sous l'égide du département de l'Énergie états-unien (DoE), décrit une approche intégrée de type projet conciliant sécurité et sûreté dans la gestion de l'arsenal nucléaire états-unien [28]. Le document s'intéresse plus particulièrement aux systèmes de sécurité devant à la fois répondre aux exigences de la DBT (*Design Basis Threat*, cf. II-1.1.1) et à des contraintes de sûreté. La démarche proposée doit permettre en plus d'établir une « boîte à outils » d'éléments réutilisables sur les différents sites du DoE. La nécessité de coordination entre les équipes, y compris en termes de concepts et de terminologie, est soulignée. Une annexe complète met en correspondance les principaux éléments de vocabulaire employés par les acteurs sûreté et sécurité du DoE. Plus largement, les éventuelles contradictions entre sûreté et sécurité doivent pouvoir être résolues par la démarche de sélection d'alternatives techniques également spécifiée dans le document.

Novak *et al.* développent en deux temps une approche visant aussi à tenir compte des interactions entre sûreté et sécurité : en 2007, ils s'intéressent aux phases amonts de spécification et de conception [450] ; en 2008, ils développent un modèle de cycle de vie complet [451, 29]. Dans les deux cas, le contexte thématique est celui des automatismes du bâtiment (BACS, pour *Building Automation Control Systems*) mais leurs réflexions sont généralisables. Leur démarche s'inspire à la fois du modèle de développement porté par la norme IEC 61508 et sur la démarche mise en œuvre par les Critères Communs. Hélas, la méthode d'identification de conflits potentiels et de résolution reste très imprécise : les exigences sûreté ou sécurité sont privilégiées sur la base des enjeux qui leur sont associés mais sans préciser les modalités d'un tel arbitrage. Le contexte des BACS amène ainsi les auteurs à systématiquement privilégier les exigences de sûreté sur celles de sécurité [29]. Des exemples de développement de protection cryptographique sont abordés dans ce contexte, montrant comment leurs spécifications prennent en compte sûreté et sécurité.

En 2009, l'Autorité de sûreté états-unienne a publié un guide de mise en œuvre du code réglementaire fédéral sur l'interface entre sûreté et sécurité pour les centrales nucléaires [30]. Les interactions entre sûreté et sécurité y sont reconnues, couvrant de façon implicite les quatre types identifiés en Section 1.2. Pour les identifier et les gérer globalement, le guide présente en quelques pages plusieurs recommandations d'ordre organisationnel et procédural. En outre, une formalisation adaptée des échanges d'information et une coordination spécifique des acteurs permettent de décloisonner les deux problématiques. Une liste de questions types est fournie pour aider dans l'évaluation de la mise en œuvre de la démarche préconisée ; les aspects audit et formation sont également traités.

Dans l'ensemble, les contributions évoquées dans cette sous-section restent très génériques. En particulier, elles ne précisent pas les méthodes de modélisation, d'analyse qualitative et d'analyse quantitative à employer pour caractériser les interdépendances qu'elles mentionnent.

2.2 Contributions ciblées

Un second type de contributions regroupe des travaux plus ciblés, ne couvrant qu'un sous ensemble de la question des interdépendances entre sûreté et sécurité. Par exemple, l'effet des approches de diversification en sûreté et en sécurité a été étudié par Littlewood et Stringini dans [31], et plus récemment par Gerstinger et Novak [528] ou Komari *et al.* [529]. Encore plus ciblés, les travaux de Cho *et al.* analysent l'incidence des systèmes de détection d'intrusions sur la fiabilité de communications de groupe orientées par mission (*mission-oriented group communications*) [530]. Ceux de Wu et Chen [531] s'intéressent au nombre de clés d'un schéma de cryptographie à seuil permettant d'optimiser niveau de protection du schéma et niveau de fiabilité du système. Dans [33], Levitin et Hausken proposent un modèle visant à choisir les meilleures stratégies d'allocation de ressources pour protéger des systèmes redondants en pa-

1. Des documents beaucoup plus complets que les articles cités sont d'ailleurs disponibles sur le site d'Altran-Praxis (`http://www.praxis-his.com/safSecStatus.aspx`).
2. *Energy Facility COntractors Group* (EFCOG, `http://www.efcog.org`).

rallèle face à différents types de risque. Ils montrent que dans les configurations considérées, elles varient considérablement selon que le risque est accidentel, malveillant ou mixte.

Globalement, les travaux mentionnés dans cette sous-section ne couvrent que des aspects très spécifiques de la problématique des interactions sûreté-sécurité. On peut également s'étonner de leur nombre relativement limité compte-tenu de la diversité des aspects à traiter et des cas d'application envisageables.

2.3 Utilisation de méthodes « formelles » et *model-checking*

Une troisième catégorie de contributions correspond aux approches formelles permettant l'utilisation de techniques de *model-checking*. En 2005, Zafar et Dromey [32] mettent à profit pour cela des techniques issues de l'ingénierie génétique logicielle (*genetic software engineering*) [532] : les exigences sûreté et sécurité de haut niveau sont formalisées graphiquement par des arbres comportementaux, ils peuvent ensuite être traduits en CSP [3] ou dans l'atelier logiciel d'analyse statique SAL [4] pour faire l'objet de traitement par *model-checking*. Le cas d'étude correspond au système de contrôle d'un dispositif médical de perfusion mobile, pour lequel le déroulement de la méthode proposée permet de vérifier le respect des deux types d'exigences. Sun *et al.* s'appuient sur le langage de réécriture logique *Maude* [533] pour modéliser le système de contrôle de porte automatisée évoqué en Section 1.1 et ses contraintes de sûreté et de sécurité [34]. Les éventuelles contradictions peuvent ensuite être identifiées par *model-checking*.

Malheureusement, ces approches ne semblent envisageables que pour des systèmes très simples ou à un niveau d'abstraction élevé, étant donné les efforts de formalisation nécessaires. Elles ne s'intéressent de plus qu'aux éventuels conflits entre exigences, mais ne permettent pas dans leur forme actuelle d'identifier les autres types d'interdépendance comme le renforcement ou la dépendance conditionnelle.

2.4 Utilisation de formalismes de modélisation graphique

À notre connaissance, il n'y a qu'une seule publication décrivant une formalisation graphique caractérisant explicitement des interactions entre sûreté et sécurité (en l'occurrence de type dépendance). En effet, Fovino *et al.* décrivent dans [534] comment arbres de défaillances, modélisant des scénarios accidentels, et arbres d'attaque, modélisant des scénarios malveillants, peuvent être intégrés dans une même structure, dénommée arbre de défaillances étendu (*extended fault tree*). La modélisation par BDMP que nous décrivons dans la prochaine section peut être vue commune une alternative permettant de dépasser bon nombre de limites de l'approche de Fovino *et al.* Nous développons cette comparaison en Section 4.

3 Les BDMP comme formalisme intégrateur

Nous avons présenté les BDMP dans le Chapitre III, et exposé comment ils pouvaient être avantageusement adaptés et employés dans le domaine de la sécurité. Dans cette section, nous mettons à profit cette adaptation pour faire des BDMP un formalisme intégrateur permettant de représenter et traiter des aspects de sûreté et de sécurité dans un même modèle graphique. Pour cela, nous nous appuyons sur les feuilles BDMP telles que classiquement employées dans les études de sûreté de fonctionnement, augmentées des feuilles définies pour la modélisation sécurité dans le Chapitre III. Les premières sont utilisées pour modéliser des événements accidentels associés à la dimension sûreté, tandis que les secondes caractérisent des actions malveillantes, relevant donc d'une problématique de sécurité.

La Section 3.1 décrit une hypothèse théorique et pratique nécessaire à une telle intégration, elle rappelle ensuite sommairement la sémantique des feuilles du formalisme d'origine. La Section 3.2 présente le cas d'étude choisi pour illustrer l'intérêt d'une telle intégration. La Section 3.3 aborde dans un premier temps la modélisation BDMP du cas d'étude d'un point de vue sûreté uniquement, puis sous l'angle sécurité. Elle montre ensuite comment les BDMP permettent de modéliser les interactions entre sûreté et sécurité, intégrant les situations considérées de façon isolée en début de section. La caractérisation quantitative de ces interactions est enfin abordée en Section 3.4.

3. *Communicating Sequential Processes (CSP)* [441].
4. *Symbolic Analysis Lab* (SAL, `http://sal.csl.sri.com`).

3.1 Considérations préliminaires

3.1.1 Hypothèse nécessaire

Le formalisme original des BDMP associe à chaque élément, porte ou feuille, trois fonctions booléennes du temps : S_i, fonction de structure indiquant l'état de réalisation de l'élément, X_i, sélecteur de mode indiquant dans quel mode l'élément doit être considéré et Y_i, indicateur de pertinence utilisé pour le mécanisme de filtrage des événements pertinents lors du traitement quantitatif du modèle. Ces fonctions sont définies précisément dans le papier de référence des BDMP [238] et dans le Chapitre III de ce mémoire, étant aussi impliquées dans l'adaptation des BDMP à la sécurité. Du point de vue théorique, la principale différence entre les BDMP « classiques » et les BDMP sécurité correspond à l'introduction d'un booléen supplémentaire pour ces derniers : l'indicateur de statut de détection D_i (cf. Chap. III Section 3.2). Dans la pratique, notamment dans la mise en œuvre logicielle présentée dans le Chapitre III, il suffit de n'instancier les règles *Figaro* associées à D_i que dans les feuilles sécurité et les portes du modèle. Sa dynamique globale reste alors définie par les fonctions S_i, X_i, Y_i, alors que la dynamique spécifique de détection est circonscrite aux seules feuilles concernées.

3.1.2 Rappels sur les feuilles : sémantique, paramètres et représentations

Les descriptions des feuilles modélisant les aspects sécurité sont données au Chapitre III Section 3, nous invitons le lecteur à s'y référer. Les feuilles correspondant aux aspects accidentels sont celles décrites dans [238], nous indiquons la représentation des deux les plus employées par la suite dans la Table IV.1 et en rappelons sommairement leurs caractéristiques.

TABLE IV.1 – Deux feuilles des BDMP classiques pour modéliser les événements accidentels

Type de feuille et représentation	**Comportement modélisé**
 Défaillance accidentelle en opération	Cette feuille modélise une défaillance accidentelle en opération, qui ne peut survenir que quand le composant est en mode « sollicité ». Ce mode, équivalent du mode Actif des feuilles sécurité, signifie que le composant est en fonctionnement « utile » pour le système (une redondance à chaud requiert un autre type de feuille). La défaillance arrive après un temps distribué exponentiellement selon une loi de paramètre λ, et peut également être réparée en un temps exponentiel (paramètre μ).
 Défaillance accidentelle à la sollicitation	Cette feuille modélise une défaillance accidentelle à la sollicitation (*on-demand failure*), qui peut arriver instantanément lorsque la feuille change de mode, avec une probabilité γ. La défaillance peut être réparée en un temps distribué exponentiellement selon une loi de paramètre μ.

3.2 Cas d'étude

Nous prenons pour cas d'étude un système très proche de celui modélisé par Fovino *et al.* dans [534] : il est constitué d'une canalisation que l'on suppose transporter une substance polluante (*e.g.* un produit chimique pour suivre l'exemple de Fovino), surveillée par un système instrumenté de sûreté (SiS) en charge d'arrêter le flux en cas de problème (fuite, surpression, etc.). L'événement redouté est la pollution de l'environnement. Au niveau d'abstraction choisi, le système n'a donc que deux composants, la canalisation et le SiS, mais suffit à illustrer la richesse des interactions entre sûreté et sécurité et montrer en quoi les BDMP peuvent fournir un outil d'intérêt pour mieux les caractériser.

3.3 Modélisation des interactions entre sûreté et sécurité par les BDMP

Dans cette section, nous allons examiner le cas d'étude en BDMP sous une double perspective, prenant en compte les défaillances de cause accidentelle et la malveillance. Le sommet du BDMP reste dans tous les modèles la pollution, à la fois événement redouté d'un point de vue sûreté et objectif de l'attaquant. Une telle approche, englobant risque sûreté et risque sécurité, peut être abordée de façon incrémentale. À cette fin, nous distinguons trois perspectives et types de modèles associés : les modèles « purs », les modèles « hybrides » et les modèles « intégrés ». Ces modèles, bien que progressivement modifiés et enrichis, restent dans l'ensemble à un niveau très macroscopique. La nature hiérarchique des BDMP permettrait de développer les feuilles et de restituer explicitement la diversité des alternatives offertes à l'attaquant et/ou celle des séquences accidentelles. Nous avons cependant préféré nous en abstenir à des fins explicatives et d'appropriation du formalisme.

3.3.1 Modèles purs

Nous désignons par modèles purs les modèles ne prenant en compte que des événements accidentels ou uniquement des actions malveillantes. La Figure IV.1 représente trois modèles BDMP purs orientés sûreté, utilisant par conséquent les feuilles de la Table IV.1. Ils modélisent de trois façons différentes le comportement dynamique du système d'un point de vue accidentel et les défaillances associées. Un arbre de défaillances standard n'aurait pu faire une telle distinction. Dans la partie a) de la figure, le SiS a une probabilité γ de défaillir au moment où il est sollicité suite à une rupture de la canalisation. Dans la partie b), le SiS peut cesser d'être opérationnel avant une telle rupture (pour cause de défaillance « silencieuse » au repos, maintenance, etc.) ; la porte *Priority AND* (PAND) impose un tel ordre pour que la pollution ait lieu. Dans la partie c), les deux comportements sont modélisés [5].

Le BDMP de la Figure IV.2 modélise le même système que précédemment, avec le même événement redouté, mais dans une perspective malveillante. En d'autres termes, la défaillance du SiS et la rupture de la canalisation menant à la pollution sont maintenant considérées comme provoquées volontairement. Il est remarquable que ce seul changement de perspective a une incidence concrète en termes de modélisation : la seule façon logique de procéder pour un attaquant intelligent est de désactiver en premier lieu le SiS avant d'attaquer la canalisation, alors que la perspective accidentelle nous amenait à considérer plusieurs types de séquences comme illustré par la Figure IV.1. Ceci souligne la pertinence et l'importance de la capacité à modéliser de tels aspects : les modèles statiques comme les arbres de défaillances et les arbres d'attaque n'en sont, comme déjà signalé, pas capables.

3.3.2 Modèles hybrides

Les modèles hybrides correspondent à des modèles combinant événements de base de type accidentel et de type malveillant. Des exemples représentatifs sont donnés par la Figure IV.3.

Le BDMP de la Figure IV.3a) modélise un comportement opportuniste de l'attaquant, évoqué de façon générique en Section 1.2.1 : l'attaquant attend que le SiS soit hors service pour attaquer la canalisation. Plusieurs hypothèses sont associées à ce modèle : tout d'abord, la canalisation doit fonctionner malgré l'arrêt du SiS, ce qui peut ne pas être réaliste selon le niveau de sûreté et de procédure en place ; deuxièmement, la défaillance du SiS doit durer au moins le temps nécessaire à l'attaquant pour réussir son attaque sur la canalisation ; finalement, l'attaquant doit être capable de repérer une telle défaillance, ce qui peut impliquer des collaborations internes ou de tierces parties associées aux activités de maintenance.

Le BDMP de la Figure IV.3b) modélise un scénario hybride dans lequel l'attaquant désactive le SiS, mais choisit de laisser une défaillance accidentelle achever la séquence menant à son objectif final, la pollution. Ceci peut être fait pour réduire les risques de détection, et dépend également de la durée de la défaillance du SiS. Notons que comme dans la Figure IV.1, une porte PAND modélise le fait que la pollution ne peut avoir lieu que si le SiS est désactivé avant la rupture de canalisation.

3.3.3 Modèles intégrés

Finalement, il est possible de combiner les modèles purs et les modèles hybrides dans des modèles que nous qualifions d'intégrés, comme illustré par la Figure IV.4. Dans la partie a) de la figure, le BDMP

5. Dans cette dernière partie, l'emploi de deux feuilles distinctes pour modéliser la défaillance d'un même composant (SiS) pourrait surprendre le lecteur. La construction du modèle et le mécanisme de filtrage des événements pertinents des BDMP assurent qu'elles ne peuvent être considérées simultanément.

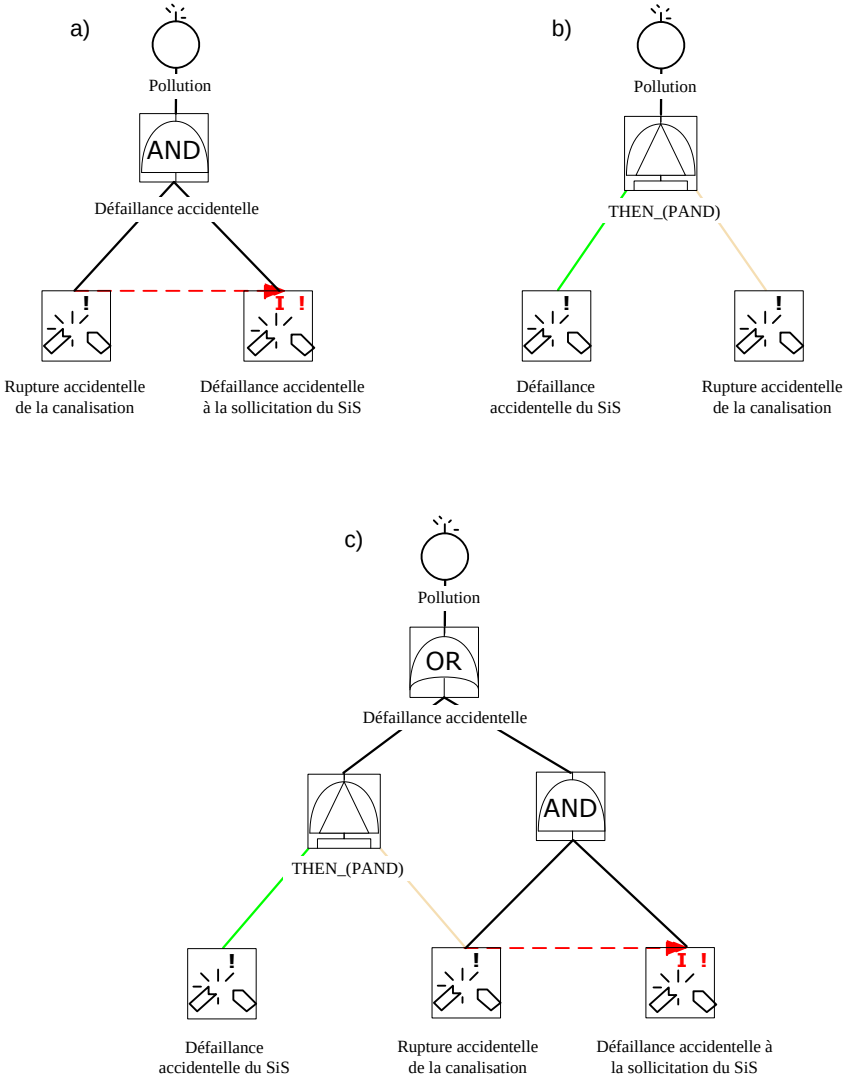


FIGURE IV.1 – Défaillances accidentelles menant à la pollution

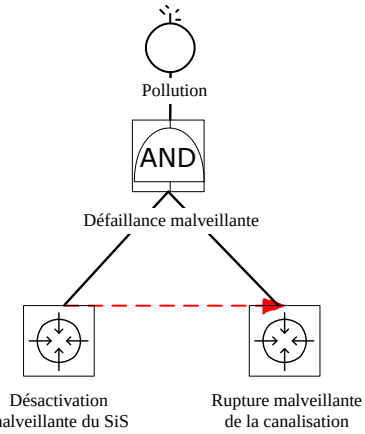


FIGURE IV.2 – Défaillance purement malveillante

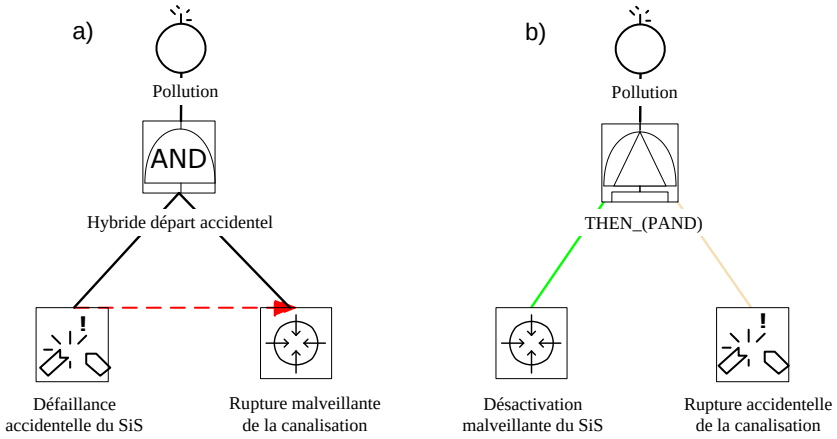


FIGURE IV.3 – Exemples de modèles hybrides

permet de distinguer deux valeurs pour la probabilité de défaillance à la sollicitation du SiS, selon qu'il est attaqué ou pas ; une telle modélisation est plus fine que les précédentes où la compromission du SiS menait directement à une défaillance certaine. La partie b) de la Figure IV.4 ne prend pas en compte cette distinction, mais couvre un spectre plus large de scénarios, incluant notamment le scénario opportuniste de la Figure IV.3a) et aussi bien les défaillances à la sollicitation qu'en opération du SiS. De plus, on peut souligner que l'origine et la cible de la gâchette dans la partie gauche de la Figure IV.4b) peuvent être changées, reflétant alors différentes configurations. Par exemple, une gâchette partant de la feuille Défaillance à la sollicitation du SiS (quand attaqué) au lieu de la porte OU Défaillance du SiS exclurait le scénario opportuniste de la Section 3.3.2. Dans le cas présent, l'attaquant tente de corrompre le SiS mais pourrait également remarquer et tirer bénéfice d'une défaillance accidentelle.

3.4 Caractérisation et comparaison de scénarios par analyse quantitative

3.4.1 Quantifications et choix du référentiel temporel

Les capacités de quantification des BDMP ont été présentées dans le Chapitre III Section 2.4. Quantifications temporelles et non-temporelles nourrissent les analyses d'un point de vue quantitatif comme qualitatif. Ces capacités peuvent être exploitées pour caractériser et comparer les modèles purs, hybrides et intégrés. Les comparaisons sont utiles entre alternatives appartenant à la même catégorie, en changeant la valeur des paramètres et les configurations, mais aussi entre catégories. Les dimensions spécifiques à chaque domaine, telles que les notions de maintenance et de réparation pour les BDMP classiques ou celles de détection/réaction pour les BDMP sécurité, peuvent être prises en compte.

Pour le traitement des modèles hybrides ou intégrés, différents référentiels de temps sont envisageables. La différence de nature entre événements accidentels et actions malveillantes peut en effet être prise en compte sur le plan de la modélisation stochastique. Plus concrètement, les modèles peuvent prendre comme référentiel temporel le début de l'attaque (c'est le cas des exemples précédents et de la Fig. IV.5a)), mais il est aussi possible de modéliser une durée d'attente précédant le début de l'attaque, par rapport à une origine des temps arbitraire. Ceci peut être fait, comme l'illustre la Figure IV.5b), par l'emploi d'une feuille supplémentaire d'occurrence d'attaque et d'une gâchette. Le temps d'arrivée de l'attaque est alors modélisé par une loi exponentielle de paramètre λ, qui peut être déterminé sur les bases d'une estimation de la fréquence d'occurrence de ce type d'attaque. Une telle approche est plus homogène avec la modélisation stochastique des événements accidentels, reposant sur le même genre de considérations.

3.4.2 Application au cas d'étude

Nous illustrons ici l'intérêt des quantifications temporelles pour la caractérisation des interactions entre sûreté et sécurité sur la base du cas d'étude précédemment développé. Dans cette perspective, les feuilles des modèles associés doivent être paramétrées selon leurs spécifications, décrites respectivement dans le Chapitre III pour les feuilles orientées sécurité, et dans [238] pour les feuilles orientées sûreté (la

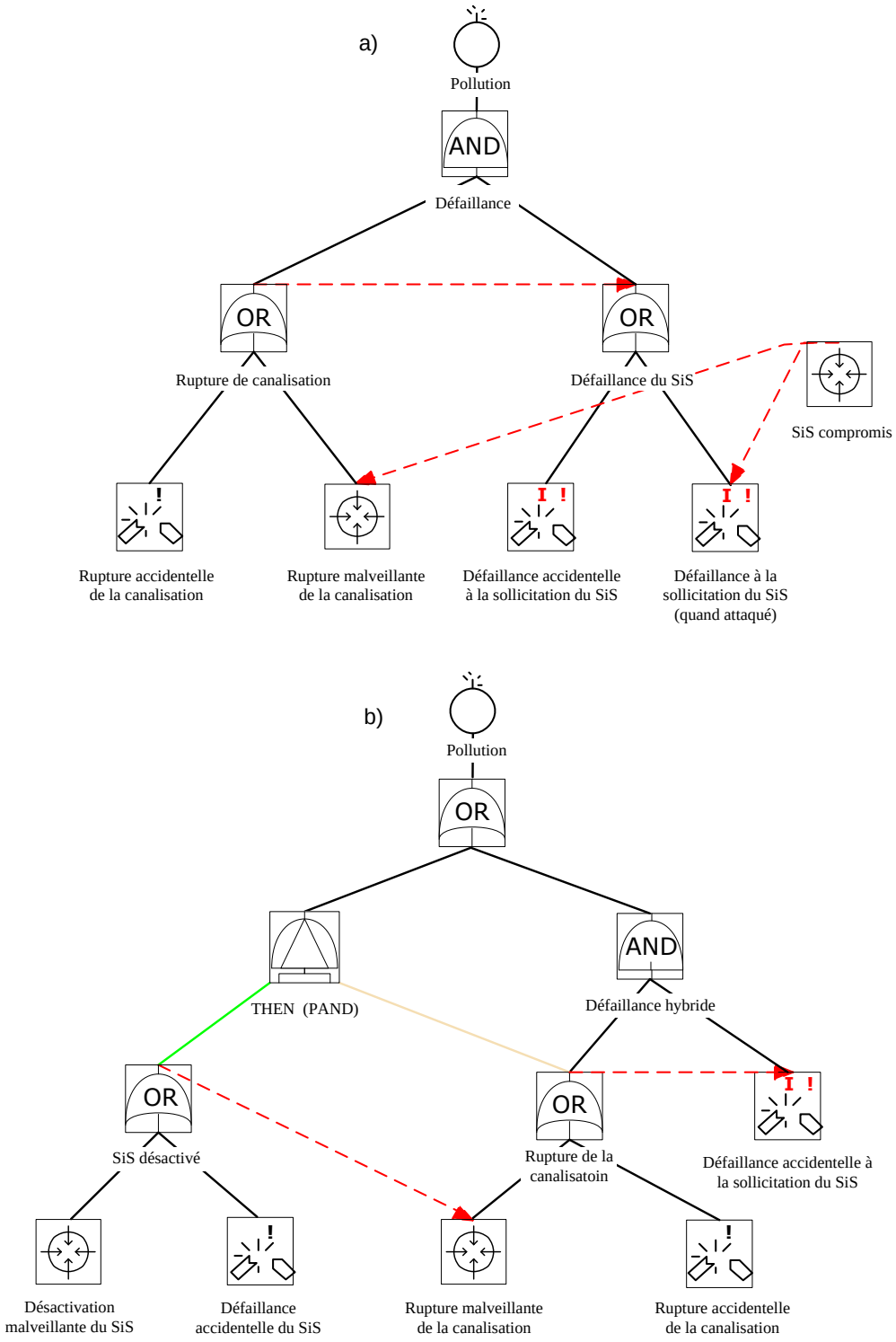


FIGURE IV.4 – Modèles intégrés

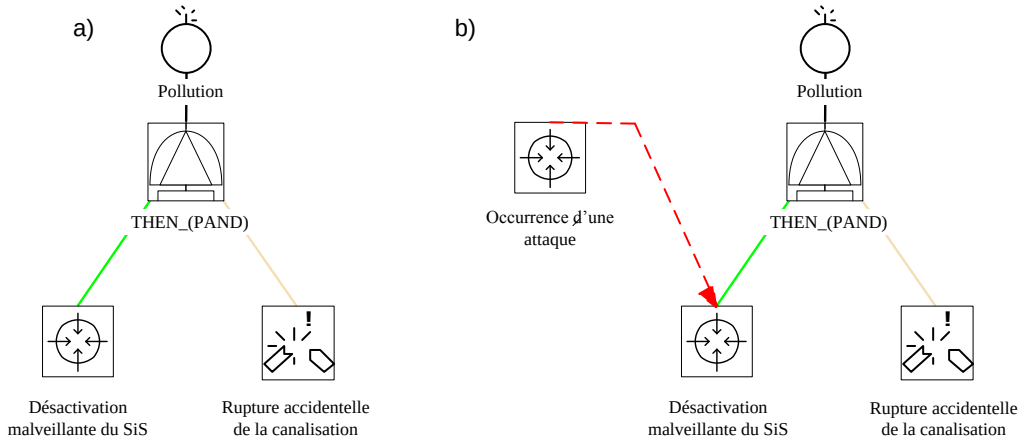


FIGURE IV.5 – Choix du référentiel temporel

Table IV.1 a synthétisé la description des feuilles classiques ici employées). Un tel paramétrage consiste dans notre cas à définir principalement les valeurs de taux de défaillance accidentelle en opération et à la sollicitation, et les taux de réussite de l'attaquant. De nombreux autres paramètres sont décrits dans le Chapitre III et dans [238], permettant une modélisation fine des réparations, détections et réactions. En cohérence avec le relative simplicité de notre cas d'étude, nous ne considérerons ici que les possibilités de détection de type I et de type E : l'attaquant peut être détecté à chaque début d'action modélisée avec des probabilités $\gamma_{D(I)}$ différenciées par feuille, il peut aussi être détecté pendant le déroulement de ses actions, avec des taux de détection $\lambda_{D(E)}$, également différenciés selon les feuilles. La Table IV.2 précise l'ensemble des paramètres choisis pour caractériser le modèle purement malveillant, les modèles hybrides et un des modèles intégrés précédemment exposés, correspondant respectivement aux Figures IV.2, IV.3a), IV.3b) et IV.4b). Les valeurs sont ici choisies arbitrairement, mais pour un cas d'étude réel, elles devraient être définies selon les approches classiques de la sûreté pour les feuilles modélisant des événements accidentels, et selon des avis d'expert pour les feuilles modélisant les actions de l'attaquant (cf. Chap. III Sections 2.4.3 et 6.1 pour une discussion sur ces aspects).

TABLE IV.2 – Paramétrage des modèles

Feuille	**Figures**	**Paramètres (unité de temps = heure)**
Désactivation malveillante du SiS	IV.2, IV.3b), IV.4b)	Si non détecté : $\lambda_{S/ND} = 4,166 \times 10^{-2}$ (MTTS $\approx$ 1 jour) ; Paramètres de détection : $\gamma_{D(I)} = 0,5$; $\lambda_{D(E)} = 5,952 \times 10^{-3}$ (MTTD $\approx$ 1 semaine) ; Une fois détecté : $\lambda_{S/D} = 1,377 \times 10^{-3}$ (MTTS $\approx$ 1 mois)
Défaillance accidentelle du SiS (en fonctionnement)	IV.3a), IV.4b)	$\lambda = 5,741 \times 10^{-5}$ (MTTF $\approx$ 2 ans)
Rupture malveillante de la canalisation	IV.2, IV.3a), IV.4b)	Si non détecté : $\lambda_{S/ND} = 5,952 \times 10^{-3}$ (MTTD $\approx$ 1 semaine) ; Paramètres de détection : $\gamma_{D(I)} = 0,1$; $\lambda_{D(E)} = 5,952 \times 10^{-3}$ (MTTD $\approx$ 1 semaine) ; Une fois détecté : $\lambda_{S/D} = 0$ (stop)
Rupture accidentelle de la canalisation	IV.3b), IV.4b)	$\lambda = 1,148 \times 10^{-4}$ (MTTF $\approx$ 1 an)
Défaillance accidentelle à la sollicitation du SiS	IV.4b)	$\gamma = 10^{-4}$

Avec ces paramètres, nous pouvons calculer la probabilité de pollution pour chaque modèle selon différents temps de mission. Le choix du temps de mission dépend du cadre de la situation à modéliser : dans notre cas, la substance polluante circule par exemple uniquement durant une période bien définie, ou l'attaquant a une contrainte de temps à respecter. La Figure IV.6 représente la probabilité d'occurrence de la pollution pour le modèle purement malveillant (Fig. IV.2) et les deux modèles hybrides (Fig. IV.3), selon trois temps de mission distincts. Elle montre que cette durée a pour ces cas une inci-

dence significative, notamment vis-à-vis de la probabilité de pollution dans les modèles hybrides. Dans la perspective d'un attaquant visant la pollution, il est plus efficace de se comporter selon le modèle purement malveillant si le temps de mission est court (cf. série pour le temps de mission d'une semaine), alors qu'une approche hybride peut être plus intéressante quand l'attaquant n'a pas de contrainte de temps. En effet, pour la série correspondant à un temps de mission d'une année, la stratégie optimisant les chances de réussite correspond au modèle hybride avec un départ malveillant : le SiS est désactivé de façon malveillante, mais la rupture de canalisation menant à la pollution arrive de façon accidentelle. Cette situation peut s'expliquer par le choix des paramètres de détection et de réaction : dans notre cas, la détection de l'attaque de la canalisation annule toute chance de succès, reflétant par exemple un durcissement radical de la défense. La détection de l'attaque du SiS amène seulement une augmentation de la difficulté. Ceci peut se justifier si l'on considère par exemple un premier type d'attaque du SiS de nature distante et informatique, vecteur préféré par l'attaquant mais remplacé dans un second temps par une attaque physique, plus risquée pour l'attaquant mais rendue nécessaire par une nouvelle protection logique du SiS (voire son isolement) provoquée par la détection du premier type d'attaque.

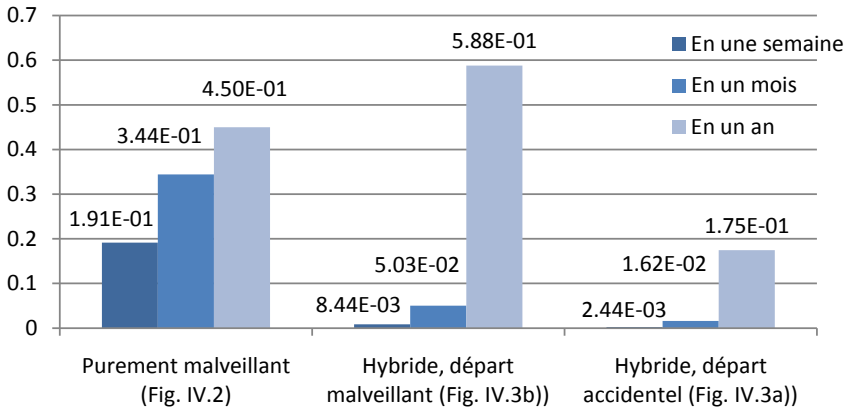


FIGURE IV.6 – Probabilité de l'événement redouté pour différents modèles et temps de mission

À côté de ce genre de résultats globaux, les BDMP permettent aussi l'identification et la quantification de toutes les séquences menant à l'événement redouté, ordonnées selon leur probabilité d'occurrence durant le temps de mission. Les modèles purs et hybrides de notre cas d'étude sont trop simples pour que de tels résultats présentent un intérêt, mais la liste des séquences procure des informations qualitatives et quantitatives pertinentes pour l'analyse de cas plus complexes. Bien que de taille encore modeste, le modèle intégré de la Figure IV.4b) convient pour l'illustrer. Son traitement mène à l'identification de 27 séquences menant à la pollution (en prenant en compte les événements de détection). Pour un temps de mission d'un an, avec les paramètres de la Table IV.2, la pollution a une probabilité d'occurrence de $0,75$. Cette valeur très élevée est liée en partie à la dimension globale de l'analyse, intégrant malveillance et défaillances accidentelles, mais également au référentiel de temps adopté, basé de façon déterministe sur le départ certain d'une attaque. Comme discuté en Section 3.4.1, les notions de fréquence d'attaque et de temps moyen avant la première attaque auraient également pu être modélisées : la probabilité d'occurrence de la pollution aurait alors été plus faible [6]. La Table IV.3 donne un extrait représentatif de la liste complète des séquences, ordonnées par contribution. Les deux premières sont purement malveillantes et représentent environ 52 % de la probabilité globale de pollution. Les deux séquences suivantes sont de nature hybride, avec un départ malveillant et contribuent pour 32 % aux chances de pollution. Il faut ensuite attendre la séquence n° 9 pour trouver une séquence hybride avec un départ accidentel, suivie par la première séquence purement accidentelle (n° 10). Assez naturellement, les séquences purement accidentelles ont un poids très limité dans la probabilité de pollution, écrasées par les séquences purement malveillantes et les séquences hybrides.

6. Rappelons aussi que les paramètres ont été choisis arbitrairement et ne reflètent pas une situation réelle.

TABLE IV.3 – Sélection de séquences et quantifications associées pour le modèle intégré de la Fig. IV.4b)

N°	Séquence	Probabilité (1 semaine)	Durée moyenne (h)	Contrib.
1	Détection initiale attaque du SiS, Succès attaque du SiS (détecté), Rupture malveillante de la canalisation (non détecté)	$1,981 \times 10^{-1}$	$7,23 \times 10^{2}$	26,2 %
2	Succès attaque du SiS (non détecté), Rupture malveillante de la canalisation (non détecté)	$1,943 \times 10^{-1}$	$1,04 \times 10^{2}$	25,7 %
3	Succès attaque du SiS (non détecté), Détection en cours de tentative de rupture malveillante de la canalisation, Rupture accidentelle de la canalisation	$1,222 \times 10^{-1}$	$8,80 \times 10^{3}$	16,2 %
4	Détection initiale attaque du SiS, Succès attaque du SiS (détecté), Détection en cours de tentative de rupture malveillante de la canalisation, Rupture accidentelle de la canalisation	$1,187 \times 10^{-1}$	$9,42 \times 10^{3}$	15,7 %
...				
9	Détection initiale attaque du SiS, Défaillance accidentelle du SiS, Rupture malveillante de la canalisation (non détecté)	$8,20 \times 10^{-3}$	$7,29 \times 10^{2}$	1,1 %
10	Détection initiale attaque du SiS, Défaillance accidentelle du SiS, Détection en cours de tentative de rupture malveillante de la canalisation, Rupture accidentelle de la canalisation	$4,92 \times 10^{-3}$	$9,42 \times 10^{3}$	0,6 %
...				
14	Détection initiale de l'attaque du SiS, Défaillance accidentelle du SiS, Détection initiale de tentative de rupture malveillante de la canalisation, Rupture accidentelle de la canalisation	$1,11 \times 10^{-3}$	$9,34 \times 10^{3}$	0,1 %
...				

4 Positionnement de la contribution

Comme nous l'avons mentionné en Section 2.4, l'état de l'art en termes de modèles graphiques capables de prendre en compte les interdépendances entre sûreté et sécurité est très restreint. Nous n'avons en fait identifié que l'approche proposée par Fovino *et al.* [534]. Elle décrit comment fusionner arbres de défaillances et arbres d'attaque dans une structure nommée arbre de défaillances étendu (*extended fault tree*), intégrant événements accidentels et actes malveillants dans un arbre logique menant à un événement redouté. La fusion s'articule autour d'événements de l'arbre de défaillances pouvant être provoqués de façon malveillante : des arbres d'attaque dont les sommets correspondent à ces événements sont rattachés à l'arbre de défaillances principal. Intégrant feuilles sûreté et feuilles sécurité dans un même BDMP, notre approche peut sembler en première analyse très similaire ; elle possède en fait plusieurs avantages significatifs.

Tout d'abord, l'utilisation des BDMP permet de prendre en compte la dimension dynamique du déroulement des scénarios tout en restant visuellement proche des arbres d'attaque et de défaillance. Nous avons pu souligner à diverses reprises à quel point une telle prise en compte importait dans la modélisation d'actions malveillantes, mais également pour rendre compte de la diversité des interactions entre sûreté et sécurité. Le simple cas d'étude décliné en Section 3.3 en donne une bonne illustration. La modélisation de ces aspects dynamiques n'est pas possible dans l'approche de Fovino *et al.* Ceux-ci mentionnent l'utilisation de DFT en perspective : nous renvoyons alors le lecteur vers le Chapitre III Section 4.4 pour une comparaison entre DFT et BDMP.

De plus, notre approche permet diverses quantifications temporelles d'intérêt, en plus des quantifications classiquement offertes par les arbres statiques. L'approche de Fovino *et al.* restreint implicitement les possibilités à ce dernier type. Aucun exemple de quantification n'est cependant donné dans [534]. Par ailleurs, la différence de nature entre probabilité de défaillance accidentelle et probabilité de réussite d'attaque, évoquée en Section 3.4.1 pour notre cadre dynamique, ne semble pas avoir été considérée. Pour finir, l'intégration entre événements accidentels et actions malveillantes s'appuie sur des portes dites de fusion (*merge gates*), permettant de connecter les objectifs d'arbres d'attaque aux feuilles d'un arbre de défaillance. Présentées dans le corps de l'article comme clés de construction des modèles, elles ne sont en fait pas mises en œuvre dans le cas d'étude proposé. Nous avons pour notre part choisi une approche plus souple, permettant la combinaison de feuilles de BDMP sécurité et de feuilles de BDMP classiques au gré des besoins de modélisation.

Au-delà de la comparaison avec l'approche de Fovino *et al.*, il peut être intéressant d'adopter un point de vue plus large, et d'examiner les autres catégories de travaux s'intéressant aux interactions entre sûreté et sécurité décrits dans la Section 2. Vis-à-vis des propositions faites en termes d'aide générique à la spécification et à la conception de la Section 2.1, notre approche trouverait sa place dans de telles méthodologies comme outil avancé de caractérisation des interactions sûreté-sécurité, éventuellement identifiées au préalable par les méthodologies mentionnées. Les contributions plus ciblées de la Section 2.2 pourraient aussi tirer bénéfice de notre proposition afin de modéliser et comparer différentes configurations pour des architectures données. Finalement, les approches de la Section 2.3 sont complémentaires : contrairement à celles-ci, la modélisation par BDMP ne permet pas d'identifier automatiquement des situations conflictuelles, ni plus largement de couvrir exhaustivement le champ complet des interactions sûreté-sécurité envisageables. Par contre, notre approche permet de modéliser des systèmes plus complexes, grâce à la nature hiérarchique des BDMP, et constitue un bon moyen d'approfondir l'étude de situations problématiques identifiées par des approches automatisées.

5 Vers une approche systématisée

5.1 Un retour au cadre référentiel SEMA

Comme montré en Section 3.3, les événements de sûreté et de sécurité peuvent être combinés de multiples manières dans des scénarios pertinents pour une analyse de risques. Il peut s'avérer difficile de couvrir une telle diversité de situations de façon rigoureuse et complète. De plus, sûreté et sécurité peuvent être considérées dans d'autres perspectives que celles que nous avons ici adoptées. Ces concepts peuvent en effet être définis de différentes façons par rapport aux distinctions Malveillance-Accidentel (M-A) et Système-Environnement (S-E) introduites dans le Chapitre I. Le référentiel SEMA qui y est exposé peut alors s'avérer utile. En effet, en plus d'éviter les ambiguïtés terminologiques typiquement associées aux mots sûreté et sécurité, SEMA peut également être vu comme une décomposition de l'espace des risques de type sûreté et sécurité considérés d'un point de vue holistique. Le cas d'étude choisi dans ce chapitre est trop simple pour tirer un bénéfice clair de la démarche associée à une telle décomposition, mais elle prend de l'intérêt pour des cas plus élaborés ; aussi, nous la décrivons ici rapidement. Dans un premier temps, le système est considéré successivement selon les perspectives correspondant aux six notions du référentiel SEMA. Dans chaque cas, des modèles BDMP sont construits indépendamment, avec l'appui des parties-prenantes et des spécialistes les plus appropriés selon le type de risque considéré. Notons qu'une telle approche est en phase avec l'avis d'Eames et Moffet [26], et dans une certaine mesure celui de Leveson [18], pour qui les idiosyncrasies de la sûreté et de la sécurité, y compris d'un point de vue méthodologique, ne doivent pas être dissoutes dans une approche uniformisante. Après un tel découpage, hybridations et intégrations des modèles BDMP produits (ou de certaines de leurs parties) peuvent être considérées, permettant de couvrir de façon incrémentale les interdépendances entre les sous-notions SEMA. Durant ce processus, l'événement redouté peut varier : l'objectif de la démarche n'est en effet pas d'aboutir à un gros modèle unique, mais plutôt de construire un ensemble de modèles pertinents, couvrant au mieux la diversité des scénarios d'interactions entre sûreté et sécurité.

5.2 Appui à l'identification des effets de bord

Les modèles BDMP hybrides et intégrés permettent de mieux caractériser les situations où sûreté et sécurité interagissent étroitement. Ils peuvent constituer un appui précieux dans une analyse de risques couvrant les problématiques de sûreté et de sécurité [26, 27, 29]. Une fois les risques identifiés et caractérisés, des décisions peuvent être prises, qu'elles correspondent à des acceptations de risque, des décisions de réduction de risque par mise en place de contre-mesures, ou toute autre alternative déjà discutée dans le Chapitre II. Cependant, les modèles BDMP hybrides ou intégrés ne permettent pas d'identifier des effets de bord imprévus ou des dépendances cachées qui affecteraient les contre-mesures mises en place : les interdépendances caractérisées par l'approche décrite en Section 3 sont les interdépendances explicitement modélisées. Nous esquissons dans cette section une méthode complémentaire simple pour assister le concepteur dans ce but.

Différentes stratégies de réduction de risque peuvent être quantifiées et comparées en changeant le paramétrage des feuilles dans les modèles BDMP, modélisant les effets de contre-mesures. Afin de repérer des interdépendances cachées, nous proposons de prendre compte de façon plus explicite les composants

du système et leurs dépendances vis-à-vis de contre-mesures évaluées indépendamment dans des modèles BDMP « purs ». La Figure IV.7 permet d'exposer concrètement cette proposition. La partie a) de la figure représente schématiquement les composants d'un système qui correspond au SiS du cas d'étude précédent. Il est constitué d'un automate programmable (ou PLC, noté P), d'un capteur (C) permettant la détection de conditions anormales dans le flux de la canalisation, et d'un actionneur (A) capable d'arrêter le flux en cas d'incident. Le BDMP de la partie b) de la figure modélise une défaillance accidentelle dans une approche sûreté, tandis que le BDMP de la partie c) modélise des scénarios d'attaque sur le SiS. L'approche proposée s'appuie sur l'ajout à ces modèles des deux types d'information suivants, représentés également sur la Figure IV.7 :

- sous chaque feuille, une liste des composants concernés par la feuille est indiquée. Pour le BDMP orienté sûreté, il s'agit directement du composant modélisé par la feuille ; pour le BDMP sécurité, la correspondance peut être moins directe, même si le cas ici reste simple ;
- dans une notation similaire à celle proposée par Bistarelli [318] pour les arbres de défense et employée dans d'autres références [332, 282], des boîtes en pointillé indiquent les contre-mesures imaginées par l'analyste dans ses tentatives de réduction de risque pour le modèle BDMP examiné. Ces boîtes sont reliées graphiquement aux feuilles dont elles influencent la valeur des paramètres, reflet des effets des contre-mesures.

Nous pouvons maintenant expliquer comment ces deux ajouts vont nous permettre d'identifier des interdépendances entre sûreté et sécurité. Supposons que l'analyse du BDMP orienté sûreté ait montré que le risque de défaillance accidentelle du SiS pouvait être largement limité par une réduction du temps moyen de réparation du PLC (paramètre μ de la feuille de défaillance en opération PLC). Une telle réduction peut être obtenue par une amélioration de la surveillance du composant et par une intervention plus rapide des experts. Dans cette optique, l'analyste propose d'équiper le dispositif de surveillance du PLC d'une connexion par modem sur le réseau téléphonique commuté (RTC) public. Cette proposition est représentée par la boîte en trait pointillé positionnée dans la Figure IV.7b) sous la feuille PLC. Avant toute décision, l'impact d'un tel changement peut être analysé sur le BDMP orienté sécurité, en mettant à profit la liste des composants associés à ses feuilles. Les effets du changement sont en effet estimés sur le BDMP sécurité pour toutes les feuilles mentionnant un lien avec le composant PLC. Concrètement, une telle connexion conduit à une baisse du niveau de sécurité et à des paramètres de réussite pour l'attaquant plus élevés pour les feuilles Attaque de dénis de service sur l'automate, Changement des réglages de seuil et Usurpation d'identité du capteur (en désactivant par exemple un contrôle sur l'automate). Il est alors possible de prendre une décision quant au déploiement du modem, sachant de façon qualitative qu'il influencera le niveau de sûreté et de sécurité du système, mais également, en prenant en compte les éventuelles analyses quantitatives associées, réalisées grâce aux modèles BDMP. Réciproquement, l'analyse du BDMP sécurité peut mener au projet de déployer une authentification entre le PLC et le capteur pour réduire les possibilités d'usurpation (feuille Usurpation d'identité du capteur) : une démarche similaire à celle précédemment décrite permet d'identifier, et si nécessaire de quantifier, les impacts potentiels sur le BDMP sûreté.

6 Limites et perspectives

6.1 *Bis repetita placent*

Les BDMP constituant le socle de l'approche proposée dans ce chapitre, nous retrouvons les limites, mais également les perspectives, énoncées dans le Chapitre III dédié à ce formalisme. Ainsi, la modélisation de comportements cycliques reste délicate, ce qui peut entraver la modélisation de certains systèmes particuliers et de scénarios d'attaques présentant des boucles. De plus, si l'utilisation d'un cadre markovien et plus globalement celle d'une approche quantitative sont communément admises pour la caractérisation des aspects de sûreté, leur emploi pour les aspects sécurité mérite justifications et précisions. Elles sont développées en III-6.1. La formalisation d'un cadre élargi (renommé BDSP dans le Chap. III) aurait dans le contexte du présent chapitre également tout son sens. De même, les limites et perspectives associées à l'analyse de sensibilité des feuilles, discutées en III-6.3.7, ont aussi une pertinence directe pour l'étude des interactions sûreté-sécurité. Par ailleurs, les intégrations potentielles avec d'autres formalismes graphiques mentionnées en III-6.3.6 trouvent un intérêt encore renforcé sous ce nouvel angle. En

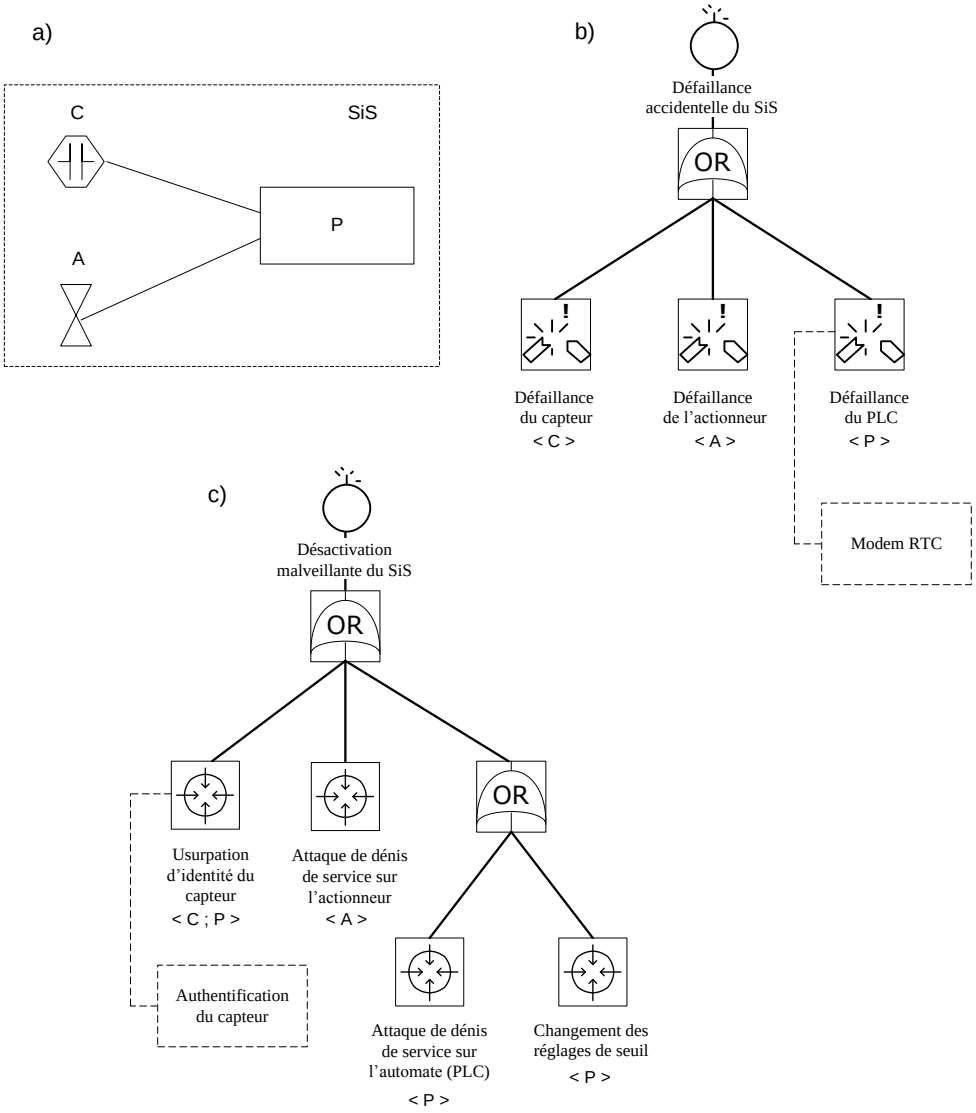


FIGURE IV.7 – Augmentation des modèles BDMP pour une meilleure identification des interdépendances

particulier, l'intégration avec les réseaux bayésiens enrichirait non seulement la démarche de paramétrage des BDMP mais aussi la méthode d'identification d'effets de bord proposée dans la Section 5.2, en l'état strictement qualitative. Enfin, l'intégration avec les diagrammes de *misuse case* aurait également un attrait consolidé par l'existence d'approches basées sur UML et ses différentes représentations graphiques en sûreté comme en sécurité [535, 536].

6.2 Améliorations spécifiques aux modèles hybrides et intégrés

La Section 5 n'a fait qu'esquisser des améliorations visant à systématiser d'une part la couverture des interactions entre sûreté et sécurité, et d'autre part l'identification de leurs éventuels effets de bord. Nous prévoyons de mieux formaliser ces premiers développements, et les tester sur des modèles plus élaborés.

Par ailleurs, la quantification des modèles intégrant à la fois composants réparables et prise en compte des quatre modalités de détection/réaction (IEFA) nécessitent des ajustements d'un point de vue algorithmique : l'algorithme SN (cf. III-5.1.2), utilisé pour traiter les modèles présentés dans ce mémoire, n'est en effet plus adapté aux caractéristiques du processus de Markov sous-jacent dans ce cas. L'adéquation des autres algorithmes mis à disposition par la plate-forme *KB3* reste à étudier : les transitions retours impliquées par les réparations, mais aussi par les détections *a posteriori*, couplées à une répartition des temps de séjour spécifique au mélange d'événements sûreté et sécurité, rendent la situation particulière par rapport aux hypothèses retenues pour le traitement de modèles par *KB3* en sûreté de fonctionnement. Ceci-dit, il est toujours possible de quantifier les modèles par simulation de Monte-Carlo.

6.3 Une contribution à intégrer avec d'autres approches

La Section 4 a permis de mieux situer notre contribution par rapport à l'état de l'art du domaine. Elle semble complémentaire à la grande majorité des contributions identifiées. En effet, la modélisation par BDMP pourrait permettre d'approfondir des situations identifiées par des démarches de plus haut niveau comme celles mentionnées en Section 2.1. Elle pourrait également compléter les analyses et appuyer des choix d'architecture discutés dans les contributions plus ciblées de la Section 2.2. Enfin, elle serait avantageusement couplée avec des approches de type *model-checking*, dans l'esprit de celles évoquées en Section 2.3, permettant d'explorer de façon plus systématique les interactions entre aspects sûreté et sécurité. En l'état actuel, le nombre de scénarios couvert par notre approche est d'une part très dépendant des compétences et de la rigueur de l'analyste ; il se limite d'autre part aux interactions explicitement modélisées par celui-ci. Le rapprochement avec les approches de *model-checking* pourrait bénéficier des travaux de Chaux [537], qui s'intéresse aux apports du *model-checking* pour l'analyse qualitative des BDMP, et qui se poursuivent actuellement à travers une thèse à EDF R&D.

Globalement, ces différentes complémentarités, identifiées *a priori*, restent à explorer afin de mieux cibler les intégrations les plus prometteuses, et d'initier leur concrétisation.

7 Conclusion

La polyvalence des BDMP permet d'intégrer événements accidentels et actions malveillantes dans un même modèle dynamique, facilitant la caractérisation des interdépendances entre sûreté et sécurité. Nous avons, à travers un cas d'étude simple, identifié différents types de situations où s'entremêlent sûreté et sécurité, et avons montré comment les capacités de modélisation et de quantification des BDMP permettaient de les analyser. De plus, l'approche proposée s'inscrit de façon très complémentaire avec les travaux déjà menés sur l'interdépendance entre sûreté et sécurité. Une exploration méthodique de ces complémentarités constitue une perspective engageante vers une maîtrise nécessaire de cette problématique encore largement ouverte.

Conclusion générale

Rappel des objectifs de la thèse

Historiquement séparées, sûreté et sécurité sont aujourd'hui devenues deux problématiques intimement liées. Les disciplines associées ont chacune développé indépendamment différentes approches et méthodes. Depuis les années 80, les trop rares initiatives de décloisonnement, adaptant notamment des outils d'un domaine à l'autre, ont abouti à des résultats convaincants. Cependant, ce décloisonnement reste encore timide et le potentiel des adaptations considérable. De plus, la récente convergence de risques de sécurité et de sûreté impliquent des interactions inédites entre exigences et mesures auparavant indépendantes, dont la caractérisation implique une perspective globale. Dans ces conditions, nos travaux de thèse ont d'abord visé à mieux cerner les notions de sûreté et de sécurité, sur le fond comme sur la forme, pour contribuer au décloisonnement nécessaire des communautés du risque. Nous avons ensuite voulu tirer parti de la complémentarité des outils de chaque domaine pour faire progresser l'état de l'art sur les aspects pour lesquels le potentiel de fertilisation croisée nous semblait le plus prometteur. Enfin, nous désirions contribuer au défi de la modélisation des nouvelles situations où risques sûreté et risques sécurité pèsent conjointement sur les mêmes systèmes. Chacun de ces objectifs s'inscrit dans une perspective dépassant le cadre de cette thèse, appelant à une indispensable intensification du mouvement de rapprochement entre sûreté et sécurité.

Principales contributions

Ces contributions peuvent être regroupées en trois points :

- **Le cadre référentiel SEMA.** Un panorama des normes et de la littérature scientifique amène à plusieurs dizaines de définitions différentes des termes sécurité et sûreté. Ces différences vont de légères nuances jusqu'à de complètes inversions selon les domaines considérés ; elles sont à l'origine de nombreux équivoques et incompréhensions. Nous avons élaboré un outil conceptuel dénommé SEMA (Système-Environnement Malveillant-Accidentel) offrant un cadre de référence positionnant ces concepts l'un par rapport à l'autre selon leur contexte d'utilisation, et rendant explicites les différences de signification cachées derrière l'emploi de ces termes en apparence anodins. De plus, SEMA permet de souligner les éventuelles incohérences ou recouvrements dans les définitions existantes, et de considérer les risques de sûreté et de sécurité d'un point de vue holistique pour assurer une meilleure couverture des analyses.

  Ces travaux ont fait l'objet de deux publications en conférence [36, 128] et d'un article de journal [37].

- **L'adaptation du formalisme BDMP en sécurité.** Un état de l'art des outils d'évaluation en sûreté et sécurité, appuyé par l'inventaire des adaptations déjà menées entre les deux domaines, nous ont conduits à explorer plus particulièrement les modélisations graphiques d'attaques. L'existant, bien que foisonnant, manquait d'un formalisme offrant un juste compromis entre prise en compte du caractère dynamique des attaques, lisibilité et capacités d'aide à la décision. L'adaptation des BDMP (*Boolean logic Driven Markov Processes*), issus du domaine des études de fiabilité et de sûreté, au domaine de la sécurité permet de dépasser bien des limites inhérentes aux techniques classiques de modélisation d'attaques. L'aspect visuel des BDMP, proche de celui des arbres d'attaque, hérite de leur lisibilité et de leur facilité d'appropriation. Cependant, les capacités de modélisation des BDMP sont bien supérieures, prenant en compte dépendances simples

et caractéristiques dynamiques. Par ailleurs, leurs propriétés mathématiques permettent une identification et une quantification temporelle efficaces des scénarios modélisés. En plus du travail de transposition du formalisme, nous avons étendu ses bases théoriques pour y intégrer des notions de sécurité telles que détection et réaction. L'adaptation des BDMP à la sécurité a enfin été concrétisée dans une mise en œuvre logicielle opérationnelle.

Ces travaux ont fait l'objet de trois articles de conférence [480, 464, 479].

- **La caractérisation et l'étude des interdépendances sûreté-sécurité avec les BDMP.** L'adaptation des BDMP à la sécurité a été mise à profit pour servir de formalisme de modélisation intégrateur, permettant de capturer graphiquement et caractériser des situations relevant à la fois de la sécurité et de la sûreté. Nous avons montré comment cette approche se positionnait dans l'état de l'art et en quoi ses spécificités et ses capacités de quantification en faisaient un outil d'intérêt dans l'étude des interdépendances entre sûreté et sécurité.

  Ces travaux ont fait l'objet d'un article de conférence [25].

Bilan et perspectives

Si les trois contributions précédemment exposées répondent aux objectifs de notre thèse, elles ne prétendent pas le faire de façon complète et définitive. Pour chacune d'entre elles, différentes limites et pistes d'amélioration ont ainsi pu être identifiées, ouvrant la voie à de futurs développements. Certaines correspondent à des travaux ciblés et sont susceptibles d'aboutir à des résultats à court terme dans la continuité de cette thèse ; d'autres s'inscrivent dans des horizons plus larges, combinant décloisonnement des communautés sûreté et sécurité et maîtrise de la convergence de ces problématiques. Nos contributions ne peuvent alors y être considérées que comme des briques élémentaires.

Nous synthétisons ici les différentes perspectives évoquées au fil des chapitres du mémoire et donnons ainsi une vision globale des pistes ouvertes par ces travaux :

- les premières utilisations de SEMA ont permis de recueillir différentes suggestions et d'identifier plusieurs évolutions potentielles. Nous ne mentionnons ici que les deux principales. La première relève plutôt d'une extension de son périmètre d'application : si celui-ci a été défini par et pour une analyse concernant les termes sûreté et sécurité, il pourrait également s'avérer utile pour expliciter des termes connexes, ayant des significations changeantes selon les contextes comme les mots sûreté et sécurité, ou mettre en évidence des concepts au contraire communs à différents domaines, mais faisant l'objet d'appellations disparates (*e.g.* « menace », « danger », « vulnérabilité »). La seconde perspective correspond à une évolution plus profonde puisqu'elle consiste à distinguer dans le référentiel SEMA dimension physique et dimension informatique. En outre, l'objectif serait d'affiner la décomposition des types de risques couverts par les termes sûreté et sécurité pour refléter les attributs de la sécurité informatique (confidentialité, intégrité, disponibilité) et lever les ambiguïtés y attenant (notamment pour les notions d'intégrité et de disponibilité, employées également en sûreté de fonctionnement pour les systèmes classiques ou informatiques) ;

- si nous nous sommes attachés à mener à bien la transposition des BDMP à la sécurité, l'état de l'art du Chapitre II nous a conduits à l'identification d'autres pistes d'adaptation prometteuses entre outils du domaine de la sûreté et outils de la sécurité. Tout d'abord, d'une façon générale, la grande majorité des adaptations ont été menées à partir de la sûreté vers la sécurité : la démarche inverse présente un potentiel encore relativement peu exploré. À titre d'exemple, les modèles formels et les méthodes d'analyse de risques constituent deux thématiques pour lesquelles la sûreté pourrait avantageusement puiser de nouvelles idées de la sécurité et enrichir ses propres outils. Réciproquement, la sécurité a encore beaucoup à apprendre de la sûreté. Les modèles graphiques, employés et affinés depuis les années 60 en sûreté, constituent un gisement d'inspirations encore loin d'être tari. L'adaptation des BDMP en est une bonne illustration, mais d'autres formalismes graphiques peuvent s'avérer pertinents en sécurité. En particulier, les approches déductives, de type arbres d'événements, ou intégrant raisonnements inductifs et déductifs, comme l'analyse Cause-Conséquence, n'ont pas encore été attentivement considérées en sécurité informatique. Pourtant, elles y trouveraient une place utile en modélisant les conséquences associées aux scénarios d'attaque à l'étude ;

- l'adaptation et l'extension des BDMP menées dans cette thèse ont abouti à un formalisme déjà opérationnel, mais pour lequel différentes évolutions et améliorations ont pu être identifiées. Sur le plan théorique d'abord, l'utilisation de lois exponentielles pourrait être complétée par d'autres types de modélisations stochastiques en généralisant les BDMP en BDSP (*Boolean logic Driven Stochastic Processes*). Par ailleurs, la définition des paramètres pourrait s'appuyer sur des éléments de théorie des jeux, reflétant de façon plus formelle les intérêts de l'attaquant et/ou du défenseur du système ; elle pourrait également impliquer l'emploi de nombres flous, modélisant l'incertitude sur la valeur des paramètres. La logique floue pourrait également intervenir à travers des portes logiques spécifiques, traduisant alors l'incertitude sur le nombre ou la nature des actions à réaliser pour un objectif d'attaque donné. Enfin, les BDMP s'intégreraient avantageusement avec d'autres formalismes graphiques, dont les *misuse cases* et les réseaux bayésiens. Cette dernière piste nous semble particulièrement pertinente. Sur un plan plus appliqué, nous avons prévu de faciliter l'utilisation des BDMP en sécurité par la constitution d'une bibliothèque de motifs d'attaques facilitant la construction des modèles. Nous comptons également enrichir les outils d'analyse et d'aide à la décision pour mieux exploiter les modèles. Un tel enrichissement passera notamment par la mise à profit du booléen de détection pour fournir de nouveaux types de quantification, et par la réalisation d'outils d'étude de sensibilité visant à identifier plus efficacement les éléments et paramètres clés vis-à-vis des objectifs de l'analyse ;

- si les BDMP permettent de caractériser de façon inédite des situations où enjeux de sûreté et de sécurité se côtoient, la finesse et la complétude des modélisations restent dans l'état de nos travaux très dépendantes de l'analyste. SEMA pourrait contribuer à systématiser l'approche et à construire des modèles de telle sorte que les différentes facettes des risques de sûreté et de sécurité ainsi que leurs interactions soient couvertes. De plus, l'identification des éventuels renforcements ou antagonismes dans ce processus n'a été traitée que par une proposition relativement élémentaire que nous projetons de formaliser et de tester de façon plus approfondie. Plus globalement, la modélisation par BDMP semble être complémentaire avec les démarches de plus haut niveau (*e.g. SafSec*) et avec celles reposant sur des techniques de type *model-checking* identifiées durant nos travaux. L'exploration effective de ces complémentarités reste à mener ;

- enfin, durant nos recherches sur les adaptations méthodologiques entre sûreté et sécurité, nous avons pu repérer, au delà des formalismes graphiques précédemment évoqués, d'autres approches susceptibles de contribuer à l'étude de leurs interdépendances. En fait, toutes les techniques d'analyse de risques et de modélisation ayant été adaptées d'un domaine à l'autre présentent un potentiel évident pour intégrer considérations de sûreté et de sécurité. On peut ainsi mentionner HAZOP, l'AMDEC ou encore la notation GSN. De telles approches intégratives (encore inexplorées à notre connaissance pour les trois exemples cités), aboutiraient certainement à des résultats complémentaires à notre proposition reposant sur les BDMP. Enfin, des langages de modélisation informatique génériques comme UML ou plus spécialisés comme AADL nous paraissent également de bons candidats pour l'étude des interdépendances entre sûreté et sécurité. En particulier, AADL a déjà été adapté à la sûreté de fonctionnement [246] et employé pour modéliser des politiques de sécurité informatique [538] ; la convergence des deux aspects semble donc être une perspective séduisante, dont les premières explorations viennent d'être initiées [539].

Les travaux présentés dans ce mémoire ont ainsi conduit à l'identification de nombreuses perspectives, allant de la continuation directe de nos recherches à l'exploration de voies alternatives. Dans tous les cas, elles ne se concrétiseront et prendront tout leur sens que si les communautés de la sûreté et de la sécurité accentuent leur rapprochement, dont cette thèse se veut à la fois une manifestation et une invitation.

Annexe A

Notes sur les référentiels de conformité et les certifications en sécurité

Historique

L'emploi de référentiels de conformité et de certification (désignés aussi par « approches par critères ») a un riche historique en matière de sécurité informatique. Comme le rapporte Dacier [257], on peut situer ses origines au rapport Ware [248], qui en 1970 regroupa les conclusions d'un groupe de travail animé par le département de la Défense états-unien (DoD) dès 1967 sur les nouvelles problématiques posées par le traitement informatisé d'informations sensibles. Parmi les diverses conclusions et recommandations, d'une actualité encore surprenante, certaines aboutirent en 1983 aux critères TCSEC (*Trusted Computer Security Evaluation Criteria*) auxquels devaient se conformer les systèmes informatiques du DoD. Il faut attendre le début des années 90 pour trouver des démarches similaires ou améliorées dans les autres pays (ITSEC en Europe, JCSEC au Japon ou CTCPEC au Canada) [386]. Au début des années 90, les États-Unis proposent une nouvelle approche, nommée *Federal Criteria* (FC), qui introduit le concept de profil de protection. Sous l'égide de l'ISO, les différentes initiatives sont ensuite rapprochées, confrontées et donnent naissance en 1996 à la première version des Critères Communs (CC) aujourd'hui dans leur troisième version majeure.

Les Critères Communs

Les CC constituent la norme mondiale de référence pour l'évaluation des produits de sécurité. Documentés dans l'IEC/ISO 15408 [258], leur formalisme lourd et complexe distingue notamment :

- la cible d'évaluation (ToE, pour *Target of Evaluation*), qui décrit le périmètre du produit à certifier. Les exigences fonctionnelles de sécurité à évaluer, qui caractérisent la ToE en termes de fonctions de sécurité soumises à l'analyse, sont choisies dans un catalogue normalisé ;
- le profil de protection (PP, pour *Protection Profile*), qui correspond à un ensemble type d'exigences de sécurité pour une catégorie de produits (*e.g.* coupe-feu, cartes à puce, etc.). Si les exigences élémentaires sont définies dans la norme, leur assemblage peut être fait de multiples manières. Il existe ainsi de très nombreux PP, généralement proposés par les organismes gouvernementaux ;
- la cible de sécurité (ST, pour *Security Target*), qui adapte spécifiquement un ou plusieurs PP au produit évalué en précisant les menaces et les objectifs de sécurité pris en compte dans l'évaluation ;
- le niveau d'assurance d'évaluation (EAL, pour *Evaluation Assurance Level*). Les Critères Communs en définissent sept, qui assemblent de façon cumulative des « paquets d'assurance » normalisés permettant de donner un niveau de confiance croissant au processus d'évaluation. La Table A.1 les présente sommairement.

TABLE A.1 – les 7 niveaux d'assurance d'évaluation des CC

EAL1	Testé fonctionnellement
EAL2	Testé structurellement
EAL3	Testé et vérifié méthodiquement
EAL4	Conçu, testé et vérifié méthodiquement
EAL5	Conçu de façon semi-formelle et testé
EAL6	Conception vérifiée de façon semi-formelle et testée
EAL7	Conception vérifiée de façon formelle et testée

Les évaluations CC sont menées par des centres d'évaluation spécialisés, agréés au niveau de chaque état ; les certificats délivrés sont alors reconnus par tous les signataires d'un traité de reconnaissance mutuelle, le CCRA [1]. La lourdeur et la complexité du processus ont éveillé de nombreuses critiques [260]. Le coût et la durée des certifications ont poussé certains pays comme la France à proposer en complément des CC des schémas de certification alternatifs allégés (CSPN [2]). Soulignons pour finir que si le domaine d'application des CC n'est plus limité aux systèmes militaires ou gouvernementaux, ils ne concernent par contre que les produits en charge de fonction de sécurité.

Autres référentiels

En parallèle du développement des Critères Communs, d'autres référentiels plus sectoriels ont vu le jour : on peut citer notamment les certifications VISA ou EMV bien établies dans le domaine bancaire. Plus récemment, des schémas de certification pour la sécurité des systèmes d'informatique industrielle ont fait leur apparition (*e.g. ISA Secure* [259] décrit ci-dessous, ou encore les certifications Achille [3] de la société Wurldtech). Encore balbutiantes, il est pour l'instant délicat de juger de leur pertinence. Elles gagneront dans tous les cas à éviter les pièges et difficultés rencontrés par leur prédécesseurs comme les CC [260].

Certification *ISA Secure*

Le programme de certification *ISA Secure* s'appuie sur un consortium industriel et sur la série de normes ISA 99 [112]. La première certification sécurité, lancée fin 2009 et nommée EDSA [4], vise les systèmes embarqués utilisés dans l'informatique industrielle [259]. La certification EDSA distingue trois niveaux d'assurance de sécurité, prenant en compte l'évaluation de trois aspects : la robustesse de la pile de communication (dont les exigences sont communes aux trois niveaux), le cycle de développement du produit, et les fonctions de sécurité mises en œuvre par le produit. Les exigences sont croissantes avec les niveaux sur ces deux derniers aspects.

1. *Common Criteria Recognition Arrangement* (CCRA)[540].
2. Certification de Sécurité de Premier Niveau (CSPN)[541].
3. http://www.wurldtech.com/cyber-security/achilles-certification/achilles-certification.aspx
4. *Embedded Device Security Assurance* (EDSA, http://www.isasecure.org).

Annexe B

Notes sur les modèles formels de politique de sécurité

Politiques et modèles de contrôle d'accès

Avertissement. Nous ne donnons dans cette annexe qu'un aperçu de ce vaste domaine. Le lecteur intéressé pourra se référer à [383, 384, 382] pour une vue plus complète et plus approfondie.

Éléments historiques et notions fondamentales. Les TCSEC, évoqués dans le Chapitre II Section 2.3.2 et dans l'Annexe A, ont introduit divers concepts fondamentaux en termes de politique de sécurité et de modèles formels. En outre, ils distinguaient deux catégories de politiques de contrôle d'accès : les politiques de type discrétionnaire (communément désignées par l'acronyme DAC, pour leur désignation en anglais *Discretionnary Access Control*) et les politiques d'accès de type obligatoire, aussi parfois appelées politiques par mandat ou mandataires (et communément désignées par l'acronyme MAC, pour *Mandatory Access Control*). De façon simplifiée, dans la première, les droits d'accès sont spécifiés individuellement par les propriétaires des ressources et peuvent être transmis, alors que dans la seconde, ils sont attribués par une autorité globale au système sur la base de niveaux de sensibilité des ressources considérées. La mise en œuvre la plus courante des politiques MAC correspond à la politique dite multi-niveaux (ou MLS pour *Multi-Level Security*) ; elle consiste à attribuer des niveaux de sécurité aux ressources, et gérer leur accès par un système d'habilitation. Les TCSEC imposaient ce type de politique pour les niveaux de sécurité les plus élevés du référentiel, et le caractérisaient par une modélisation formelle définie par Bell et Lapadula quelques années plus tôt [542]. Il s'agit d'un des premiers exemples de modélisation de politique de contrôle d'accès, permettant de vérifier formellement que des règles de contrôle d'accès respectent la politique choisie.

Bell-Lapadula (BLP). Ce modèle repris par les TCSEC est un modèle de politique de contrôle d'accès obligatoire, orienté confidentialité. Il repose sur une machine à états permettant de s'assurer qu'une politique de droits d'accès donnée, représentée sous la forme d'une matrice statique, ne laisse pas des informations classées à un niveau de confidentialité donné être accessibles par des sujets ayant une habilitation d'accès d'un niveau inférieur. On vérifie pour cela les propriétés dites « *no-read-up* » (ou *ss-*, pour *simple security*), « *no-write-down* » (**-*), et de contrôle d'accès discrétionnaire (*ds-*). La première correspond à l'interdiction pour un sujet de lire un objet classifié à un niveau d'habilitation supérieur à son propre niveau d'habilitation ; la seconde lui interdit d'écrire dans un objet classifié à un niveau inférieur à sa propre habilitation ; la troisième correspond simplement au respect des droits indiqués dans la matrice de contrôle d'accès, spécifiant les droits des sujets en lecture et en écriture pour tous les objets du système. Hélas, BLP permet de ne traiter que des droits d'accès statiques, uniquement dans une optique de confidentialité, et ne rend pas compte d'éventuels canaux cachés (un refus d'accès conforme à BLP peut par exemple informer de l'existence même d'un objet). La Figure B.1 illustre de façon simplifiée les accès permis, dans la partie a), et les accès interdits, dans la partie b), d'une politique de contrôle d'accès type BLP.

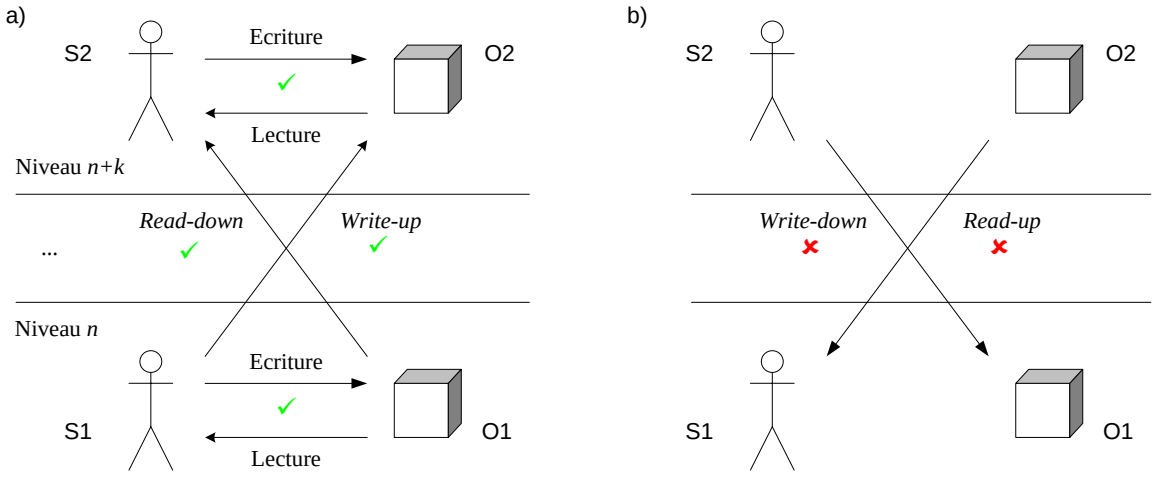


FIGURE B.1 – Lecture/écriture entre sujets et objets « hauts » et « bas »

Biba. Le modèle de Biba [543] (1977) transpose l'approche BLP, avec ses défauts, à des objectifs d'intégrité ; il s'appuie sur la vérification de propriétés inversées par rapport à celui-ci, de type *no-read-down* et *no-write-up*, permettant de s'assurer que des informations « propres » de niveau élevé ne peuvent être corrompues par des informations « sales » de niveau inférieur.

Modèles de politique discrétionnaire à base de matrices de contrôle d'accès. Les modèles de Graham-Denning [544] (1971) puis de Harrison-Ruzzo-Ullman (dit HRU, 1976) [545] sont des modèles de politique discrétionnaires, permettant de prendre en compte les changements de droits, la création et la destruction d'objets. Ils introduisent pour cela de nouvelles propriétés (dont une dénommée *safety* pour HRU, avec une signification encore différente de celles discutées dans le Chapitre I) mais qui ne peuvent plus être systématiquement prouvées, certaines situations devenant indécidables. Dans le même type d'approches, on peut aussi citer *Take & Grant* (1977)[546], basée sur des graphes orientés, et TAM (*Typed Access Matrix*, 1992) [547], largement inspirée de HRU.

Clark-Wilson. Plus tardivement, le modèle de Clark-Wilson [548] (1987), également orienté intégrité et conçu pour les applications commerciales, propose de façon moins formelle mais plus souple des vérifications du même ordre que celles permises par les modèles du précédent paragraphe.

Muraille de Chine. Parmi les grands classiques des modèles de contrôle d'accès, on peut encore citer le modèle de la Muraille de Chine (1989) [549] qui permet de vérifier qu'aucune information susceptible de provoquer des conflits d'intérêt ne circule. Il est particulièrement adapté aux organisations manipulant des données à caractère commercial ou financier.

***Role-Based Acces Control* (RBAC).** Dans le modèle RBAC (*Role-based Acces Control*) [550], les droits d'accès ne sont plus directement rattachés aux utilisateurs, mais à des rôles abstraits, définissant un profil type établi par rapport à une fonction donnée dans l'organisation. Les utilisateurs sont ensuite rattachés à un ou plusieurs rôles au gré de leurs évolutions dans l'organisation, évitant d'avoir à redéfinir pour chaque utilisateur la liste complète des droits d'accès lors d'un changement.

***Organization-Based Access Control* (OrBAC).** Le modèle OrBAC, plus récent (2003), est centré sur le concept d'organisation. Il s'appuie sur les concepts de rôles du modèle RBAC, mais aussi d'activités, de vues et d'organisations, correspondant aux abstractions des actions possibles dans le système et de ses objets. OrBAC permet de définir dynamiquement les droits d'accès selon des notions de contextes et autorise la définition de permissions, mais également d'interdictions et d'obligations.

Modèles d'autorisation à base de flux et de traces d'exécution. Ces modèles ont été initialement élaborés pour répondre aux limites des modèles à base de matrices d'accès ou de treillis comme BLP et Biba, qui en limitant l'analyse aux notions de sujets et d'objets ne permettaient pas de prendre en compte l'existence de canaux cachés.

La vue ici adoptée est plus générale que dans les modèles précédents, intégrant les notions de flux d'information et de comportement global du système (par une formalisation de ses états successifs sous la forme d'une trace d'exécution). Historiquement, la propriété de *non-interference*, introduite par Goguen et Meseguer [440] en 1982, joue un rôle fondamental. Elle inaugure une série de modèles, complémentaires à ceux déjà mentionnés, basés sur la notion d'*information-flow* [551]. La *non-interference* correspond à l'idée suivante : un utilisateur A interfère avec un utilisateur B dans un système si et seulement si les actions de A sur le système peuvent affecter ce que B peut observer ou faire de celui-ci. Dans un environnement multi-niveaux, la *non-interference* correspond globalement au fait que les utilisateurs d'un niveau donné ne doivent rien pouvoir connaître de l'activité des utilisateurs des niveaux supérieurs. S'appuyant sur des bases proches de celles de BLP, la propriété de *non-interference* formalise ceci-dit une politique de sécurité plus contraignante, garantissant notamment l'absence de canaux cachés. Si elle est considérée aujourd'hui par certains comme préhistorique [552], elle a depuis donné jour à de nombreuses autres « propriétés » pour certaines directement dérivées en allégeant ou nuançant la contrainte exprimée, pour d'autres plus éloignées (on peut citer ainsi la *non-deduciblity,* la *correcteability*, la *causality*, la *non-influence*, la *non-inference*...). Focardi et Gorrieri formalisent et comparent une douzaine de ces propriétés dans [553]. Elles se distinguent par l'exigence sécurité modélisée, mais aussi par le formalisme utilisé ou par exemple leur facilité à être composée, ou encore à prendre en compte des systèmes non-déterministes. Ryan propose dans [554] une bonne synthèse permettant de saisir la philosophie générale de ces différentes propriétés.

Modèles cryptographiques

La cryptographie moderne, construite sur des bases mathématiques rigoureuses, constitue un sous-domaine particulier des techniques de sécurité ; modèles d'évaluation et preuves théoriques y sont très avancés et sans doute mieux structurés qu'ailleurs. Les propriétés attendues de primitives cryptographiques, que ce soit pour le chiffrement symétrique, asymétrique, ou la signature électronique, sont définies en rapport à leur résistance aux attaques permettant de « casser » l'algorithme plus efficacement que par recherche exhaustive. Si l'évaluation de cette résistance peut se limiter dans certains cas au constat qu'aucune attaque significative n'a pu être proposée par la communauté mondiale de la cryptanalyse (*e.g.* pour RSA et AES, les deux standards de fait), elle peut être dans certains cas formalisée dans un système de preuves théoriques. Les propriétés font alors l'objet d'une nomenclature communément admise. A titre d'exemple, pour les schémas de cryptographie asymétrique, on distingue les propriétés suivantes :

- *One-Wayness* (OW) : il est impossible de retrouver le clair [1] sans la clé privée ;
- *Indistinguishability* (IND) / *Semantic Security* : l'adversaire ne peut pas obtenir de l'information d'un message chiffré, même s'il sait qu'il a été choisi parmi plusieurs messages clairs connus de lui ;
- *Non-Malleability* (NM) : un attaquant ne peut pas créer un nouveau message signifiant en fonction d'un message chiffré ;
- *Chosen-Plaintext Attacks* (CPA) *Resistance* : l'attaquant peut chiffrer n'importe quel message avec la clé publique ayant chiffré le message attaqué ;
- *Adaptive Chosen Ciphertext Attacks* (CCA) *Resistance* : l'attaquant a accès à un « oracle » de déchiffrement, pouvant chiffrer à sa guise tout message, avant d'attaquer le chiffré ciblé (CCA1) ou même après (CCA2).

Ces propriétés sont articulées et hiérarchisées dans la Figure B.2.

On retrouve des systèmes de propriétés équivalents pour la signature électronique (avec la notion de forge universelle, ou existentielle) et le chiffrement symétrique, loin de la situation confuse décrite

1. Le « clair » désigne en cryptographie le message non-chiffré, par opposition au « chiffré ».

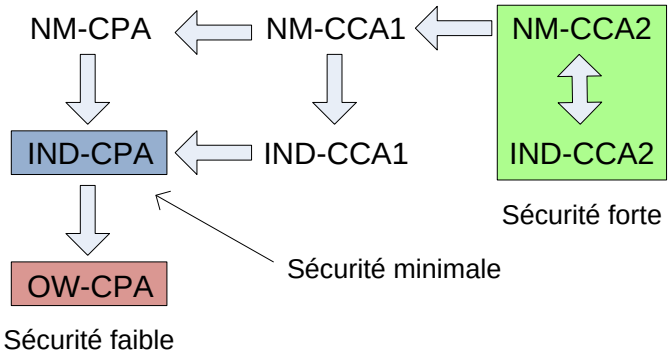


FIGURE B.2 – Propriétés cryptographiques prouvables pour le chiffrement asymétrique (d'après [555])

précédemment pour les modèles formels de politique sécurité. Pour le lecteur intéressé, Pointcheval propose un cours complet sur le sujet [555]. Notons toutefois que, en cryptographie, les preuves des propriétés évoquées reposent généralement sur des modèles idéalisés par rapport aux implémentations réelles : le hachage y est par exemple remplacé par une fonction d'aléas (*random oracle model*) et le chiffrement par blocs par une permutation strictement aléatoire (*ideal cipher model*). Si la démonstration de ces propriétés dans de tels modèles est incontestablement utile pour améliorer les schémas cryptographiques, elle ne peut pas être considérée comme suffisante pour garantir de façon absolue la sécurité de schémas cryptographiques dans leur déclinaison réelle [556], nonobstant les failles de mise en œuvre logicielle. Plus généralement, ces preuves ne doivent pas faire oublier que les mécanismes cryptographiques, aussi solides soient-ils, ne constituent qu'un élément parmi de nombreux autres dans la chaîne de mesures techniques et organisationnelles nécessaires au maintien d'un niveau satisfaisant de sécurité.

Annexe C

Données numériques des études de sensibilité

- Données utilisées pour tracer la courbe de la Figure III.19, p. 96 :

Feuille Force brute			
MTTS (jrs)	λ	P_s	λ_0/λ $(MTTS/MTTS_0)$
60	$1{,}93\times10^{-7}$	$3{,}53\times10^{-1}$	1,97
54	$2{,}14\times10^{-7}$	$3{,}61\times10^{-1}$	1,77
50	$2{,}31\times10^{-7}$	$3{,}68\times10^{-1}$	1,64
46	$2{,}52\times10^{-7}$	$3{,}76\times10^{-1}$	1,51
42	$2{,}76\times10^{-7}$	$3{,}85\times10^{-1}$	1,38
38	$3{,}05\times10^{-7}$	$3{,}95\times10^{-1}$	1,25
34	$3{,}40\times10^{-7}$	$4{,}08\times10^{-1}$	1,12
30,4	$3{,}80\times10^{-7}$	$4{,}22\times10^{-1}$	1,00
26	$4{,}45\times10^{-7}$	$4{,}45\times10^{-1}$	$8{,}54\times10^{-1}$
22	$5{,}26\times10^{-7}$	$4{,}71\times10^{-1}$	$7{,}23\times10^{-1}$
18	$6{,}43\times10^{-7}$	$5{,}07\times10^{-1}$	$5{,}91\times10^{-1}$
14	$8{,}27\times10^{-7}$	$5{,}59\times10^{-1}$	$4{,}60\times10^{-1}$
10	$1{,}16\times10^{-6}$	$6{,}39\times10^{-1}$	$3{,}29\times10^{-1}$
6	$1{,}93\times10^{-6}$	$7{,}74\times10^{-1}$	$1{,}97\times10^{-1}$
2	$5{,}79\times10^{-6}$	$9{,}78\times10^{-1}$	$6{,}57\times10^{-2}$

- Données utilisées pour tracer la courbe de la Figure III.20, p. 96 :

Feuille Préparation de la charge			
MTTS (h)	λ	P_s	λ_0/λ $(MTTS/MTTS_0)$
96	$2{,}89\times10^{-6}$	$4{,}18\times10^{-1}$	2,00
90	$3{,}09\times10^{-6}$	$4{,}18\times10^{-1}$	1,87
84	$3{,}31\times10^{-6}$	$4{,}19\times10^{-1}$	1,75
78	$3{,}56\times10^{-6}$	$4{,}19\times10^{-1}$	1,63
72	$3{,}86\times10^{-6}$	$4{,}20\times10^{-1}$	1,50
66	$4{,}21\times10^{-6}$	$4{,}20\times10^{-1}$	1,38
60	$4{,}63\times10^{-6}$	$4{,}21\times10^{-1}$	1,25
54	$5{,}14\times10^{-6}$	$4{,}22\times10^{-1}$	1,13
48	$5{,}79\times10^{-6}$	$4{,}22\times10^{-1}$	1,00
42	$6{,}61\times10^{-6}$	$4{,}23\times10^{-1}$	$8{,}75\times10^{-1}$
36	$7{,}72\times10^{-6}$	$4{,}25\times10^{-1}$	$7{,}50\times10^{-1}$
30	$9{,}26\times10^{-6}$	$4{,}26\times10^{-1}$	$6{,}25\times10^{-1}$
24	$1{,}16\times10^{-5}$	$4{,}28\times10^{-1}$	$5{,}00\times10^{-1}$
18	$1{,}54\times10^{-5}$	$4{,}30\times10^{-1}$	$3{,}75\times10^{-1}$
12	$2{,}31\times10^{-5}$	$4{,}32\times10^{-1}$	$2{,}50\times10^{-1}$
6	$4{,}63\times10^{-5}$	$4{,}36\times10^{-1}$	$1{,}25\times10^{-1}$

- Données utilisées pour tracer la courbe de la Figure III.21, p. 97 :

Feuille Reconnaissance générique			
MTTS (h)	λ	P_s	λ_0/λ $(MTTS/MTTS_0)$
48	$5{,}79\times10^{-6}$	$3{,}91\times10^{-1}$	2,00
44	$6{,}31\times10^{-6}$	$3{,}95\times10^{-1}$	1,83
40	$6{,}94\times10^{-6}$	$4{,}00\times10^{-1}$	1,67
36	$7{,}72\times10^{-6}$	$4{,}05\times10^{-1}$	1,50
32	$8{,}68\times10^{-6}$	$4{,}10\times10^{-1}$	1,33
28	$9{,}92\times10^{-6}$	$4{,}16\times10^{-1}$	1,17
24	$1{,}16\times10^{-5}$	$4{,}22\times10^{-1}$	1,00
20	$1{,}39\times10^{-5}$	$4{,}30\times10^{-1}$	$8{,}33\times10^{-1}$
16	$1{,}74\times10^{-5}$	$4{,}38\times10^{-1}$	$6{,}67\times10^{-1}$
12	$2{,}31\times10^{-5}$	$4{,}47\times10^{-1}$	$5{,}00\times10^{-1}$
8	$3{,}47\times10^{-5}$	$4{,}57\times10^{-1}$	$3{,}33\times10^{-1}$
4	$6{,}94\times10^{-5}$	$4{,}68\times10^{-1}$	$1{,}67\times10^{-1}$
1	$2{,}78\times10^{-4}$	$4{,}77\times10^{-1}$	$4{,}17\times10^{-2}$

- Données utilisées pour tracer la courbe de la Figure III.22, p. 97 :

Feuille Installation locale du keylogger			
MTTS (h)	λ	P_s	λ_0/λ $(MTTS/MTTS_0)$
48	$5{,}79\times10^{-6}$	$4{,}01\times10^{-1}$	2,00
44	$6{,}31\times10^{-6}$	$4{,}04\times10^{-1}$	1,83
40	$6{,}94\times10^{-6}$	$4{,}07\times10^{-1}$	1,67
36	$7{,}72\times10^{-6}$	$4{,}10\times10^{-1}$	1,50
32	$8{,}68\times10^{-6}$	$4{,}14\times10^{-1}$	1,33
28	$9{,}92\times10^{-6}$	$4{,}18\times10^{-1}$	1,17
24	$1{,}16\times10^{-5}$	$4{,}22\times10^{-1}$	1,00
20	$1{,}39\times10^{-5}$	$4{,}28\times10^{-1}$	$8{,}33\times10^{-1}$
16	$1{,}74\times10^{-5}$	$4{,}34\times10^{-1}$	$6{,}67\times10^{-1}$
12	$2{,}31\times10^{-5}$	$4{,}41\times10^{-1}$	$5{,}00\times10^{-1}$
8	$3{,}47\times10^{-5}$	$4{,}49\times10^{-1}$	$3{,}33\times10^{-1}$
4	$6{,}94\times10^{-5}$	$4{,}59\times10^{-1}$	$1{,}67\times10^{-1}$
1	$2{,}78\times10^{-4}$	$4{,}67\times10^{-1}$	$4{,}17\times10^{-2}$

- Les Figures III.23 et III.24 p. 98 ont été tracées grâce aux données consignées dans les colonnes λ/λ_0 des quatre tableaux précédents.

- Données utilisées pour tracer la courbe de la Figure III.25, p. 99 :

Feuille Utilisateur piégé		**Feuille Bonne exécution de la charge**	
γ	P_s	γ	P_s
0	$2{,}64\times10^{-1}$	0	$4{,}07\times10^{-1}$
0,1	$3{,}12\times10^{-1}$	0,1	$4{,}22\times10^{-1}$
0,2	$3{,}60\times10^{-1}$	0,2	$4{,}38\times10^{-1}$
0,3	$4{,}08\times10^{-1}$	0,3	$4{,}54\times10^{-1}$
0,4	$4{,}56\times10^{-1}$	0,4	$4{,}70\times10^{-1}$
0,5	$5{,}04\times10^{-1}$	0,5	$4{,}85\times10^{-1}$
0,6	$5{,}52\times10^{-1}$	0,6	$5{,}01\times10^{-1}$
0,7	$6{,}00\times10^{-1}$	0,7	$5{,}17\times10^{-1}$
0,8	$6{,}49\times10^{-1}$	0,8	$5{,}32\times10^{-1}$
0,9	$6{,}97\times10^{-1}$	0,9	$5{,}48\times10^{-1}$
1	$7{,}45\times10^{-1}$	1	$5{,}64\times10^{-1}$

- Données utilisées pour tracer les courbes des Figures III.26 et III.27, p.107 :

Proba. détection qd le piège échoue		**Proba. détection charge exécutée**		**Proba. détection charge défectueuse**	
$\gamma_{D/NR}$	P_s	$\gamma_{D/R}$	P_s	$\gamma_{D/NR}$	P_s
0	$3{,}31\times10^{-1}$	0	$3{,}62\times10^{-1}$	0	$3{,}55\times10^{-1}$
0,1	$3{,}38\times10^{-1}$	0,1	$3{,}64\times10^{-1}$	0,1	$3{,}58\times10^{-1}$
0,2	$3{,}44\times10^{-1}$	0,2	$3{,}65\times10^{-1}$	0,2	$3{,}60\times10^{-1}$
0,3	$3{,}51\times10^{-1}$	0,3	$3{,}66\times10^{-1}$	0,3	$3{,}63\times10^{-1}$
0,4	$3{,}57\times10^{-1}$	0,4	$3{,}67\times10^{-1}$	0,4	$3{,}65\times10^{-1}$
0,5	$3{,}64\times10^{-1}$	0,5	$3{,}69\times10^{-1}$	0,5	$3{,}68\times10^{-1}$
0,6	$3{,}70\times10^{-1}$	0,6	$3{,}70\times10^{-1}$	0,6	$3{,}70\times10^{-1}$
0,7	$3{,}77\times10^{-1}$	0,7	$3{,}71\times10^{-1}$	0,7	$3{,}73\times10^{-1}$
0,8	$3{,}83\times10^{-1}$	0,8	$3{,}72\times10^{-1}$	0,8	$3{,}75\times10^{-1}$
0,9	$3{,}90\times10^{-1}$	0,9	$3{,}74\times10^{-1}$	0,9	$3{,}78\times10^{-1}$
1	$3{,}96\times10^{-1}$	1	$3{,}75\times10^{-1}$	1	$3{,}81\times10^{-1}$

Sigles et acronymes

Note : la signification des sigles et acronymes employés dans ce mémoire est systématiquement indiquée à la première occurrence. Nous listons ici seulement ceux faisant l'objet d'une utilisation multiple, leur signification n'étant alors pas toujours rappelée après leur première apparition.

AA	*Attacker Action*
AADL	*Architecture Analysis & Design Language*
AIEA	Agence International de l'Énergie Atomique
AMDE	Analyse des Modes de Défaillance et de leurs Effets
AMDEC	Analyse des Modes de Défaillance, de leurs Effets et de leurs Criticités
ANSI	*American National Standards Institute*
ANSSI	Agence Nationale de la Sécurité des Systèmes
API	*American Petroleum Institute*
ASN	Autorité de Sûreté Nucléaire
BDMP	*Boolean logic Driven Markov Process*
BDSP	*Boolean logic Driven Stochastic Process*
BGP	*Border Gateway Protocol*
BLP	Bell-Lapadula
CC	Critères Communs
CCA	*Cause-Consequence Analysis*
CEI	Commission Électrotechnique Internationale
CERT	*Computer Emergency Response Team*
CFR	*Code of Federal Regulations*
CIIP	*Critical Information Infrastructure Protection*
CIP	*Critical Infrastructure Protection*
CNRS	Centre National de la Recherche Scientifique
COTS	*Commercial-Off-The-Shelf*
CSP	*Communicating Sequential Processes*
CVSS	*Common Vulnerability Scoring System*
DBT	*Design Basis Threat*
DFT	*Dynamic Fault Tree*
DHS	*Department of Homeland Security*
DoD	*Department of Defense*
DoE	*Department of Energy*
EAL	*Evaluation Assurance Levels*
EDSA	*Embedded Device Security Assurance*
E/E/EP	Électriques/Électroniques/Électroniques Programmables
EID	*Extended Influence Diagrams*
EMV	Europay Mastercard Visa
ENISA	*European Network and Information Security Agency*
EPS	Etude Probabiliste de Sûreté
EuroCAE	*European Organisation for Civil Aviation Equipment*
FMEA	*Failure Mode and Effects Analysis*
FRS	*Fragmentation-Redundancy-Scattering*
GEMS	*General Error Modeling System*
GSN	*Goal Structured Notation*
GSPN	*Generalized Stochastic Petri Nets*
HAZOP	*HAZard and OPerability studies*
IATA	*International Air Transport Association*
ICAO	*International Civil Aviation Organization*
IEC	*International Electrotechnical Committee*
IEEE	*Institute of Electrical and Electronics Engineers*
IEFA	Initiale, En-cours, Finale, *A posteriori*
IETF	*Internet Engineering Task Force*
IFIP	*International Federation for Information Processing*
INL	*Idaho National Laboratories*
IP	*Internet Protocol*
ISA	*International Society of Automation*
ISE	*Instantaneous Security Event*
ISO	*International Organization for Standardization*
ITSEC	*Information Technology Security Evaluation Criteria*
JRC	*Joint Research Center*

KINS *Korea Institute of Nuclear Safety*

LAAS Laboratoire d'Analyse et d'Architecture des Systèmes

MORDA *Mission Oriented Risk and Design Analysis*

MTBF *Mean Time Before Failure*

MTTD *Mean Time To Detection*

MTTF *Mean Time To Failure*

MTTR *Mean Time To Realization*

MTTS *Mean Time To Success*

NASA *National Aeronautics and Space Administration*

NERC *North American Electric Reliability Corporation*

NIST *National Institute of Standards and Technology*

NRC *Nuclear Regulatory Commission*

NSA *National Security Agency*

OLF *Oljeindustriens Landsforening*

OSSTMM *Open Source Security Testing Methodology Manual*

OTAN Organisation du Traité de l'Atlantique Nord

PCRD Programme Cadre de Recherche et Développement

PLC *Programmable Logic Controller*

RBAC *Role-based access control*

RFC *Request For Comments*

RTC Réseau Téléphonique Commuté

RTCA *Radio Technical Commission for Aeronautics*

SAIV Secteurs d'Activité d'Importance Vitale

SAL *Security Assurance Level*

SAN *Stochastic Activity Networks*

SCADA *Supervisory Control And Data Acquisition*

SEMA Système-Environnement/Malveillant-Accidentel

SIL *Safety Integrity Level*

SiS Système instrumenté de Sûreté

SN Séquences Normales

SPN *Stochastic Petri Nets*

SQUARE *Security QUality Requirements Engineering*

SRI Sans Retour à l'état Initial

TCP *Transmission Control Protocol*

TCSEC *Trusted Computer System Evaluation Criteria*

TSE *Timed Security Event*

TSN Transparence et Sécurité en matière Nucléaire

TTC *Time To Compromise*

UCTE Union pour la Coordination du Transport de l'Électricité

UML *Unified Modeling Language*

VISA *VISA International Service Association*

WG *Working Group*

XML *Extensible Markup Language*

ZHA *Zonal Hazard Analysis*

ZSA *Zonal Safety Analysis*

Bibliographie

[1] Commission européenne, « Critical infrastructure protection in the fight against terrorism ». COM(2004)702, oct. 2004.

[2] « Code de la Défense, articles R.1332-1 à 1332-42 et L.1332-1 à 1332-7 ». Version consolidée au 6 juin 2010.

[3] Premier Ministre, Secrétariat Général de la Défense Nationale (SGDN), « Instruction générale interministérielle relative à la sécurité des activités d'importance vitale ». IGI N°6600/SGDN/PSE/PPS du 26 septembre 2008.

[4] Commission européenne, « Council directive 2008/114/EC of 8 December 2008 on the identification and designation of European critical infrastructures and the assessment of the need to improve their protection ». Journal officiel de l'Union européenne, déc. 2008.

[5] Maison Blanche, « HSPD-7 Homeland Security Presidential Directive for critical infrastructure identification, prioritization, and protection ». Directive présidentielle états-unienne, déc. 2003.

[6] E. M. Brunner et M. Suter, *International CIIP Handbook 2008/2009*. Center for Security Studies, ETH Zurich, 2008.

[7] L. Pièvre-Cambacédès, T. Kropp, J. Weiss et R. Pellizzoni, « Cybersecurity standards for the electric power industry - a survival kit », dans *Proceedings of the 42nd CIGRE Session*, Paris, France, p. D2–217, août 2008.

[8] A. Bartels, L. Pièvre-Cambacédès et S. Duckworth, « Security technologies guideline - practical guidance for deploying cyber security technology within electric utility data networks », *Electra*, vol. 244, p. 11–17, 2009.

[9] L. Pièvre-Cambacédès et P. Sitbon, « An analysis of two new directions in control system perimeter security », dans *Proceedings of the 3rd SCADA Security Scientific Symposium (S4)*, Miami, États-Unis, p. 4.1–4.30, jan. 2009.

[10] L. Pièvre-Cambacédès, M. Tritschler et G. Ericsson, « Cyber security myths on power control systems : 21 misconceptions and false beliefs », *IEEE Transactions on Power Delivery*, 2010. Accepté pour publication.

[11] K. Stouffer, J. Falco et K. Scarfone, « Guide to industrial control systems (ICS) security ». U.S. National Institute of Standards and Technology (NIST), SP 800-82, sept. 2008. Final public draft.

[12] L. Pièvre-Cambacédès et P. Sitbon, « Cryptographic key management for SCADA systems - issues and perspectives », dans *Proceedings of the 2nd International Conference on Information Security and Assurance (ISA'08)*, Pusan, Corée, p. 156–161, avril 2008.

[13] International Atomic Energy Agency (IAEA), « Safety glossary : terminology used in nuclear safety and radiation protection ». Ref. STI/PUB/1290, édition 2007.

[14] J.-C. Laprie, J. Arlat, J.-P. Blanquart, A. Costes, Y. Crouzet, Y. Deswarte, J.-C. Fabre, H. Guillermain, M. Kaâniche, K. Kanoun, C. Mazet, D. Powell, C. Rabéjac et P. Thévenod, *Guide de la sûreté de fonctionnement*. Cépaduès Éditions, Toulouse, 2e édition, 1996.

[15] J.-C. Laprie, « Sûreté de fonctionnement des systèmes : concepts de base et terminologie », *Revue de l'Electricité et de l'Électronique*, vol. 11, p. 95–105, déc. 2004.

[16] A. Avizienis, J.-C. Laprie, B. Randell et C. Landwehr, « Basic concepts and taxonomy of dependable and secure computing », *IEEE Transactions on Dependable and Secure Computing*, vol. 1, no 1, p. 11–33, 2004.

[17] International Electrotechnical Commission (IEC), « International electrotechnical vocabulary – chapter 191 : Dependability and quality of service ». IEC 60500-191 et premier amendement, mars 1999.

[18] N. Leveson, « Software safety : Why, what, and how », *ACM Computing Surveys*, vol. 18, p. 125–163, juin 1986.

[19] N. G. Leveson, *Safeware : System Safety and Computers*. Addison-Wesley Professional, 1995.

[20] E. Schoitsch, « Design for safety and security of complex embedded systems : a unified approach », dans *Proceedings of the NATO Advanced Research Workshop on Cyberspace Security and Defense : Research Issues*, Gdansk, Pologne, p. 161–174, sept. 2004.

[21] M. B. Line, O. Nordland, L. Røstad et I. A. Tøndel, « Safety vs. security ? », dans *Proceedings of the 8th International Conference on Probabilistic Safety Assessment and Management (PSAM 2006)*, Nouvelle-Orléans, États-Unis, mai 2006.

[22] B. Schneier, « Attack trees : Modeling security threats », *Dr. Dobb's Journal*, vol. 12, n° 24, p. 21–29, 1999.

[23] Direction Centrale de la Sécurité des Systèmes d'Information (DCSSI), « La défense en profondeur appliquée aux systèmes d'information ». Memento du SGDN/DCSSI, juil. 2004. Version 1.1.

[24] G. Deleuze, E. Châtelet, L. Pièvre-Cambacédès et P. Laclémence, « Les paradoxes de la sécurité industrielle », dans *Actes (électroniques) du 1er Workshop Interdisciplinaire sur la Sécurité Globale (WISG'07)*, Troyes, France, jan. 2007.

[25] L. Pièvre-Cambacédès et M. Bouissou, « Modeling safety and security interdepedencies with BDMP (Boolean logic Driven Markov Processes) », dans *IEEE International Conference on Systems, Man, and Cybernetics (SMC 2010)*, Istanbul, Turquie, oct. 2010. Accepté pour publication.

[26] D. P. Eames et J. Moffett, « The integration of safety and security requirements », dans *Proceedings of the 18th International Conference on Computer Safety, Reliability and Security (SAFECOMP'99), LNCS1698*, Toulouse, France, p. 468–480, sept. 1999.

[27] S. Lautieri, D. Cooper et D. Jackson, « SafSec : Commonalities between safety and security assurance », dans *Proceedings of the 13th Safety Critical Systems Symposium (SSS'05)*, Southampton, Royaume-Uni, p. 65–75, fév. 2005.

[28] Energy Facility Contractors Group (EFCOG), « Topical Report on Security and Safety Integration (TROSSI) », sept. 2006.

[29] T. Novak et A. Treytl, « Functional safety and system security in automation systems - a life cycle model », dans *Proceedings of the 13th IEEE Conference on Emerging Technologies and Factory Automation (ETFA'08)*, Hambourg, Allemagne, p. 311–318, sept. 2008.

[30] U.S. Nuclear Regulatory Commission (NRC), « Managing the safety/security interface ». Regulatory Guide 5.74, juin 2009.

[31] B. Littlewood et L. Strigini, « Redundancy and diversity in security », dans *Proceedings of the European Symposium on Research in Computer Security (ESORICS'04), LNCS 3193*, Sophia-Antipolis, France, p. 423–438, sept. 2004.

[32] S. Zafar et G. R. Dromey, « Integrating safety and security requirements into design of an embedded system », dans *Proceedings of the 12th Asia-Pacific Software Engineering Conference (APSEC'05)*, Taipei, Taiwan, p. 629–636, déc. 2005.

[33] G. Levitin et K. Hausken, « Redundancy vs. protection in defending parallel systems against unintentional and intentional impacts », *IEEE Transactions on Reliability*, vol. 58, p. 679–690, déc. 2009.

[34] M. Sun, S. Mohan, L. Sha et C. Gunter, « Addressing safety and security contradictions in cyber-physical systems », dans *Proceedings of the 1st Workshop on Future Directions in Cyber-Physical Systems Security (CPSSW'09)*, Newark, États-Unis, juil. 2009.

[35] European Network of Transmission System Operators for Electricity, « UCTE operation handbook - glossary ». Version 2.2, juil. 2004.

[36] L. Pièvre-Cambacédès et C. Chaudet, « The SEMA referential framework : avoiding ambiguities when dealing with security and safety issues », dans *4th Annual IFIP WG 11.10 International Conference on Critical Infrastructure Protection (CIP 2010)*, Washington D.C., États-Unis, mars 2010. Hors proceedings (sélection pour publication journal).

[37] L. Pièvre-Cambacédès et C. Chaudet, « The SEMA referential framework : avoiding ambiguities in the terms "security" and "safety" », *International Journal of Critical Infrastructure Protection*, vol. 3, n° 2, p. 55–66, 2010.

[38] M. Van Der Meulen, *Definitions for Hardware and Software Safety Engineers*. Springer, 1re édition, avril 2000.

[39] A. Burns, J. McDermid et J. Dobson, « On the meaning of safety and security », *The Computer Journal*, vol. 35, n° 1, p. 3–15, 1992.

[40] International Organization for Standardization (ISO) et International Electrotechnical Commission (IEC), « Safety aspects – guidelines for their inclusion in standards ». ISO/IEC Guide 51, 2e édition, jan. 1999.

[41] J. Arlat, Y. Crouzet, Y. Deswarte, J.-C. Fabre, J.-C. Laprie et D. Powell, *Encyclopédie de l'informatique et des systèmes d'information*, chap. Tolérance aux fautes, p. 240–270. Paris, France : Vuibert, 2006.

[42] Y. Laarouchi, *Sécurités (immunité et innocuité) des architectures ouvertes à niveaux de criticité multiples : application en avionique.* Thèse de doctorat, Institut National des Sciences Appliquées (INSA) de Toulouse et Laboratoire d'Analyse et d'Architecture des Systèmes du CNRS (LAAS), 2009.

[43] J. Rushby, « Critical system properties : Survey and taxonomy », *Reliability Engineering & System Safety*, vol. 43, no 2, p. 189–219, 1994.

[44] D. G. Firesmith, « Common concepts underlying safety, security, and survivability engineering », Rapport Technique CMU/SEI-2003-TN-033, Université Carnegie Mellon, déc. 2003.

[45] D. M. Nicol, W. H. Sanders et K. S. Trivedi, « Model-based evaluation : From dependability to security », *IEEE Transactions on Dependable and Secure Computing*, vol. 1, no 1, p. 48–65, 2004.

[46] M. Al-Kuwaiti, N. Kyriakopoulos et S. Hussein, « A comparative analysis of network dependability, fault-tolerance, reliability, security, and survivability », *IEEE Communications Surveys and Tutorials*, vol. 11, no 2, p. 106–124, 2009.

[47] International Air Transport Association (IATA), « IATA security manual ». 6e édition, avril 2010.

[48] International Civil Aviation Organization (ICAO), « Security manual for safeguarding civil aviation against acts of unlawful interference ». Doc. 8973, 2002.

[49] International Atomic Energy Agency (IAEA), « Computer security at nuclear facilities ». Manuel de référence, version préliminaire non publiée, 2009.

[50] International Atomic Energy Agency (IAEA), « Fundamental safety principles ». Safety Fundamentals No. SF-1, 2006.

[51] International Atomic Energy Agency (IAEA), « Software for computer based systems important to safety in nuclear power plants ». Safety Guide No. NS-G-1.1, sept. 2000.

[52] International Electrotechnical Commission (IEC), « Nuclear power plants – instrumentation and control important to safety – requirements for computer security programmes ». IEC Committee Draft 62645), avril 2010.

[53] International Atomic Energy Agency (IAEA), « Instrumentation and control systems important to safety in nuclear power plants ». Safety Guide No. NS-G-1.3, mars 2002.

[54] International Atomic Energy Agency (IAEA), « Safety of nuclear power plants : Design ». Safety Guide No. NS-R-1, sept. 2000.

[55] U.S. Office of the Federal Register, « Title 10 : Energy, part 73.54 : Protection of digital computer and communication systems and networks ». Code of Federal Regulations (10 CFR 73.54).

[56] International Atomic Energy Agency (IAEA), International Nuclear Safety Group (INSAG), « Basic safety principles for nuclear power plants ». 75-INSAG-3, Rev. 1, oct. 1999.

[57] U.S. Nuclear Regulatory Commission (NRC), « Criteria for use of computers in safety systems of nuclear power plants ». Regulatory Guide 1.152, Revision 2, jan. 2006.

[58] International Electrotechnical Commission (IEC), « Nuclear power plants – instrumentation and control for systems important to safety – general requirements for systems ». IEC 61513, mars 2001.

[59] International Electrotechnical Commission (IEC), « Nuclear power plants – instrumentation and control systems important to safety – classification of instrumentation and control functions ». IEC 61226, 2e édition, fév. 2005.

[60] U.S. Nuclear Regulatory Commission (NRC), « Cyber security programs for nuclear facilities ». Regulatory Guide 5.71, jan. 2010.

[61] International Electrotechnical Commission (IEC), « Nuclear power plants – instrumentation and control systems important to safety – software aspects for computer-based systems performing category A functions ». IEC 60880, 2e édition, mai 2006.

[62] International Electrotechnical Commission (IEC), « Nuclear power plants – instrumentation and control important for safety – software aspects for computer-based systems performing category B or C functions ». IEC 62138, jan. 2001.

[63] Institute of Electrical and Electronics Engineers (IEEE), « IEEE standard criteria for security systems for nuclear power generating stations ». IEEE Std 692-2010, fév. 2010.

[64] Korea Institute of Nuclear Safety (KINS), « Cyber security of digital instrumentation and control systems in nuclear facilities ». Guide réglementaire KINS/GT-N09-DR, jan. 2007.

[65] Institute of Electrical and Electronics Engineers (IEEE), « IEEE standard criteria for safety systems for nuclear power generating stations ». IEEE Std 603-1998, juil. 1998.

[66] Institute of Electrical and Electronics Engineers (IEEE), « IEEE standard criteria for digital computers in safety systems of nuclear power generating stations ». IEEE Std 7-4.3.2-2003, déc. 2003.

[67] International Electrotechnical Commission (IEC), « Power systems management and associated information exchange – data and communications security ». Série de Technical Specifications (TS) IEC 62351 (parties 1 à 8), 2007 à 2009.

[68] North American Electric Reliability Council (NERC), « Reliability standards for the bulk electric systems of North America », nov. 2009.

[69] North American Electric Reliability Council (NERC), « Cyber security standards ». CIP-002-1 à CIP-009-1, 2006.

[70] U.S. National Institute of Standards and Technology (NIST), « Smart grid cyber security – strategy and requirements ». NISTIR 7628 (draft), sept. 2009.

[71] Institute of Electrical and Electronics Engineers (IEEE), « IEEE guide for electric power substation physical and electronic security ». IEEE Std 1402-2000, jan. 2000.

[72] Institute of Electrical and Electronics Engineers (IEEE), « IEEE standard for substation intelligent electronic devices (IEDs) cyber security capabilities ». IEEE Std 1686-2007, déc. 2007.

[73] Institute of Electrical and Electronics Engineers (IEEE), « IEEE trial use standard for a cryptographic protocol for cyber security of substation serial links ». IEEE P1711 (draft), 2007.

[74] Commission européenne, « Regulation (EC) No 2320/2002 of the European parliament and of the council of 16 december 2002 establishing common rules in the field of civil aviation security ». Journal officiel de l'Union européenne, déc. 2002.

[75] International Civil Aviation Organization (ICAO), « Safety oversight audit manual ». Doc. 9735, 2e édition, 2006.

[76] Radio Technical Commission for Aeronautics (RTCA) et European Organisation for Civil Aviation Equipment (EUROCAE), « Software considerations in airborne systems and equipment certification ». DO-178B/ED-12B, jan. 1992.

[77] Maison Blanche, « National strategy for aviation security ». Directive présidentielle états-unienne, mars 2007.

[78] European Organisation for the Safety of Air Navigation, « ESARR 4 - risk assessment and mitigation in ATM », avril 2001.

[79] European Organisation for the Safety of Air Navigation, « ESARR 6 - software in ATM systems », nov. 2003.

[80] Commission européenne, « Regulation (EC) No 216/2008 of the European parliament and of the Council on common rules in the field of civil aviation and establishing a European aviation safety agency ». Journal officiel de l'Union européenne, mars 2008.

[81] U.S. Office of the Federal Register, « Title 49 : Transportation, parts 1520 and 1580 : Rail transportation security ». Code of Federal Regulations (14 CFR 1520 & 1580).

[82] International Electrotechnical Commission (IEC), « Railway applications – specification and demonstration of reliability, availability, maintainability and safety (RAMS) ». IEC 62278, sept. 2002.

[83] International Electrotechnical Commission (IEC), « Railway applications – communications, signalling and processing systems – software for railway control and protection systems ». IEC 62279, sept. 2002.

[84] U.S. Office of the Federal Register, « Title 14 : Aeronautics and space, part 1203 : Information security program ». Code of Federal Regulations (14 CFR 1203).

[85] U.S. Office of the Federal Register, « Title 14 : Aeronautics and space, part 1203a : NASA security areas ». Code of Federal Regulations (14 CFR 1203a).

[86] U.S. Office of the Federal Register, « Title 14 : Aeronautics and space, part 1203b : Security programs ; arrest authority and use of force by NASA security force personnel ». Code of Federal Regulations (14 CFR 1203a).

[87] European Cooperation for Space Standardization (ECSS), « Glossary of terms ». ECSS-P-001B, juil. 2004.

[88] U.S. National Aeronautics and Space Administration (NASA), « Standard for integrating applications into the NASA access management, authentication, and authorization infrastructure ». EA-STD-0001, juil. 2008.

[89] U.S. National Aeronautics and Space Administration (NASA), « NASA security program procedural requirements w/change 2 ». NASA Procedural Requirements 1600.1, nov. 2004.

[90] U.S. National Aeronautics and Space Administration (NASA), « Software safety standard ». NASA-STD-8719.13B w/Change 1, juil. 2004.

[91] Oljeindustriens Landsforening (OLF), « Information security baseline requirements for process control, safety, and support ICT systems ». OLF Guideline No. 104, déc. 2006.

[92] International Organization for Standardization (ISO), « Petroleum and natural gas industries – offshore production installations – basic surface process safety systems ». ISO 10418, 2e édition, oct. 2003.

[93] American Petroleum Institute (API), « Pipeline SCADA security ». STD 1164, juil. 2009.

[94] International Organization for Standardization (ISO), « Petroleum and natural gas industries – control and mitigation of fires and explosions on offshore production installations – requirements and guidelines ». ISO 13702, mars 1999.

[95] International Organization for Standardization (ISO), « Petroleum and natural gas industries – offshore production installations – guidelines on tools and techniques for hazard identification and risk assessment ». ISO 17776, oct. 2000.

[96] NORSOK, « Technical safety ». NORSOK Standard S-001, jan. 2000.

[97] NORSOK, « Safety and automation system (SAS) ». NORSOK Standard I-002, mai 2001.

[98] Oljeindustriens Landsforening (OLF), « Application of IEC 61508 and IEC 61511 in the Norwegian petroleum industry ». OLF Guideline No. 70, oct. 2004.

[99] Oljeindustriens Landsforening (OLF), « Recommended guidelines : Common model for safe job analysis (SJA) ». OLF Guideline No. 90, mars 2006.

[100] U.S. Office of the Federal Register, « Title 6 : Homeland security, part 27 : Chemical facility anti-terrorism standards ». Code of Federal Regulations (6 CFR 27).

[101] American Institute of Chemical Engineers (AIChE), Center for Chemical Process Safety (CCPS), « Combined glossary of terms », mars 2005.

[102] U.S. Department of Homeland Security (DHS), « Risk-based performance standards guidance ». Chemical Facility Anti-Terrorism Standards, mai 2009.

[103] North Atlantic Treaty Organization (NATO) Standardization Agency (NSA), « NATO glossary of terms and definitions (English and French) ». AAP-6, 2009.

[104] North Atlantic Treaty Organization (NATO), « NATO R & M terminology applicable to ARMPs ». AMRP-7, août 2008.

[105] U.S. Department of Defense (DoD), « Standard practice for system safety ». MIL-STD-882D, jan. 1993.

[106] U.K. Ministry of Defence (MoD), Directorate of Standardization, « Safety management requirements for defence systems – part 1 – requirements ». MoD-Def-Stan-00-56/1, juin 2007.

[107] U.K. Ministry of Defence (MoD), Directorate of Standardization, « Safety management requirements for defence systems – part 2 – guidance on establishing a means of complying with part 1 ». MoD-Def-Stan-00-56/2, juin 2007.

[108] International Electrotechnical Commission (IEC), « Industrial communication networks – network and system security – part 1-1 : Terminology, concepts and models ». Technical Specification IEC/TS 62443-1-1, juil. 2009.

[109] International Electrotechnical Commission (IEC), « Industrial communication networks – network and system security – part 3-1 : Security technologies for industrial automation and control systems ». Technical Specification IEC/TS 62443-1-1, juil. 2009.

[110] International Electrotechnical Commission (IEC), « Functional safety of electrical/electronic/programmable electronic safety-related systems ». Série de normes internationales IEC 61508 (parties 1 à 7), 1998 à 2005.

[111] U.S. National Institute of Standards and Technology (NIST), « Security controls for federal information systems and organizations ». NIST Special Publication 800-53, revision 3, août 2009.

[112] American National Standards Institute (ANSI) et International Society of Automation (ISA), « Security for industrial automation and control systems – part 1 : Terminology, concepts, and models ». ANSI/ISA–99.00.01, oct. 2007. (Équivalent à l'IEC/TS62433-1-1 :2009).

[113] UK Centre for the Protection of National Infrastructure (CPNI), « Process control and SCADA security, good practice guide », juin 2008. Version 2 (guide en 8 parties).

[114] International Organization for Standardization (ISO) et International Electrotechnical Commission (IEC), « Information technology – security techniques – information security management systems – overview and vocabulary ». ISO/IEC 27000, mai 2009.

[115] International Organization for Standardization (ISO) et International Electrotechnical Commission (IEC), « Information technology – security techniques – information security management systems ». ISO/IEC 27001, déc. 2007.

[116] International Electrotechnical Commission (IEC), « Information technology equipment – safety – part 1 : General requirements ». IEC 60950-1, déc. 2005. 2e édition.

[117] International Organization for Standardization (ISO) et International Electrotechnical Commission (IEC), « Information technology – security techniques – code of practice for information security management ». ISO/IEC 27002, juin 2005.

[118] International Organization for Standardization (ISO) et International Electrotechnical Commission (IEC), « Information technology – Security techniques – Information security risk management ». ISO/IEC 27005, juin 2008.

[119] International Organization for Standardization (ISO) et International Electrotechnical Commission (IEC), « Information technology – security techniques – management of information and communications technology security – part 1 : Concepts and models for information and communications technology security management ». ISO/IEC 13335, nov. 2004.

[120] R. Shirey et Internet Engineering Task Force (IETF), « Internet security glossary ». RFC 4949, août 2007. Version 2.

[121] International Telecommunication Union (ITU-T), « Information technology – security techniques – information security management guidelines for telecommunications organizations based on ISO/IEC 27002 ». ITU-T X.1051, fév. 2008. 2e édition.

[122] U.S. National Institute of Standards and Technology (NIST), « Standards for security categorization of federal information and information systems ». FIPS PUB 199, fév. 2004.

[123] U.S. National Institute of Standards and Technology (NIST), « Glossary of key information security terms ». NIST IR 7298, avril 2006.

[124] U.S. Committee on National Security Systems (CNSS), « National information assurance (IA) ». CNSS Instruction No. 4009, juin 2006.

[125] International Organization for Standardization (ISO) et International Electrotechnical Commission (IEC), « Guidelines for the inclusion of security aspects in standards ». ISO/IEC Guide 81, déc. 2009. Draft.

[126] International Organization for Standardization (ISO) et International Electrotechnical Commission (IEC), « Standardization and related activities – general vocabulary ». ISO/IEC Guide 2, nov. 2004. 8e édition.

[127] International Electrotechnical Commission (IEC), « Power systems management and associated information exchange – data and communications security part 1 : Communication network and system security – introduction to security issues ». IEC 62351-1, mai 2007.

[128] L. Pièvre-Cambacédès et C. Chaudet, « Disentangling the relations between safety and security », dans *Proceedings of the 9th WSEAS International Conference on Applied Informatics and Communications (AIC'09)*, Moscou, Russie, p. 156–161, août 2009.

[129] G. N. Ericsson, « Information security for Electric Power Utilities (EPUs) - CIGRÉ developments on frameworks, risk assessment, and technology », *IEEE Transactions on Power Delivery*, vol. 24, no 3, p. 1174–1181, 2009.

[130] V. Madani et R. King, « Strategies to meet grid challenges for safety and reliability », *International Journal of Reliability and Safety*, vol. 2, no 1-2, p. 146–165, 2008.

[131] U.S. National Grid, « Electric safety ». Site web (consulté le 8 juin 2010) - http://www.nationalgridus.com/masselectric/safety_electric.asp.

[132] American Naional Standards Institute (ANSI) et Institute of Electrical and Electronics Engineers (IEEE), « National electrical safety code (NESC) ». Accredited Standards Committee C2-2007, 2007.

[133] S. Abraham, « National transmission grid study ». U.S. Department of Energy, mai 2002.

[134] CIGRÉ et Institute of Electrical and Electronics Engineers (IEEE), « Definition and classification of power system stability ». Technical Brochure No. 231, juin 2003.

[135] A. Gheorghe, M. Masera, M. Weijnen et L. De Vries, éds, *Critical Infrastructures at Risk : Securing the European Electric Power System*. Springer, 2006.

[136] C. Tranchita, *Risk Assessment for Power System Security with Regard to Intentional Events*. Thèse de doctorat, Institut Polytechnique de Grenoble et Université de Los Andes, 2008.

[137] « Strategic research agenda for Europe's electricity networks of the future ». Commission européenne, EUR 22580, 2007.

[138] U.S. National Institute of Standards and Technology (NIST), « NIST framework and roadmap for smart grid interoperability standards ». Draft Publication, Standards Release 1.0, sept. 2009.

[139] N. Hadjsaid, J.-C. Sabonnadière et J.-P. Angelier, « La distribution électrique de l'avenir : les Smart Grids », *Revue de l'Électricité et de l'Électronique (REE)*, jan. 2010.

[140] « Google Map of AMI & smart metering programmes across the world ». Carte électronique maintenue par l'Energy Retail Association du Royaume Uni, http://tinyurl.com/AMI-projects-worldmap (consultée le 30 déc. 2009).

[141] « Loi 2006-686 du 13 juin 2006 relative à la transparence et à la sécurité en matière nucléaire ». Journal officiel de la République française du 14 juin 2006, juin 2006.

[142] J. Jalouneix, P. Cousinou, J. Couturier et D. Winter, « Approche comparative entre sûreté et sécurité nucléaires », Rapport Technique 2009/117, Institut de Radioprotection et de Sûreté Nucléaire (IRSN), avril 2009.

[143] Nuclear Energy Institute, « Cyber security plan for nuclear power reactors (revision 6) ». NEI 08-09, avril 2010.

[144] Commission européenne, « Protecting Europe from large scale cyber-attacks and disruptions : enhancing preparedness, security and resilience ». Communications SEC(2009)399 et SEC(2009)400, mars 2009.

[145] U.S. Department of Homeland Security (DHS), « Roadmap to secure control systems in the water sector ». Water Sector Coordinating Council Cyber Security WG, mars 2008.

[146] U.S. Department of Homeland Security (DHS), « LOGIIC - linking the oil and gas industry to improve cybersecurity », sept. 2006.

[147] International Electrotechnical Commission (IEC), « Risk management – vocabulary – guidelines for use in standards ». IEC Guide 73, juin 2002.

[148] U.S. Federal Aviation Administration (FAA), « FAA system safety handbook », déc. 2000.

[149] U.S. National Institute of Standards and Technology (NIST), « Minimum security requirements for federal information and information systems ». FIPS PUB 200, mars 2006.

[150] S. Kaplan et B. Garrick, « On the quantitative definition of risk », *Risk Analysis*, vol. 1, n° 1, p. 11–27, 1981.

[151] L. Magne et D. Vasseur, éds, *Risques industriels. Complexité, incertitudes et décision : une approche interdisciplinaire*. TEC & DOC, Collection EDF R&D, Lavoisier, 2006.

[152] E. Zio, *An introduction to the basics of reliability and risk analysis*, vol. 13 dans *Series on Quality, Reliablity and Engineering Statistics*. World Scientific Publishing, 2007.

[153] International Atomic Energy Agency (IAEA), « The physical protection of nuclear material and nuclear facilities ». INFCIRC/225/Rev.4, juin 1999.

[154] R. G. Johnston, « Adversarial safety analysis : Borrowing the methods of security vulnerability assessments », *Journal of Safety Research*, vol. 35, p. 245–248, 2004.

[155] S. Tom, D. Christiansen et D. Berrett, « Recommended pratice for patch management of control systems ». DHS Control System Security Program (CSSP) Recommended Practice, déc. 2008.

[156] B. Schneier, *Secrets & Lies : Digital Security in a Networked World*. John Wiley & Sons, 2000.

[157] J. Viega et G. Mc Graw, *Building Secure Software*. Addison-Wesley, 2002.

[158] R. J. Anderson, *Security Engineering : A Guide to Building Dependable Distributed Systems*. John Wiley & Sons, 2e édition, avril 2008.

[159] G. Deleuze, E. Châtelet, P. Laclémence, J. Piwowar et B. Affeltranger, « Are safety and security in industrial systems antagonistic or complementary issues ? », dans *Proceedings of the17th European Safety and Reliability Conference (ESREL'08)*, Valence, Espagne, sept. 2008.

[160] A. J. Menezes, P. C. van Oorschot et S. A. Vanstone, *Handbook of Applied Cryptography*. CRC Press, 2001.

[161] J. Jürjens, « Composability of secrecy », dans *Proceedings of the 1st International Workshop on Methods, Models, and Architectures for Network Security (MMM-ACNS'01), LNCS 2052*, Saint-Pétersbourg, Russie, p. 28–38, mai 2001.

[162] J. Alves-Foss, « Computer security aspects of dependable avionics systems », dans *Proceedings of National Workshop on Aviation Software Systems : Design for Certifiably Dependable Systems (A Workshop on Research Directions and State of Practice of High Confidence Software Systems) NITRD HCSS-AS*, Alexandria, États-Unis, oct. 2006.

[163] L. Yang et S. Yang, « A framework of security and safety checking for internet-based control systems », *International Journal of Computer Security*, vol. 1, n° 1/2, p. 185–200, 2007.

[164] J. N. Sorensen, « Safety culture : a survey of the state-of-the-art », *Reliability Engineering & System Safety*, vol. 76, n° 2, p. 189–204, 2002.

[165] « BP Texas City, refinery explosion and fire (15 killed, 180 injured) », Rapport d'enquête 2005-04-I-TX, U.S. Chemical Safety and Hazard Investigation Board, mars 2007.

[166] C. W. Choo, « Information failures and organizational disasters », *MIT Sloan Management Review*, vol. 46, n° 3, p. 7–10, 2005.

[167] B. Schneier, « The psychology of security », dans *Proceedings of the 1st International Conference on Cryptology in Africa (AfricaCrypt 2008), LNCS 5023*, Casablanca, Maroc, p. 50–79, juin 2008.

[168] B. Schneier, *Beyond Fear : Thinking Sensibly About Security in an Uncertain World*. Springer, 2003.

[169] R. J. Anderson, *Security Engineering : A Guide to Building Dependable Distributed Systems*, chap. 2 - Usability and Pyschology, p. 17–62. John Wiley & Sons, 2e édition, avril 2008.

[170] International Atomic Energy Agency (IAEA), « Nuclear security culture ». IAEA Nuclear Security Series No. 7, Implementing Guide, 2008.

[171] K. Mitnick, W. Simon et S. Wozniak, *The art of deception*. Wiley, 2002.

[172] J. Dobson, « New security paradigms : what other concepts do we need as well ? », dans *Proceedings of the 1993 Workshop on New Security Paradigms (NSPW'93)*, Little Compton, États-Unis, p. 7–18, août 1993.

[173] International Atomic Energy Agency (IAEA), « The International Nuclear Event Scale (INES) user's manual », fév. 2001. Préparation conjointe AIEA et OCDE.

[174] G. Deleuze, « Un cadre conceptuel pour la comparaison sûreté et sécurité de filières industrielles », dans *Actes (électroniques) du 2e Workshop Interdisciplinaire sur la Sécurité Globale (WISG'08)*, Troyes, France, jan. 2008.

[175] O. Nordland, « Safety and security - two sides of the same medal ». European CIIP Newsletter (ECN), vol. 3, no. 2, juin 2007.

[176] M. Bouissou, *Gestion de la complexité dans les études quantitatives de sûreté de fonctionnement de systèmes*. TEC & DOC, Collection EDF R&D, Lavoisier, 2008.

[177] B. Littlewood, S. Brocklehurst, N. Fenton, P. Mellor, S. Page, D. Wright, J. Dobson, J. McDermid et D. Gollmann, « Towards operational measures of computer security », *Journal of Computer Security*, vol. 2, p. 211–229, 1993.

[178] B. Littlewood, « Dependability assessment of software-based systems : State of the art », dans *Proceedings of the 27th International Conference on Software Engineering (ICS'05)*, Saint-Louis, États-Unis, p. 6–7, mai 2005.

[179] I. Bloch, « Incertitude, imprécision et additivité en fusion de données : point de vue historique », *Traitement du Signal*, vol. 13, n° 4, p. 267–288, 1996.

[180] M. Marseguerra et E. Zio, « Basics of the Monte-Carlo method with application to system reliability ». LiLoLe Publishing, Hagen, Allemagne, 2002.

[181] A. Marquez, A. Sanchez et B. Iung, « Monte-Carlo-based assessment of system availability : A case study for cogeneration plants », *Reliability Engineering & System Safety*, vol. 88, n° 3, p. 273–289, 2005.

[182] D. E. Peplow, C. D. Sulfredge, R. L. Sanders, R. H. Morris et T. A. Hann, « Calculating nuclear power plant vulnerability using integrated geometry and event/fault-tree models », *Nuclear Science and Engineering*, vol. 146, n° 1, p. 71–87, 2004.

[183] C. R. H. Morris, D. Sulfredge, R. L. Sanders et H. S. Rydell, « Using the VISAC program to calculate the vulnerability of nuclear power plants to terrorism », *International Journal of Nuclear Governance, Economy and Ecology*, vol. 1, n° 2, p. 193–211, 2006.

[184] G. Stoneburner, « Toward a unified security-safety model », *IEEE Computer*, vol. 39, p. 96–97, août 2006.

[185] N. Leveson, « White paper on approaches to safety engineering ». Disponible en ligne sur le site de l'auteur (sunnyday.mit.edu/caib/concepts.pdf), avril 2003.

[186] International Organization for Standardization (ISO) et International Electrotechnical Commission (IEC), « Risk management – risk assessment techniques ». IEC/ISO 31010, nov. 2009.

[187] International Electrotechnical Commission (IEC), « Gestion de la sûreté de fonctionnement partie 3-1 : Guide d'application – techniques d'analyse de la sûreté de fonctionnement – guide méthodologique ». IEC 60300-3-1, 2003.

[188] M. Rausand et A. Høyland, *System Reliability Theory : Models and Statistical Methods*. Wiley, 2e édition, 2004.

[189] International Organization for Standardization (ISO) et International Electrotechnical Commission (IEC), « Analysis techniques for system reliability – procedure for failure mode and effects analysis (FMEA) ». IEC/ISO 60812, jan. 2006.

[190] The Chemical Industry Safety and Health Council of the Chemical Industries Association (CISHEC), « A guide to hazard and operability studies », 1977.

[191] T. Kletz, *HAZOP and HAZAN : Identifying and Assessing Process Industry Hazards*. Institution of Chemical Engineers, 4e édition, 2002.

[192] International Electrotechnical Commission (IEC), « Hazard and operability studies (HAZOP) – application guide ». IEC 61882, mai 2001. 1re édition.

[193] D. J. Pumfrey, *The Principled Design of Computer System Safety Analyses*. Thèse de doctorat, Université de York, 1999.

[194] F. Redmill, M. Chudleigh et J. Catmur, *System Safety : HAZOP and Software HAZOP*. Wiley, avril 1999.

[195] J. P. Rankin et C. F. White, « Sneak circuit analysis handbook », Rapport Technique D2-118341-1 / NASA-CR-108721, U.S. National Aeronautics and Space Administration (NASA), 1970.

[196] U.S. National Aeronautics and Space Administration (NASA), « Sneak circuit analysis guideline for electromechanical systems ». Preferred Reliability Practices, Practice No PD-AP-1314, oct. 2005.

[197] European Cooperation for Space Standardization (ECSS), « Sneak analysis ». ECSS-Q-TM-40-04 Part 1A and Part 2A, avril 2010.

[198] J. Miller, « Sneak circuit analysis for the common man », Rapport Technique RADC-TR-89-223, Rome Air Development Center, oct. 1989.

[199] J. P. Rankin, « Sneak circuit analysis », *Nuclear Safety*, vol. 14, n° 5, p. 461–469, 1973.

[200] Society of Automotive Engineers (SAE), « Guidelines and methods for conducting the safety assessment process on civil airborne systems and equipment ». ARP4761, déc. 1996.

[201] International Electrotechnical Commission (IEC), « Analysis techniques for dependability – reliability block diagram and boolean methods ». IEC 61078, jan. 2006. 2e édition.

[202] S. Distefano et A. Puliafito, « Reliability and availability analysis of dependent-dynamic systems with DRBDs », *Reliability Engineering & System Safety*, vol. 94, p. 1369–1498, sept. 2009.

[203] C. A. Ericson, « Fault tree analysis - a history », dans *Proceedings of the 17th International System Safety Conference (ISSC)*, Orlando, États-Unis, août 1999.

[204] W. Vessely, F. F. Glodberg, N. H. Roberts et D. F. Haasl, « Fault tree handbook ». U.S. Nuclear Regulatory Commission (NRC), NUREG-0492, jan. 1981.

[205] M. Stamatelatos, W. Vesely, J. Dugan, J. Fragola, J. I. Minarick et J. Railsback, « Fault tree handbook with aerospace applications ». U.S. National Aeronautics and Space Administration (NASA) Handbook, août 2002. Version 1.1.

[206] N. G. Leveson et P. R. Harvey, « Software fault tree analysis », *The Journal of Systems and Software*, vol. 3, n° 2, p. 173–181, 1983.

[207] International Electrotechnical Commission (IEC), « Fault tree analysis (FTA) ». IEC 61025, déc. 2006. 2e édition.

[208] S. Contini et V. Matuzas, « ASTRA 3.0 : Logical and probabilistic analysis methods », Rapport Technique JRC56318, European Commission Joint Research Centre (JRC), Institute for the Protection and Security of the Citizen (IPSC), 2010.

[209] U.S. Nuclear Regulatory Commission (NRC), N. Rassmussen et al., « Reactor safety study : an assessment of accident risk in US commercial nuclear plants ». NUREG-75/014 / WASH-1400, oct. 1975.

[210] D. J. Wakefield et S. Epstein, « Fault-tree linking vs. event tree linking », dans *Actes du 12e congrès de fiabilité et maintenabilité de l'IMdR (λμ14)*, Lyon, France, oct. 2002.

[211] American Institute of Chemical Engineers (AIChE), Center for Chemical Process Safety (CCPS), *Guidelines for Hazard Evaluation Procedures*. Wiley, 2008.

[212] D. S. Nielsen, « The cause-consequence diagram method as a basis for quantitative accident analysis », Rapport Technique RISØ-M-1374, Danish Atomic Energy Commission, Danemark, 1971.

[213] D. S. Nielsen, O. Platz et e. B. Rung, « A cause-consequence chart of a redundant protection system », *IEEE Transactions on Reliability*, vol. 24, no 1, p. 8–13, 1975.

[214] G. Burdick et J. Fussell, « On the adaption of cause-consequence analysis to U.S. nuclear power systems reliability and risk assessment », Rapport Technique, Idaho National Engineering Laboratory, 1976.

[215] J. Taylor, « Fault tree and cause consequence analysis for control software validation », Rapport Technique RISØ-M-2326, RISØ Labs, jan. 1982.

[216] Rail Safety and Standards Board, « Engineering Safety Management Volume 1 & 2 ». The Yellow Book, 2007. Issue 4.

[217] J. D. Andrews et L. M. Ridley, « Application of the cause-consequence diagram method to static systems », *Reliability Engineering & System Safety*, vol. 75, no 1, p. 47–58, 2002.

[218] H. Langseth et L. Portinale, « Bayesian networks in reliability », *Reliability Engineering & System Safety*, vol. 92, p. 92–108, 2007.

[219] J. Pearl, « Bayesian networks : a model of self-activated memory for evidential reasoning. », Rapport Technique CSD-850017, Université de Californie, Los Angeles (UCLA), avril 1985.

[220] T. Bayes, « Essay towards solving a problem in the doctrine of chances », *The Philosophical Transactions of the Royal Society of London*, 1763.

[221] F. Jensen et T. Nielsen, *Bayesian networks and decision graphs*. Springer, 2e édition, 2007.

[222] A. Bobbio, L. Portinale, M. Minichino et E. Ciancamerla, « Improving the analysis of dependable systems by mapping fault trees into Bayesian networks », *Reliability Engineering & System Safety*, vol. 71, no 3, p. 249–260, 2001.

[223] International Electrotechnical Commission (IEC), « Application of Markov techniques ». IEC 61165, mai 2006. 2e édition.

[224] E. Zio, *Computational Methods for Reliability and Risk Analysis*, vol. 14 dans *Series on Quality, Reliablity and Engineering Statistics*. World Scientific Publishing, 2009.

[225] J. B. Dugan, S. Bavuso et M. Boyd, « Fault trees and sequence dependencies », dans *Proceedings of the Reliability and Maintainability Annual Symposium (RAMS'90)*, Los Angeles, États-Unis, p. 286–293, jan. 1990.

[226] J. B. Dugan, S. Bavuso et M. Boyd, « Dynamic fault tree models for fault tolerant computer systems », *IEEE Transactions on Reliability*, vol. 41, no 3, p. 363–377, 1992.

[227] R. Manian, D. Coppit, K. J. Sullivan et J. B. Dugan, « Bridging the gap between systems and dynamic fault tree models », dans *Proceedings of the 45th Reliability and Maintainability Annual Symposium (RAMS'99)*, Washington D.C., États-Unis, p. 105–111, jan. 1999.

[228] J. B. Dugan, K. J. Sullivan et D. Coppit, « Developing a low-cost high-quality software tool for dynamic fault-tree analysis », *IEEE Transactions on Reliability*, vol. 49, p. 49–59, 2000.

[229] H. Xu, J. B. Dugan et L. Meshkat, « A dynamic fault tree model of a propulsion system », dans *Proceedings of the 8th International Conference on Probabilistic Safety Assessment & Management*, Nouvelle-Orléans, États-Unis, mai 2006.

[230] C. A. Petri, *Kommunikation mit Automaten*. Thèse de doctorat, Technische Hochschule Darmstadt, 1962.

[231] P. Merlin et D. Farber, « Recoverability of communication protocols – implications of a theoretical study », *IEEE Transactions on Communications*, vol. 24, no 9, p. 1036–1043, 1976.

[232] S. Natkin, *Les Réseaux de Petri Stochastiques*. Thèse de doctorat, Conservatoire National des Arts et Métiers (CNAM), 1980.

[233] M. Molloy, *On the integration of delay and troughput in distributed processing models*. Thèse de doctorat, Université de Californie, Los Angeles (UCLA), 1981.

[234] M. A. Marsan, G. Conte et G. Balbo, « A class of generalized stochastic Petri nets for the performance evaluation of multiprocessor systems », *ACM Transactions on Computer Systems (TOCS)*, vol. 2, p. 93–122, mai 1984.

[235] W. H. Sanders et J. F. Meyer, *Lectures on formal methods and performance analysis : first EEF/Euro summer school on trends in computer science*, chap. Stochastic activity networks : formal definitions and concepts, p. 315–343. Springer-Verlag, 2002.

[236] N. G. Leveson et J. L. Stolzy, « Safety analysis using Petri nets », *IEEE Transactions on Software Engineering*, vol. 13, n° 3, p. 386–397, 1987.

[237] É. Châtelet et J.-F. Aubry, « Sûreté de fonctionnement des systèmes de commande - exemple d'application et rappels sur les RdP ». Techniques de l'Ingénieur, S 8263, sept. 2008.

[238] M. Bouissou et J.-L. Bon, « A new formalism that combines advantages of fault-trees and Markov models : Boolean logic driven Markov processes », *Reliability Engineering & System Safety*, vol. 82, p. 149–163, nov. 2003.

[239] M. Bouissou, H. Bouhadana, M. Bannelier et N. Villatte, « Knowledge modelling and reliability processing : Presentation of the FIGARO language and associated tools », dans *Proceedings of the 10th International Conference on Computer Safety, Reliability and Security (SAFECOMP'91)*, Trondheim, Norvège, p. 69–75, nov. 1991.

[240] A. Arnold, G. Point, A. Griffault et A. Rauzy, « The AltaRica formalism for describing concurrent systems », *Fundamenta Informaticae*, vol. 40, n° 2-3, p. 109–124, 1999.

[241] M. Bouissou et C. Seguin, « Comparaison des langages de modélisation AltaRica et FIGARO », dans *Actes du 14e congrès de fiabilité et maintenabilité de l'IMdR ($\lambda\mu$14)*, Lille, France, oct. 2006.

[242] M. Boiteau, Y. Dutuit, A. Rauzy et J. Signoret, « The AltaRica Dataflow language in use : Assessment of production availability of a multistates system », *Reliability Engineering & System Safety*, vol. 91, p. 747–755, 2006.

[243] M. Bouissou, « Automated dependability analysis of complex systems with the KB3 workbench : the experience of EDF R&D », dans *Proceedings of the International Conference on Energy and Environment (CIEM'05)*, Buharest, Roumanie, oct. 2005.

[244] M. Bouissou, « KB3-BDMP, un nouveau couteau suisse pour vos études de fiabilité de systèmes ». Transparents présentés au 16e congrès de fiabilité et maintenabilité de l'IMdR ($\lambda\mu$16), Avignon, France, oct. 2008.

[245] SAE International, « Architecture Analysis & Design Language (AADL) ». AS5506, jan. 2009. Rév. A.

[246] A.-E. Rugina, *Modélisation et évaluation de la sûreté de fonctionnement - De AADL vers les réseaux de Pétri stochastiques*. Thèse de doctorat, Institut National Polytechnique de Toulouse (INPT) et Laboratoire d'Analyse et d'Architecture des Systèmes du CNRS (LAAS), 2007.

[247] P. Haas, *Stochastic Petri nets*. Springer, 2002.

[248] Defense Science Board Task Force on Computer Security, Office of the U.S. Secretary of Defense, « Security controls for computer systems ». http://csrc.nist.gov/publications/history/ware70.pdf, fév. 1970. (ed. W. H. Ware).

[249] U.S. DHS InfoSec Research Council, « A roadmap for cybersecurity research ». http://www.cyber.st.dhs.gov/docs/DHS-Cybersecurity-Roadmap.pdf, nov. 2009.

[250] A. Jaquith, *Security metrics : Replacing fear, uncertainty, and doubt*. Addison-Wesley Professional, 1re édition, avril 2007.

[251] R. B. J. Vaughn, R. Henning et A. Siraj, « Information assurance measures and metrics - state of practice and proposed taxonomy », dans *Proceedings of the 36th Annual Hawaii International Conference on System Sciences (HICSS-36)*, Hawaï, États-Unis, p. 10–20, jan. 2003.

[252] R. M. Savola, « Towards a taxonomy for information security metrics », dans *Proceedings of the 2007 ACM Workshop on Quality of Protection (QoP'07)*, Alexandria, États-Unis, oct. 2007.

[253] A. Hecker, « On system security metrics and the definition approaches », dans *Second International Conference on Emerging Security Information, Systems and Technologies (SECURWARE'08)*, Cap Esterel, France, p. 412–419, août 2008.

[254] W. Jansen, « Directions in security metrics research ». U.S. National Institute of Standards and Technology (NIST), NISTIR 7564, avril 2009.

[255] International Organization for Standardization (ISO) et International Electrotechnical Commission (IEC), « Information technology – security techniques – measurement ». ISO/IEC 27004, déc. 2009.

[256] E. Chew, M. Swanson, K. Stine, N. Bartol, A. Brown et W. Robinson, « Performance measurement guide for information security ». U.S. National Institute of Standards and Technology (NIST), SP 800-55 Rev. 1, juil. 2009.

[257] M. Dacier, *Vers une évaluation quantitative de la sécurité informatique*. Thèse de doctorat, Laboratoire d'Analyse et d'Architecture des Systèmes du CNRS (LAAS), 1994.

[258] International Standardization Organisation (ISO), « Information technology – security techniques – evaluation criteria for IT security ». IEC 15408, parties 1 à 3, 2008 à 2009. Édition 3.0.

[259] ECSI et ISA Secure certification, « Embedded Device Security Assurance (EDSA) brochure ». http://www.isasecure.org/PDFs/ISASecure-EDSA-Certification-March-2010.aspx, mars 2010.

[260] J. R. Anderson et T. Moore, « Information security : where computer science, economics and psychology meet », *Philosophical Transactions of the Royal Society*, vol. 367, p. 2717–2727, juil. 2009.

[261] N. L. Foster, *The application of software and safety engineering techniques to security protocol development*. Thèse de doctorat, Université de York, 2002.

[262] M. Cheminod, I. Cibrario Bertolotti, L. Durante, R. Sisto et A. Valenzano, « Tools for cryptographic protocols analysis : A technical and experimental comparison », *Computer Standards & Interfaces*, vol. 31, n° 5, p. 954–961, 2009.

[263] O. Sheyner, *Scenario Graphs and Attack Graphs*. Thèse de doctorat, Université Carnegie Mellon, 2004.

[264] Agence Nationale de la Sécurité des systèmes d'information (ANSSI), « Expression des Besoins et Identification des Objectifs de Sécurité (EBIOS) - Méthodes de gestion des risques ». http://www.ssi.gouv.fr/IMG/pdf/EBIOS-1-GuideMethodologique-2010-01-25.pdf, jan. 2010.

[265] Agence Nationale de la Sécurité des systèmes d'information (ANSSI), « Expression des Besoins et Identification des Objectifs de Sécurité (EBIOS) - Bases de connaissances ». http://www.ssi.gouv.fr/IMG/pdf/EBIOS-2-BasesDeConnaissances-2010-01-25.pdf, jan. 2010.

[266] S. Evans, D. Heinbuch, E. Kyule, J. Piorkowski et J. Wallner, « Risk-based systems security engineering : stopping attacks with intention », *IEEE Security and Privacy*, vol. 2, n° 6, p. 59–62, 2004.

[267] N. Mead, E. Hough et T. Stehney, « Security quality requirements engineering (SQUARE) methodology », Rapport Technique CMU/SEI-2005-TR-009, Université Carnegie Mellon, 2005.

[268] P. Herzog, « OSSTMM 3.0 LITE - Introduction to the Open Source Security Testing Methodology Manual », août 2008.

[269] Open Web Application Security Project (OWASP) et M. Meucci, « OWASP testing guide », 2008. Version 3.

[270] D. Farmer et E. Spafford, « The COPS security checker system », dans *Proceedings of the Summer Usenix Conference*, Anaheim, États-Unis, p. 165–170, juin 1990.

[271] W. Venema et D. Farmer, « Improving the Security of your site by breaking into it ». Posté sur Usenet, déc. 1993.

[272] J. Wack, M. Tracy et M. Souppaya, « Guideline on network security testing ». U.S. National Institute of Standards and Technology (NIST), SP 800-42, oct. 2003.

[273] M. Howard, J. Pincus et J. M. Wing, « Measuring relative attack surfaces », dans *Proceedings of Workshop on Advanced Developments in Software and Systems Security*, Taipei, Taiwan, p. 109–137, déc. 2003.

[274] P. K. Manadhata, *An Attack Surface*. Thèse de doctorat, Université Carnegie Mellon, déc. 2008.

[275] P. Mell, K. Scarfone et S. Romanosky, « A complete guide to the Common Vulnerability Scoring System (CVSS) », juin 2007. Version 2.

[276] J. D. Weiss, « A system security engineering process », dans *Proceedings of the 14th National Computer Security Conference (NCSC)*, Washington D.C., États-Unis, p. 572–581, oct. 1991.

[277] E. G. Amoroso, *Fundamentals of computer security technology*, chap. 2 - Threat Trees, p. 15–29. Prentice-Hall, États-Unis, 1994.

[278] S. Kumar et E. H. Spafford, « A pattern-matching model for instrusion detection », dans *Proceedings of the 17th National Computer Security Conference (NCSC'94)*, Baltimore, États-Unis, p. 11–21, oct. 1994.

[279] Y. Ho, D. Frincke et D. Tobin, « Planning, Petri nets, and intrusion detection », dans *Proceedings of the 21st National Information System Security Conference (NISCC'98)*, Arlington, États-Unis, p. 346–361, oct. 1998.

[280] S. T. Smith et J. J. Lim, « Risk analysis in computer systems - an automated procedure », *Information Age*, vol. 7, p. 15–18, jan. 1985.

[281] U.S. Department of Defense (DoD), « Standard practice for system safety ». MIL-STD-882D, juin 1988.

[282] M. Howard et D. LeBlanc, *Writing Secure Code*. Microsoft Press, 2e édition, 2002.

[283] C. Salter, O. S. Saydjari, B. Schneier et J. Wallner, « Toward a secure system engineering methodology », dans *Proceedings of the 1998 Workshop on New Security Paradigms (NSPW '98)*, Charlottesville, États-Unis, p. 2–10, sept. 1998.

[284] K. S. Edge, *A Framework for Analyzing and Mitigating the Vulnerabilities of Complex Systems via Attack and Protection Trees*. Thèse de doctorat, Air Force Institute of Technology, juil. 2007.

[285] A. P. Moore, R. J. Ellison et R. C. Linger, « Attack modeling for information security and survivability », Rapport Technique CMU/SEI-2001-TN-001, Université Carnegie Mellon, mars 2001.

[286] T. Tidwell, R. Larson, K. Fitch et J. Hale, « Modeling internet attacks », dans *Proceedings of the 2nd IEEE Systems, Man and Cybernetics Information Assurance Workshop (IAW '01)*, West Point, États-Unis, p. 54–59, juin 2001.

[287] A. Opel, « Design and implementation of a support tool for attack trees », rapport de stage, Université Technique d'Eindhoven, mars 2005.

[288] G. Helmer, J. Wong, M. Slagell, V. Honavar, L. Miller et R. Lutz, « A software fault tree approach to requirements analysis of an intrusion detection system », *Requirements Engineering Journal*, vol. 7, p. 207–220, déc. 2002.

[289] G. Helmer, J. Wong, M. Slagell, V. Honavar, L. Miller, Y. Wang, X. Wang et N. Stakhanova, « Software fault tree and coloured Petri net-based specification, design and implementation of agent-based intrusion detection systems », *International Journal of Information and Computer Security*, vol. 1, no 1/2, p. 109–142, 2007.

[290] S. A. Camtepe et B. Yener, « Modeling and detection of complex attacks », dans *Proceedings of the 3rd International Conference on Security and Privacy in Communications Networks (SecureComm 2007)*, Nice, France, p. 234–243, sept. 2007.

[291] I. N. Fovino et M. Masera, « Through the description of attacks : A multidimensional view », dans *Proceedings of the 25th International Conference on Computer Safety, Reliability and Security (SAFECOMP'06), LNCS 4166*, Gdansk, Pologne, p. 15–28, sept. 2006.

[292] E. Byres, M. Franz et D. Miller, « The use of attack trees in assessing vulnerabilities in SCADA systems », dans *Proceedings of the International Infrastructure Survivability Workshop (IISW 2004)*, Lisbonne, Portugal, déc. 2004.

[293] C. D. Convery, S. et M. Franz, « An attack tree for the border gateway protocol ». IETF Internet Draft, fév. 2004.

[294] K. Edge, R. Raines, M. Grimaila, R. Baldwin, R. Bennington et C. Reuter, « The use of attack and protection trees to analyze security for an online banking system », dans *Proceedings of the 40th Annual Hawaii International Conference on System Sciences (HICSS-40)*, Hawaï, États-Unis, p. 144b, jan. 2007.

[295] A. Buldas et T. Mägi, « Practical security analysis of e-voting systems », dans *Advances in Information and Computer Security, Proceedings of the 2nd International Workshop on Security (IWSEC), LNCS 4752*, Nara, Japon, p. 320–335, oct. 2007.

[296] V. Higuero, J. J. Unzilla, E. Jacob, P. Sáiz et D. Luengo, « Application of 'attack trees' technique to copyright protection protocols using watermarking and definition of a new transactions protocol SecDP (secure distribution protocol) », dans *Proceedings of the 2nd International Workshop on Multimedia Interactive Protocols and Systems (MIPS'04), LNCS 3311*, Grenoble, France, p. 264–275, sept. 2004.

[297] R. Cowan, M. Grimaila et R. Patel, « Using attack and protection trees to evaluate risk in an embedded weapon system », dans *Proceedings of the 3rd International Conference on Information Warfare and Security (ICIW 2008)*, Omaha, États-Unis, p. 97–108, avril 2008.

[298] K. Karppinen, « Security measurement based on attack trees in a mobile ad hoc network environment », Mémoire de master, VTT et Université de Oulu, 2005.

[299] P. A. Khand et P. H. Seong, « An attack model development process for the cyber security of safety related nuclear digital I&C systems », dans *Proceedings of the Korean Nucleary Society (KNS) Fall meeting*, Corée, oct. 2007.

[300] G.-Y. Park, C. K. Lee, J. G. Choi, D. H. Kim, Y. J. Lee et K.-C. Kwon, « Cyber security analysis by attack trees for a reactor protection system », dans *Proceedings of the Korean Nuclear Society (KNS) Fall Meeting*, Pyeong Chang, Corée, oct. 2008.

[301] S. C. Patel, J. H. Graham et P. A. Ralston, « Quantitatively assessing the vulnerability of critical information systems : A new method for evaluating security enhancements », *International Journal of Information Management*, vol. 28, p. 483–491, déc. 2008.

[302] C.-W. Ten, C.-C. Liu et M. Govindarasu, « Vulnerability assessment of cybersecurity for SCADA systems using attack trees », dans *Proceedings of the IEEE Power Engineering Society General Meeting*, Tampa, États-Unis, p. 1–8, juin 2007.

[303] S. McLaughlin, P. McDaniel et D. Podkuiko, « Energy theft in the advanced metering infrastructure », dans *Proceedings of the 4th International Workshop on Critical Information Infrastructure Security (CRITIS'09)*, Bonn, Allemagne, 2009.

[304] J. H. Espedalen, « Attack trees describing security in distributed internet-enabled metrology », Mémoire de master, Université de Gjøvik, 2007.

[305] D. L. Buckshaw, G. S. Parnell, W. L. Unkenholz, D. L. Parks, J. M. Wallner et O. S. Saydjari, « Mission oriented risk and design analysis of critical information systems », *Military Operations Research*, vol. 10, n° 2, p. 19–38, 2005.

[306] K. Clark, E. Singleton, S. Tyree et J. Hale, « Strata-gem : risk assessment through mission modeling », dans *Proceedings of the 4th ACM Workshop on Quality of Protection (QoP'08)*, Alexandria, Virginia, États-Unis, p. 51–58, oct. 2008.

[307] I. Ray et N. Poolsapassit, « Using attack trees to identify malicious attacks from authorized insiders », dans *Proceedings of the 10th European Symposium on Research in Computer Security (ESORICS'05), LNCS 3679*, Milan, Italie, p. 231–246, 2005.

[308] I. Ray et N. Poolsapassit, « Investigating computer attacks using attack trees », dans *Proceedings of the 3rd IFIP WG 11.9 International Conference on Digital Forensics, (IFIP Volume 242)*, Orlando, États-Unis, p. 331–343, jan. 2007.

[309] K. Reddy, H. Venter, M. Olivier et I. Currie, « Towards privacy taxonomy-based attack tree analysis for the protection of consumer information privacy », dans *Proceedings of the 6th Annual Conference on Privacy, Security and Trust (PST '08)*, New Brunswick, Canada, p. 56–64, oct. 2008.

[310] S. Mauw et M. Oostdijk, « Foundations of attack trees », dans *Proceedings of the 8th Annual International Conference on Information Security and Cryptology (ICISC'05), LNCS 3935*, Séoul, Corée, p. 186–198, déc. 2005.

[311] A. Buldas, P. Laud, J. Priisalu, M. Saarepera et J. Willemson, « Rational choice of security measures via multi-parameter attack trees », dans *Proceedings of the 1st International Workshop on Critical Information Infrastructure Security (CRITIS'06), LNCS 4347*, Îles de Samos, Grèce, p. 235–248, sept. 2006.

[312] A. Jürgenson et J. Willemson, « Computing exact outcomes of multi-parameter attack trees », dans *Proceedings of the Confederated International Conferences on the Move to Meaningful Internet Systems (OTM 2008/IS 2008), LNCS 5332*, Monterrey, Mexique, p. 1036–1051, nov. 2008.

[313] A. Jürgenson et J. Willemson, « Processing multi-parameter attacktrees with estimated parameter values », dans *Advances in Information and Computer Security, Proceedings of the 2nd International Workshop on Security (IWSEC), LNCS 4752*, Nara, Japon, p. 308–319, oct. 2007.

[314] S. Bistarelli, M. Dall'Aglio et P. Peretti, « Strategic games on defense trees », dans *Proceedings of the 4th International Workshop on Formal Aspects in Security and Trust (FAST 2006), LNCS 4691*, Hamilton, Ontario, Canada, p. 1–15, août 2006.

[315] R. R. Yager, « OWA trees and their role in security modeling using attack trees », *Information Sciences*, vol. 176, p. 2933–2959, oct. 2006.

[316] K. Daley, R. Larson et J. Dawkins, « A structural framework for modeling multi-stage network attacks », dans *Proceedings of the 31st International Conference on Parallel Processing Workshops (ICPPW'02)*, Vancouver, Canada, p. 5–10, août 2002.

[317] L. Grunske et D. Joyce, « Quantitative risk-based security prediction for component-based systems with explicitly modeled attack profiles », *The Journal of Systems & Software*, vol. 81, n° 8, p. 1327–1345, 2008.

[318] S. Bistarelli, F. Fioravanti et P. Peretti, « Defense trees for economic evaluation of security investments », dans *Proceedings of the 1st International Conference on Availability, Reliability and Security (ARES'06)*, Vienne, Autriche, p. 416–423, avril 2006.

[319] A. Jürgenson et J. Willemson, « Serial model for attack tree computations », dans *Proceedings of the 12th Annual International Conference on Information Security and Cryptology (ICISC'09), LNCS 5984*, Séoul, Corée, p. 118–128, déc. 2009.

[320] P. A. Khand, « System level security modeling using attack trees », dans *Proceedings of the 2nd International Conference on Computer, Control and Communication (IC4)*, Karachi, Pakistan, p. 1–6, fév. 2009.

[321] J. P. McDermott, « Attack net penetration testing », dans *Proceedings of the 2000 Workshop on New Security Paradigms (NSPW'00)*, Cork, Irlande, p. 15–21, sept. 2000.

[322] V. Horvath et T. Dörges, « From security patterns to implementation using Petri nets », dans *Proceedings of the 4th International Workshop on Software Engineering for Secure Systems (SESS'08)*, Leipzig, Allemagne, p. 17–24, 2008.

[323] J. Steffan et M. Schumacher, « Collaborative attack modeling », dans *Proceedings of the 2002 ACM Symposium on Applied Computing (SAC'02)*, Madrid, Espagne, p. 253–259, mars 2002.

[324] C.-W. Ten, C.-C. Liu et M. Govindarasu, « Vulnerability assessment of cybersecurity for SCADA systems », *IEEE Transactions on Power Systems*, vol. 23, n° 4, p. 1836–1846, 2008.

[325] D. Xu et K. Nygard, « Threat-driven modeling and verification of secure software using aspect-oriented Petri nets », *IEEE Transactions on Software Engineering*, vol. 32, n° 4, p. 265–278, 2006.

[326] V. Gupta, V. Lam, H. Ramasamy, W. Sanders et S. Singh, « Dependability and performance evaluation of intrusion-tolerant server architectures », dans *Proceedings of the 1st Latin America Dependability Conference (LADC-1), LNCS 2847*, São Paulo, Brésil, p. 81–101, oct. 2003.

[327] F. Stevens, T. Courtney, S. Singh, A. Agbaria, J. Meyer, W. Sanders et P. Pal, « Model-based validation of an intrusion-tolerant information system », dans *Proceedings of the 23rd IEEE International Symposium on Reliable Distributed Systems (SRDS)*, Florianpolis, Brésil, p. 184–194, oct. 2004.

[328] J. L. Rrushi et R. H. Campbell, « Detecting cyber attacks on nuclear power plants », dans *Proceedings of the 2nd Annual IFIP WG 11.10 International Conference on Critical Infrastructure Protection (CIP 2008)*, Washington D.C., États-Unis, p. 41–54, mars 2008.

[329] J. Rrushi, « Exploiting physical process internals for network intrusion detection in process control networks », dans *Proceedings of the 6th American Nuclear Society International Topical Meeting on Nuclear Plant Instrumentation, Control, and Human-Machine Interface Technologies (NPIC-HMIT 2009)*, Knoxville, États-Unis, avril 2009.

[330] G. Dalton, R. Mills, J. Colombi et R. Raines, « Analyzing attack trees using generalized stochastic petri nets », dans *Proceedings of the 7th IEEE Systems, Man and Cybernetics Information Assurance Workshop (IAW '06)*, West Point, États-Unis, p. 116–123, juin 2006.

[331] S. Pudar, G. Manimaran et C.-C. Liu, « PENET : a practical method and tool for integrated modeling of security attacks and countermeasures », *Computers & Security*, vol. 28, p. 754–771, mai 2010.

[332] W. J. Caelli, D. Longley et A. B. Tickle, « A methodology for describing information and physical security architectures », dans *Proceedings of the 8th IFIP TC11 International Conference on Information Security (SEC'92)*, vol. A-15 dans *IFIP Transactions*, Singapoure, p. 277–296, May 1992.

[333] C. Meadows, « A representation of protocol attacks for risk assessment », dans *Proceedings of the DIMACS Workshop on Network Threats*, New Brunswick, États-Unis, p. 1–10, déc. 1996.

[334] I. Cervesato et C. Meadows, « One picture is worth a dozen connectives : A fault-tree representation of NPATRL security requirements », *IEEE Transactions on Dependable and Secure Computing*, vol. 4, p. 216–227, juil. 2007.

[335] I. S. Moskowitz et M. H. Kang, « An insecurity flow model », dans *Proceedings of the 1997 Workshop on New Security Paradigms (NSPW'97)*, Langdale, Royaume-Uni, p. 61–74, sept. 1997.

[336] M. A. McQueen, W. F. Boyer, M. A. Flynn et G. A. Beitel, « Quantitative cyber risk reduction estimation methodology for a small SCADA control system », dans *Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS-39)*, vol. 9, Hawaï, États-Unis, p. 226–237, jan. 2006.

[337] E. Byres et D. Leversage, « Estimating a system's mean time-to-compromise, », *IEEE Security & Privacy Magazine*, vol. 6, p. 52–60, 2008.

[338] D. J. Leversage et E. J. Byres, « Comparing electronic battlefields : Using mean time-to-compromise as a comparative security metric », dans *Proceedings of the 4th International Conference on Methods, Models, and Architectures for Network Security (MMM-ACNS'07)*, Saint-Pétersbourg, Russie, p. 213–227, sept. 2007.

[339] C. Phillips et L. P. Swiler, « A graph-based system for network-vulnerability analysis », dans *Proceedings of the 1998 Workshop on New Security Paradigms (NSPW'98)*, Charlottesville, États-Unis, p. 71–79, sept. 1998.

[340] L. P. Swiler, C. Phillips, D. Ellis et S. Chakerian, « Computer-attack graph generation tool », dans *Proceedings of the DARPA Information Survivability Conference and Exposition II (DISCEX'01)*, vol. 2, Anaheim, États-Unis, p. 307–321, 2001.

[341] O. Sheyner, J. Haines, S. Jha, R. Lippmann et J. Wing, « Automated generation and analysis of attack graphs », dans *Proceedings of the IEEE Symposium on Security and Privacy (S&P'02)*, Oakland, États-Unis, p. 273–284, mai 2002.

[342] P. Ammann, D. Wijesekera et S. Kaushik, « Scalable, graph-based network vulnerability analysis », dans *Proceedings of the 9th ACM Conference on Computer and Communications Security (CCS'02)*, Washington D.C., États-Unis, p. 217–224, nov. 2002.

[343] S. Noel, S. Jajodia, B. O'Berry et M. Jacobs, « Efficient minimum-cost network hardening via exploit dependency graphs », dans *Proceedings of the 19th Annual Computer Security Applications Conference (ACSAC'03)*, Las Vegas, États-Unis, p. 86–95, déc. 2003.

[344] N. S. Jajodia, S. et B. O'berry, *Managing Cyber Threats : Issues, Approaches, and Challenges*, chap. 9, Topological Analysis of Network Attack Vulnerability, p. 247–266. Springer US, 2005.

[345] X. Ou, W. F. Boyer et M. A. McQueen, « A scalable approach to attack graph generation », dans *Proceedings of the 13th ACM conference on Computer and Communications Security (CCS'06)*, Alexandria, États-Unis, p. 336–345, nov. 2006.

[346] L. R. Ingols, K. et K. Piwowarski, « Practical attack graph generation for network defense », dans *Proceedings of the 22nd Annual Computer Security Applications Conference (ACSAC'06)*, Washington D.C., États-Unis, p. 121–130, déc. 2006.

[347] R. Lippmann et K. Ingols, « An annotated review of past papers on attack graphs », Rapport Technique ESC-TR-2005-054, Massachusetts Institute of Technology (MIT), mars 2005.

[348] V. Mehta, C. Bartzis, H. Zhu, E. Clarke et J. Wing, « Ranking attack graphs », dans *Proceedings of the 9th International Symposium on Recent Advances in Intrusion Detection (RAID'06), LNCS 4219*, Hambourg, Allemagne, p. 127–144, sept. 2006.

[349] B. S. Malhotra, S. et S. K. Ghosh, « A vulnerability and exploit independent approach for attack path prediction », dans *Proceedings of the IEEE 8th International Conference on Computer and Information Technology Workshops*, Sydney, Australie, p. 282–287, juil. 2008.

[350] L. Wang, T. Islam, T. Long, A. Singhal et S. Jajodia, « An attack graph-based probabilistic security metric », dans *Proceeedings of the 22nd Annual IFIP WG 11.3 Working Conference on Data and Applications Security (DAS'2008), LNCS 5094*, Londres, Royaume-Uni, p. 283–296, juil. 2008.

[351] S. Noel et S. Jajodia, « Managing attack graph complexity through visual hierarchical aggregation », dans *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security (VizSEC'04)*, Fairfax, États-Unis, p. 109–118, oct. 2004.

[352] S. Noel, M. Jacobs, P. Kalapa et S. Jajodia, « Multiple coordinated views for network attack graphs », dans *Proceedings of the 2005 IEEE Workshop on Visualization for Computer Security (VizSEC 05)*, Minneapolis, États-Unis, p. 99–106, oct. 2005.

[353] L. Williams, R. Lippmann et K. Ingols, « An interactive attack graph cascade and reachability display », dans *Proceedings of the 2007 Workshop on Visualization for Computer Security (VizSEC'07)*, Sacramento, États-Unis, p. 221–236, oct. 2007.

[354] V. A. O. X. Homer, J. et M. A. Mcqueen, « Improving attack graph visualization through data reduction and attack grouping », dans *Proceedings of the 5th international Workshop on Visualization For Computer Security (VizSEC'08)*, Cambridge, États-Unis, p. 68–79, sept. 2008.

[355] I. Kotenko et M. Stepashkin, « Analyzing network security using malefactor action graphs », *International Journal of Computer Science and Network Security*, vol. 6, n° 6, p. 226–236, 2006.

[356] S. Braynov et M. Jadliwala, « Representation and analysis of coordinated attacks », dans *Proceedings of the 2003 ACM Workshop on Formal Methods in Security Engineering (FMSE'03)*, Washington D.C., États-Unis, p. 43–51, 2003.

[357] J. Dawkins et J. Hale, « A systematic approach to multi-stage network attack analysis », dans *Proceedings of the 2nd IEEE International Information Assurance Workshop (IAWA'04)*, Charlotte, États-Unis, p. 48–56, avril 2004.

[358] K. Clark, S. Tyree, J. Dawkins et J. Hale, « Qualitative and quantitative analytical techniques for network security assessment », dans *Proceedings of the 5th IEEE Systems, Man and Cybernetics Information Assurance Workshop (IAW'04)*, West Point, États-Unis, p. 321–328, juin 2004.

[359] Y. Liu et H. Man, « Network vulnerability assessment using Bayesian networks », dans *Proceedings of SPIE Data Mining, Intrusion Detection, Information Assurance, and Data Networks Security 2005*, vol. 5812, Orlando, États-Unis, p. 61–71, mars 2005.

[360] M. Frigault, L. Wang, A. Singhal et S. Jajodia, « Measuring network security using dynamic Bayesian network », dans *Proceedings of the 4th ACM Workshop on Quality of Protection (QoP'08)*, Alexandria, États-Unis, p. 23–30, oct. 2008.

[361] S. H. Houmb, V. N. Franqueira et E. A. Engum, « Quantifying security risk level from CVSS estimates of frequency and impact », *Journal of Systems and Software*, 2009. Sous presse (disponible en ligne).

[362] X. An, D. Jutla et N. Cercone, « Privacy intrusion detection using dynamic Bayesian networks », dans *Proceedings of the 8th International Conference for Electronic Commerce (ICEC'06)*, Fredericton, Canada, p. 208–215, août 2006.

[363] T. Sommestad, M. Ekstedt et P. Johnson, « Combining defense graphs and enterprise architecture models for security analysis », dans *Proceedings of the 12th IEEE International Conference on Enterprise Distributed Object Computing (EDOC'08)*, Munich, Allemagne, p. 349–355, sept. 2008.

[364] U. Franke, T. Sommestad, M. Ekstedt et P. Johnson, « Defense graphs and enterprise architecture for information assurance analysis », dans *Proceedings of the 26th Army Science Conference*, Orlando, États-Unis, déc. 2008.

[365] T. Sommestad, M. Ekstedt et P. Johnson, « Cyber security risks assessment with bayesian defense graphs and architectural models », dans *Proceedings of the 42nd Annual Hawaii International Conference on System Sciences (HICSS-42)*, Hawaï, États-Unis, p. 1–10, jan. 2009.

[366] M. Ekstedt et T. Sommestad, « Enterprise architecture models for cyber security analysis », dans *Proceedings of the IEEE/PES Power System Conference and Exposition (PSCE'09)*, Seattle, États-Unis, p. 1–6, mars 2009.

[367] T. Sommestad, M. Ekstedt et L. Nordström, « Modeling security of power communication systems using defense graphs and influence diagrams », *IEEE Transactions on Power Delivery*, vol. 24, p. 1801–1808, oct. 2009.

[368] P. Johnson, R. Lagerström, P. Närman et M. Simonsson, « Enterprise architecture analysis with extended influence diagrams », *Information Systems Frontiers*, vol. 9, p. 163–180, juil. 2007.

[369] J. McDermott et C. Fox, « Using abuse case models for security requirements analysis », dans *Proceedings of the 15th Annual Computer Security Applications Conference (ACSAC'99)*, Phoenix, États-Unis, p. 55–64, déc. 1999.

[370] D. J. Firesmith, « Security use cases », *Journal of Object Technology*, vol. 2, p. 53–64, mai 2003.

[371] G. Sindre et A. L. Opdahl, « Eliciting security requirements by misuse cases », dans *Proceedings of 37th International Conference on Technology of Object-Oriented Languages and Systems (TOOLS-PACIFIC 2000)*, Sydney, Australie, p. 120–131, nov. 2000.

[372] G. Sindre et A. L. Opdahl, « Templates for misuse case description », dans *Proceedings of the 7th International Working Conference on Requirements Engineering : Foundation for Software Quality (REFSQ 2001)*, Interlaken, Suisse, p. 125–136, juin 2001.

[373] G. Sindre, A. L. Opdahl et G. F. Brevik, « Generalization/specialization as a structuring mechanism for misuse cases », dans *Proceedings of the 2nd Symposium on Requirements Engineering for Information Security (SREIS'02)*, Raleigh, États-Unis, oct. 2002.

[374] I. Alexander, « Misuse cases : Use cases with hostile intent », *IEEE software*, vol. 20, n° 1, p. 58–66, 2003.

[375] G. Sindre et A. L. Opdahl, « Eliciting security requirements with misuse cases », *Requirements Engineering*, vol. 10, n° 1, p. 34–44, 2005.

[376] L. Røstad, « An extended misuse case notation : Including vulnerabilities and the insider threat », dans *Proceedings of the 12th International Working Conference on Requirements Engineering : Foundation for Software Quality (REFSQ 2006)*, Luxembourg, Grand-Duché de Luxembourg, juin 2006.

[377] A. L. Opdahl et G. Sindre, « Experimental comparison of attack trees and misuse cases for security threat identification », *Information and Software Technology*, vol. 51, n° 5, p. 916–932, 2009.

[378] I. Tøndel, J. Jensen et L. Røstad, « Combining misuse cases with attack trees and security activity models », dans *The 4th International Conference on Availability, Reliability and Security (ARES 2010)*, Cracovie, Pologne, p. 438–445, fév. 2010.

[379] P. H. Meland, I. A. Tøndel et J. Jensen, « Idea : Reusability of threat models - two approaches with an experimental evaluation », dans *International Symposium on Engineering Secure Software and Systems (ESSoS)*, Pise, Italie, p. 114–122, fév. 2010.

[380] G. Sindre, « Mal-activity diagrams for capturing attacks on business processes », dans *Proceedings of the 13th International Working Conference on Requirements Engineering : Foundation for Software Quality (REFSQ 2007), LNCS 4542*, Trondheim, Norvège, p. 355–366, juin 2007.

[381] H. D. Mamadou, J. Romero-Mariona, E. S. Sim et D. J. Richardson, « A comparative evaluation of three approaches to specifying security requirements », dans *Proceedings of the 12th International Working Conference on Requirements Engineering : Foundation for Software Quality (REFSQ 2006)*, Luxembourg, Grand-Duché du Luxembourg, juin 2006.

[382] F. Cuppens et N. Cuppens-Boulahia, *La sécurité des réseaux et systèmes répartis*, chap. Les modéles de sécurité F. Cuppens, N. Cuppens-Boulahia. Hermès, 2003.

[383] M. Bishop, *Computer Security : Art and Science*. Addison Wesley Professional, 2003.

[384] R. Anderson, F. Stajano et J. Lee, *Advances in Computer*, vol. 55, chap. Security policies, p. 186–237. Academic Press, juil. 2001.

[385] E. Jonsson et T. Olovsson, « On the integration of security and dependability in computer systems », dans *Proceedings of the IASTED International Conference on Reliability, Quality Control and Risk Assessment*, Washington D.C., États-Unis, p. 93–97, 1992.

[386] D. F. C. Brewer, « Applying security techniques to achieve safety », dans *Proceedings of the 3rd Safety-critical Systems Symposium (SSS'93)*, Bristol, U.K., p. 246–256, fév. 1993.

[387] J.-C. Laprie et Y. Deswarte, « Saturne : systèmes répartis tolérant les fautes et les intrusions », Rapport Technique 84.023, Laboratoire d'Analyse et d'Architecture des Systèmes du CNRS (LAAS), avril 1984.

[388] J. Fraga et D. Powell, « A fault and intrusion-tolerant file system », dans *Proceedings of the IFIP 3rd International Conference on Computer Security (SEC'85)*, Dublin, Irlande, p. 203–218, août 1985.

[389] Y. Koga, E. Fukushima et K. Yoshihara, « Error recoverable and securable data communication for computer network », dans *Proceedings of the 12th IEEE International Symposium on Fault-Tolerant Computing (FTCS-12)*, Santa-Monica, États-Unis, p. 183–186, juin 1982.

[390] Y. Deswarte, L. Blain et J.-C. Fabre, « Intrusion tolerance in distributed systems », dans *Proceedings of the 1991 IEEE Symposium on Research in Security and Privacy (S&P'91)*, Oakland, États-Unis, p. 110–121, mai 1991.

[391] J. E. Dobson et B. Randell, « Building reliable secure computing systems out of unreliable insecure components », dans *Proceedings of the 1986 IEEE Symposium on Security and Privacy (S&P'86)*, Oakland, États-Unis, p. 187–193, avril 1986.

[392] C. Meadows, « Applying the dependability paradigm to computer security », dans *Proceedings of the 1995 Workshop on New Security Paradigms (NSPW'95)*, La Jolla, États-Unis, p. 75–79, août 1995.

[393] P. Veríssimo, N. Neves et M. Correia, *Architecting Dependable Systems, LNCS 2677*, chap. Intrusion-Tolerant Architectures : Concepts and Design, p. 3–36. Springer, 2003.

[394] D. Powell, A. Adelsbasch, C. Cachin, S. Creese, M. Dacier, Y. Deswarte, T. McCutcheon, N. Neves, B. Pfitzmann, B. Randell, R. Stroud, P. Veríssimo et M. Waidner, « MAFTIA (Malicious- and Accidental-Fault Tolerance for Internet Applications) », dans *Proceedings of the 31st Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2001), Supplemental Volume*, Göteborg, Suède, p. 32–35, juil. 2001.

[395] Y. Deswarte et D. Powell, « Intrusion tolerance for internet applications », dans *Proceedings of the IFIP World Computer Congress*, vol. IFIP 156/2004, Toulouse, France, p. 241–256, août 2004.

[396] R. J. Ellison, D. A. Fisher, R. C. Linger, H. F. Lipson, T. Longstaff et N. R. Mead, « Survivable network systems : An emerging discipline », Rapport Technique CMU/SEI-97-TR-013, Université Carnegie Mellon, mai 1997.

[397] J.-C. Laprie, « From dependability to resilience », dans *Proceedings of the 38th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2008), Supplemental Volume*, Anchorage, États-Unis, juin 2008.

[398] P. Verissímo, N. Neves et M. Correia, « The CRUTIAL reference critical information infrastructure architecture : a blueprint », *International Journal of System of Systems Engineering*, vol. 1, n° 1-2, p. 78–95, 2008.

[399] P. Sousa, A. Bessani, W. Dantas, F. Souto, M. Correia et N. Neves, « Intrusion-tolerant self-healing devices for critical infrastructure protection », dans *Proceedings of the 39th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2009)*, Estoril, Portugal, p. 217–222, juil. 2009.

[400] É. Totel, F. Majorczyk et L. Mé, « COTS diversity based intrusion detection and application to Web servers », dans *Proceedings of the 8th International Symposium on Recent Advances in Intrusion Detection (RAID'05), LNCS 3858*, Seattle, États-Unis, p. 43–62, sept. 2005.

[401] D. Quéniart, « Analyse de sûreté. Principes et pratiques ». Techniques de l'Ingénieur, B 3 810, mai 1996.

[402] International Atomic Energy Agency (IAEA), International Nuclear Safety Group (INSAG), « Defence in depth in nuclear safety ». INSAG-10, STI/PUB/1013, 1996.

[403] U.S. National Security Agency (NSA), « Defense in depth, a practical strategy for achieving information assurance in today's highly networked environments ». Guide NSA, Information Assurance Mission, http://www.nsa.gov/ia/_files/support/defenseindepth.pdf.

[404] K. Dauch, A. Hovak et R. Nestler, « Information assurance using a defense in-depth strategy », dans *Proceedings of the Cybersecurity Applications & Technology Conference for Homeland Security (CATCH)*, Washington D.C., États-Unis, p. 267–272, mars 2009.

[405] J. Reason, *Human Error*. Cambridge University Press, 1990.

[406] R. Winther, O.-A. Johnsen et B. A. Gran, « Security assessments of safety critical systems using HAZOPs », dans *Proceedings of the 20th International Conference on Computer Safety, Reliability and Security (SAFECOMP 2001), LNCS 2187*, Budapest, Hongrie, p. 14–24, sept. 2001.

[407] N. Foster et J. Jacob, « Hazard analysis for security protocol requirements », dans *Proceedings of the 1st International IFIP Working Conference on Network Security*, Louvain, Belgique, p. 75–92, nov. 2001.

[408] T. Srivatanakul, *Security Analysis with Deviational Techniques*. Thèse de doctorat, Université de York, 2005.

[409] P. Baybutt, « Sneak path analysis (SPSA) for industrial cyber security », Rapport Technique, Primatech, 2003.

[410] P. Baybutt, « Sneak path analysis : Security application finds cyber threats, then works to protect a system », *ISA InTech*, vol. 51, sept. 2004.

[411] A. Gorbenko, V. Kharchenko, O. Tarasyuk et A. Furmanov, *Rigorous Development of Complex Fault-Tolerant Systems (LNCS 4157)*, chap. F(I)MEA-Technique of Web-services Analysis and Dependability Ensuring, p. 153–167. Springer, 2006.

[412] E. Babeshko, V. Kharchenko et A. Gorbenko, « Applying F(I)MEA-technique for SCADA-based industrial control systems dependability assessment and ensuring », dans *Proceeding of the 3rd International Conference on Dependability of Computer Systems (DepCoS-RELCOMEX)*, Szklarska Poreba, Pologne, p. 309–315, juin 2008.

[413] J. R. Inge, « The safety case, its development and use in the United Kingdom », dans *Proceedings of the 25th International System Safety Conference (ISSC)*, Baltimore, États-Unis, p. 725–730, août 2007.

[414] P. G. Bishop et R. E. Bloomfield, « A methodology for safety case development », dans *Proceedgins of the 6th Safety-critical Systems Symposium (SSS'98)*, Birmingham, Royaume-Uni, fév. 1998.

[415] R. E. Bloomfield, S. Guerra, M. Masera, A. Miller et C. B. Weinstock, « International working group on assurance cases (for security) », *IEEE Security & Privacy*, vol. 4, p. 66–68, mai 2006.

[416] R. Bloomfield et P. Bishop, « Safety and assurance cases : Past, present and possible future - an Adelard perspective », dans *Proceedings of the 18th Safety-Critical Systems Symposium (SSS 2010)*, Bristol, Royaume-Uni, p. 51–67, fév. 2010.

[417] R. Weaver, *The Safety of Software : Constructing and Assuring Arguments*. Thèse de doctorat, Université de York, 2003.

[418] J. Goodenough, H. Lipson et C. Weinstock, « Arguing security - creating security assurance cases », rapport en ligne (initiative *build security-in* du U.S. CERT), Université Carnegie Mellon, jan. 2007.

[419] H. Lipson et C. Weinstock, « Evidence of assurance : Laying the foundation for a credible security case », rapport en ligne (initiative *build security-in* du U.S. CERT), Université Carnegie Mellon, mai 2008.

[420] T. P. Kelly, *Arguing Safety - A Systematic Approach to Managing Safety Cases.* Thèse de doctorat, Université de York, 1998.

[421] K. Moleyar et A. Miller, « Formalizing attack trees for a SCADA system », dans *1st Annual IFIP WG 11.10 International Conference on Critical Infrastructure Protection (CIP 2007)*, Hanover, États-Unis, mars 2007. Session papiers courts (hors actes du congrès, disponible en ligne).

[422] S. Brostoff et M. A. Sasse, « Safe and sound : a safety-critical approach to security », dans *Proceedings of the 2001 Workshop on New Security Paradigms (NSPW'01)*, Cloudcrofl, États-Unis, p. 41–51, Sep 2001.

[423] U.K. Ministry of Defence (MoD), Directorate of Standardization, « Requirements for safety-related software in defense equirment - part 1 - requirements ». Interim MoD-Def-Stan-00-55 (Part 1)/Issue 1, avril 1991.

[424] International Society of Automation (ISA), « Application of safety instrumented systems for the process industries ». ISA–84.01-1996, avril 1996.

[425] N. Kube et B. Singer, « Security assurance levels : a SIL approach to security », dans *Procdeedings of the 2nd SCADA Security Scientific Symposium (S4)*, Miami, États-Unis, jan. 2008.

[426] P. Gruhn, « Safety, security groups form joint working group ». ISA InTech, juin 2009.

[427] D. K. Holstein et K. Stouffer, « Trust but verify critical infrastructure cyber security solutions », dans *Proceedings of the 43rd Hawaii International Conference on System Sciences (HICSS-43)*, Hawaï, États-Unis, p. 1–8, jan. 2010.

[428] J. Arlat, *Validation de la sûreté de fonctionnement par injection de fautes, méthode - mise en œuvre - application.* Thèse d'état, Laboratoire d'Analyse et d'Architecture des Systèmes du CNRS (LAAS), 1990.

[429] J. R. Voas, « Testing software for characteristics other than correctness : Safety, failure tolerance, and security », dans *Proceedings of the 10th International Conference on Testing Computer Software*, Washington D.C., États-Unis, juin 1996.

[430] M. Sutton, A. Greene et P. Amini, *Fuzzing : Brute Force Vulnerability Discovery.* Addison-Wesley Professional, 2007.

[431] A. Ozment, « Software security growth modeling : Examining vulnerabilities with reliability growth models », dans *Proceedings of the 1st Workshop on Quality of Protection (QoP'05)*, Milan, Italie, p. 25–36, sept. 2005.

[432] E. Rescorla, « Is finding security holes a good idea ? », dans *Proceedings of the 3rd Workshop on Economics and Information Security (WEIS'04)*, Minneapolis, Minnesota, États-Unis, mai 2004.

[433] M. D. Schroeder, « Engineering a security kernel for multics », dans *Proceedings of the 5th ACM Symposium on Operating Systems Principles*, Austin, États-Unis, p. 25–32, 1975.

[434] S. R. J. Ames, M. Gasser et R. R. Schell, « Security kernel design and implementation : An introduction », *Computer*, vol. 16, n° 7, p. 14–22, 1983.

[435] J. Rushby, « Kernels for safety ? », dans *Proceedgins of the Safety-critical Systems Symposium (SSS'86)*, Glasgow, Écosse, p. 210–220, oct. 1986.

[436] N. G. Leveson, T. J. Shimeall, J. L. Stolzy et J. C. Thomas, « Design for safe software », dans *Proceedings of the 21st Aerospace Sciences Meeting of the American Institute of Aeronautics and Astronautics*, Reno, États-Unis, p. 10–13, 1983.

[437] K. G. Wika et J. C. Knight, « A safety kernel architecture », Rapport Technique CS-94-04, Université de Virginie, 1994.

[438] J.-L. Boulanger, V. Delebarre, S. Natkin et J. Ozello, « Deriving safety properties of critical software from the system risk analysis, application to ground transportation systems », dans *Proceedings of the 2nd IEEE High-Assurance Systems Engineering Workshop (HASE'97)*, Washington D.C., États-Unis, p. 162–167, août 1997.

[439] A. Simpson, J. Woodcock et J. Davies, « Safety through security », dans *Proceedings of the 9th International Workshop on Software Specification and Design (IWSSD '98)*, Japon, p. 18–24, Apr 1998.

[440] J. Goguen et J. Meseguer, « Security policies and security models », dans *Proceedings of the IEEE Symposium on Security and Privacy (S&P'82)*, Oakland, États-Unis, p. 11–20, avril 1982.

[441] A. W. Roscoe, *The theory and practice of concurrency.* Prentice Hall, 1998.

[442] V. Stavridou et B. Dutertre, « From security to safety and back », dans *Proceedings of the Computer Security, Dependability, and Assurance : From Needs to Solutions (CSDA'98)*, York, Royaume-Uni, p. 182–195, juil. 1998.

[443] É. Totel, J.-P. Blanquart, Y. Deswarte et D. Powell, « Supporting multiple levels of criticality », dans *Proceedings of the 28th IEEE Symposium on Fault Tolerant Computing Systems (FTCS-28)*, Munich, Allemagne, p. 70–79, juin 1998.

[444] D. Powell, J. Arlat, L. Beus-Dukic, A. Bondavalli, P. Coppola, A. Fantechi, E. Jenn, C. Rabéjac et A. Wellings, « GUARDS : a generic upgradable architecture for real-time dependable systems », *IEEE Transactions on Parallel and Distributed Systems*, vol. 10, n° 6, p. 580–599, 1999.

[445] Y. Laarouchi, Y. Deswarte, D. Powell, J. Arlat et E. de Nadai, « Connecting commercial computer to avionics systems », dans *Proceedings of the 28th Digital Avionics System Conference (DASC'09)*, Orlando, États-Unis, p. 6.D.1.1–9, oct. 2009.

[446] G. Sindre, *Situational Method Engineering : Fundamentals and Experiences*, vol. 244/2007 dans *IFIP International Federation for Information Processing*, chap. A Look at Misuse Cases for Safety Concerns, p. 252–266. Springer Boston, 2007.

[447] I. Alexander, « Initial industrial experience of misuse cases in trade-off analysis », dans *Proceedings of the 10th Anniversary IEEE Joint International Requirements Engineering Conference (RE'02)*, Essen, Allemagne, p. 61 –70, sept. 2002.

[448] T. Stålhane et G. Sindre, « A comparison of two approaches to safety analysis based on use cases », dans *Proceedings of the 26th International Conference on Conceptual Modeling (ER 2007), LNCS 4801*, Auckland, Nouvelle-Zélande, p. 423–437, nov. 2007.

[449] J. Ridgway, « Achieving safety through security management », dans *Proceedings of the 15th Safety-Critical Systems Symposium (SSS 2007)*, Bristol, Royaume-Uni, p. 3–20, fév. 2007.

[450] T. Novak et A. Treytl, « Common approach to functional safety and system security in building automation and control systems », dans *Proceedings of the 12th IEEE Conference on Emerging Technologies and Factory Automation (ETFA'07)*, Patras, Grèce, p. 1141–1148, sept. 2007.

[451] T. Novak, A. Treytl et A. Gerstinger, « Embedded security in safety critical automation systems », dans *Proceedings of the 26th International System Safety Conference (ISSC 2008)*, Vancouver, Canada, p. S.1–11, août 2008.

[452] Y. Deswarte, M. Kaâniche, P. Corneillie et J. Goodson, « SQUALE dependability assessment criteria », dans *Proceedings of the 18th International Conference on Computer Safety, Reliability and Security (SAFECOMP'99), LNCS1698*, Toulouse, France, p. 27–38, sept. 1999.

[453] P. Corneillie, S. Moreau, C. Valentin, J. Goodson, A. Hawes, T. Manning, H. Kurth, G. Liebisch, A. Steinacker, Y. Deswarte, M. Kaâaniche et P. Benoit, « Dependability assessment criteria, SQUALE project (ACTS95/AC097) », Rapport Technique 98456, Laboratoire d'Analyse et d'Architecture des Systèmes du CNRS (LAAS), jan. 1999.

[454] P. H. Jesty et D. D. Ward, « Towards a unified approach to safety and security in automotive systems », dans *Proceedings of the 15th Safety-Critical Systems Symposium (SSS 2007)*, Bristol, Royaume-Uni, p. 21–34, fév. 2007.

[455] J. Alves-Foss, B. Rinker et C. Taylor, « Towards Common Criteria certification for DO-178B compliant airborne software systems ». Université d'Idaho, 2002.

[456] B. J. Garrick, J. Hall, M. Kilger, J. Mcdonald, T. O'Toole, P. Probst, E. R. Parker, R. Rosenthal, A. Trivelpiece, L. Van Arsdale et E. L. Zebroski, « Confronting the risks of terrorism : making the right decisions », *Reliability Engineering & System Safety*, vol. 86, n° 2, p. 129–176, 2004.

[457] R. Wilson, « Combating terrorism : an event tree approach », dans *Proceedings of the 27th International Seminar on Nuclear War and Planetary Emergencies*, Erice, Italie, p. 122–145, août 2002.

[458] G. Woo, « Quantitative terrorism risk assessment », *The Journal of Risk Finance*, vol. 4, n° 1, p. 7–14, 2002.

[459] Y. Y. Haimes, « Accident precursors, terrorist attacks, and systems engineering ». Présentation préparée pour le NAE Workshop on Project on Accident Precursors (disponible en ligne), juil. 2003.

[460] G. Cojazzi, S. Contini et G. Renda, « FT analysis in security related applications : Challenges and needs », dans *Proceedings of the 29th ESReDA Seminar on Systems Analysis for a More Secure World : Application of System Analysis and RAMS to Security of Complex System*, Ispra, Italie, p. 345–366, oct. 2005.

[461] International Society of Automation (ISA), « System security requirements and security assurance levels ». ISA–99.03.03, 2010. Draft.

[462] R. K. Flink, D. F. Spencer et R. A. Wells, « Lessons learned from cyber security assessments of SCADA and energy management systems », Rapport Technique INL/CON-06-11665, Idaho National Laboratory, sept. 2006.

[463] J. Davidson et J. Wright, « Recommended practice for securing control system modems ». U.S. Department of Homeland Security (DHS), National Cyber Security Division, jan. 2008.

[464] L. Pièvre-Cambacédès et M. Bouissou, « Beyond attack trees : dynamic security modeling with Boolean logic Driven Markov Processes (BDMP) », dans *Proceedings of the 8th European Dependable Computing Conference (EDCC-8)*, Valence, Espagne, p. 199–208, avril 2010.

[465] G. Sindre, A. L. Opdahl et D. G. Firesmith, « A reuse-based approach to determining security requirements », dans *Proceedings of the 9th International Working Conference on Requirements Engineering : Foundation for Software Quality (REFSQ 2003)*, Montpellier, France, p. 127–136, juin 2003.

[466] J. Pestourie, G. Malarange, E. Breton, S. Muffat et M. Bouissou, « Étude de la sûreté de fonctionnement d'un poste source EDF (90/20 kV) avec le logiciel OPALE », dans *Actes du 14e congrès de fiabilité et maintenabilité de l'IMdR ($\lambda\mu$14)*, Bourges, France, oct. 2004.

[467] P. Carer, J. Bellvis, M. Bouissou, J. Domergue et J. Pestourie, « A new method for reliability assessment of electrical power supplies with standby redundancies », dans *Proceedings of the 7th International Conference on Probabilistic Methods Applied to Power Systems (PMAPS'02)*, Naples, Italie, sept. 2002.

[468] E. Skoudis, *Counter hack : a step-by-step guide to computer attacks and effective defenses.* Prentice Hall PTR, 2001.

[469] M. Bouissou et Y. Dutuit, « Reliability analysis of a dynamic phased mission system », dans *Proceedings of the 4th International Conference on Mathematical Methods in Reliability (MMR'04)*, Santa Fe, New Mexique, États-Unis, juin 2004.

[470] M. Bouissou et Y. Lefebvre, « A path-based algorithm to evaluate asymptotic unavailability for large Markov models », dans *Proceedings of the 48th Reliability and Maintainability Annual Symposium (RAMS'02)*, Seattle, États-Unis, p. 32–39, 2002.

[471] M. Bouissou, « Recherche et quantification automatique de séquences accidentelles pour un système réparable », dans *Actes du 5e congrès de fiabilité et maintenabilité de l'IMdR ($\lambda\mu$5)*, Biarritz, France, oct. 1986.

[472] P. Harrison, « Laplace transform inversion and passage time distributions in Markov processes », *Journal of Applied Probability*, vol. 27, no 1, p. 74–87, 1990.

[473] J. Collet et I. Renault, « Path probability evaluation with repeated rates », dans *Proceedings of the 43rd Reliability and Maintainability Annual Symposium (RAMS'97)*, Philadelphia, États-Unis, p. 184–187, jan. 1997.

[474] Y. Lefebvre, *Nouveaux développements et justifications de méthodes de calcul de mesures de performance en sûreté de fonctionnement.* Thèse de doctorat, Université de Marne-la-Vallée, 2003.

[475] E. Jonsson et T. Olovsson, « A quantitative model of the security intrusion process based on attacker behavior », *IEEE Transactions on Software Engineering*, vol. 23, no 4, p. 235–245, 1997.

[476] K. Sallhammar, *Stochastic models for combined security and dependability evaluation.* Thèse de doctorat, Université Norvégienne de Sciences et Technologies (NTNU), 2007.

[477] R. Pullen, « Attacktree+, a computer tool for modeling attack scenarios », dans *Proceedings of the 29th ESReDA Seminar on Systems Analysis for a More Secure World : Application of System Analysis and RAMS to Security of Complex System*, Ispra, Italie, p. 333–343, oct. 2005.

[478] R. Dewri, N. Poolsappasit, I. Ray et D. Whitley, « Optimal security hardening using multi-objective optimization on attack tree models of networks », dans *Proceedings of the 14th ACM conference on Computer and Communications Security (CCS'07)*, Alexandria, États-Unis, p. 204–213, oct. 2007.

[479] L. Pièvre-Cambacédès et M. Bouissou, « Attack and defense dynamic modeling with BDMP », dans *Proceedings of the 5th International Conference on Mathematical Methods, Models, and Architectures for Computer Networks Security (MMM-ACNS-2010), LNCS 6258*, Saint-Pétersbourg, Russie, p. 86–101, sept. 2010.

[480] L. Pièvre-Cambacédès et M. Bouissou, « The promising potential of the BDMP formalism for security modeling », dans *Proceedings of the 39th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2009), Supplemental Volume*, Estoril, Portugal, juin 2009. Fast Abstract track.

[481] D. Coppit, K. J. Sullivan et J. B. Dugan, « Formal semantics of models for computational engineering : A case study on dynamic fault trees », dans *Proceedings of the 11th International Symposium on Software Reliability Engineering (ISSRE'00)*, San Jose, États-Unis, p. 270–282, oct. 2000.

[482] S. Distefano et A. Puliafito, « Dependability evaluation using dynamic reliability block diagrams and dynamic fault trees », *IEEE Transaction on Dependable and Secure Computing*, vol. 6, no 1, p. 4–17, 2008.

[483] M. Bouissou, « A generalization of dynamic fault trees through Boolean logic driven Markov processes (BDMP) », dans *Proceedings of the16th European Safety and Reliability Conference (ESREL'07)*, Stavanger, Norvège, juin 2007.

[484] M. Bouissou, S. Humbert, S. Muffat et N. Villatte, « KB3 tool : feedback on knowledge bases », dans *Proceedings of the11th European Safety and Reliability Conference (ESREL'02)*, Lyon, France, mars 2002.

[485] M. Bouissou et A. Flori, « Manuel d'utilisation de la base de connaissances BDMP pour KB3 ». Note EDF R&D HT-52/03/039/A (publique), déc. 2003.

[486] N. Villatte, « Manuel utilisateur de KB3 V3 ». Note EDF R&D HT-52/05/057/A (publique), nov. 2005.

[487] M. Bouissou, H. Chraibi et S. Muffat, « Utilisation de la simulation de Monte-Carlo pour la résolution d'un benchmark (MINIPLANT) », dans *Actes du 14^e congrès de fiabilité et maintenabilité de l'IMdR ($\lambda\mu$14)*, Bourges, France, oct. 2004.

[488] M. Bouissou et A.-S. Derode, « Approximation exponentielle de la fiabilité : cas des systèmes à réparations différées », dans *Actes du 14éme congrès de fiabilité et maintenabilité de l'IMdR ($\lambda\mu$14)*, Bourges, France, oct. 2004.

[489] J. Bon et M. Bouissou, « Fiabilité des grands systemes séquentiels : résultats théoriques et applications dans le cadre du logiciel GSI », *Revue de Statistique Appliquée*, vol. 40, n° 2, p. 45–54, 1992.

[490] J. Bon et J. Collet, « An algorithm in order to implement reliability exponential approximations », *Reliability Engineering & System Safety*, vol. 43, n° 3, p. 263–268, 1994.

[491] M. Bouissou et G. Torrente, « Méthodologie de développement de bases de connaissances pour la SdF avec l'environnement Open-source Visual Figaro », dans *Actes du 16^e congrès de fiabilité et maintenabilité de l'IMdR ($\lambda\mu$16)*, Avignon, France, oct. 2008.

[492] R. Ortalo, Y. Deswarte et M. Kaâniche, « Experimenting with quantitative evaluation tools for monitoring operational security », *IEEE Transactions on Software Engineering*, vol. 25, n° 5, p. 633–649, 1999.

[493] B. B. Madan, K. Goševa-Popstojanova, K. Vaidyanathan et K. S. Trivedi, « A method for modeling and quantifying the security attributes of intrusion tolerant systems », *Performance Evaluation*, vol. 56, p. 167–186, 2004.

[494] M. A. McQueen, W. F. Boyer, M. A. Flynn et G. A. Beitel, « Time-to-compromise model for cyber risk reduction estimation », dans *Proceedings of the 1st Workshop on Quality of Protection (QoP'05)*, Milan, Italie, p. 49–64, sept. 2005.

[495] W. Poundstone, *Prisoner's Dilemma : John von Neumann, Game Theory and the Puzzle of the Bomb*. Anchor Books, 1993.

[496] T. Sandler, G. Daniel et M. Arce, « Terrorism and game theory », *Simulation and Gaming*, vol. 34, p. 317–337, sept. 2003.

[497] A. Holmgren, E. Jenelius et J. Westin, « Evaluating strategies for defending electric power networks against antagonistic attacks », *IEEE Transactions on Power Systems*, vol. 22, p. 76 – 84, fév. 2007.

[498] S. Roy, C. Ellis, S. Shiva, D. Dasgupta, V. Shandilya et Q. Wu, « A survey of game theory as applied to network security », dans *Proceedings of the 43rd Hawaii International Conference on System Sciences (HICSS-43)*, Hawaï, États-Unis, jan. 2010.

[499] V. Bier, *Statistical Methods in Counterterrorism : Game Theory, Modeling, Syndromic Surveillance, and Biometric Authentication*, chap. Game-Theoretic and Reliability Methods in Counterterrorism and Security, p. 23–40. Springer New York, 2006.

[500] C. Griffin, B. Madan et K. Trivedi, « State space approach to security quantification », dans *Proceedgins of the 29th Annual International Computer Software and Applications Conference (COMPSAC'05) Volume 2*, Edinbourg, Écosse, p. 83–88, juil. 2005.

[501] S. Amin, A. Cárdenas et S. Sastry, « Safe and secure networked control systems under denial-of-service attacks », dans *Proceedings of the 12th International Conference on Hybrid Systems : Computation and Control (HSCC'09)*, San Francisco, États-Unis, p. 31–45, avril 2009.

[502] H. Tanaka, L. Fan, F. Lai et K. Toguchi, « Fault-tree analysis by fuzzy probability », *IEEE Transactions on Reliability*, vol. 32, n° 5, p. 453–457, 1983.

[503] D. Singer, « A fuzzy set approach to fault tree and reliability analysis », *Fuzzy Sets and Systems*, vol. 34, n° 2, p. 145–155, 1990.

[504] Y. Ou et J. B. Dugan, « Approximate sensitivity analysis for acyclic Markov reliability models », *IEEE Transactions on Reliability*, vol. 52, p. 220–230, juin 2003.

[505] X.-R. Cao et Y.-W. Wan, « Algorithms for sensitivity analysis of Markov systems through potentials and perturbation realization », *IEEE Transactions on Control Systems Technology*, vol. 6, p. 482–494, juil. 1998.

[506] B. Natvig et J. Gåsemyr, « New results on the Barlow–Proschan and Natvig measures of component importance in nonrepairable and repairable systems », *Methodology and Computing in Applied Probability*, vol. 11, n° 4, p. 603–620, 2009.

[507] P. Van, A. Barros et C. Bérenguer, « Importance measure on finite time horizon and application to Markovian multistate production systems », *Proceedings of the Institution of Mechanical Engineers, Part O : Journal of Risk and Reliability*, vol. 222, n° 3, p. 449–461, 2008.

[508] P. D. Van, A. Barros et C. Bérenguer, « Reliability importance analysis of Markovian systems at steady state using perturbation analysis », *Reliability Engineering & System Safety*, vol. 93, n° 11, p. 1605–1615, 2008.

[509] P. D. Van, A. Barros et C. Bérenguer, « From differential to difference importance measures for Markov reliability models », *European Journal of Operational Research*, vol. 204, p. 513–521, août 2009.

[510] P. D. Van, C. Bérenguer et L. Dieulle, « Comparaison et évaluation des méthodes de calcul des facteurs d'importance fiabilistes pour les systèmes dynamiques », dans *Actes du 6e Congrès international pluridisciplinaire Qualité et Sûreté de Fonctionnement (Qualita 2005)*, Bordeaux, France, mars 2005.

[511] M. Bouissou, « Détermination efficace de scenarii minimaux de défaillance pour des systèmes séquentiels », dans *Actes du 14e congrès de fiabilité et maintenabilité de l'IMdR ($\lambda\mu$14)*, Lille, France, oct. 2006.

[512] G. Renda, S. Contini et G. Cojazzi, « On the methods to model and analyze attack scenarios with fault trees », dans *Proceedings of the17th European Safety and Reliability Conference (ESREL'08)*, Valence, Espagne, p. 3135–3142, sept. 2008.

[513] International Atomic Energy Agency (IAEA), « Handbook on the physical protection of nuclear materials and facilities ». IAEA-TECDOC-1276, mars 2002.

[514] G. Cojazzi, G. Renda et S. Contini, « Qualitative and quantitative analysis of safeguards logic trees », dans *Joint proceedings of the 7th International Conference on Probabilistic Safety Assessment and Management (PSAM-7) and the13th European Safety and Reliability Conference (ESREL'04)*, Berlin, Allemagne, juin 2004.

[515] H. Ham, *An integrated methodology for quantitative assessment of proliferation resistance of advanced nuclear systems using probabilistic methods.* Thèse de doctorat, Massachusetts Institute of Technology (MIT), 2005.

[516] O. Nordland, « Making safe software secure », dans *Proceedings of the 16th Safety-Critical Systems Symposium (SSS 2008) (Improvements in System Safety)*, Bristol, Royaume-Uni, p. 15–23, fév. 2008.

[517] D.-B. Pan et F. Liu, « Influence between functional safety and security », dans *Proceedings of the 2nd IEEE Conference on Industrial Electronics and Applications (ICIEA'07)*, Harbin, Chine, p. 1323–1324, mai 2007.

[518] J. Smith, S. Russell et M. Looi, « Security as a safety issue in rail communications », dans *Proceedings of the 8th Australian Workshop on Safety Critical Systems and Software (SPS'03)*, vol. 33, Canberra, Australie, p. 79–88, oct. 2003.

[519] M. Lin, L. Xu, L. Yang, X. Qin, N. Zheng, Z. Wu et M. Qiu, « Static security optimization for real-time systems », *IEEE Transactions on Industrial Informatics*, vol. 5, p. 22 –37, fév. 2009.

[520] R. Robinson, M. Li, S. Lintelman, K. Sampigethaya, R. Poovendran, D. von Oheimb, J.-U. Busser et J. Cuellar, « Electronic distribution of airplane software and the impact of information security on airplane safety », dans *Proceedings of the 26th International Conference on Computer Safety, Reliability and Security (SAFECOMP'07), LNCS 4680*, Nuremberg, Allemagne, p. 28–39, sept. 2007.

[521] N. Neogi, « Safety and security in the next generation air transportation system », dans *National Workshop on Aviation Software Systems*, Alexandria, États-Unis, oct. 2006.

[522] M. G. Jaatun, M. B. Line et T. O. Grotan, « Secure remote access to autonomous safety systems : A good practice approach », *International Journal of Autonomous and Adaptive Communications Systems*, vol. 2, n° 3, p. 297–312, 2009.

[523] R. G. Herrtwich, « Automotive telematics - road safety vs. IT security ? (invited talk) », dans *Proceedings of the 23rd International Conference on. Computer Safety, Reliability and Security (SAFECOMP'04), LNCS 3219*, Potsdam, Allemagne, p. 239, sept. 2004.

[524] B. Chess et G. McGraw, « Static analysis for security », *IEEE Security & Privacy*, vol. 2, n° 6, p. 76–79, 2004.

[525] V. M. Igure, *A taxonomy of security vulnerabilities in SCADA protocols.* Thèse de doctorat, Université de Virginie, 2007.

[526] S. Lautieri, « De-risking safety », *Computing & Control Engineering Journal*, vol. 17, n° 3, p. 38–41, 2006.

[527] T. J. Cockram et S. R. Lautieri, « Combining security and safety principles in practice », dans *Proceedings of the 2nd Institution of Engineering and Technology International Conference on System Safety*, Londres, Royaume-Uni, p. 159–164, oct. 2007.

[528] A. Gerstinger et T. Novak, « Diversity for safety and security improvement », dans *Proceedings of the 26th International System Safety Conference (ISSC 2008)*, Vancouver, Canada, août 2008.

[529] I. E. Komari, V. Kharchenko, A. Romanovsky et E. Babeshko, « Diversity and security of computing systems : Points of interconnection (part 1 and 2) », *MASAUM Journal of Open Problems in Science and Engineering*, vol. 1, p. 28–41, 2009.

[530] J.-H. Cho, I.-R. Chen et P.-G. Feng, « Effect of intrusion detection on reliability of mission-oriented mobile group systems in mobile ad hoc networks », *IEEE Transactions on Reliability*, vol. 59, p. 231–241, mars 2009.

[531] T. Wu et K. Chen, « Reliability and key-protection for computer-security systems », *IEEE Transactions on Reliability*, vol. 36, p. 113–116, avril 1987.

[532] R. G. Dromey, « Genetic design : Amplifying our ability to deal with requirements complexity », dans *International Workshop on Scenarios : Models, Transformations and Tools, International (Revised Selected Papers, LNCS 3466)*, Dagstuhl, Allemagne, p. 95–108, sept. 2003.

[533] M. Clavel, F. Durán, S. Eker, P. Lincoln, N. Martí-Oliet, J. Meseguer et C. Talcott, « The Maude 2.0 system », dans *Proceedings of the 14th International Conference on Rewriting Techniques and Applications (RTA 2003), LNCS 2706*, Valence, Espagne, p. 76–87, juin 2003.

[534] I. N. Fovino, M. Masera et A. De Cian, « Integrating cyber attacks within fault trees », *Reliability Engineering & System Safety*, vol. 94, p. 1394–1402, sept. 2009.

[535] J. Jürjens et S. H. Houmb, « Development of safety-critical systems and model-based risk analysis with UML », dans *Proceedings of the 1st Latin America Dependability Conference (LADC-1), LNCS 2847*, São Paulo, Brésil, p. 364–365, oct. 2003.

[536] J. Jürjens, *Secure Systems Development with UML*. Loughborough : Springer, May 7–8 2005. Invited talk.

[537] P.-Y. Chaux, « Apport du model-checking pour l'analyse qualitative de BDMP », Mémoire de master, École Normale Supérieure de Cachan, 2009.

[538] J. Hansson, L. Wrage, P. Feiler, J. Morley, B. Lewis et J. Hugues, « Architectural Modeling to Verify Security and Nonfunctional Behavior », *IEEE Security and Privacy*, vol. 8, n° 1, p. 43–49, 2010.

[539] J. Delange, L. Pautet et P. Feiler, « Validating safety and security requirements for partitioned architectures », dans *Proceedings of the 14th International Conference on Reliable Software Technologies (Ada-Europe 2009), LNCS 5570*, Brest, France, p. 30–43, juin 2009.

[540] « Arrangement on the recognition of Common Criteria certificates in the field of IT security ». http://www.commoncriteriaportal.org/files/operatingprocedures/cc-recarrange.pdf, mai 2000.

[541] Direction Centrale de la Sécurité des Systèmes d'Information (DCSSI), « Certification de sécurité de premier niveau des technologies de l'information ». N°915/SGDN/DCSSI/SDR, avril 2008. Version 2.4 (phase expérimentale).

[542] D. E. Bell et L. J. La Padula, « Secure computer systems : Mathematical foundations. », Rapport Technique MTR-2547, MITRE Corporation, mars 1973.

[543] K. Biba, « Integrity considerations for secure computer systems », Rapport Technique ESD-TR 76-372, MITRE Corporation, 1977.

[544] G. Graham et P. Denning, « Protection : principles and practice », dans *Proceedings of the Fall Joint Computer Conference*, Montvale, États-Unis, p. 417–429, nov. 1971.

[545] M. Harrison, W. Ruzzo et J. Ullman, « Protection in operating systems », *Communications of the ACM*, vol. 19, n° 8, p. 461–471, 1976.

[546] R. Lipton et L. Snyder, « A linear time algorithm for deciding subject security », *Journal of the ACM*, vol. 24, n° 3, p. 455–464, 1977.

[547] R. Sandhu, « The typed access matrix model », dans *Proceedings of the 1992 IEEE Symposium on Research in Security and Privacy (S&P'92)*, Oakland, États-Unis, p. 122–136, mai 1992.

[548] D. D. Clark et D. R. Wilson, « A comparison of commercial and military computer security policies », dans *Proceedings of the 1987 IEEE Symposium on Research in Security and Privacy (S&P'87)*, Oakland, États-Unis, p. 183–194, mai 1987.

[549] D. Brewer et J. Michael, « The Chinese Wall security policy », dans *Proceedings of the 1989 IEEE Symposium on Research in Security and Privacy (S&P'89)*, Oakland, États-Unis, p. 206–214, mai 1989.

[550] R. Sandhu, E. Coyne, H. Feinstein et C. Youman, « Role-based access control models », *Computer*, vol. 29, n° 2, p. 38–47, 1996.

[551] J. McLean, *Encyclopedia of Software Engineering*, chap. Security Models. Wiley, 1994.

[552] P. Ryan, J. McLean, J. Millen, V. Gligor et C. Mellon, « Non-interference, who needs it ? », dans *Proceedings of the 14th IEEE Computer Security Foundations Workshop (CSFW-13)*, Cap Breton, Canada, p. 237–238, 2001.

[553] R. Focardi et R. Gorrieri, *Foundations of Security Analysis and Design*, chap. Classification of security properties, p. 331–396. Springer, 2001.

[554] P. Y. Ryan, « Mathematical models of computer security », dans *Revised versions of lectures given during the IFIP WG 1.7 International School on Foundations of Security Analysis and Design : Tutorial Lectures, LNCS 2171*, Bertinoro, Italie, p. 1–62, sept. 2001.

[555] D. Pointcheval, *Provable Security for Public Key Schemes*, chap. Advanced Course on Contemporary Cryptology, p. 133–189. Birkhäuser Publishers, 2005.

[556] N. Koblitz et A. Menezes, « Another look at "provable security" », *Journal of Cryptology*, vol. 20, n° 1, p. 3–37, 2007.

Index