Détection contextuelle de cyberattaques par gestion de confiance à bord d'un navire

Benjamin Coste

UNIVERSITE BRETAGNE LOIRE / MATHSTIC

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

THESE DE DOCTORAT DE

L'ÉCOLE NATIONALE SUPERIEURE MINES-TELECOM ATLANTIQUE
BRETAGNE PAYS DE LA LOIRE - IMT ATLANTIQUE
COMUE UNIVERSITE BRETAGNE LOIRE

ECOLE DOCTORALE N° 601
Mathématiques et Sciences et Technologies de l'Information et de la Communication
Spécialité : *Informatique*

Par
Benjamin COSTE

Détection contextuelle de cyberattaques par gestion de confiance à bord d'un navire

Thèse présentée et soutenue à Brest, le 18/12/2018
Unité de recherche : IRENav et LaTIM (INSERM UMR 1101)
Thèse N° : 2018IMTA0106

Rapporteurs avant soutenance :

Aldo Napoli	Maître de Recherche, MINES ParisTech
Anne-Laure Jousselme	Scientist, NATO STO Centre for Maritime Research and Experimentation

Composition du Jury :

Président :	Laurent Nana	Professeur, Université de Bretagne Occidentale
Examinateurs :	Anne-Laure Jousselme	Scientist, NATO STO CMRE
	Aldo Napoli	Maître de Recherche, MINES ParisTech
	François Pinet	Directeur de Recherche, IRSTEA
Dir. de thèse :	Gouenou Coatrieux	Professeur, IMT Atlantique
Co-dir. de thèse :	Cyril Ray	Maître de conférence, IRENav

Invité(s)
Alain Deturche Chef Service Logiciel, Thales DMS

N° d'ordre : 2018IMTA0106

Sous le sceau de l'Université Bretagne Loire

IMT Atlantique
Bretagne-Pays de la Loire

En accréditation conjointe avec l'Ecole Doctorale MathSTIC

Détection contextuelle de cyberattaques par gestion de confiance à bord d'un navire

---

Thèse de Doctorat

Spécialité: Informatique

Présentée par **Benjamin Costé**

Département: Images et Traitement de l'Information

Laboratoire: Institut de Recherche de l'École Navale / LaTIM Inserm, UMR 1101, IMT Atlantique

Thèse réalisée au sein de la Chaire de cyberdéfense des systèmes navals

Directeurs de thèse : Gouenou Coatrieux & Cyril Ray

Soutenue le 18 décembre 2018

Jury :

Aldo Napoli, Maître de recherches, HDR, Mines ParisTech (Rapporteur)
Anne-Laure Jousselme, Senior researcher, CMRE (Rapporteuse)
Gouenou Coatrieux, Professeur, IMT Atlantique (Directeur de thèse)
Cyril Ray, Maître de Conférences, École Navale (Encadrant)
Laurent Nana, Professeur des Universités, Université de Bretagne Occidentale (Examinateur)
François Pinet, Directeur de recherches, IRSTEA Clermont-Ferrand (Examinateur)
Alain Deturche, Chef Service Logiciel, Thales Defense Mission Systems (Invité)

Table des matières

Introduction générale **1**

1 Sécurité des systèmes d'information navals **7**

1.1 Menace cyber 7

1.2 Systèmes navals 10

1.3 Typologie des moyens de sécurisation 14

1.3.1 Identifier 14

1.3.2 Protéger 15

1.3.3 Détecter 17

1.3.4 Répondre & Récupérer 19

1.4 Détection d'attaques cyber : cas des systèmes navals 19

2 La confiance : définitions et modélisations **25**

2.1 Une notion complexe 25

2.1.1 Un concept économique 26

2.1.2 Un concept social 28

2.1.3 Confiance et réseaux informatisés 33

2.1.4 Gestion de confiance dans les systèmes d'information 37

2.2 Modéliser et mesurer la confiance 40

2.2.1 Modèles de confiance 40

2.2.2 Mesures 44

2.2.3 Propagation 46

2.3 Discussion 49

3 Modélisation pour la sécurité d'un système d'information **55**

3.1 Systèmes d'information et complexité 55

3.1.1 Modéliser un système d'information : pré-requis 56

3.1.2 Caractérisation des systèmes complexes 57

3.1.3 Principes de modélisation des systèmes complexes 62

3.2 Modélisation d'un système d'information 64

4 Modéliser et mesurer la confiance au sein d'un système d'information 87

4.1 Modélisation de la confiance 88

4.1.1 Compétence 88

4.1.2 Sincérité 90

4.1.3 Risque et connaissance 91

4.2 Mesures de la confiance 93

4.2.1 Mesurer la compétence des sources 95

4.2.2 Mesurer la sincérité des sources 98

4.2.3 Mesurer la confiance des sources 101

4.3 Propagation de confiance dans un système d'information 103

4.3.1 Propagation horizontale 106

4.3.2 Propagation verticale 109

4.4 Discussion 111

5 Simulations et expérimentations 115

5.1 Conception d'un simulateur 115

5.1.1 Conception d'un système d'information de scénarisation de données et d'évaluation de la confiance 116

5.1.2 Implémentation 120

5.2 Expérimentations 122

5.2.1 Scénarios 123

5.2.2 Résultats 127

5.3 Discussion 137

Conclusion 139

Résumé 157

Table des figures

1 Diagramme de Venn du travail de thèse. *Sécurité des Systèmes d'Information, Trust-Based Access Control et Trust Aware Recommender Systems* 3

1.1 Classification des approches de détection basées sur l'analyse des flux réseaux 20

2.1 Recommandation et confiance directe 35

2.2 Théories utilisées en fusion de données (KHALEGHI et al., 2011) . . 42

2.3 Exemple d'un modèle utilisant 6 arguments et 6 attaques entre ces arguments. 45

2.4 Opérateurs de propagation de confiance et leurs propriétés associées 47

3.1 Structure hiérarchique d'un système complexe 59

3.2 Une source S distribue une information I 65

3.3 Un collecteur C reçoit des informations $(I_i)_{i=1..n}$ provenant de une ou plusieurs sources 67

3.4 Un bloc de traitement T reçoit des informations $(I_i)_{i=1..n}$ provenant de une ou plusieurs sources puis transmet le résultat de son traitement 67

3.5 Deux blocs ayant les mêmes entrées mais des sorties différentes . . . 68

3.6 Un bloc de rétroaction F traite une une même information plusieurs fois par itérations successives 68

3.7 Exemple de système modélisé par un ensemble de bloc 70

3.8 Relations binaires 72

3.9 Modélisation d'une mesure numérique via un canal gaussien 78
3.10 Exemple de simulation de mesures de vitesse 80
3.11 Types d'attaques pouvant cibler un bloc 81
3.12 Exemple d'une attaque par offset sur 10 informations 82
3.13 Exemple d'une attaque incrémentale sur 10 informations 83
3.14 Exemple de rejeu de 10 informations 84

4.1 Exemple de la modélisation d'un système d'information simple constitué de blocs fonctionnels 94
4.2 Modélisation d'une mesure numérique via un canal gaussien 95
4.3 Comparaison de plusieurs fonctions possibles pour une mesure compétence 97
4.4 Évolution des fonctions puissance selon le paramètre n 98
4.5 Différentes fonctions de confiance pour une source de compétence $Comp = 0.9$ 103
4.6 Sens de propagation de la confiance au sein d'un système d'information 105
4.7 Vue locale d'acquisition d'informations par un bloc 107
4.8 Illustration de la propriété de commutativité : les confiances de la source S_1 et du bloc T_2 se propagent simultanément au bloc T_1. . . 107
4.9 Impact des informations transmises par une source au travers d'une chaîne 109

5.1 Méthodologie de scénarisation des données 118
5.2 Chaîne de traitement des données pour produire des indices de confiance 119
5.3 Diagramme de structure 122
5.4 Écran de construction du scénario : choix de l'attaque 122
5.5 Trajectoire et positions du navire étudié 123
5.6 Extrait des informations contenues dans la base de données 124
5.7 Modèle de système utilisant 3 sources de vitesse employé dans les simulations 125

5.8 Modèle de système de navigation employé dans les simulations . . . 127
5.9 Évolution de trois mesures de confiance en présence de trois sources de vitesse sur un navire 129
5.10 Évolution de trois mesures de confiance en présence d'une falsification incrémentale des informations d'une source parmi trois 130
5.11 Évolution des mesures de confiance en présence d'un rejeu des informations d'une source parmi trois 131
5.12 Évolution des mesures de confiance en présence de deux attaques sur deux sources différentes 132
5.13 Propagation horizontale en présence d'une attaque sur un GPS . . . 133
5.14 Propagation horizontale en présence d'une attaque sur un Loch . . 134
5.15 Propagation horizontale en présence d'une attaque sur composant interne du SI 135
5.16 Propagation verticale de confiance dans un GPS 136

Liste des tableaux

1.1 Secteur maritime : les infrastructures sensibles face aux cybermenaces 11

2.1 Matrice des gains pour un dilemme du prisonnier 26
2.2 Typologie de la confiance selon Lewis 29
2.3 Diverses représentation de la confiance (De Cock et Da Silva, 2006) 44
2.4 Règles de propagation de la confiance selon différents comportements 49
2.5 Synthèse des définitions de la confiance et des critères associés . . . 51
2.6 Critères de choix pour mesurer la confiance dans une entité inconnue au sein d'un réseau 53

3.1 Liste des préceptes des approches analytique et systémique 63
3.2 Tableau récapitulatif des relations 77

4.1 Tableau des propriétés des mesures associées à la compétence, la sincérité et la confiance 104
4.2 Propriétés des opérateurs de propagation de la confiance 106

5.1 Tableau récapitulatif des scénarios testés 128

Remerciements

Loin d'être un travail individuel, une thèse cristallise les contributions, qu'elles aient été directes ou indirectes, de nombreuses personnes ayant mené à sa réussite. Bien que ces personnes soient trop nombreuses pour être chacune citées individuellement dans ce qui suit, je ne leur en suis pas moins reconnaissant.

Parmi les premières personnes dont je souhaite honorer la contribution, il y a bien sûr mes directeurs de thèse, Cyril Ray et Gouenou Coatrieux. Malgré des disponibilités souvent réduites, ils ont toujours su partager leur expérience, me guider et m'aider, parfois sans me le dire, à avancer sereinement. Cette thèse leur doit beaucoup et ils ont pour cela toute ma gratitude.
Je remercie également Alain Deturche pour avoir toujours manifesté le plus vif intérêt envers mes travaux et m'avoir apporté son expérience sur la mise en œuvre dans des situations concrètes.

Un grand merci à Anne-Laure Jousselme et Aldo Napoli pour avoir accepté de rapporter cette thèse, leurs commentaires constructifs et leur intérêt pour mes travaux, mais aussi aux autres membres du jury, Laurent Nana et François Pinet, pour leurs questions et les perspectives ouvertes.

J'ai eu le privilège de connaître à ses débuts celle que ceux qui la connaissent nomment simplement « la Chaire ». Pedro, Guillaume, Bastien, Thibaud, Etienne, Arthur, merci pour tous ces bons moments passés à travailler, échanger, jouer, et hacker. Ces moments passés ensemble m'ont beaucoup apporté. Je remercie également David et Yvon pour leur investissement dans la construction et l'animation de la Chaire et leurs conseils précieux ainsi que Xavier, Erwan, Thomas, Gaël et Olivier pour m'avoir fait bénéficier de leur temps et de leur expérience.

Merci à l'ensemble du personnel de l'IRENav, notamment les doctorants et post-doctorants qui se sont succédés, pour m'avoir accueilli toutes ces années et qui

ont rendu cette aventure à nulle autre pareille. Je garderai longtemps les souvenirs de ces belles années passées en votre compagnie. Je remercie tout particulièrement Loïc et Léa pour leur énergie et leur enthousiasme communicatif à animer le laboratoire tant à l'intérieur qu'à l'extérieur.

Cette thèse a été l'occasion de rencontres et même si nos échanges se font trop rares, je remercie Pierre-Emmanuel et Káthia pour leur bienveillance et leurs conseils.

En dehors du cadre officiel, je remercie Solenn, alias Binôme, qui a contribué à cette thèse à bien des niveaux sans jamais se départir de sa bonne humeur, pour me rappeler qu'il faut parfois prendre le temps de se poser (autour d'un chocolat par exemple).

Pour les parties de cartes, les courses de stylo, les anecdotes rocambolesques sur la vie des célébrités ou de notre département de mathématiques préféré, les échanges tout à la fois sérieux et décalés, leur enthousiasme, leur soutien et leur amitié, je remercie Marion, Flo et Aurélie.

Enfin, pour son indéfectible soutien tout au long de ces années, je remercie ma famille sans qui cette aventure n'aurait jamais commencé.

Introduction générale

Les technologies de l'information sont aujourd'hui omniprésentes dans nos sociétés. Ordinateurs, smartphones et objets communiquant de tous types sont de plus en plus interconnectés ce qui amène de nouvelles problématiques quant à leur sécurisation. Par voie de conséquence, tous les domaines nouvellement numérisés (industrie, médecine, transport, etc.) font désormais face à de nouvelles menaces contre lesquelles leurs acteurs n'étaient pas préparés.

Besoins

Dans le domaine maritime, la maîtrise de la navigation et de la conduite d'un navire sont deux aspects essentiels pour la bonne marche et la sécurité du navire, des personnels et la préservation de l'environnement maritime. Or, les navires modernes (notamment les navires militaires mais aussi les grands navires de croisière), embarquent de plus en plus de technologies informatisées et automatisées standards et à haut niveau d'intégration pour gérer ces fonctions primordiales. Les bénéfices de cette informatisation sont multiples : réduction des effectifs, accroissement des capacités du navire, augmentation de l'efficacité des opérateurs, réduction des coûts de maintenance, interchangeabilité des équipements, ...

Cependant, cela génère aussi de nombreuses faiblesses et failles de sécurité bien souvent accrues par la standardisation massive des équipements embarqués et leur complexité croissante. Les problématiques de sécurité (les attaques) peuvent affecter l'ensemble des éléments du navire, notamment les systèmes de navigation (passerelle intégrée et système de navigation, positionnement et cartographie numérique ECDIS, autopilote, ...) et les systèmes de plateforme (systèmes de gestion intégrés de la plateforme (IPMS), gestion des réseaux et des équipements

électriques, la propulsion, la manœuvre ...). Les vulnérabilités affectant un navire moderne, qu'il soit militaire ou civil, peuvent intervenir à trois niveaux : d'une part, les systèmes d'information et de communication « classiques » à base de systèmes d'exploitation usuels tels que Windows ou Linux et pour lesquels les vulnérabilités font l'objet de travaux actifs ; d'autre part, les systèmes d'armes (seule exclusivité des navires militaires) et les systèmes de contrôle et d'acquisition de données (SCADA), tous deux très mal protégés et qui font l'objet d'attaques croissantes. Le fonctionnement de ces systèmes repose en partie sur les capacités du navire à percevoir son environnement interne et externe aux travers de capteurs (*e.g.* radar) et par les informations qu'il reçoit (*e.g.* mise à jour d'une carte nautique). Les perturbations ou malversations de ces capteurs ont été récemment démontrées et soulèvent évidement des inquiétudes car les mécanismes de détection et protection n'existent pas ou peu.

Dans le contexte des systèmes d'information (SI), la confiance est considérée comme un paramètre important de sa sécurité présumée. De fait, les multiples mécanismes de sécurité d'un système visent, entre autres choses, à restaurer la confiance des utilisateurs dans celui-ci. Cette confiance est en effet considérée perdue avec l'émergence grandissante de menaces variées telles que celles citées précédemment.

Démarche et objectifs de la thèse

Les acteurs qui produisent et traitent l'information utilisée par un SI peuvent également se révéler malveillants à son égard. Du point de vue du système, ces acteurs sont soit des sources d'information, c.-à-d. identifiés par le SI comme producteurs de l'information, quelle qu'en soit la provenance initiale, soit des composants internes. Cependant, le faible recours à des mesures de protection, dû à des contraintes opérationnelles ou à un manque de sensibilisation et de formation des équipages, nous amène à considérer la sécurisation des navires modernes sous l'angle de la détection des attaques affectant les informations de navigation manipulées par le système d'information du bateau.

Nous proposons dans cette thèse d'aborder la problématique de la détection des falsifications d'informations de navigation sous l'angle de la **confiance** dans le système d'information. Notre démarche se situe à la confluence de trois domaines (cf. figure 1) : la sécurité, les systèmes d'information et l'étude de la confiance. Le

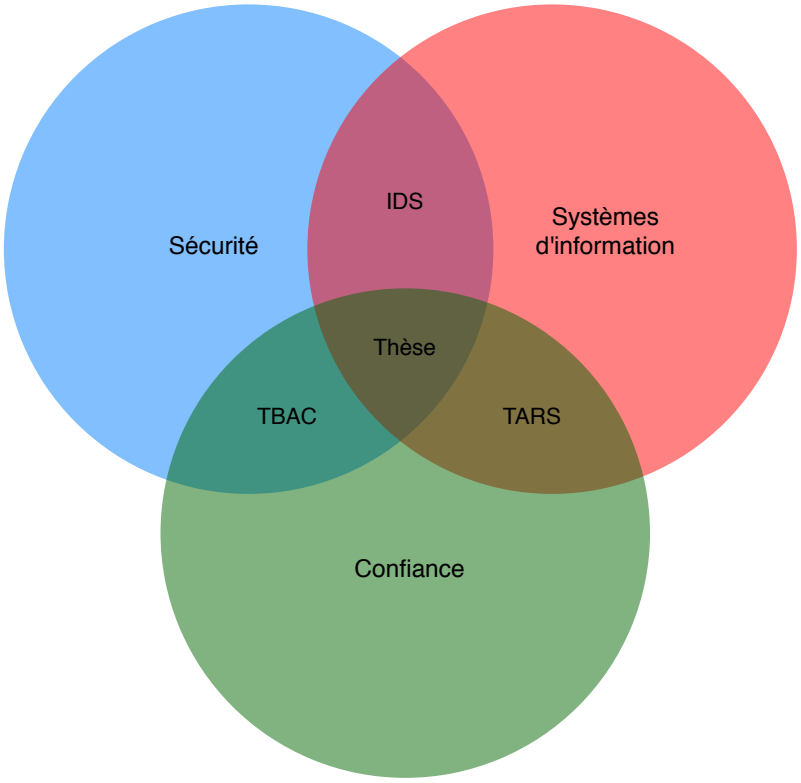


FIGURE 1 – Diagramme de Venn du travail de thèse. *Sécurité des Systèmes d'Information, Trust-Based Access Control et Trust Aware Recommender Systems*

domaine de la sécurité recouvre l'ensemble des moyens permettant de se placer dans un « état d'esprit confiant et tranquille, qui résulte du sentiment, bien ou mal fondé, que l'on est à l'abri de tout danger » (Centre National de Ressources Textuelles et Lexicales). En particulier, l'application de techniques de sécurité aux systèmes d'information (SSI) a donné naissance à de nombreux outils et techniques visant à protéger et défendre le SI contre les menaces tant extérieures qu'intérieures. L'utilisation de la confiance dans des outils dédiés à la sécurité est apparue avec le *Trust-Based Access Control* (TBAC). Le contrôle d'accès a pour objectif de maitriser les accès aux ressources d'un système d'information. Il repose pour cela sur des mécanismes d'authentification et d'autorisation effectués par des éléments spécifiques du SI (SINCLAIR et SMITH, 2008). La particularité des systèmes basés sur la confiance est de permettre une décentralisation du processus d'autorisation : plusieurs éléments peuvent appliquer séparément des règles de contrôle d'accès tout en garantissant une cohérence globale (c.-à-d. qu'une même entité ne bénéficie pas d'accès différents selon qui donne l'autorisation).

En dehors du contexte de la sécurité, l'utilisation de la confiance a déjà été abordée par l'étude des systèmes de recommandation. Les systèmes de recommandation sont des logiciels et des techniques qui suggèrent à un utilisateur donné des objets, contenus ou services en fonction de leur utilité ou pertinence pour celui-ci (Adomavicius et al., 2011). L'intégration de la confiance dans de tels systèmes, appelés alors *Trust Aware Recommender System* (TARS), a permis d'augmenter la pertinence des suggestions dans les approches collaboratives (Massa et Avesani, 2007a) ; approches dont les recommandations sont basées sur l'étude des similarités entre utilisateurs.

Notre démarche se situe à contre-pied des approches classiques qui augmentent le niveau de sécurité des équipements pour restaurer la confiance que leur accordent leurs utilisateurs. En effet, nous proposons une approche qui repose sur la confiance en tant que moyen de sécurisation des systèmes d'information. Pour cela, nous montrons comment construire un système de détection à partir d'une modélisation de la confiance. Dans ce contexte, les questions suivantes demeurent ouvertes :

- Comment définir la confiance dans le contexte des systèmes d'information ?
- Comment construire une mesure de confiance qui tient compte de la possibilité d'une falsification de l'information ?
- Comment évaluer la confiance dans un système d'information autant que dans ses composants ?

Nous ne considérons pas le consommateur des informations produites par notre système comme faisant partie de notre problème d'étude. Par conséquent, les aspects relatifs à la visualisation des informations et l'identification des attaques ainsi que des attaquants ne sont pas pris en compte dans notre travail.

Plan

Ce document est structuré en 5 chapitres. Le premier chapitre présente le contexte spécifique de la cybersécurité, en particulier dans le domaine naval. Il permet de comprendre les différents moyens et motivations d'un attaquant ainsi que les besoins et contraintes spécifiques des systèmes navals en matière de cybersécurité. De plus, les solutions de sécurité existantes, notamment les moyens de détection, seront présentées.

Le deuxième chapitre introduit la notion de confiance à travers différents domaines d'étude et en particulier ses évolutions vis-à-vis des réseaux modernes (nouveaux acteurs et moyens de communication). Il pose également les bases des problématiques de modélisation, de mesure et de propagation de la confiance telles qu'elles sont connues dans la littérature.

Le troisième chapitre expose nos choix de modélisation pour la sécurité d'un système d'information. En s'appuyant sur la théorie des systèmes complexes, nous proposons un modèle de système composé de blocs fonctionnels et leurs relations. Une modélisation des informations ainsi que des attaques pouvant cibler un SI est également présentée.

Le quatrième chapitre propose notre définition de la confiance dans le contexte des systèmes d'information. L'élaboration de mesures de confiance dans des sources d'information puis de propagation au sein du système est abordée.

Le cinquième et dernier chapitre expose les résultats expérimentaux obtenus et les conclusions qui en découlent.

Chapitre 1

Sécurité des systèmes d'information navals

Les océans recouvrent presque les trois quarts de la surface terrestre et le transport maritime représente plus de 90% du total mondial. Cet espace constitue autant une importante ressource (économique, environnementale, scientifique, etc.) qu'il est une source de conflits, notamment de type cyber. Ce chapitre dresse un état des lieux de la cybersécurité dans le domaine maritime et positionne notre approche de détection par rapport à celles déjà connues. Tout d'abord, nous introduisons en section 1.1 la problématique de la menace cyber ainsi que les moyens dont dispose un éventuel attaquant. Nous poursuivons en section 1.2 en présentant la surface d'attaque des systèmes navals : leurs spécificités ainsi que leurs besoins en matière de cybersécurité. Une typologie des différentes solutions disponibles, envisagées voire mises en œuvre pour protéger et défendre de tels systèmes est ensuite exposée en section 1.3. Enfin, la section 1.4 présente les différentes approches de la littérature en matière de détection d'attaques cyber.

1.1 Menace cyber

Le 20e siècle a connu un essor important des technologies numériques. D'abord exploitées pour leur efficience à réaliser des tâches requérant une automatisation des moyens de calcul, elles ont aussi amené de nouveaux usages notamment grâce à l'ajout d'interfaces de communications standardisées et peu chères (Atzori et al., 2010). Toutefois, la prolifération massive de ces technologies a engendré de nouveaux risques pour la sécurité de leurs usagers, des équipements et des données. En effet, les outils numériques n'ont pas toujours été conçus en prenant en compte la malveillance possible d'un ou plusieurs utilisateurs. De plus, leur mise en réseau

massive expose désormais ces outils numériques à des menaces mondiales ; par opposition aux menaces physiques [1] qui sont plutôt locales, c'est-à-dire que l'attaquant a un accès direct à sa cible. Encouragés par cette déformation des distances et l'anonymat relatif qu'elle procure, les attaques utilisant les technologies numériques ont connu une importante évolution tant en volume qu'en sophistication (Choo, 2011).

Dans ce travail, nous considérons comme étant un attaquant toute entité ayant des intentions malveillantes à l'égard d'un système d'information. Qu'il s'agisse de groupes ou d'individus isolés, les attaquants sont le plus souvent motivés par des raisons idéologiques ou financières. Les activistes (ou hacktivistes) cherchent à perturber les activités ou entacher la réputation de leur cible (ANSSI, 2017). Les criminels recherchent le gain financier en rançonnant leurs victimes ou en revendant les données dérobées (*e.g.* espionnage industriel) (Mohurle et Patil, 2017). Les états s'espionnent (Langner, 2011) ou cherchent à bouleverser l'équilibre politique (Untersinger, 2017). Quelques soient leurs motivations, les attaquants potentiels disposent de plus en plus de moyens techniques, financiers et humains pour arriver à leurs fins.

Techniquement, une attaque est une action ou une suite d'actions qui consiste à exploiter une ou plusieurs vulnérabilités du système ciblé afin d'en perturber le fonctionnement. Cette perturbation altère la confidentialité, l'intégrité, la disponibilité voire la traçabilité de l'information manipulée par le système (Disterer, 2013). Selon la complexité de l'attaque et les défenses mises en œuvre, différents moyens peuvent être utilisés parmi lesquels :

- Attaques matérielles : elles ciblent directement le composant physique et ont pour but principal l'extraction d'informations sensibles. Ces attaques sont spécifiques à une plateforme donnée et peuvent se révéler onéreuses. Exemples : Attaques par canaux cachés (Kocher et al., 1999 ; Genkin et al., 2014), démarrage à froid (Yitbarek et al., 2017), attaques ciblant les réseaux sans-fils (Juels, 2006 ; Shiu et al., 2011).
- Attaques logicielles : elles ciblent les vulnérabilités résultant de la conception du programme. Ces vulnérabilités sont dues à une mauvaise programmation (*e.g.* utilisation de fonctions non sécurisées) ou une sous-utilisation des options de sécurité du compilateur (ASLR, stack protector). Exemples : injection de code (Halfond et al., 2006), dépassement de tampon, XSS/CSRF.

1. c.-à-d. qui ne nécessitent pas de recours à un outil informatique

- Attaques réseau : elles exploitent les failles dans la conception des protocoles. Nous regroupons sous cette appellation les attaques qui détournent la sémantique du protocole autant que celles qui utilisent le séquencement des actions permises par celui-ci. Exemples : DNS/IP spoofing (Ehrenkranz et Li, 2009), Homme du milieu, Déni de service (Srivastava et al., 2011)
- Attaques cryptographiques : elles ciblent directement les primitives cryptographiques utilisées (Boneh, 1999).
- Ingénierie sociale : elles exploitent des biais du comportement humain. Elles ne sont pas directement considérées comme des cyberattaques mais sont souvent employées pour faciliter une intrusion. Exemples : hameçonnage, biais cognitifs, point d'eau, typosquatting.

Enfin, selon les objectifs de l'attaquant, ses actions peuvent être ciblées ou non (BIMCO et al., 2016). Par exemple, une attaque par point d'eau ou par hameçonnage s'adressent à un nombre de cibles le plus important possible afin d'en maximiser la portée. Au contraire, les attaques ciblées nécessitent plus d'investissement financier et humain pour produire des effets. Ces attaques sont toutefois plus difficiles à détecter car elles sont spécifiques à la cible : le contournement de ses moyens de défense est inclus dans la méthodologie de l'attaquant. Une attaque ciblée nécessite de contourner plusieurs protections. La méthodologie poursuivie par l'attaquant peut alors se résumer selon la *cyber-killchain* (Yadav et Rao, 2015) :

1. Reconnaissance : l'attaquant s'informe sur sa cible afin de déterminer les vulnérabilités exploitables.
2. Armement : L'attaquant prépare une arme qui exploite les vulnérabilités découvertes.
3. Livraison : L'attaquant envoie son arme à l'environnement ciblé.
4. Exploitation : L'arme se déclenche sur le système cible et exploite les vulnérabilités.
5. Installation : L'attaquant prend possession du système et s'octroie des accès durables.
6. Contrôle : L'attaquant met en place un accès privilégié vers un système sous son contrôle pour envoyer des commandes ou stocker les information exfiltrées.
7. Action : L'attaquant réalise son objectif (*e.g.* exfiltration de données, destruction du système, effacement de traces, etc.)

D'autres méthodologies existent mais reposent également sur les phases de reconnaissance-armement, de livraison de la charge, de l'implantation dans le système puis de la réalisation de l'attaque proprement dite (BIMCO et al., 2016).

Sans prétendre à l'exhaustivité, cette section a présenté une liste des motivations et moyens d'une entité cherchant à compromettre un système d'information. Les propos développés dans cette section ne sont pas spécifiques au domaine maritime. En effet, les motivations d'un attaquant ne dépendent que de lui-même. La concrétisation de ses objectifs est quant à elle dépendante des caractéristiques de la cible. C'est l'objet de la section suivante qui expose la surface d'attaque des infrastructures du domaine maritime, en particulier les navires.

1.2 Systèmes navals

Le domaine maritime n'a pas échappé à la numérisation mais a accordé peu d'attention à l'aspect sécurité (Cimpean et al., 2011). Le tableau 1.1 montre les menaces pouvant peser sur les systèmes d'information dans le domaine maritime. Ces menaces peuvent se décomposer en trois grands axes selon les cibles (Bothur et al., 2017) : les systèmes industriels et automates embarqués, les infrastructures portuaires ainsi que les systèmes liés à la navigation, à son contrôle et à la sécurité nautique, avec des vulnérabilités particulières autour du système de positionnement GPS et de l'Automatic Identification System (AIS).

Chaque cible comporte des contraintes spécifiques qui rendent parfois difficiles la mise en œuvre d'outils classiques de sécurité. Par exemple, les systèmes industriels embarqués sur les navires sont soumis à de fortes contraintes de disponibilité, voire de performance. En effet, les automates industriels sont chargés d'assurer la régulation des processus, l'acquisition et le traitement de données ainsi que le contrôle des équipements physiques (*e.g.* valves, pompes, turbines, etc.) (ANSSI, 2015). Selon les fonctions assurées par l'automate, son dysfonctionnement peut entraîner la perte du contrôle de la propulsion, de l'énergie voire du système de combat dans le cas d'un navire militaire. Du fait des fortes contraintes de disponibilité imposées par ces systèmes, les dispositifs de sécurité intégrés doivent répondre aux mêmes contraintes, y compris lorsque l'automate doit répondre dans un temps donné (*e.g.* commande de refroidissement du moteur). Par conséquent, les dispositifs classiques de sécurité ne peuvent pas toujours être utilisés. Il convient alors d'assurer une défense à la périphérie de ces systèmes industriels.

Infrastructures sensibles	Systèmes d'information utilisés	Risques en cas de cyberattaque
• Infrastructure portuaire • Navire de pêche • Navire marchand • Grand navire de tourisme • Bâtiment militaire • Câbles sous-marins et satellites	• Maintenance des navires • Gestion automatique des installations (mécanique, carburant ...) • Gestion automatique de la logistique (gestion des containers ...) • GPS et cartes maritimes électroniques • Système d'alerte incendie • AIS : système d'échange d'informations sur l'identité d'un bâtiment, sa position, sa route • Système de combat • Système d'arme • Télécommunications	• Perte de la marchandise • Retard d'approvisionnement • Perte de contrôle du navire • Contrôle maritime faussé • Déclenchement constant : altération de l'image de l'entreprise • Vol de données • Impact mission

TABLE 1.1 – Secteur maritime : les infrastructures sensibles face aux cybermenaces

De plus, ces derniers sont parfois soumis à des conditions extrêmes de fonctionnement (températures, poussières, vibrations). Afin de réduire les risques liés à une intervention humaine et de faciliter leur administration, celle-ci est parfois réalisée par un opérateur distant : un moteur à fond de cale n'est pas facilement accessible mais nécessite une supervision régulière du fait de son importance critique pour le bon fonctionnement du navire. Enfin, les navires ont souvent une très longue durée de vie pouvant atteindre plusieurs dizaines d'années. De ce fait, en l'absence de mises à jour, les logiciels utilisés deviennent rapidement obsolètes au regard de l'évolution des technologies de l'information. Pour des raisons de stabilité puis de compatibilité, les logiciels embarqués sont peu mis à jour (*e.g.* présence durable de Windows XP). Au contraire, l'écosystème numérique se renouvelle très rapidement ce qui conduit à des incidents tels que celui subi par l'armateur Maersk, victime d'un rançongiciel ciblant des machines Windows non mises à jour.

Les systèmes industriels embarqués posent de nombreux défis techniques du point de vue de la sécurisation des navires mais ce ne sont pas les seuls : les systèmes de navigation sont également vulnérables (Schmidt et al., 2016). Le système de navigation permet au navire de se mouvoir en mer sur la base d'informations fournies par les capteurs. La plupart des informations acquises par le système de navigation sont ensuite diffusées au sein du système. Par exemple, la mesure du temps à partir des horloges des satellites du système GPS sert de référence au reste des équipements du navire. Ces équipements (GPS, AIS, ...) n'ont cependant pas été conçus en tenant compte des possibilités de malversations et leurs informations ont été prouvées falsifiables à peu de frais (Balduzzi et al., 2014). Par ailleurs, la mise à jour des logiciels de passerelle, quand ceux-ci ne sont pas reliés à un réseau, est réalisée via des médias amovibles provenant de réseaux parfois non contrôlés (*e.g.* prestataire externe) (BIMCO et al., 2016). En conséquence, du fait de leur rôle prépondérant au sein des navires, les systèmes de navigation constituent des cibles de choix pour des pirates et ce d'autant plus qu'il n'offrent peu ou pas de protection contre les malversations dont il pourraient être victimes.

Enfin, les infrastructures portuaires posent également de multiples défis concernant leur sécurité ainsi que celle des navires. En effet, le navire n'est plus seul, isolé en pleine mer, mais reste en contact régulier avec de multiples acteurs du monde maritime (Fitton et al., 2015).

L'information circule au travers de multiples canaux ayant chacun ses spécificités en termes d'accessibilité, de fiabilité et de sécurité. Par exemple, un câble sous-marin est moins sujet aux altérations de l'information qu'une liaison satellite

mais il est plus difficile de relier un navire à un câble que d'établir une connexion satellite. Au contraire, les sites terrestres privilégient les communications par câble pour leur fiabilité. Tous les acteurs échangent donc de l'information au travers de divers moyens de communication, chacun disposant de ses propres vulnérabilités. Cette forte hétérogénéité des moyens de communications est combinée à un nombre élevé d'acteurs interagissant simultanément, notamment dans les ports (*e.g.* chargement/déchargement, mise à quai, transport de containers). Dans ces conditions, il devient difficile de déterminer d'où provient une information et donc *a fortiori* de lui faire confiance. Chaque acteur manipulant une information possède en effet des vulnérabilités qui lui sont spécifiques ainsi que sa propre politique de sécurité associée (Tam et Jones, 2018) ; au sein d'un système, une politique de sécurité spécifie *qui* doit accéder à *quoi* et dans *quelles circonstances* (Dacier, 1994). Chacun est donc soumis à diverses menaces qui peuvent altérer ses informations qui sont ensuite diffusées aux acteurs à proximité. À chaque étape de son acheminement, vers le navire dans notre contexte, l'information est donc soumise à une multitude de risques menaçant son intégrité, ce qui remet en cause la confiance que l'on peut lui accorder.

Qu'il soit à terre pour gérer les marchandises et la mise à quai des navires ou embarqué en tant que membre d'équipage ou touriste en croisière, l'humain est partie prenante dans la sécurité du navire. En effet, il produit, traite et consomme de l'information au même titre que les autres éléments du navire. La formation de l'équipage à la gestion des incidents liés aux technologies numériques est donc primordiale afin d'assurer la sécurité du navire. De plus, l'automatisation des navires modernes réduit la taille des équipages et augmente donc la responsabilité de chacun de leurs membres. Un incident de sécurité est le signe d'une défaillance des moyens (automatisés) de protection et demande donc une importante contribution humaine pour être traité. Par conséquent, la formation en cybersécurité des personnes devient d'autant plus nécessaire au regard de leur rôle prépondérant dans la gestion des cyber-risques. Outre les erreurs (non-intentionnelles) du personnel naviguant, ceux-ci peuvent également délibérément porter atteinte à la sécurité du système en ignorant la politique de sécurité (non-malveillance) ou en y contrevenant directement dans l'intention de nuire (Arduin, 2018). L'humain fait donc partie intégrante du système d'information du navire et doit à ce titre être pris en compte lors de l'élaboration de systèmes de sécurité.

Cette section a montré différents aspects de la sécurité des systèmes d'information dans le domaine maritime. Ces systèmes se caractérisent par une forte hé-

térogénéité (capteurs, équipements informatiques, automates industriels, humains) ainsi que par une très longue durée de vie pouvant aller jusqu'à plusieurs décennies. La numérisation croissante des infrastructures maritimes, en particulier des navires, accroit significativement les risques auxquels elles sont confrontées.

1.3 Typologie des moyens de sécurisation

Face à la menace sans cesse grandissante des attaques numériques ciblant les systèmes d'information maritimes, des solutions émergent. Le *National Institute of Standards and Technology* est un organisme américain rattaché au département du Commerce et chargé de la promotion de l'innovation et de la compétitivité économique en développant des technologies et standards, notamment sur la cybersécurité. Il publie la première version du *Cybersecurity Framework* en 2014. Celle-ci propose une méthodologie, articulée autour de cinq fonctions, pour évaluer et gérer les vulnérabilités liées aux outils numériques (National Institute of Standards and Technology, 2014). En 2017, la version 1.1 est soumise à une relecture publique ainsi qu'aux propositions d'évolutions. À travers le prisme du *Cybersecurity Framework*, nous présentons dans cette section les solutions émergentes dans le domaine de la cybersécurité maritime.

1.3.1 Identifier

Afin de se prémunir efficacement contre des menaces, il faut d'abord pouvoir les identifier. La caractérisation des menaces passe par une prise de conscience au niveau global puis, plus précisément, par une analyse de risques menée en amont de la mise en place de mesures de sécurité adéquates.

En juin 2016, l'Organisation Maritime Internationale a publié des « Directives intérimaires sur la gestion des cyber-risques maritimes » (MSC.1/Circ.1526). Ces directives reconnaissent l'existence (et l'importance) des cyber-risques maritimes. Elles fournissent des recommandations de haut niveau sur leur gestion, lesquelles visent à protéger les transports maritimes contre les cyber-menaces et vulnérabilités actuelles et émergentes. Elles comprennent également les éléments fonctionnels sur lesquels repose une gestion efficace des cyber-risques. Elles proposent une approche top-down de la gestion des cyber-risques : les cadres dirigeants doivent être sensibilisés à la problématique pour créer une culture de sensibilisation et mettre

en place puis entretenir les mesures de gestion de ces risques. Outre le contenu des directives, leur importance tient surtout à la reconnaissance, par un organisme officiel et reconnu, des risques liés à l'utilisation des technologies numériques en milieu maritime.

Par ailleurs, le Comité de la sécurité maritime a également adopté la résolution MSC.428(98) sur la gestion des cyber-risques maritimes dans le cadre des systèmes de gestion de la sécurité. Cette résolution marque la prise de conscience des spécificités des cyber-risques dans le cadre maritime ainsi que la nécessité de les prendre en considération lors de l'évaluation des système de gestion de la sécurité [2].

La prise de conscience des organismes de référence d'organismes tels que l'OMI permet de se sentir concerné par les cyber-menaces émergentes et donc d'initier une démarche de sécurisation des systèmes d'information maritimes. Toutefois, les préconisations en matières de sécurité sont nécessairement génériques et ne permettent pas de mettre en place une défense adaptée. L'identification des menaces pouvant cibler un système ou une organisation précise se réalise à l'aide d'analyses qui identifient les risques et estiment leurs niveaux qualitativement ou quantitativement (Mayer, 2009). Ces analyses de risques permettent de caractériser les vulnérabilités spécifiques des composants des systèmes mais également leur occurrence en établissant la facilité d'accès d'un attaquant aux différents éléments du système d'information (Polatidis et al., 2018).

1.3.2 Protéger

Afin de préserver les navires des multiples risques menaçant leur sécurité, des moyens de protection, plus ou moins spécifiques au domaine maritime, émergent.

Depuis 2015, la Direction des Affaires Maritimes (DAM), administration française chargée d'élaborer et de mettre en œuvre au niveau national la réglementation dans le domaine maritime, a entamé une démarche de sensibilisation et de sécurisation des navires. Menée conjointement avec l'Agence Nationale pour la Sécurité des Systèmes d'Information (ANSSI), cette démarche a abouti à la production de trois rapports servant de base à toute démarche ultérieure en matière de sécurité.

Le premier rapport est le résultat d'une enquête menée par la DAM ainsi que d'un audit conduit par l'ANSSI sur 68 navires sous pavillon français. Il conclut sur 7 recommandations pour améliorer la protection des navires :

2. Au sens de l'anglais *safety*, c.-à-d. qui ne concerne que les risques accidentels.

1. Confiance numérique : réaliser une évaluation de la sécurité des systèmes d'information du navire.
2. Gouvernance : rédiger une politique de sécurité spécifique à la compagnie.
3. Gestion des accès et des fuites de données : appliquer des mesures d'hygiène en matière de gestion des systèmes d'information du navires.
4. Sécurisation des transactions : contrôler les échanges des systèmes d'information du navire.
5. Continuité de service : mettre en place un plan de continuité de fonctionnement après un incident.
6. Traçabilité et audit : contrôler et gérer les incidents des systèmes d'information du navire.
7. Confidentialité : appliquer des mesures de protections physiques des systèmes.

Ces mesures ne sont pas spécifiques au domaine maritime. Néanmoins, l'intérêt est avant tout leur transposition dans le domaine naval. L'ensemble de ces mesures a pour but de limiter voire d'empêcher toute intrusion dans le système d'information. Elles peuvent s'appliquer tant à des systèmes existant qu'à la mise en place de nouveaux SI. Elles nécessitent toutefois la réalisation d'une analyse des risques menaçant la sécurité du système.

Le second rapport porte sur l'analyse des risques et les moyens à mettre en œuvre dans le cas spécifique des systèmes industriels embarqués à bord des navire. Il statue sur 3 niveaux de protection d'une installation industrielle par une compagnie maritime. Le premier niveau consiste à adopter une approche globale de sécurisation du système d'information. Lors de l'élaboration d'une politique de sécurité des systèmes d'information, les systèmes industriels embarqués doivent être considérés au même titre que les systèmes bureautiques ou d'administration. Ils doivent donc faire partie des procédures en cas d'incident et être soumis aux mêmes mesures de protection que celles définies dans le premier rapport (cf paragraphe précédent). Le second niveau préconise l'utilisation d'outils de surveillance passive au niveau soit du réseau avec des sondes de détection d'intrusion soit du système en surveillant les changements de configurations des automates. Le troisième niveau concerne le durcissement des systèmes : utilisation d'équipement certifiés et préparés à faire face à un acte de malveillance. Par exemple, la gamme d'automates Simatic S7-1500 de Siemens ainsi que l'automate M580 de Schneider Electric ont récemment obtenu la Certification de Sécurité de Premier Niveau (CSPN) décer-

née par l'ANSSI [3] [4]. Cette certification valide des exigences minimales de sécurité telles que le stockage sécurisé des données utilisateurs, l'authentification à l'interface d'administration ou encore l'intégrité des programmes exécutés. L'utilisation de tels automates durcis, couplée à celle d'équipements de sécurité (pare-feux, antivirus, logiciels de supervision), est donc une bonne pratique en matière de protection des systèmes d'information industriels. Outre les automates, des pistes de sécurisation des capteurs utilisé par les systèmes de navigation ont également été proposées. Elles s'appuient principalement sur l'utilisation de moyens de chiffrement et d'authentification dans des systèmes tels que le GPS (Kröner et Dimc, 2010) et l'AIS (Goudossis et Katsikas, 2018).

Enfin, l'Agence Nationale pour la Sécurité des Systèmes d'Information a publié un guide de recommandations pour la gestion des systèmes d'information dans le domaine maritime [5]. Ce guide s'adresse à la fois aux utilisateurs des systèmes ainsi qu'à leurs administrateurs. Il reprend essentiellement les principes exposés dans leur guide d'hygiène numérique [6] : cloisonnement des réseaux, sécurisation des réseaux sans-fils, mises à jour et sauvegardes régulières, utilisation de mots de passe robustes, maîtrise des logiciels installés, prudence dans l'utilisation de sa messagerie électronique, etc. Ce rapport vise à sensibiliser les équipages et les compagnies maritimes afin d'appréhender les risques liés aux technologies numériques des navires. Il complète donc les deux rapports précédents en intégrant le facteur humain à toute démarche en matière de cybersécurité maritime.

1.3.3 Détecter

Les moyens de protection servent à réduire les risques identifiés par une analyse préalable. Néanmoins, ceux-ci souffrent de plusieurs limitations.

D'une part, les mesures de protection imposent des modifications parfois importantes en termes d'architecture ou de traitement. Par exemple, la mise en place d'un cloisonnement réseau suppose d'en modifier la topologie, ce qui implique que les administrateurs disposent de la cartographie du système d'information et

3. https://www.ssi.gouv.fr/uploads/2017/12/anssi_sde_pss_bqa_5479_pj_decisionqualificationsimatic.vfp_.pdf
4. https://www.ssi.gouv.fr/uploads/2017/10/anssi-cspn-2017_25fr.pdf
5. https://www.ssi.gouv.fr/uploads/2016/10/bonnes_pratiques_securite_informatique_maritime_anssi.pdf
6. https://www.ssi.gouv.fr/uploads/2017/01/guide_hygiene_informatique_anssi.pdf

peuvent modifier, comme bon leur semble, les liens réseaux déjà établis. D'autre part, les mesures de protection peuvent fortement impacter les traitements réalisés au sein du SI. L'ajout d'équipements de sécurité (pare-feux, architecture AAA), l'utilisation de mécanismes cryptographiques ou de protections systèmes (*e.g.* contrôle d'accès, isolation mémoire) augmentent la quantité de calculs effectués ce qui augmente le risque de latence (Diallo et Feuillet, 2014). Dans le cas de systèmes industriels, cela n'est parfois pas tolérable, eu égard à leurs contraintes de disponibilité et de performance. Il est alors indispensable d'avoir prévu l'intégration des dispositifs de protection en amont de la conception du système d'information, ce qui n'est possible que pour les futurs navires. La problématique principale demeure la gestion des systèmes existant : intégrer des dispositifs de sécurité au sein des navires qui n'ont pas été conçus avec une prise en compte a priori des cyber-risques.

D'autre part, la rapide évolution et sophistication des attaques conduit à une obsolescence, totale ou partielle, des mesures de protection. En effet, un équipement de sécurité non entretenu peut devenir lui-même vulnérable à des attaques (*e.g.* utilisation des accès d'un ancien employé, factorisation de clés cryptographiques, vulnérabilité permettant une évasion du « bac à sable »). Pour des raisons de compatibilité, la mise à jour des mesures de protection peut se révéler difficile voire impossible. Par exemple, l'utilisation d'une ancienne version d'un système d'exploitation limite les bibliothèques disponibles et, par conséquent, les technologies de protection à disposition.

Quand une attaque ne peut être stoppée, il faut donc pouvoir la détecter afin de réagir et, a minima, ralentir sa progression. En cas d'attaque, plus le délai de réaction est court et plus celle-ci sera facile à contrer. Une intervention rapide limite en effet les risques d'implantation de l'attaquant au sein du réseau. L'utilisation d'outils passifs de supervision et de détection, sans éviter les incidents ni remplacer une mise à jour régulière des moyens de sécurité, supplée les mesures de protection et est donc nécessaire (mais non suffisante) pour une gestion efficiente de la sécurité et un traitement approprié des incidents.

Lorsqu'il n'est pas possible de suffisamment protéger le système d'information, il est donc toutefois possible d'opérer une surveillance passive de celui-ci afin de détecter toute activité malveillante. Une détection aboutit à l'intervention des responsables de la sécurité lesquels peuvent alors traiter l'incident.

1.3.4 Répondre & Récupérer

Lorsque les mesures de protection se sont révélées insuffisantes, une intervention humaine est nécessaire afin de rétablir la situation du système d'information et le ramener à un état nominal. Cette phase nécessite une forte implication humaine : intervention des équipes de sécurité, dialogues avec les utilisateurs et les sous-traitants, décisions des dirigeants. C'est à cette étape que l'ensemble des moyens organisationnels (sensibilisations, analyses de risques, chaînes de décision) employés révèle toute son utilité. Toutefois, le monde maritime n'a pas encore atteint la maturité suffisante en gestion des cyber-risques pour employer les outils produits par les spécialistes des technologies de l'information. Ces outils sont en effet très lourds dans leur mise en œuvre et pré-supposent une vision claire des risques et des besoins de sécurité. Ce n'est pas encore le cas dans le domaine maritime à cause du manque de culture en sécurité lié à l'éloignement et l'isolement par la mer qui donne l'illusion que le navire est hors d'atteinte.

1.4 Détection d'attaques cyber : cas des systèmes navals

Nous choisissons d'aborder la détection des attaques ciblant un système de navigation. Tout système est soumis à des menaces de différentes ampleurs dépendant des moyens et opportunités d'un attaquant. Ces menaces posent la question de la confiance qui peut être accordée au système dans son ensemble ainsi qu'à chacun de ses éléments. Dans ce travail, nous souhaitons montrer la possibilité d'utiliser la confiance en tant que moyen de détection.

Selon les contraintes imposées par le système et les attaques à détecter, diverses approches sont possibles. Les moyens de détection peuvent être classés en deux catégories : les outils orientés hôte et ceux orientés réseau (Garcia-Teodoro et al., 2009). Les systèmes de détection d'intrusion orientés hôte se focalisent sur le fonctionnement des entités qui constituent le système d'information. Au contraire, les systèmes de détection orientés réseau analysent les flux entre ces entités pour détecter les falsifications d'information ou l'envoi de charges malveillantes. Dans ce travail de thèse, nous nous plaçons dans ce dernier contexte. En effet, les systèmes orientés hôtes supposent une connaissance approfondie du fonctionnement interne des systèmes étudiés. Par ailleurs, la surveillance exploite les

ressources de l'hôte et peut introduire une dégradation de ses performances. Nous privilégions une surveillance passive qui ne nécessite pas de modifier les composants du système de navigation. C'est pourquoi nous adoptons l'approche réseau. De plus, en accord avec ce choix, nous considérons les composants du système comme des boîtes noires : seules leurs interactions avec les autres éléments du système sont connus, ce qui n'est pas le cas de leur fonctionnement interne.

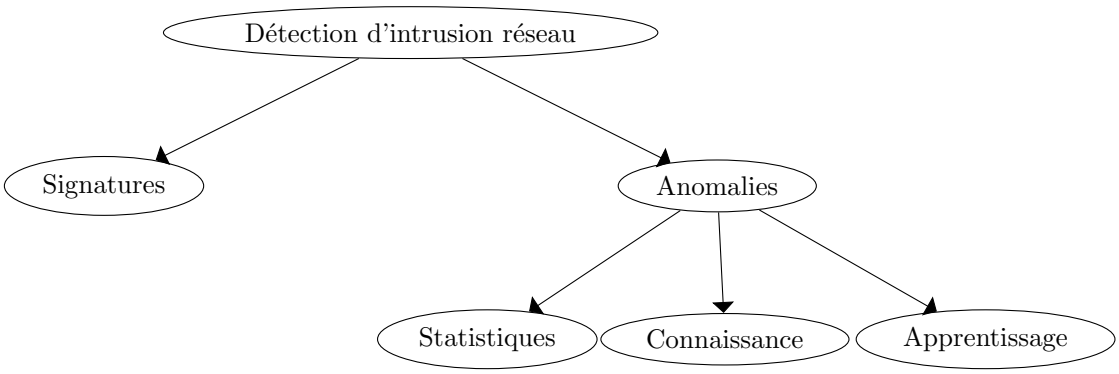


FIGURE 1.1 – Classification des approches de détection basées sur l'analyse des flux réseaux

Les systèmes de détection d'intrusion orientés réseau utilisent deux approches différentes : la détection par signatures et la détection par anomalies.

La détection par signatures se base sur des caractéristiques spécifiques à des attaques préalablement identifiées. Elle consiste à caractériser, *a priori* et de manière exhaustive, les attaques pouvant menacer la sécurité du système d'information. Cela revient à modéliser les actions d'un attaquant, lequel a un nombre restreint d'options pour détourner le système de son fonctionnement normal, *a fortiori* lorsque ce dernier est sécurisé. Cette approche a l'avantage d'être précise : la signature d'une attaque l'identifie uniquement. Ainsi, seules les attaques contre le SI seront détectées par le système de détection. Cependant, comme tout système, ce dernier peut faire des erreurs : soit en levant une alerte en l'absence d'attaque (*e.g.* à cause d'une signature mal construite) soit en restant silencieux en présence d'attaque (*e.g.* attaque innovante pour laquelle il n'existe pas de signature). Ce dernier cas, en particulier, constitue la faiblesse des systèmes à base de signatures : toute attaque non anticipée ne peut être détectée. C'est sur ce constat que repose une approche radicalement différente : la détection par anomalies.

Au contraire de la détection par signatures qui se base sur une modélisation des actions d'un attaquant, la détection d'anomalies cherche les déviations

du système par rapport à son comportement considéré normal (Chandola et al., 2007). C'est donc le comportement du système qui sert dans ce cas de modèle. Au lieu de chercher à exploiter la connaissance sur les attaquants, laquelle est souvent sommaire, il s'agit ici d'utiliser une connaissance bien plus accessible : celle sur le système. La force de cette approche est de pouvoir détecter des évènements qui n'ont pas été anticipés. En effet, puisque c'est le comportement, connu ou non, du système qui est modélisé, toute attaque, qu'elle soit prévue ou non par les concepteurs du système de détection, ne fera pas partie du modèle et donc sera détectée. De fait, les alertes levées par ces systèmes de détection sont bien plus nombreuses que lors de l'utilisation de signatures. Tout comportement du système d'information, même légitime, qui n'a pas été qualifié au préalable de normal sera classé comme étant une anomalie. La détection par anomalies génère donc un nombre important de faux-positifs : des alertes liées à une utilisation légitime du système. Par ailleurs, les outils de détection d'anomalie recourent à des méthodes mathématiques parfois complexes qui nécessitent une importante puissance de calcul. Au contraire, les systèmes à base de signatures se contentent de vérifier des règles et sont donc bien plus faciles à déployer car nécessitant moins de puissance de calcul. Enfin, en s'appuyant sur la connaissance du système plutôt que de l'attaquant, la détection par anomalies s'expose à un problème de *démarrage à froid*. En effet, les données extraites du système d'information destinées à la construction du modèle de normalité peuvent être corrompues : le système subit peut-être déjà une attaque, à l'insu des administrateurs. Selon la taille du système d'information, une procédure exhaustive d'assainissement peut se révéler difficile, voire impossible, et l'apprentissage considérera l'attaque que subit le système comme normale, ce qui pose d'évidents problèmes de sécurité.

Les méthodes de détection d'anomalies peuvent être classés selon 3 approches (Garcia-Teodoro et al., 2009) : détection par des méthodes *statistiques*, détection basé sur une *connaissance* du système et détection par *apprentissage automatique*. La hiérarchie des systèmes de détection réseau est résumée sur la figure 1.1.

La détection statistique considère le comportement du système comme un ensemble de phénomènes aléatoires. Un tel système de détection d'intrusion va extraire des modèles statistiques des flux d'échanges entre les composants du système. Il construit dynamiquement le modèle de fonctionnement normal du système d'information et ne nécessite donc pas une connaissance a priori de ce qui constitue la normalité. Cependant, ces systèmes ont l'inconvénient de pouvoir être entraînés

par les attaquants qui peuvent, sur le long-terme, faire accepter des actions malveillantes comme faisant partie du modèle construit par le système de détection. Ils ne souffrent cependant pas du problème du démarrage à froid : pour entraîner de tels systèmes, il faut que l'attaquant en ait une connaissance afin de l'entraîner selon ses souhaits, ce qui se réalise alors dans le temps. Par exemple en envoyant des mails frauduleux à un seul utilisateur du SI simultanément pour faire reconnaitre l'adresse d'expédition comme légitime.

La détection à base de connaissance utilise la formalisation des spécifications du système (à l'aide de langages type UML ou de machines à états) ou l'expertise humaine pour qualifier les évènements affectant le comportement du SI. Cette approche permet une détection efficace et très fiable des attaques. Néanmoins, ceci est possible au prix d'une acquisition puis de la mise à jour des connaissances qui se révèlent extrêmement coûteuses à la fois en moyens financiers et humains ainsi qu'en temps. En effet, cette connaissance nécessite une intervention humaine, bien souvent experte. Par exemple, la construction formelle d'un protocole réseau à l'aide de machines à états finis (Sekar et al., 2002).

La dernière approche est la détection à l'aide d'apprentissage automatisé. Les techniques d'apprentissage constituent un compromis entre les deux approches précédentes : le modèle de comportement du système est construit à la fois à l'aide de connaissances et de l'analyse des données. Elles mobilisent d'importantes ressources de calcul car elles font appel à des outils mathématiques complexes. De plus, à l'instar des systèmes à base de connaissance, les techniques d'apprentissage automatisé nécessitent une intervention humaine. Par exemple, les méthodes d'apprentissage supervisé fournissent une grande quantité de données à l'algorithme qui va construire le modèle du système. Pour fiabiliser l'apprentissage en caractérisant correctement des erreurs du système ou des comportements inattendus (*e.g.* la connexion légitime d'un employé en dehors des heures de bureau), les données peuvent être labellisées : un expert caractérise les données ou des groupes de données par des marqueurs connus de l'algorithme.

Nous choisissons de baser notre approche sur les systèmes statistiques. En effet, de tels systèmes ont déjà été proposés (Ristic et al., 2008 ; Psiaki et al., 2013) afin de détecter les malversations des capteurs embarqués à bord des navires mais ceux-ci sont spécifiques à un capteur particulier et, à notre connaissance, aucune approche générale permettant d'appréhender l'hétérogénéité des systèmes navals n'a été proposée. Par ailleurs les autres types de systèmes de détection d'anomalies ne conviennent pas à notre approche.

Tout d'abord, un système de détection par signatures n'est pas adapté à notre contexte. D'une part, la longue durée de vie du navire ainsi que les faibles mesures de protection existantes engendrent un accroissement important des vulnérabilités au cours de l'existence du SI. En effet, les correctifs de sécurité ne sont pas toujours applicables car ceux-ci peuvent dégrader les performances du système et impacter ses missions (Sultan et al., 2017). De plus, pour des raisons d'interopérabilité, les équipements ne peuvent pas toujours être changés. Le navire accumule donc des vulnérabilités tout au long de sa vie. D'autre part, la composition exotique des systèmes d'information embarqués constitue une importante surface d'attaque ce qui implique un grand nombre de signatures à entretenir. Par exemple, les signatures spécifiques aux systèmes industriels ne sont pas ré-exploitables pour des systèmes d'information. En conséquence, une base de signatures servant à défendre un système naval nécessite de nombreuses connaissances sur des systèmes variés ainsi qu'un entretien régulier par un ou plusieurs experts en sécurité. Enfin, une approche par signatures permet de détecter l'exploitation d'une vulnérabilité connue mais pas de repérer une attaque innovante.

Nous souhaitons par ailleurs limiter les besoins en connaissance du système. L'architecture du système est stable, ce qui nécessite un faible renouvellement des connaissances qui y sont liées. Cependant, l'environnement marin est dynamique et extrêmement variable (*e.g.* apparition soudaine d'une tempête) ce qui rend difficile la maintenance d'une connaissance précise du comportement du navire dans ce contexte imprévisible. Pour toutes ces raisons, nous privilégions une approche statistique.

Conclusion

Nous avons vu dans ce chapitre les diverses menaces pouvant cibler les systèmes d'information maritimes ainsi que les moyens, émergents, permettant de s'en protéger. Face à des attaques en perpétuelle évolution, ces mesures sont néanmoins limitées et ce d'autant plus que la durée de vie des équipements utilisés s'étend sur plusieurs dizaines d'années. La forte hétérogénéité des systèmes d'information maritime les rend par ailleurs difficiles à sécuriser car ceux-ci demandent des mesures spécifiques dans le sens où celles-ci doivent être capables de résister aux contraintes de fonctionnement du système. Si le SI ne peut être suffisamment protégé, ce qui est souvent le cas pour les navires conçus sans prise en compte a priori des menaces cyber, il peut toutefois être l'objet d'une surveillance passive visant à détecter les

évènements malveillants. La difficulté voire l'impossibilité d'anticiper les attaques pouvant cibler le SI conduit à envisager un moyen de gérer l'inconnu, c'est-à-dire de maintenir un certain niveau de sécurité malgré un manque de connaissance. De plus, l'humain fait partie intégrante du système d'information et doit donc pouvoir être intégré dans les modèles de sécurité afin que celle-ci soit efficiente. Les informations échangées entre les différents acteurs du monde maritime, dont le niveau de sécurité est variable, rend toute confiance dans celles-ci relative. Cependant, pour des besoins évidents de fonctionnement, les décisions prises, automatiquement ou non, à bord du navire sont basées sur ces informations. Afin d'améliorer la sécurité des systèmes d'information et de navigation embarqués à bord des navires, nous souhaitons montrer une approche permettant d'élaborer un système de détection d'attaques qui puisse s'adapter aux particularités de tels systèmes. Nous proposons pour cela de baser ce système sur l'évaluation de la confiance dans les multiples composants du SI du navire qui émettent, reçoivent et traitent de l'information.

Chapitre 2

La confiance : définitions et modélisations

Issue de l'étymologie latine *confidere* qui signifie « croire ensemble », la confiance est une notion fondamentale dans les interactions humaines. Elle est une condition indispensable à l'établissement de relations durables. Selon le Larousse, elle se définit comme un « sentiment de quelqu'un qui se fie entièrement à quelqu'un d'autre, à quelque chose ». Cette première définition montre l'aspect relationnel de la confiance (deux entités impliquées) mais également son aspect subjectif (perception de la première entité par la seconde). Ces aspects constituent les bases de la confiance et sont repris par l'ensemble des travaux cités dans ce chapitre. La section 2.1 présente un ensemble de définitions usuelles de la confiance dans trois domaines : économie, sociologie puis Internet, en particulier le e-commerce. La section 2.2 expose les problématiques liées à la mesure de la confiance ainsi que le phénomène de propagation. La section 2.3 conclut ce chapitre par une analyse des différentes contributions de la littérature. En particulier, elle rapproche la confiance des problématiques de détection de cyber-attaques.

2.1 Une notion complexe

La confiance est un élément essentiel de la vie quotidienne. Bien qu'intuitivement compréhensible, elle reste une notion difficile à délimiter. De nombreuses définitions sont apparues ces dernières années, particulièrement dans les sphères économiques (pour maximiser le gain) et sociales (pour améliorer les relations). Toutes différentes, il est néanmoins possible d'identifier certaines caractéristiques communes qui conviennent à notre cadre d'étude.

2.1.1 Un concept économique

En économie, les définitions favorisent généralement la **coopération** entre les agents (Lorenz, 1988 ; Sabel, 1990 ; Lorini et Demolombe, 2008), c.-à-d. que les agents « travaillent et agissent ensemble sur une tâche, et partagent les bénéfices qui en découlent » (Marsh, 1994b).
Dans (Lorini et Demolombe, 2008), la confiance est définie comme étant la « croyance que l'autre veut agir dans notre intérêt pour un but donné ». Autrement dit, la confiance d'un individu A, qui poursuit un certain objectif, dans un individu B est l'inclinaison du premier à croire que l'autre souhaite lui venir en aide, c.-à-d. lui apporter sa contribution pour atteindre l'objectif poursuivi.
Un exemple célèbre d'une mise en application de la confiance est le dilemme du prisonnier, tiré de la théorie des jeux et introduit par Tucker (Tucker, 1983). Cet exemple met en scène deux bandits, arrêtés par la police qui n'a malheureusement trouvé que peu de preuves pour les inculper. Le procureur en charge de l'affaire se voit donc contraint d'obtenir des aveux d'au moins l'un des deux suspects pour obtenir une condamnation. Les deux prisonniers sont gardés dans deux salles différentes sans pouvoir ou avoir eu la possibilité de se parler depuis l'arrestation.
Le procureur fait alors une offre, la même, à chacun des suspects :

- Si l'un des deux prisonniers dénonce l'autre, le premier est remis en liberté tandis que le second écope de la peine maximale correspondant à dix années de prison.
- Si les deux se dénoncent entre eux, ils seront condamnés à une peine plus légère de cinq années.
- Les prisonniers en concluent alors que si aucun ne dénonce l'autre, les deux seront condamnés pour une peine de six mois, faute de preuves solides.

La situation est résumée dans le tableau 2.1.

Décision prisonniers	Ne dénonce pas	Dénonce
Ne dénonce pas	$(\frac{1}{2},\frac{1}{2})$	(10,0)
Dénonce	(0,10)	(5,5)

Table 2.1 – Matrice des gains pour un dilemme du prisonnier

Chacun des prisonniers, ayant quelques minutes pour réfléchir, en arrive à la conclusion logique que, quelle que soit la décision de son comparse, le trahir mène à une peine plus légère : $0 < \frac{1}{2}$ et $5 < 10$.
Bien que la trahison soit le choix le plus profitable pour chacun des prisonniers pris indépendamment, la coopération entre les prisonniers, c.-à-d. qu'aucun ne dénonce l'autre, mène à un résultat bien plus profitable pour les deux, la somme de leurs peines ne dépassant pas un an. Cette situation ne se rencontre malheureusement pas avec des joueurs rationnels, lesquels se trahiront mutuellement. Pour outrepasser la trahison logique et coopérer, les deux joueurs doivent avoir une confiance réciproque. Ce comportement est d'autant plus important quand les deux joueurs sont confrontés successivement à plusieurs situations de type « dilemme du prisonnier ». La décision prise à la n-ième itération aura des conséquences sur les suivantes (Axelrod, 1984).
La confiance, en favorisant la coopération, va alors entrainer les joueurs dans un cercle vertueux qui minimisera leur temps à passer en prison. En économie, la confiance est donc perçue avant tout comme un rapport bénéfices/risques, la coopération étant alors une alternative plus profitable que la compétition qui est un modèle souvent répandu.

Très proche de la coopération, la confiance n'en demeure pas moins distincte (Mayer et Davis, 1995). En effet, la coopération n'implique pas une confiance mutuelle entre les deux parties et, bien que souvent vérifiée en pratique, la réciproque est également fausse. En effet, la coopération peut être forcée, c.-à-d. qu'un intérêt particulier ou une autorité supérieure peut imposer de coopérer avec une personne dont on se méfie. Que cela vienne de sa propre initiative ou qu'elle soit imposée par une volonté extérieure, la coopération demeure indépendante de la confiance. De fait, tandis que la coopération est une action nécessairement symétrique (Williams, 2000), la confiance est quant à elle asymétrique. La différence entre les deux concepts vient du **risque** induit par la situation (Mayer et Davis, 1995). La confiance de A à B est dépendante du risque encouru par A. Ainsi, cela est mis en évidence dans le dilemme du prisonnier : si A et B travaillent ensemble (coopération), ils partagent les bénéfices (peine réduite pour les deux) tandis que si l'un des deux se fait doubler par son comparse (confiance sans coopération) il sera seul à endurer les conséquences (peine maximale). Malgré des liens étroits, les deux notions sont ainsi bien distinctes.

En approfondissant le lien entre coopération et confiance, (Lorini et Demolombe, 2008) suggère la pertinence de la **sincérité** dans de tels modèles. En

effet, les auteurs font remarquer que la confiance d'un individu A envers un individu B n'est possible que si B est sincère aux yeux de A, c.-à-d. qu'il ne dissimule aucune information ayant un intérêt pour A (Demolombe, 2001, 2004 ; Lorini et Demolombe, 2009). Cependant B n'a peut-être aucun intérêt à se montrer sincère. La sincérité est dépendante des objectifs ou intérêts de chacun. Si A et B ont des intérêts contraires, alors il est du bon sens que chacun se méfie de l'autre, la coopération étant dès lors impossible.
Par exemple, si A et B souhaitent chacun acquérir une même ressource, sans vouloir la partager avec l'autre, alors leurs intérêts divergent. Tout avis ou conseil que l'un peut donner à l'autre sur le « bon moyen » d'acquérir la ressource doit alors être considéré non sincère. Aucun n'a cependant de raison de mentir sur des sujets non liés à la ressource en question (Paglieri et al., 2014). La sincérité introduit alors les intentions des agents dans les modèles économiques.

La confiance est utilisée en économie pour renforcer la coopération et gérer le risque afin de maximiser les gains. Cependant, ses définitions les plus fécondes sont issues de l'étude des relations sociales.

2.1.2 Un concept social

La confiance est répandue en économie pour faciliter la coopération et maximiser les bénéfices, mais c'est avant tout un élément important des relations sociales. La conception économique de la confiance est généralement purement **rationnelle** (Axelrod, 1984 ; Gambetta, 1988 ; Blomqvist, 1997 ; Teacy et al., 2006). Cette vision est critiquée car elle considère que toute personne dans la même situation prendrait la même décision. De nombreux travaux considèrent que la confiance dépend de l'individu (Deutsch, 1958 ; Luhmann, 1979 ; Lewis et Weigert, 1985 ; Castelfranchi et Falcone, 2000).

La première définition de la confiance est née des travaux de Deutsch (Deutsch, 1958). Il a étudié la confiance sous l'angle de la psychologie. Pour lui, une situation de confiance est un choix ayant deux alternatives possibles : l'une des deux a des conséquences bénéfiques (Va^+) et l'autre des conséquences négatives (Va^-). Ce choix est tel que le risque est plus important que les bénéfices espérés ($Va^- > Va^+$). La confiance réside dans l'acceptation de ce choix et la **méfiance** dans le refus de choisir. En fait, cela peut se représenter par un jeu de Pile ou Face. Avant de lancer une pièce (non truquée), le joueur doit choisir une face. Si le résultat est conforme à son choix, il reçoit 2 €. Dans le cas contraire, il

perd 10 €. Les risques sont plus importants que les bénéfices espérés ($10 > 2$) et le résultat est inconnu a priori. Il n'existe aucun argument qui pourrait lui faire préférer une face plutôt que l'autre. Il s'agit bien d'une situation de confiance selon Deutsch. Accepter de jouer place donc le joueur dans une situation de confiance, laquelle est avant tout un état d'esprit. La décision ne sera pas basée sur des choix rationnels mais plutôt sur des **sentiments** personnels.

Cette opposition entre rationalité pure et confiance est au cœur des travaux de Lewis (Lewis et Weigert, 1985). Il définit la confiance selon deux composantes principales : **rationalité** et **émotions**. Il définit ainsi différents types de confiance selon les degrés de rationalité et d'émotions qui entrent en jeu. Ces différents types sont résumés dans le tableau 2.2. Ainsi la foi est un état de confiance uniquement basée sur de fortes émotions. À l'inverse, la prédiction est basée sur un raisonnement objectif mais pas sur des émotions.

		Émotions		
		Hautes	Basses	Absentes
Rationalité	Haute	Confiance idéologique	Confiance cognitive	Prédiction rationnelle
	Basse	Confiance émotionnelle	Confiance ordinaire	Anticipation probable
	Absente	Foi	Destin	Incertitude, panique

Table 2.2 – Typologie de la confiance selon Lewis

La partie rationnelle de la confiance est résumée ainsi : « Nous choisissons les personnes à qui nous faisons confiance, par rapport à quoi et dans quelles circonstances. Nous basons nos choix sur ce que nous considérons comme étant de « bonnes raisons », lesquelles constituent les preuves de la crédibilité. ». Lewis met ici en évidence que la confiance n'est pas un concept absolu : celui-ci dépend d'un **objectif** (le *quoi*) mais également de circonstances ainsi que de nos **connaissances** (l'accumulation des « bonnes raisons »). La confiance dépend donc, sur le plan cognitif, de critères tant intrapersonnels (connaissance, objectif) qu'extrapersonnels (circonstances).

Contrairement à la définition de Deutsch, celle de Lewis fait intervenir un paramètre **temporel** au travers de la connaissance. La confiance évolue donc différemment, même si les situations l'impliquant sont identiques. Un fait que l'on

retrouve dans des situations quotidiennes : imaginons un bus qui arrive systématiquement en retard de 10 minutes. La première fois qu'une personne le prendra, celle-ci sera surprise et aura une appréciation négative de l'évènement. En revanche, au fil du temps, la même personne pourra en tirer parti. Elle sera en confiance vis-à-vis de ce bus car elle aura acquis la conviction que le bus arrivera en retard. Au sens de Deutsch, les situations seront toutes identiques les unes aux autres. L'itération de la situation ne modifiera pas l'appréciation que l'on peut en avoir. Au contraire, la connaissance permet de tirer parti de la répétabilité d'un évènement et de modifier sa confiance en conséquence.

D'autres travaux ont étudié le lien entre confiance et connaissance. Par exemple, (Mayer et Davis, 1995) et (Gambetta, 1988) mettent en avant l'attente par une entité d'une action particulière de la part d'une autre. La confiance évolue en fonction de l'occurrence ou non de ces actions, c.-à-d. en comparant ce qui est attendu (basé sur les connaissances personnelles) à ce qui arrive effectivement. Dans (Deutsch, 1958), la situation d'une entité va s'améliorer ou se dégrader a posteriori, c.-à-d. que la situation est dans un état stable tant qu'aucune décision n'est prise ou qu'aucune action n'est effectuée. Au contraire, la confiance est définie dans (Gambetta, 1988) comme étant la « probabilité subjective par laquelle un individu A s'attend à ce qu'un autre individu B accomplisse une action donnée de laquelle dépend son bien-être ». Dans ce cas, la situation va donc se dégrader à moins que l'objectif ne soit atteint (phénomène de dépendance) : l'action de B est nécessaire au bien-être de A.

Rationalité, émotions, connaissance, la confiance est reliée à toutes ces notions à des degrés divers selon les définitions. Ces qualités se retrouvent en particulier dans les travaux de Luhmann (Luhmann, 1979, 2000). Luhmann introduit cependant une nouvelle composante de la confiance qu'il appelle *complexité*. Cette complexité est une matérialisation de l'ensemble des alternatives possibles qui s'imposent à quelqu'un lorsqu'il doit choisir de faire confiance ou non.
De manière semblable à celle de Deutsch, la définition de Luhmann met un individu, désireux d'atteindre un certain objectif, face à un ensemble de choix. Cet ensemble représente les divers futurs possibles pour un évènement. Certains de ces futurs mènent à l'objectif recherché, les autres non. L'ensemble des possibles est si grand qu'il est impossible d'arriver à un choix rationnel, lequel devrait considérer l'ensemble exhaustif des solutions. La confiance est alors « une méthode subjective pour réduire la complexité perçue du futur en supposant, sur la base d'une connaissance personnelle limitée, des actions bénéfiques de la part d'acteurs

indépendants. » (Luhmann, 1979). Contrairement aux définitions précédentes, la connaissance est ici explicitement mentionnée. La confiance est alors vue comme un processus qui s'affine au cours du temps et qui permet de simplifier la prise de décision. C'est un aspect important de la confiance où les émotions et les expériences personnelles prennent le pas sur la raison pure afin de réduire les possibilités. Sans avoir tous les éléments à disposition, les émotions permettent de trancher et de prendre une décision qui ne serait peut-être pas la meilleure si l'on disposait de toutes les informations. La confiance, en mêlant rationnel et émotionnel, permet donc de s'adapter au manque de connaissance et d'appréhender plus simplement la prise de décision (Marsh, 1994b). Ce principe est illustré par exemple lors de la réception d'une information à la sortie d'une longue chaîne de traitement. Il est bien souvent impossible de maîtriser tous les éléments de la chaîne à cause de leur nombre ou de leur complexité. Il est donc indispensable de faire confiance à chacun des acteurs de la chaîne pour le correct acheminement de l'information, ce qui permet de prendre des décisions sur la base des informations fournies.

Comme il l'a été expliqué, la confiance n'est pas une notion purement rationnelle. En effet, Lewis soutient que lorsqu'une personne A a confiance dans une autre personne B, les actions de ce dernier qui ne sont pas conformes aux intérêts de A sont interprétées bien plus négativement qu'elles ne devraient. Ainsi, apprendre qu'une personne ment n'est pas quelque chose de particulièrement choquant en soi. Cependant cela impacte bien plus lorsque la personne est une personne de confiance. Au-delà de l'aspect moralement répréhensible de la chose, c'est la trahison qui est particulièrement mal vécue. La confiance est ainsi plus qu'un simple calcul logique. Elle fait entrer en compte nos idées, nos préconceptions, nos valeurs et tout ce qui a trait à notre jugement. Elle suppose ainsi la bienveillance de l'autre, c.-à-d. la croyance qu'il se soucie des autres et qu'il souhaite agir dans leur intérêt, sans agir de manière opportuniste (McKnight et Chervany, 2000).
La confiance est une perception des actions des autres. Selon que ces actions sont bienveillantes ou démontrent au contraire une trahison, la confiance sera affectée positivement ou négativement. Cela modifie le comportement de l'entité, c.-à-d. les actions qu'elle est susceptible d'effectuer. Par exemple, si A est trahi par B, il sera moins enclin à partager ses ressources ou à lui apporter son aide et ce même dans des domaines qui n'ont rien à voir avec la trahison. Ainsi, si une personne A découvre qu'une personne B l'a volontairement induit en erreur pour un placement financier alors A ne tiendra pas compte voire prendra une décision opposée à tous les conseils ultérieurs que B pourra lui donner, même si ceux-ci n'ont pas de rapport avec la finance. Le raisonnement évoqué étant qu'une entité pouvant trahir

sur un sujet (ici un placement) le peut sur tous les autres (indication d'itinéraire, renseignements personnels, etc.), et ce même si elle n'y a aucun intérêt perçu.

À l'instar des définitions de la sphère économique, le lien entre confiance et risque est souvent mis en avant (Deutsch, 1958 ; Luhmann, 1979 ; Jøsang et Presti, 2004). Dans (Mayer et Davis, 1995), la confiance est définie comme étant « la volonté d'une partie A d'être vulnérable aux actions d'une autre partie B. Cette volonté est basée sur l'attente que B accomplisse une action spécifique importante pour A. Elle ne tient pas compte de la capacité de surveiller ou de contrôler B ». Comme pour (Deutsch, 1958), la confiance repose sur la notion de risque (vulnérabilité) mais également sur un objectif particulier (l'action que A souhaite voir B réaliser). L'objectif sert de base de référence pour évaluer les actions d'autrui. Par exemple, dans la définition de Deutsch, les alternatives Va^+ sont considérées bénéfiques, c.-à-d. comme servant les intérêts du sujet. Cependant cette considération est dépendante de la volonté du sujet. Pour reprendre l'exemple de la pièce, si l'on souhaite perdre de l'argent alors l'alternative de perdre 10 € est plus séduisante que celle de gagner 2 €. Cela inverse alors complètement la vision du jeu et le risque de « gagner » est plus important que le risque de « perdre ». Sans objectif, il est impossible de dire si un évènement ou une action est positif ou négatif (Castelfranchi et Falcone, 1998).

La possibilité de surveiller ou de contrôler B n'est pas prise en compte dans la définition de (Mayer et Davis, 1995). En effet, Mayer et al. appliquent leurs définitions aux relations d'un individu avec d'autres identifiables qui sont perçus comme agissant et réagissant de leur propre volonté envers le premier. Le fait que les actions des individus résultent de leur libre-arbitre lie fortement la confiance à la notion de risque. lesquelles demeurent toutefois différentes. Supposons que A soit un piéton et B l'ensemble des automobilistes. Alors A, lorsqu'il traverse un passage piéton, a confiance en B pour respecter le code de la route et être en capacité de s'arrêter. Il est bien dans une situation de risque puisqu'un automobiliste peut le renverser si l'attente de A n'est pas satisfaite (le respect du code et les capacités de freinage dans ce contexte). Cependant, si A dispose d'un dispositif qui active les freins de toutes les voitures dans un rayon de 50 mètres, alors il force B à s'arrêter. A peut alors traverser sans courir le moindre risque. Au sens de (Mayer et Davis, 1995), le situation présentée ci-dessus n'est pas une situation de confiance puisque le fait que B s'arrête ou non ne dépend pas de sa volonté.

L'étude de la confiance et son impact sur les relations sociales a mis en évidence de nombreuses caractéristiques sur lesquelles elle repose : émotions, connais-

sance, risque. Le réseau de confiance d'une entité est alors un réseau bien connu et peu risqué, dont les membres agissent de leur propre volonté dans l'intérêt de l'entité. Toutefois, la constitution de ce réseau est coûteuse et la confiance nécessite donc d'autres critères pour s'adapter aux réseaux modernes.

2.1.3 Confiance et réseaux informatisés

Au cœur des théories économiques, phénomène social ou psychologique, la confiance est également de plus en plus présente depuis l'émergence d'Internet. En effet, l'entourage d'une personne s'est considérablement élargi. Un individu n'est plus seulement en relation avec d'autres personnes de son entourage mais est connecté à une multitude d'entités ; entités qui peuvent être tout aussi bien des personnes que des services. Cette nouvelle masse de relations diminue considérablement le nombre d'interactions entre chaque entité et limite du même coup l'accroissement du niveau de connaissance d'un individu sur ses relations. Ainsi un entourage réduit mais bien connu a laissé place à de nombreuses relations méconnues (Grandison et Sloman, 2000). Dans le même temps, les nouveaux moyens de communication empêchent parfois d'identifier la nature du correspondant (un être humain, un automate ou une intelligence artificielle ?). Le peu d'interactions couplé à l'anonymat relatif d'Internet rend cette connaissance difficile à acquérir.

Un cas particulièrement étudié est celui du e-commerce : la croissance exponentielle des sites de vente en ligne amène aujourd'hui les consommateurs à se poser la question de savoir à qui faire confiance pour obtenir l'objet de leur convoitise. Le défi est d'autant plus grand qu'il n'est pas possible de juger la marchandise avant d'acheter. Afin de réduire les risques d'escroquerie, il convient alors d'entamer un long travail d'investigation pour recueillir des avis sur tel ou tel marchand, tel ou tel produit. Cette méthode n'est cependant pas très fiable. En effet, la rédaction de faux avis ou de biais dans la publication est devenu une pratique courante dans ce domaine. Il arrive ainsi que pour promouvoir un site particulier, les avis négatifs à son encontre soient supprimés ou publiés en différé. Il arrive également que le marchand en question rédige lui-même de nombreux avis positifs ou paie des internautes pour le faire. Face à ces pratiques déloyales, de nouveaux projets utilisent la confiance (Massa et Avesani, 2007a ; Cofta, 2004 ; Bizer et Oldakowski, 2004 ; Avesani et al., 2005 ; Massa et Avesani, 2007b) ou la **réputation** (Abdul-Rahman et Hailes, 1997 ; Yu et Singh, 2002 ; Kamvar et al., 2003 ; Grishchenko, 2004) pour identifier les sites ou avis les plus fiables.

Pour s'adapter à ce nouvel environnement où se confondent humains et machines [1], Grandison et Sloman ont introduit la **compétence** comme critère de base de la confiance. Ils ont défini celle-ci comme « la croyance ferme en la compétence d'une entité à agir de manière fiable au travers d'un contexte spécifique » (Grandison et Sloman, 2000). Une définition dans laquelle la notion de compétence s'exprime comme « la capacité d'une entité à assurer les fonctions qui lui sont attribuées ». La compétence se retrouve également dans les travaux de (Falcone et al., 2003) et de (McKnight et Chervany, 1996). Dans chacun de ces cas, celle-ci est liée à l'efficacité de l'entité, c.-à-d. sa capacité à effectuer la bonne action au bon moment. Une autre approche pour mesurer la confiance dans ces réseaux complexes est d'utiliser la réputation.

La confiance et la réputation sont deux notions proches et pourtant différentes. Elles sont proches dans le sens où les deux se basent sur une perception de l'autre. Cette perception est liée à un moment, à un endroit, à un contexte particulier. Cependant la principale différence entre les deux notions provient de la façon dont celles-ci sont évaluées. En effet, la confiance est estimée à partir de critères propres, c.-à-d. dépendants de l'individu. Pour illustrer ces propos, considérons trois entités : E_1, E_2 et E_3. Lorsque E_1 veut initier une relation de confiance avec E_2, elle va déterminer un niveau de confiance à accorder à E_2 selon sa perception de certaines caractéristiques de E_2. Sa perception n'est évidemment pas la même que celle de E_3 qui aura peut-être un avis très différent de celui de E_1 sur la confiance qui peut être accordée à E_2. Ainsi chacun a sa propre méthode pour évaluer la confiance. Ceci dépend de la disposition à faire confiance (Mayer et Davis, 1995 ; McKnight et Chervany, 2000 ; Falcone et al., 2003 ; McKnight et Chervany, 1996, 2001a), c.-à-d. de la tendance naturelle d'un individu à faire confiance à une personne inconnue.

La réputation quant à elle, ne repose pas directement sur de telles méthodes propres. Elle est en fait la « perception qu'une partie crée, à travers ses actions passées, à propos de ses intentions et normes » (Mui et al., 2002). Contrairement à la confiance qui est une perception *locale* entre deux entités (une qui effectue des actions et l'autre qui juge ces actions), la réputation d'une entité est *globale* : c'est la perception par le réseau tout entier de ses actions. La confiance est alors une relation entre deux individus tandis que la réputation n'est qu'un attribut d'un individu. Cette différence fondamentale entre les deux concepts fait qu'il est parfois bien plus facile de se baser au premier abord sur la réputation plutôt que

1. Une machine peut être logicielle ou simplement mécanique

sur la confiance (Teacy et al., 2006 ; Fullam et Barber, 2007). En effet, la confiance nécessite une interaction directe entre les deux parties concernées tandis que la réputation s'apprécie par l'intermédiaire de tierces personnes. Lorsqu'aucune relation de confiance n'existe entre deux entités, le niveau de confiance de A en B s'évalue au travers de la réputation (resp. confiance) de B communiquée par d'autres personnes inconnues (resp. connues) de A. Ce processus de communication est appelé **recommandation**.

Lorsque l'on aborde la confiance et la réputation, la notion de recommandation entre bien souvent en jeu (Lin et al., 2005 ; Artz et Gil, 2007 ; Jøsang et al., 2007). La recommandation sert à pallier le manque d'expériences ou de connaissances au sujet d'un individu. Ainsi lorsqu'on ne connaît pas suffisamment une personne pour se forger un avis sur le niveau de confiance à lui accorder, le premier réflexe est de consolider ses connaissances en interrogeant son entourage au sujet de la personne en question. L'entourage va alors émettre une recommandation (positive ou négative) selon ses propres expériences (Jøsang et al., 2006b) comme illustré sur la figure 2.1.

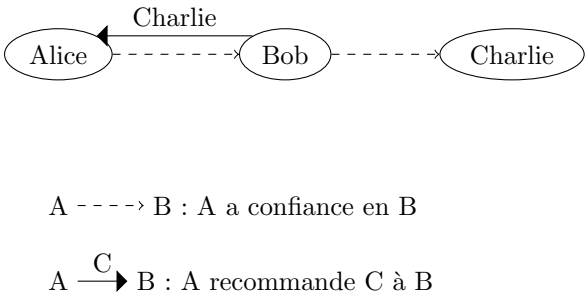


Figure 2.1 – Recommandation et confiance directe

Au fil des recommandations, la confiance peut donc se construire sans interaction directe avec l'objet de la confiance. La recommandation permet de propager la confiance à travers un réseau. Cependant, cette propagation n'est pas absolue. Elle est cantonnée à un domaine spécifique. En effet, le fait que B soit reconnu par A pour lui recommander un garagiste ne garanti en aucun cas la qualité des conseils de B en pâtisserie (Jøsang et al., 2006b). Ainsi la recommandation, à l'instar de la confiance, est dépendante de l'objectif recherché par A.
La recommandation est le processus permettant de propager la confiance dans un réseau mais de nombreux problèmes se posent :

- A et B ne partagent pas nécessairement le même point de vue sur tout. A peut aimer jouer aux cartes pour le seul plaisir de jouer tandis que B

peut simplement apprécier le côté social du jeu. A sait-il faire des recommandations qui prennent en compte les intérêts de B ?

- Que dire de B si celui-ci recommande positivement C alors que ce dernier n'est pas apprécié de A ?
- Que dire d'une recommandation qui n'est pas directe, c.-à-d. si B ne connaît pas C et qu'il doit donc demander à son ami D ?
- Comment agréger les avis de B et D, qui connaissent tous les deux C mais ne lui font pas confiance pour les mêmes sujets ou qu'ils lui accordent des niveaux de confiance différents ?

Toutes ces questions seront explorées dans la sous-section 2.2.3 lorsque nous aborderons la problématique de la propagation de confiance.

Une autre approche utilisée pour modéliser les réseaux hybrides (humains et entités logiques) est celle des systèmes multi-agents. Elle modélise des systèmes complexes par des agents autonomes et leurs croyances, désirs et intentions (Wooldridge, 2000) ainsi que les actions qu'ils peuvent effectuer, ceci dans le but d'observer l'émergence de comportements de groupes. Les systèmes multi-agents sont appropriés pour analyser les interactions des différents agents et ont été utilisés dans un contexte économique (Lorini et Demolombe, 2008) pour étudier l'influence des croyances et de la confiance sur les relations entre les agents et en particulier la validité des informations qu'ils s'échangent. Dans (Marsh, 1994a), Marsh a étudié l'impact de la confiance sur les relations entre les agents. Il a montré que la confiance est un outil particulièrement adapté pour de tels systèmes lorsque les agents sont dans un environnement non maîtrisé. Dans ce cas, la confiance permet de modéliser un manque de connaissance et de faciliter la prise de décision basée sur des informations incertaines ou incomplètes. Les systèmes multi-agents, comme la plupart des systèmes distribués, reposent sur l'hypothèse que chaque élément du système a de nombreuses interactions avec les autres mais surtout qu'il a la possibilité d'interagir avec un grand nombre d'agents. Cette hypothèse conduit aux problématiques de coopération et de consensus : comment faire travailler des agents ensemble pour accomplir une tâche lorsque certains d'entre eux ont des intérêts contraires ? Ce problème a notamment été abordé sous la forme du problème des généraux byzantins (Lamport et al., 1982). Dans ce cas, la solution proposée montre qu'il est possible d'obtenir un réseau résilient si chaque général peut transmettre sa décision à tous les autres.

2.1.4 Gestion de confiance dans les systèmes d'information

La confiance se rencontre également dans l'étude des systèmes d'information. Dans ce domaine, elle est cependant considérée comme étant un état plus qu'un moyen : un objet ou une entité est considéré de confiance lorsque des preuves de sécurité permettent d'en garantir le bon fonctionnement et que celui-ci ne sera pas détourné de son objectif initial. Par exemple, un *Trust Execution Environment* (TEE) permet d'effectuer une suite d'actions au sein d'un environnement sécurisé par divers mécanismes tels que l'isolation mémoire ou des protocoles cryptographiques. La sécurité du TEE garantit que les actions ne peuvent pas être détournées de leur objectif (intégrité) ni espionnées (confidentialité).

Pour un système d'information [2], une des problématiques est le contrôle d'accès : comment prouver qu'une entité E est autorisée à effectuer l'action a ? Par exemple, autoriser une personne à payer avec une carte bancaire ou bien autoriser un ordinateur à imprimer. Cela repose sur deux phases distinctes : vérifier l'identité de E et autoriser E à effectuer a.

La vérification de l'identité d'une entité est appelé **authentification**. Elle repose sur l'utilisation d'un ou plusieurs facteurs parmi (Choi et Zage, 2012) :

- Ce que l'entité est : empreinte biométrique (digitale, rétinienne, vocale, etc.), frappe au clavier, réseau veineux.
- Ce que l'entité sait : mot de passe, question secrète, code PIN, signature.
- Ce que l'entité possède : badge ou clé USB renfermant une clé cryptographique, téléphone (code SMS).

Ces facteurs permettent à E de prouver son identité en résolvant un ou plusieurs défis. Par exemple, pour authentifier un acheteur lors d'un paiement en ligne, le système électronique de paiement envoie un code sur le téléphone du possesseur de la carte servant au paiement. Seul la personne en possession du téléphone dispose donc du code et peut valider la transaction. Pour authentifier E, le système doit donc posséder un moyen de vérifier son facteur d'authentification. Dans le cas du paiement en ligne, c'est le numéro de téléphone qui permet d'envoyer le code. Pour authentifier un acheteur, il faut donc posséder son numéro de téléphone avant d'effectuer la transaction et pouvoir certifier que le numéro est bien celui qui va avec la carte servant au paiement. En pratique, un vendeur ne peut pas rencontrer physiquement chaque acheteur potentiel pour récupérer ses numéros de carte et

2. Le contrôle d'accès n'est pas spécifique au domaine des SI mais ceux-ci en ont popularisé l'étude, notamment en cherchant à l'automatiser.

de téléphone. Il délègue cette responsabilité à la banque qui lui sert dans ce cas de **tiers de confiance**.
Un tiers de confiance est une entité externe considérée sûre qui authentifie le demandeur auprès du système. Le demandeur E s'authentifie une fois auprès du tiers qui lui fournit un certificat en retour. À chaque demande d'authentification, le demandeur fournit le certificat au système qui demande au tiers d'en confirmer la validité. Ainsi le système n'a pas à stocker les informations relatives à chaque demandeur : c'est le rôle du tiers de confiance. L'inconvénient de cette méthode est qu'un tiers concentre toutes les informations permettant d'authentifier un grand nombre de personnes. Si le tiers est corrompu, il peut authentifier n'importe qui pour n'importe quelle action. En centralisant les informations, il constitue donc un élément extrêmement critique qui présente un risque élevé d'être la cible d'attaques.

Pour pallier les faiblesses d'un tel système centralisé, une approche basée sur la signature de clés cryptographiques a été proposée dans (Abdul-Rahman, 1997). Dans ce modèle, appelé **toile de confiance**, chaque utilisateur joue le rôle de tiers de confiance et fournit des certificats à ceux qu'il peut authentifier. Par exemple, si un utilisateur A peut authentifier un autre utilisateur B qui lui-même peut vérifier l'identité d'un utilisateur C alors A peut authentifier C via B. De même, A peut authentifier toute personne déjà identifiée par C.

La deuxième phase d'un contrôle d'accès est l'**autorisation**. Les modalités d'accès à des ressources ou plus largement les actions autorisées au sein d'un système sont définies par une politique de sécurité. Cette dernière précise *qui* peut accéder à *quoi*. Les premiers systèmes mettant en œuvre les politiques de sécurité sont appelés *système de protection* (Denning, 1982). Ils définissent un ensemble restreint d'actions possibles (*e.g.* lecture, écriture et exécution). Pour ces systèmes, la problématique peut se formuler ainsi : l'entité E est-elle autorisée à effectuer l'action a sur l'objet O ? Les premières approches étaient alors basées sur des matrices d'accès définissant l'ensemble des états possibles du système. L'ensemble total des états étant connu, le sous-ensemble des états (E, O, a) dangereux, c.-à-d. contraires à la politique de sécurité, est dans ce cas assez faible. Cela n'est cependant pas applicable dans le cas où l'ensemble des objets et des entités est important, dynamique et inconnu a priori (Blaze et al., 1998).

Blaze et al. ont alors introduit le problème de la gestion de confiance sous la forme suivante : les informations c prouvent-elles que l'action a est autorisée par la politique de sécurité P ? (Blaze et al., 1999) Contrairement aux systèmes clas-

siques de contrôle d'accès, ces systèmes contournent le problème de l'identification. En effet, il n'est pas systématiquement nécessaire de prouver l'identité d'une entité pour autoriser son action : seules les informations c requises par la politique sont nécessaires, quelque soit le nombre des entités souhaitant agir. Une caractéristique de ces systèmes est que l'authentification n'est plus nécessaire : toute la sécurité repose sur l'unique phase d'autorisation.

Les systèmes de gestion de confiance, ou systèmes *Trust Based Access Control*, répondent à quatre problématiques auxquelles se heurtent les systèmes classiques, dits de protection :

- Authentification : les entités du système sont en nombre inconnu et variable. Un système de protection doit connaître l'intégralité des actions possibles à un instant donné ce qui nécessite une mise à jour constante. Pour de grands réseaux, cette mise à jour nécessite une forte puissance de calcul ainsi qu'une faible latence.
- Délégation : une entité fait confiance à une autre pour effectuer une action donnée à sa place. La politique de sécurité doit donc être appliquée à la fin de la chaîne de délégation : le dernier élément de la chaîne prend la décision d'autoriser ou non l'action. Ceci nécessite la connaissance de toute la chaîne et donc, par assemblage de toutes les chaînes, une vision globale du système.
- Expressivité : les conditions et les restrictions d'accès à des ressources évoluent dans le temps. La formulation d'une politique de sécurité doit s'adapter à ces changements. Dans les systèmes de protection, cela requiert souvent la reconfiguration, la reconstruction voire la réécriture complète du système de contrôle.
- Politique locale de confiance : chaque entité du système applique une politique de sécurité qui lui est propre.

Les systèmes de gestion de confiance facilitent la délégation par la mise en place et l'application de politiques locales de sécurité. En effet, certaines règles ne concernent pas nécessairement tous les éléments d'un réseau. Par exemple, dans un réseau scolaire, les professeurs et les élèves doivent respecter une politique de sécurité commune à tous les utilisateurs (*e.g.* être membre de l'établissement) mais également des politiques spécifiques (*e.g.* respect de certaines plages horaires ou connexion dans des lieux réservés comme la salle des professeurs). Ces politiques locales peuvent être appliquées par des acteurs différents de la politique globale : un enseignant qui autorise ses élèves à accéder à certaines ressources tandis que les administrateurs donnent accès ou non au réseau de l'école. Cela se révèle efficace en

cas de changement d'autorité (*e.g.* remplacement d'un enseignant ou du directeur) ou de rôle (*e.g.* un enseignant qui devient professeur principal d'une classe) : seules les politiques dont elle faisait partie doivent être mises à jour. Les politiques locales permettent donc de raccourcir les chaines de délégation et par voie de conséquence, faciliter le maintien de la politique de sécurité.

Pour pouvoir exprimer plusieurs politiques de sécurité s'appliquant à des contextes (i.e. des personnes, des usages, des lieux et des instants) différents, l'entité ou le système chargé de spécifier la politique de sécurité doit disposer d'un langage suffisamment expressif. Blaze et al. ont proposé une une version dérivée du langage AWK pour exprimer les autorisations (Blaze et al., 1998). Les auteurs se sont toutefois concentrés sur l'algorithme d'autorisation (i.e. apporter une réponse à la problématique de la gestion de confiance) plutôt que sur le langage lui-même, sujet à de nombreuses évolutions.

2.2 Modéliser et mesurer la confiance

Comme l'a montré la section précédente, la confiance est une notion complexe. Celle-ci peut s'adapter à de multiples domaines mais est à l'origine destinée aux relations humaines. De ce fait, ce n'est pas une notion naturelle dans un cadre informatique. C'est pourquoi il existe une abondante littérature à ce sujet. La manipulation de la confiance par un ordinateur nécessite de pouvoir la modéliser. Nous explorons dans cette section les différents moyens de modéliser la confiance. Dans un deuxième temps, les diverses stratégies permettant de mesurer la confiance seront présentées. Enfin, une propriété particulière de la confiance sera mise en avant : la propagation.

2.2.1 Modèles de confiance

Afin de pouvoir manipuler la confiance, des modèles mathématiques simples sont généralement utilisés. Le plus simple de ces modèles consiste à considérer le niveau de confiance comme un élément d'un ensemble fini qui constitue alors des classes de confiance. Un exemple est l'ensemble { « Méfiance totale », « Ignorance », « Confiance minimale », « Confiance moyenne », « Beaucoup de confiance », « Confiance aveugle »} utilisé dans (Abdul-Rahman et Hailes, 1997). Bien qu'ayant l'avantage de la simplicité, cette représentation ne permet

pas une grande souplesse. Elle est en effet peu sensible aux changements. Une telle rigidité n'est pas des plus adaptées pour manipuler une notion ayant une forte composante humaine. En effet, elle ne permet pas de représenter des préférences comme dans la situation suivante :

- A a une confiance élevée en B et en C
- A préfère B à C

Il est dans ce cas impossible de déduire la préférence de A pour B sans devoir soit surclasser B soit sous-classer C : cette préférence n'est pas représentable dans le modèle de (Abdul-Rahman et Hailes, 1997).

Une représentation plus souple et plus courante consiste à assimiler le niveau de confiance à une valeur comprise dans l'intervalle $[0; 1]$. Cet intervalle est choisi par simplicité mais les possibilités peuvent naturellement être étendues par bijection à tout intervalle $[a; b]$. Le procédé est également valable pour tout intervalle, qu'il soit ouvert ou fermé, à une borne ou aux deux. Afin de ne pas surcharger les explications, seul le cas fermé sera détaillé, celui-ci étant le plus couramment utilisé. Le passage d'un ensemble fini à un intervalle, c.-à-d. d'une représentation discrète à une représentation continue, étend considérablement la sensibilité du modèle. En effet, une variable continue permet de prendre en compte les légères différences ou, comme souligné précédemment, les préférences pour telle ou telle entité, qui ne seraient pas flagrantes par ailleurs. Cette sensibilité s'adapte naturellement à la confiance. Cette forme mathématique est, de plus, aisément manipulable. C'est la raison de son adoption dans la majorité des travaux du domaine (Teacy et al., 2006 ; Avesani et al., 2005 ; Massa et Avesani, 2007b ; Kamvar et al., 2003 ; Massa et Avesani, 2009).

Comme expliqué dans la section 2.1, la confiance a des liens étroits avec la connaissance et l'incertitude. La façon la plus commune de représenter l'incertitude est la théorie des probabilités qui permet un raisonnement plus souple que le déterminisme en modélisant des connaissances partielles (Khaleghi et al., 2011). Par exemple, dans un jeu de Pile ou Face, il est difficile voire impossible de calculer toutes les variables qui entrent en jeu lors du lancement de la pièce. En revanche, il est possible de modéliser des informations statistiques (connaissance du passé) pour inférer les résultats des lancers futurs, sans toutefois avoir la garantie d'un résultat conforme au modèle.

Un défaut de connaissance peut être dû à différents facteurs : incertitude, imprécision, ambiguité ou d'autres formes d'imperfection (Khaleghi et al., 2011). Pour modéliser ces multiples aspects, d'autres théories sont parfois utilisées parmi

lesquelles la logique floue (Falcone et al., 2003), la logique modale (Lorini et Demolombe, 2008 ; Demolombe, 2001 ; Paglieri et al., 2014 ; Demolombe, 2011), la théorie de Dempster-Shafer (Yu et Singh, 2002) ou la logique subjective (Jøsang et al., 2006b ; Jøsang, 1999, 2001 ; Jøsang et al., 2006a ; Alhadad et al., 2014). Si ces théories expriment la confiance sous la même forme mathématique (une valeur entre 0 et 1), elles ne l'interprètent néanmoins pas de la même façon. On retrouve là une problématique similaire au domaine de la fusion de données : exprimer un défaut et le manipuler à l'aide de la théorie adéquate (Khaleghi et al., 2011). Différentes solutions selon le problème considéré sont représentées sur la figure 2.2.

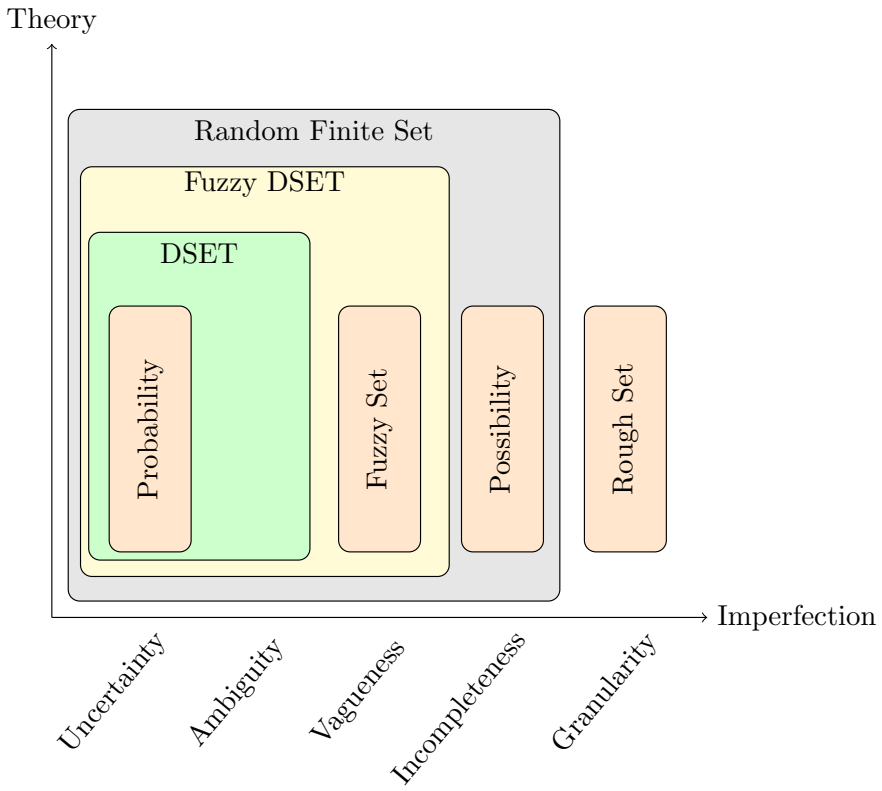


Figure 2.2 – Théories utilisées en fusion de données (Khaleghi et al., 2011)

Une imperfection en particulier a permis l'émergence de nouveaux modèles de confiance. En effet, une alternative aux représentations de type $[0; 1]$, dérivée de la théorie de Dempster-Shafer, a été proposée par Jøsang (Jøsang, 2001). Dans les modèles classiques où la confiance est représentée par une valeur entre 0 et 1, que représente la valeur $\frac{1}{2}$? Un avis mitigé suite à de nombreuses interactions autant

positives que négatives ? Un manque de connaissance due à l'absence totale d'interactions ? Afin de pallier ce manque, la logique subjective représente la confiance par un triplet de valeurs [3], toutes comprises entre 0 et 1 et telles que la somme des trois vaut 1. Ces valeurs forment ensemble une opinion sur une proposition telle que, par exemple, « B est de confiance ». Ces trois composantes de l'opinion sont respectivement la croyance que la proposition est vraie (confiance), la croyance qu'elle est fausse (méfiance) et l'inconnu. Par exemple, le triplet $(0,7, 0, 1, 0, 2)$ à propos d'une proposition P représente une confiance de $0,7$ dans la proposition, une méfiance de $0,1$ et un manque de connaissance de $0,2$. Cette représentation permet donc de représenter plus précisément le niveau de connaissance d'une entité, rendant ainsi le modèle plus expressif. Étant une représentation plus riche et par conséquent plus complexe que les précédentes, son utilisation est à privilégier lorsque l'inconnu a besoin d'une modélisation explicite, c.-à-d. lorsque l'on veut faire la différence entre « ne pas avoir confiance » et « se méfier ».
Cette volonté de marquer la différence entre « non-confiance » et « méfiance » se retrouve également dans les travaux (Lewicki et al., 1998 ; McKnight et Chervany, 2001b ; Da Costa Pereira, 2009). En particulier, les auteurs de (De Cock et Da Silva, 2006) font remarquer que l'incertitude peut se représenter sous forme d'intervalles. La confiance est dans ce cas une valeur, non connue a priori, comprise entre une valeur minimale (crédibilité) et une maximale (plausibilité). La taille de l'intervalle (i.e. la distance entre ses bornes) représente le degré d'hésitation associé à la confiance accordée. Divers exemples montrant les différences entre les modèles présentés sont résumés dans le tableau 2.3.

Indépendamment des caractéristiques qui fondent la confiance, la représentation de la méfiance est ainsi un atout considérable pour l'expressivité d'un modèle basé sur ces notions. Nous venons de voir comment est représentée la confiance dans la littérature. Ces divers modèles permettent de réaliser des calculs avec la confiance mais ne résolvent néanmoins pas la problématique de sa mesure, c.-à-d. comment produire de la confiance (numérique).

3. En réalité, c'est une représentation à quatre valeurs, la quatrième servant à représenter le comportement par défaut. Cette valeur étant décorrélée des trois autres, son emploi ne sera pas détaillé ici. Pour plus de précision voir (Jøsang et al., 2006b ; Jøsang, 2001 ; Jøsang et al., 2006a)

3. Intuitionistic Fuzzy Set theory et Interval Valued Fuzzy Set theory, variantes de la théorie des ensembles flous prouvées formellement équivalentes (Deschrijver et Kerre, 2003).

	Confiance	Confiance et méfiance		
		IFS [1]	IVFS[1]	Guha et al., 2004
	t	(t, d)	$[t; 1-d]$	$t-d$
Confiance totale	1	(1, 0)	[1; 1]	1
Méfiance totale	0	(0, 1)	[0; 0]	-1
Aucune connaissance	0	(0, 0)	[0; 1]	0
Confiance partielle	0.2	(0.2, 0)	[0.2; 1]	0.2
Confiance et méfiance partielles	0.6	(0.6, 0.4)	[0.6; 0.6]	0.2

Table 2.3 – Diverses représentation de la confiance (De Cock et Da Silva, 2006)

2.2.2 Mesures

Plusieurs travaux ont cherché à mesurer la confiance. La plupart de ces contributions sont basées sur un modèle de réseau dans lequel les divers nœuds interagissent. Les interactions sont alors sources de recommandations faites par les divers membres du réseau pour calculer leurs indices de confiance (Yu et Singh, 2002 ; Yan et al., 2003 ; Das et Islam, 2012 ; Jøsang et al., 2015). La recommandation est le processus par lequel un nœud i va communiquer sa confiance $C_{i,j}$ dans le nœud j. Très utilisée, cette mesure suppose cependant que les nœuds ont conscience les uns des autres. Si chaque nœud est isolé des autres et n'a pas conscience du réseau alors la recommandation est impossible. Bien que cette hypothèse soit vérifiée dans les réseaux sociaux ou sur le web, elle ne l'est cependant pas en général. Par exemple, la recommandation est difficile dans les réseaux dits centralisés où un serveur communique avec plusieurs clients qui ne se connaissent pas entre eux.

Lorsqu'il n'est pas possible d'obtenir les diverses appréciations des sources entre elles, il est encore possible de mesurer la confiance à partir de l'analyse des informations transmises par la source (Matt et al., 2010). Beaucoup de ces mesures sont construites sur la base de la théorie de l'argumentation (Dung, 1993) qui modélise un ensemble de propositions appelées *arguments* et d'*attaques* entre ces arguments. Les arguments sont assimilés aux nœuds d'un réseau et les attaques à des arêtes unidirectionnelles. La théorie de l'argumentation cherche à établir quels arguments sont rationnellement acceptables. Plus clairement, et comme illustré en figure 2.3, l'argument d est acceptable puisqu'il n'est attaqué par aucun autre. Il

en est de même pour l'argument f. En revanche, l'argument e est contesté à la fois par b et f.

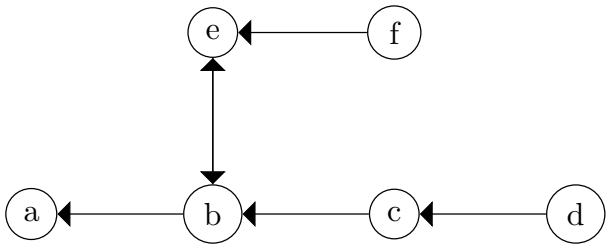


FIGURE 2.3 – Exemple d'un modèle utilisant 6 arguments et 6 attaques entre ces arguments.

Sur la base de cette théorie, divers modèles de confiance qui utilisent des sources d'informations ont été suggérés (PAGLIERI et al., 2014 ; STRANDERS et al., 2008 ; PARSONS et al., 2011 ; VILLATA et al., 2011). Ces modèles reposent sur deux hypothèses : l'ensemble des arguments utilisables ainsi que leurs liens (i.e. les attaques) sont connus et sont en nombre fini. Pour juger de la recevabilité d'un argument, il est donc nécessaire de pouvoir comparer l'ensemble de ceux à disposition et donc de pouvoir clairement identifier les attaques. Cela n'est pas toujours possible, notamment en présence d'incertitude. En effet, les arguments peuvent ne pas formellement s'opposer. Par exemple, les deux assertions « il fait chaud » et « il fait froid » ne s'opposent pas nécessairement : elles peuvent indiquer une température modérée, intermédiaire. La théorie de l'argumentation n'est donc pas adaptée lorsqu'un conflit entre informations n'est pas clairement identifié et est donc incertain.

Néanmoins, plusieurs travaux ont cherché à pallier cette faiblesse. Parmi ceux-là, (DA COSTA PEREIRA et al., 2011) proposent un modèle dans lequel l'acceptabilité des arguments (i.e. le degré de croyance qu'ils sont vrais) est évaluée selon la confiance attribuée à la source. Contrairement aux modélisations de (DUNG, 1993) et (VILLATA et al., 2013) où un argument est soit accepté soit rejeté, l'acceptabilité d'un argument est ici continue. Cependant, la confiance est considérée comme un concept unidimensionnel alors qu'elle est multidimensionnelle pour (VILLATA et al., 2013) qui la modélisent à partir de la compétence et de la sincérité de la source.

D'autres théories plus adaptées à la gestion de l'incertitude ont été utilisées (CAPRA et MUSOLESI, 2006 ; SUN et al., 2006 ; WANG et SINGH, 2007). En particulier, Sun et al. arguent que la confiance est une mesure de l'incertitude et

définissent ainsi leur mesure de confiance à partir de la probabilité qu'une entité effectue une certaine action (Sun et al., 2006). De même, (Wang et Singh, 2007) considèrent l'importance de la prise en compte de la certitude comme critère pour mesurer la confiance. Malgré une gestion efficiente des grandeurs réelles, ces travaux se basent cependant exclusivement sur une confiance monodimensionnelle.

2.2.3 Propagation

Comme cela a été exposé en section 2.1.3, la confiance se forme au fil des interactions entre les individus et plus largement entre des entités de toutes sortes, qu'elles soient humaines ou non. Cette confiance repose sur la connaissance que chaque entité a de l'autre. Cependant, l'acquisition d'un niveau de connaissance suffisant pour établir une relation de confiance peut être coûteux en temps. Par leurs échanges, les entités constituent des réseaux ou des communautés dans lesquelles elles se lient. Ces réseaux apportent alors un ensemble de ressources qui étaient inconnues ou inaccessibles. Afin d'accéder puis exploiter ces nouvelles ressources, une entité va avoir besoin d'agrandir son réseau de confiance. Elle va donc interagir avec de nouvelle entités pour les connaître et construire de nouveaux liens. Pour élargir son réseau plus rapidement, une entité peut s'appuyer sur les liens déjà établis par ses « amis », c.-à-d. qu'elle va accorder sa confiance à de nouvelles entités sur la base de la confiance qu'elle accorde à ses premières relations connues. La confiance se propage alors dans les réseaux via les liens et les niveaux de confiance déjà établis. La propagation répond donc à la problématique de la mesure de la confiance dans une entité en l'absence d'interactions ou d'éléments concernant directement l'entité ciblée.

De nombreux travaux ont étudié la propagation de confiance dans les réseaux. Chacun des modèles proposés s'appuie sur l'architecture du réseau et développe différent modes de propagation comme présenté dans le tableau 2.4. Les deux modes de propagation les plus couramment utilisés sont la *propagation en série* et la *propagation parallèle*.

La propagation en série, aussi appelée *concatenation*, sert à propager la confiance au travers d'une chaîne arbitrairement longue d'acteurs. Ce processus permet d'agrandir le réseau de confiance d'une entité en sollicitant les connaissances de ses amis (i.e. les membres de son propre réseau de confiance). Plusieurs travaux ont utilisé ce mode de propagation (Massa et Avesani, 2007b ; Kamvar et al., 2003 ; Demolombe, 2011 ; Esfandiari et Chandrasekharan, 2001)

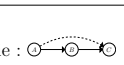
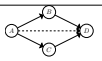
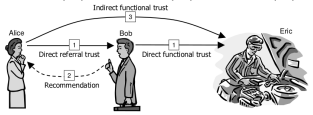
Propagation / Source	Série : 	Parallèle : 
Esfandiari et al. 2001	$Conf_A(C) = Conf_A(B) * Conf_B(C)$	$Conf_A(D) = [min(T); max(T)]$ avec $T = \{Conf_A(B) * Conf_B(D), Conf_A(C) * Conf_C(D)\}$
Mui et al. 2002	$Conf_A(C) = Conf_A(B) * Conf_B(C)$	
Kamvar et al. 2003	$Conf_A(C) = Conf_A(B) * Conf_B(C)$	$Conf_A(D) = Conf_A(B) * Conf_B(D) + Conf_A(C) * Conf_C(D)$
Jøsang et al. 2006	Typologie de la confiance $Conf_A(C) = Conf_A(B) * Conf_B(C)$  Figure 1: Transitive trust principle	Dépend de l'incertitude u t.q. $0 \leq Conf + u < 1$
Massa et al. 2007	$Conf_A(C) = Conf_A(B) * Conf_B(C)$	$Conf_A(D) = \frac{Conf_A(B) * Conf_B(D) + Conf_A(C) * Conf_C(D)}{Conf_A(B) + Conf_A(C)}$ $\forall x, y\ Conf_x(y) > \delta_A$, δ_A le seuil de confiance minimal de A
Wang et al. 2006	$Conf_A(C) = Conf_A(B) \otimes Conf_B(C)$ $\otimes$ est associatif, non-distributif sur $\oplus$	$Conf_A(C) = (Conf_A(B) \otimes Conf_B(C)) \oplus (Conf_A(C) \otimes Conf_C(D))$ $\oplus$ est commutatif et associatif
Richardson et al. 2003	$Conf_A(C) = Conf_A(B) \otimes Conf_B(C)$ $\otimes$ est associatif et distributif sur $\oplus$	$Conf_A(C) = (Conf_A(B) \otimes Conf_B(C)) \oplus (Conf_A(C) \otimes Conf_C(D))$ $\oplus$ est commutatif et associatif

Figure 2.4 – Opérateurs de propagation de confiance et leurs propriétés associées

mais, par souci de simplicité, l'opérateur choisi est systématiquement la multiplication. En effet, un opérateur mathématique $\otimes$ de propagation en série doit disposer des propriétés d'associativité (Richardson et al., 2003 ; Wang et Singh, 2006) mais doit également diminuer la confiance avec la taille de la chaîne (Sun et al., 2006).
Par exemple, pour la chaîne $A \to B \to C \to D$:

- la confiance $Conf_A(D)$ de A dans D est inférieure aux confiances intermédiaires de la chaîne :
$$Conf_A(D) \leq min(Conf_A(B), Conf_B(C), Conf_C(D))$$
- $(Conf_A(B) \otimes Conf_B(C)) \otimes Conf_C(D)$
$= Conf_A(B) \otimes (Conf_B(C) \otimes Conf_C(D))$

La multiplication est donc un opérateur simple mais qui répond à tous les critères requis.

La propagation parallèle, aussi appelée *agrégation*, permet d'augmenter le niveau de connaissance d'une entité au sujet d'une autre en agrégeant les avis qu'ont tous les membres qui sont communs à leurs réseaux respectifs. Par exemple, si une entité A souhaite avoir un niveau de confiance au sujet de l'entité B, celle-ci va combiner les niveaux de confiance des entités connues à la fois de A et de B. À l'instar de la concaténation, un opérateur d'agrégation $\oplus$ dispose de

la propriété d'associativité mais dispose en plus de la commutativité. En effet, les entités ayant toutes interagi directement avec la cible, elles peuvent être considérées indépendantes, auquel cas l'ordre dans lequel leurs connaissances sont combinées n'a pas d'importance (RICHARDSON et al., 2003 ; WANG et SINGH, 2006). De plus, la confiance agrégée est croissante avec le nombre d'entités sollicitées (SUN et al., 2006). Par exemple, si une entité A agrège les confiances $Conf_{E_1}(B)$ et $Conf_{E_2}(B)$ d'entités E_1 et E_2 au sujet d'une dernière entité B à travers deux mêmes chemins de propagation (i.e. $Conf_A(E_1) = Conf_A(E_2)$ et $Conf_{E_1}(B) = Conf_{E_2}(B)$) alors la propagation en série (i.e. selon un seul chemin) sera inférieur à l'agrégation des confiances :

$$Conf_A(E_1) \otimes Conf_{E_1}(B) \leq (Conf_A(E_1) \otimes Conf_{E_1}(B)) \oplus (Conf_A(E_2) \otimes Conf_{E_2}(B))$$

Les liens entre propagation et recommandation sont forts. En effet, la recommandation est le moyen par lequel les différentes entités d'un réseau se communiquent leurs connaissances. C'est donc à travers la recommandation (processus local) que s'effectue la propagation (processus global). Toutefois, comme nous l'avons observé dans la section 2.1.3, les différentes connaissances peuvent ne pas concerner le même objet. Par exemple, A a confiance dans B pour réparer sa voiture et B a confiance dans C pour identifier les constellations. Cela n'apporte cependant aucune information supplémentaire à A sur la confiance qu'il peut accorder à C. Ce phénomène est appelé **non-transitivité** de la confiance (DEMOLOMBE, 2011 ; CHRISTIANSON et HARBISON, 1996).

Afin de distinguer les différents objectifs de la confiance, Jøsang et al. ont introduit une typologie de cette dernière (JØSANG et al., 2006b,a). Différentes relations de confiance se nouent alors entre les entités :

- Confiance fonctionnelle directe : c'est le cas le plus simple où A a déjà interagi avec B et s'est donc forgé son propre avis sur ses capacités.
- Confiance directe dans les recommandations : A a confiance dans B pour tenir compte de ses intérêts.
- Confiance fonctionnelle indirecte : sans avoir interagit avec, A a confiance dans C grâce à un intermédiaire B. B a une confiance fonctionnelle directe dans C et A a confiance directe dans les recommandations de B.

Grâce à cette typologie, la confiance devient transitive : une confiance directe fonctionnelle se propage grâce aux confiances dans les recommandations, ce qui préserve la cible initiale de la confiance (par exemple, les compétences en mécanique de B).

D'autres travaux ont exploré la non-transitivité au travers du prisme de la méfiance. Comme expliqué dans la section précédente (cf section 2.2.1), la confiance et son opposé, la méfiance, ne peuvent pas nécessairement être représentées par une unique valeur. Divers travaux ont proposé des solutions pour aborder non seulement la propagation de la confiance mais également celui de la méfiance et du manque de connaissances (Guha et al., 2004). Dans (De Cock et Da Silva, 2006), les auteurs ont proposé divers comportements possibles dans chaque cas. Par exemple, si A se méfie de B, elle peut soit suivre le contraire de sa recommandation, soit considérer que son avis n'est pas contributif (C restant un inconnu aux yeux de A). Dans le cas où une entité A ne connaît pas une entité B, elle peut choisir de s'en méfier ou de ne pas tenir compte de son avis. Tous ces comportements sont répertoriés dans la table 2.4.

Opinion de A au sujet de B	Opinion de B au sujet de C	Opinion de A au sujet de C
$(1, 0)$	$(Conf_B(C), Mef_B(C))$	$(Conf_B(C), Mef_B(C))$
$(0, 1)$	$(Conf_B(C), Mef_B(C))$	$(Mef_B(C), Conf_B(C))$
		$(0, 0)$
$(0, 0)$	$(Conf_B(C), Mef_B(C))$	$(0, 0)$
		$(0, Mef_B(C))$

Table 2.4 – Règles de propagation de la confiance selon différents comportements

Bien sûr ces comportements sont extrêmes. Une atténuation des opinions de l'autre est également possible. Le degré de confiance devient alors une pondération des recommandations. Ces règles reflètent le comportement de l'entité considérée (humain, capteur, système, ...). Toutes les entités n'adoptent pas nécessairement le même comportement ce qui peut conduire à des phénomènes de propagation parfois complexes.

2.3 Discussion

Cette section détaille quelques observations sur les différents aspects de la confiance présents dans la littérature. Au travers de différents domaines, la confiance s'accompagne de diverses notions. En accord avec les sections précédentes, le tableau 2.5 reprend certaines définitions majeures de la confiance et les

différentes notions qui y sont liées.

Une relation de confiance dépend de différents facteurs qui peuvent être **externes** à la relation (*i.e.* environnementaux) ou bien **internes** (i.e. qui dépendent des entités impliquées) (Falcone et al., 2003).

Les facteurs externes sont la manifestation de l'influence de l'environnement sur la relation de confiance. Par exemple, la capacité pour une personne de se rendre à un lieu donné dans un laps de temps fixé dépend directement de l'endroit où elle se trouve à l'instant présent. Une voiture, même bien entretenue, peut tomber en panne indépendamment de la volonté de son conducteur. Dans ce cas, le risque de panne dépend de facteurs extérieurs tels que l'état de la route ou les conditions météorologiques (*e.g.* grand froid). Parmi les facteurs externes que l'on retrouve dans la littérature, le risque est incontestablement celui qui revient le plus souvent.

Une relation de confiance d'une entité A vers une entité B fait intervenir deux rôles : une cible et un évaluateur qui détermine la confiance qu'il a dans la cible. La confiance dépend alors de facteurs internes qui peuvent être spécifiques à la cible ou à l'évaluateur. Par exemple, les définitions de la confiance présentées dans la section 2.1 s'appuient sur la connaissance de l'évaluateur ou la sincérité de la cible. Les facteurs spécifiques à la cible sont des caractéristiques de celle-ci, observables et évaluables par une autre entité : compétence, sincérité, actions, etc. Par nature, ces critères sont nécessairement pris explicitement en compte dans un modèle de la confiance. En effet, les critères sont internes à la relation mais externes au processus d'évaluation car faisant partie de l'entité cible de la relation. Dans les modèles de confiance, ceux-ci sont donc systématiquement exprimés en tant que grandeur séparée, laquelle est ensuite reliée à la confiance par le biais d'équations. Au contraire, les facteurs internes dépendants de l'évaluateur peuvent être pris en compte implicitement ou explicitement. En effet, ceux-ci, par définition, font partie intégrante du processus d'évaluation de la confiance puisqu'ils dépendent de l'entité qui réalise ce processus. En particulier, la connaissance est souvent considérée implicitement dans les mesures soit par des mécanismes de rétroaction (Paglieri et al., 2014), ou plus généralement en prenant en compte le temps (Seigneur, 2006), soit en prenant en compte des informations a priori considérées comme des preuves des intentions d'autrui (Jøsang et al., 2007 ; Wang et Singh, 2007 ; Ismail et Jøsang, 2002).

Auteurs	Définition	Critères internes		Critères externes
		Origine	Cible	
Deutsch, 1958	On peut dire d'un individu qu'il a confiance en l'occurrence d'un évènement s'il s'attend à cette occurrence et que ses attentes mènent à un comportement qu'il perçoit comme ayant des conséquences plus négatives, si son attente n'est pas confirmée, que positives dans le cas contraire.			Risque
Luhmann, 1979	La confiance est un moyen subjectif de réduire la complexité perçue du futur en supposant, sur la base d'une connaissance personnelle limitée, des actions bénéfiques de la part d'acteurs indépendants.	Connaissance	Bienveillance	
Lewis et Weigert, 1985	Nous choisissons les personnes à qui nous faisons confiance, par rapport à quoi et dans quelles circonstances. Nous basons nos choix sur ce que nous considérons comme étant de « bonnes raisons », lesquelles constituent les preuves de la crédibilité.	Émotions	Crédibilité	Circonstances
Mayer et Davis, 1995	La volonté d'une personne A d'être vulnérable aux actions d'une autre personne B. Cette volonté est basée sur l'attente que B accomplisse une action spécifique importante pour A. Elle ne tient pas compte de la capacité de surveiller ou de contrôler B.	Vulnérabilité, connaissance		
McKnight et Chervany, 1996	Typologie de la confiance		Compétence, bienveillance, intégrité	
Grandison et Sloman, 2000	La croyance ferme en la compétence d'une entité à agir de manière fiable au travers d'un contexte spécifique.		Compétence	
Gambetta, 1988	La confiance est la probabilité subjective par laquelle un individu A s'attend à ce qu'un autre individu B accomplisse une action donnée de laquelle dépend son bien-être.	Connaissance		
Mui et al., 2002	La confiance est l'attente personnelle qu'un agent a à propos du futur comportement d'un autre en se basant sur l'historique de leurs rencontres.	Historique		
Ruohomaa, 2004	La confiance est la limite jusqu'à laquelle une personne a la volonté de participer à une action donnée avec un partenaire donné, en considérant les risques et les motivations impliquées.			Risque
Lorini et Demolombe, 2008	Un agent i a confiance dans un agent j pour effectuer une certaine action α si et seulement si i a un certain but et pense que j va effectuer α de telle façon que son but sera atteint.		Compétence, Sincérité	

Table 2.5 – Synthèse des définitions de la confiance et des critères associés

Au travers de ses multiples définitions, la confiance s'applique à des acteurs de natures diverses. En effet, la confiance est à l'origine une composante essentiellement humaine. Elle cristallise les émotions d'un sujet en une perception des actions d'un autre individu. Cette perception influence ensuite leurs interactions et évolue avec elles. Plus récemment, de nombreux travaux ont cherché à adapter la confiance à la diversité des réseaux modernes, en particulier Internet. Les interactions n'ont plus seulement lieu entre humains mais entre une multitude d'entités dont la nature est le plus souvent inconnue.

Au sein d'un réseau, la confiance permet de faciliter les échanges en renforçant la coopération entre les entités, quelque soit leur nature. Cependant, la confiance évolue au fil du temps et des interactions. Dans des réseaux de grande taille, ces dernières sont plus diffuses : seuls quelques couples d'éléments du réseaux échangent régulièrement. La plupart des liens ne s'établissent qu'un nombre limité de fois. La confiance est alors plus difficile à établir. Dans ce cas, une entité cherchant à évaluer la confiance à accorder à une autre entité, inconnue ou dont elle sait très peu de choses, va exploiter la connaissance des autres entités du réseau. Elle peut le faire de deux façons :

1. Approche globale : évaluer la confiance à partir de la réputation de l'entité ciblée.
2. Approche locale : évaluer la confiance à partir des relations de confiance déjà établies (cf section 2.2.3).

La réputation exploite toutes les interactions entre une entité et le reste du réseau tandis que la confiance n'exploite que les interactions entre deux d'entre elles. A cause de cette perception de multiples actions non nécessairement liées entre elles (*i.e.* elle n'ont pas le même but), la réputation est très générique, au contraire de la confiance qui dépend d'un objectif précis (Jøsang et al., 2006b). De plus, la réputation n'est pertinente que lorsque l'entité interagit peu mais avec un grand nombre d'entités. Dans le cas contraire, de fortes relations de confiance se nouent et la confiance est alors une meilleure alternative. Ceci est résumé dans le tableau 2.6.

Pour conclure, la confiance constitue un moyen adapté à la gestion de tout type de conflit (Jøsang et al., 2015 ; Wang et Singh, 2007). La détection de cyberattaques est un conflit particulier qui oppose les intentions de l'attaquant aux objectifs du système, les seconds étant nécessairement contraires aux premières. Dans le cas contraire, une attaque serait inutile puisque l'attaquant serait satisfait par le fonctionnement normal du système.

Approche	Nombre moyen d'interactions par entité	Nombre d'entités
Globale (réputation)	faible	élevé
Locale (confiance)	élevé	faible

TABLE 2.6 – Critères de choix pour mesurer la confiance dans une entité inconnue au sein d'un réseau

La confiance satisfait en particulier deux propriétés adéquates pour détecter des cyberattaques :

- Elle s'adapte à des environnements mêlant des acteurs de natures variées, ce qui correspond aux systèmes d'information modernes. Les vulnérabilités d'un système sont exploitées par chacun de ses acteurs qui peuvent être logiciels (dépassement de tampon, dépassement d'entier), matériel réseau (DDoS, IP spoofing, ARP poisonning), humain (ingénierie sociale) ou plus récemment des capteurs (GPS spoofing, injection d'informations).
- Elle permet de gérer l'inconnu. Les attaques sont de plus en plus sophistiquées et les systèmes de détection doivent faire face à l'inventivité des attaquants qui tentent sans cesse de contourner les nouvelles mesures de sécurité.

La confiance dépend d'un certain nombre de notions qui peuvent être internes ou externes à la relation. Parmi ces notions, nous retrouvons la compétence, la sincérité, la connaissance et la bienveillance comme critères internes tandis que le risque ou les circonstances constituent des critères externes.

Nous avons vu que la confiance peut se propager au sein des réseaux de deux façons selon la structure du système : via la confiance ou via la réputation. Pour propager de la confiance à l'aide de la réputation, les systèmes à privilégier sont ceux où les éléments ont peu d'interactions mais interagissent avec un grand nombre d'autres éléments. Dans le contexte des systèmes embarqués (*e.g.* à bord des navires), les interactions sont nombreuses entre les entités et souvent régulières. Par exemple, un capteur envoie des informations à une certaine fréquence et à un certain destinataire qui varient peu voire pas dans le temps : l'architecture du système est stable. La réputation n'est donc pas adaptée à notre contexte.

Chapitre 3

Modélisation pour la sécurité d'un système d'information

Nous étudions la confiance dans le contexte de la sécurité des systèmes d'information (SI). Ce chapitre présente notre modélisation de tels systèmes. Sur la base de la théorie des systèmes complexes, nous modélisons un SI par assemblage de blocs fonctionnels. Ces blocs sont liés par de multiples relations qui décrivent le fonctionnement du système global. Outre la modélisation du système, ce chapitre expose la modélisation des informations numériques que celui-ci manipule ainsi que des attaques pouvant le cibler.
La section 3.1 pose les pré-requis pour la modélisation d'un SI. Elle introduit ensuite la notion de système complexe puis leur modélisation. Les sections suivantes présentent nos choix de modélisation des différents éléments clés du contexte : le système, l'information et les attaques qui menacent leur sécurité.

3.1 Systèmes d'information et complexité

Afin de modéliser les systèmes d'information, nous en analysons dans un premier temps les principales caractéristiques pertinentes dans notre contexte. En particulier, les systèmes d'information navals sont des systèmes complexes. Nous présentons donc dans un second temps les grands principes de la théorie des systèmes complexes. Enfin, les deux grandes approches de modélisation de ces systèmes seront comparées.

3.1.1 Modéliser un système d'information : pré-requis

Un système d'information peut être défini de multiples manières. Par exemple, De Courcy définit un SI comme « un ensemble organisé de ressources qui collectent, stockent, traitent et distribuent de l'information » (De Courcy, 1992). Dans ce travail de thèse, nous définissons « ressources » par « entités interconnectées », lesquelles contribuent à diverses fonctions : mesure, analyse, traitement, voire prise de décision. Chaque entité du système, quelque soit sa fonction, peut être complexe, c'est-à-dire composée de multiples capteurs, logiciels, automates, humains, etc. Chacune de leurs fonctions peut être attribuée à un *bloc fonctionnel*. Un bloc est donc une partie d'une entité qui réalise une unique fonction. Afin de ne pas alourdir le discours, nous utiliserons indifféremment les termes « bloc » et « bloc fonctionnel » dans la suite de ce document.

Chacun des blocs d'un système peut être la cible d'une ou plusieurs actions visant à perturber le fonctionnement normal du système global. Nous souhaitons adresser la problématique de la sécurité des systèmes d'information à l'aide de mesures de confiance que le système peut avoir dans ses composants ainsi que les sources d'information qui l'alimentent. Une mesure de confiance dans un SI repose sur la modélisation de ce dernier. Un modèle d'un système est une représentation simplifiée de celui-ci mettant en évidence les aspects utiles à la résolution d'un problème donné (Landry et Santerre, 1999).

Dans notre cas d'étude, la modélisation d'un SI doit inclure celle de son architecture interne. Nous nous intéressons aux possibilités de menaces internes au système. Dans ce cadre, la compromission d'un des éléments du SI peut entraîner celle d'autres éléments, a priori non exposés. La sécurité globale du système dépend alors de la sécurité de chacun de ses composants. Le composant le plus faible du système, une fois compromis, augmente la vulnérabilité du système global. Une modélisation d'un système d'information considérant la sécurité interne du système doit donc prendre en compte la modélisation de ses composants.

Par ailleurs, chaque élément du SI dispose de caractéristiques qui lui sont propres : fonction, précision, vulnérabilité, interactions avec certains composants, etc. Chacune de ces caractéristiques influencent la confiance qui peut leur être accordée. Par exemple, à bord d'un navire, deux thermomètres qui mesurent respectivement les températures du moteur et celle du frigo n'ont pas les mêmes fonctions : le premier sert à préserver le navire d'une avarie moteur, le second tend à optimiser la conservation des aliments. La confiance dans ces deux capteurs est

différente car ils affectent différemment la sécurité du SI. De leur bon fonctionnement dépendent respectivement l'état de la chaine de propulsion et la santé de l'équipage. Une modélisation de système d'information à des fins de sécurité doit donc prendre en compte les spécificités des composants. En effet, ceux-ci ne remplissent pas les mêmes fonctions et disposent de caractéristiques adaptées à leurs tâches respectives.

Enfin, nous souhaitons pouvoir modéliser et simuler la dynamique du système, c'est-à-dire l'évolution temporelle de ses constituants et de leurs interactions. Nous considérons chaque composant du système comme étant une boîte noire : le traitement de l'information effectué par le composant est inconnu. Seules ses entrées, sorties et méta-informations sont connues. L'information est alors un actif du système, c'est-à-dire une ressource qui influence son fonctionnement : le système agit sur l'information mais l'information modifie également le système (*e.g.* via des prises de décisions).

En conclusion, nous souhaitons élaborer un modèle de système d'information permettant d'en assurer la sécurité par la détection de malversations. Un tel modèle s'appuie sur l'architecture interne en considérant chaque bloc comme un acteur de la sécurité du système global. Cependant, la modélisation d'un système à des fins de sécurité doit également mettre en avant l'aspect fonctionnel du système. En effet, les relations et interactions entre les composants importent plus que leur connaissance exhaustive. Le système n'est pas qu'un agrégat de composants mais bien un ensemble cohérent centré sur l'acquisition, le stockage, le traitement et l'émission de sa ressource première : l'information. La prise en compte de ces deux aspects (structurel et fonctionnel) permet d'adresser la problématique de la falsification des informations reçues et manipulées par le système en prenant en compte d'éventuelles malversations internes.

3.1.2 Caractérisation des systèmes complexes

Comme nous l'avons montré dans la section précédente, un système d'information est un ensemble organisé d'entités complexes composées de multiples blocs fonctionnels. Il constitue donc à ce titre un système complexe dont il existe de nombreuses définitions. Cependant, des diverses approches proposées, quatre caractéristiques fondamentales peuvent être retenues (Ray, 2003) :

- Organisation : un système dispose de plusieurs niveaux hiérarchiques internes reposant sur des relations entre les nombreux éléments qui le com-

posent.

- Interactions : un système repose sur de multiples relations entre ses éléments qui peuvent être définies, en particulier mais pas seulement, par les flux d'informations tant internes (*i.e.* entre éléments du système) qu'externes (*i.e.* entre le système et son environnement).
- Globalité : un système est un tout non réductible à ses parties. Ceci est dû en particulier au phénomène d'*émergence* : le système global possède des propriétés qui sont non-déductibles de celles de ses éléments.
- Complexité : un système est complexe par son degré élevé d'organisation, l'incertitude de son environnement et des éléments et interconnexions difficiles voire impossibles à dénombrer.

Toutes ces caractéristiques reposent sur la mise en relation de multiples composants, ce qui structure le système. Ainsi la notion de système, en particulier complexe, est intiment liée à celle de structure voire de hiérarchie entre ses éléments. En effet, les propriétés de l'ensemble du système dépendent du nombre d'éléments et de leur nature mais également des relations que ceux-ci ont établis et qui peuvent évoluer. Par exemple, les isomères sont des composés chimiques constitués des mêmes éléments mais différenciés par leurs agencements structurels qui leurs confèrent des propriétés différentes. La figure 3.1 montre la structure d'un système, organisée en niveaux hiérarchiques. Les niveaux supérieurs sont les plus complexes et les moins détaillés, au contraire des niveaux inférieurs qui décomposent le système en éléments plus petits et plus simples. Cependant, aussi simples soient-ils, plus les éléments du système sont nombreux et plus le nombre d'interactions à considérer le sera également. Ainsi, la multiplicité des éléments complexifie l'étude du système global.

Outre son aspect structurel, un système se décrit également par sa dynamique, c'est-à-dire l'évolution de ses entrées et sorties (du système global mais également de chacun de ses composants) dans le temps. La dynamique d'un système repose en grande partie sur ses boucles de rétroaction lesquelles sont responsables de la non-linéarité du système (Forrester et Sylvestre-Baron, 1984).

Les boucles de rétroaction sont l'action d'un système sur lui-même, c'est-à-dire que le système global ou un sous-ensemble de ses composants se ré-alimente à partir de ses propres informations traitées. Il existe deux types de rétroaction : les boucles positives et les boucles négatives. Les boucles positives accumulent les changements du système, c'est-à-dire qu'elles accroissent au cours du temps les écarts entre les sorties et les objectifs initiaux du système, permettant alors

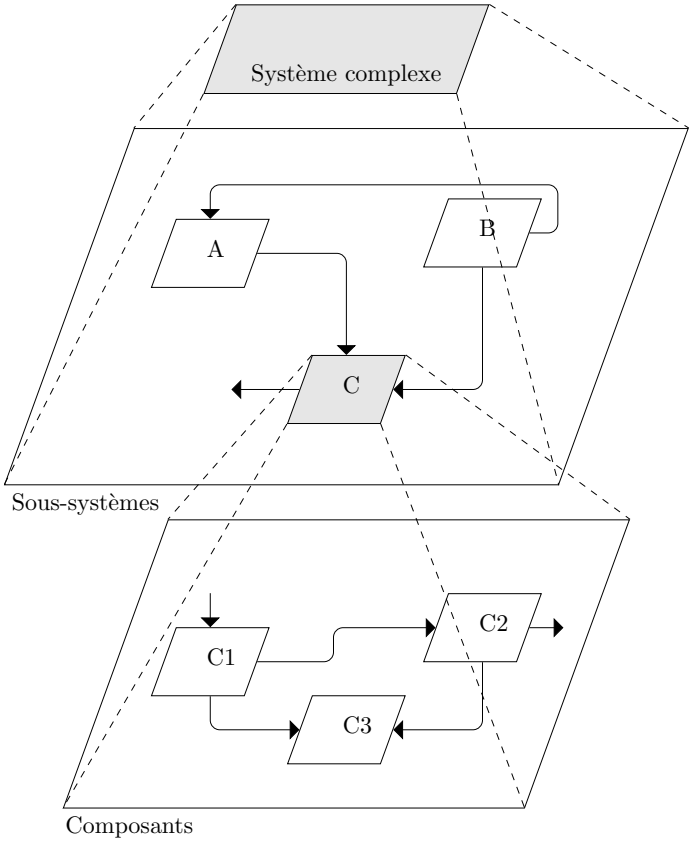


FIGURE 3.1 – Structure hiérarchique d'un système complexe

à ce dernier d'évoluer. Elles conduisent parfois à l'émergence de phénomènes ou de propriétés. Au contraire, les boucles négatives concourent à la stabilisation du système en atténuant à chaque itération l'impact sur le système des processus qui les composent. Par exemple, un thermostat régule la température d'une pièce en se basant sur celle mesurée par un thermomètre, ceci dans le but d'atteindre une température fixe de référence, par exemple 20 °. Cette dernière constitue un point d'équilibre dans l'évolution du système de régulation de la température. La boucle thermostat-thermomètre est donc une boucle négative qui stabilise le système (la pièce) dans un état particulier (*e.g.* une température de 20 °).

La dynamique d'un système dépend des interactions entre les composants lesquelles évoluent dans l'espace et le temps. En effet, le temps influence l'existence voire la nature des flux du système. Par exemple, dans le cadre des systèmes d'information, les interactions entre un client et un serveur ne sont pas durables : elles ne se produisent pas à intervalles de temps réguliers et ne se reproduiront

pas nécessairement. Ainsi, demander une page web n'engage pas à la redemander ni à en demander une autre au même serveur. La nature des interactions peut également évoluer au cours du temps : deux personnes qui ne se connaissent pas a priori peuvent, au fil de leur échanges, modifier leurs rapports pour passer d'une vague connaissance à une amitié. Ainsi, le nombre et la nature de leurs échanges évoluent en même temps que leur relation.

Une interaction dépend du temps mais également de l'espace. En effet, certaines entités ne sont mises en relation que si celles-ci sont suffisamment proches. Par exemple, un téléphone mobile est lié à la ou les antennes qui sont les plus proches de lui. Lors du déplacement de l'utilisateur, le téléphone communique avec ces antennes pour établir une communication.

Comme le montre la figure 3.1, un système complexe est structuré en différents niveaux hiérarchiques. L'étude de chaque niveau permet de comprendre de différentes manières le fonctionnement du système (voir la section 3.1.3 pour plus de précisions). Toutefois, la compréhension de ces niveaux, indépendamment les uns des autres ne permet pas de saisir le comportement global du système. L'exemple le plus couramment utilisé est le cerveau : la compréhension des interactions entre les neurones, cortex ou hémisphères ne permet pas de saisir l'émergence de la conscience. L'émergence est alors un phénomène ascendant dans la structure du système : ce sont les interactions entre les éléments les plus simples qui produisent un résultat n'affectant que des éléments de complexité supérieure. Pour reprendre l'exemple du cerveau, l'activation des synapses produit la conscience qui n'est un phénomène constaté que pour le cerveau (*i.e.* le système complet) sans être attribuable à aucun élément plus simple (*e.g.* un hémisphère).

La théorie des systèmes complexes reste centrée sur le système plus que sur la complexité (Duboz, 2004). Toutefois, d'autres domaines scientifiques ont étudié la complexité pour elle-même. Selon (Manson, 2001), il existe trois types de complexité : *algorithmique*, *déterministe* et *globale*.
La complexité algorithmique recouvre deux aspects selon son domaine d'étude :

- En informatique théorique, la complexité d'un programme est son besoin en ressources (espace mémoire et temps d'exécution) pour résoudre un problème donné. Par extension, la complexité d'un problème est la complexité du programme le plus efficace capable de le résoudre. Les problèmes sont ainsi classés selon la difficulté à les résoudre algorithmiquement.
- En théorie algorithmique de l'information, un objet est d'autant plus complexe qu'il faut beaucoup d'information pour le décrire. Plus précisément,

la complexité d'une suite de symboles est la taille du plus court programme capable d'engendrer cette suite (Kolmogorov, 1965). Ainsi, un système est considéré aussi complexe que sa description laquelle, lorsqu'elle est connue, permet de reproduire le comportement du système considéré. Cette mesure de la complexité ne reflète toutefois pas l'organisation structurelle de la suite. En effet, plus la complexité de Kolmogorov d'une suite est élevée et moins celle-ci possède de structure. Une suite complexe au sens de Kolmogorov est aléatoire et ne possède donc aucune structure. Dans (Bennett, 1988), Bennett expose une mesure de la complexité organisée appelée aussi *profondeur logique*. Elle s'exprime comme le temps que prend le plus court programme capable de décrire une suite de symboles donnée. Comme il a été évoqué ci-dessus, cette suite peut être la description d'un système. Ainsi un système structurellement complexe est facile à décrire à l'aide d'un programme (*e.g.* « afficher chaque élément du système ») mais le passage du programme à la description est long à cause du nombre important d'éléments en jeu : entités, relations et interactions du système avec lui-même ou avec son environnement. Au sens de Bennett, la complexité d'un objet est donc une mesure de son niveau d'organisation structurelle.

La complexité algorithmique, quelque soit l'angle sous lequel elle est étudiée, a l'avantage d'être bien définie formellement et constitue à ce titre un outil universel d'étude de la complexité. Toutefois, ses mesures sont non-calculables (complexité de Kolmogorov) ou asymptotiques (informatique théorique) ce qui les rend difficilement utilisables en pratique (Delahaye, 2013).

La complexité déterministe est liée à l'étude des systèmes chaotiques et en particulier au phénomène d'émergence. Un système est dit chaotique s'il est dynamique (*i.e.* évolutif au cours du temps) et possède la caractéristique d'être extrêmement sensible aux conditions initiales. C'est cet aspect qui donne au système sa complexité notamment via l'utilisation de rétroactions. En revanche, le déterminisme vient de l'utilisation d'un nombre restreint de variables-clés reliées entre elles par des équations connues qui paramètrent le système et permettent d'en reproduire le comportement.

Enfin, la complexité totale concerne l'étude des systèmes composés d'entités inter-reliées. La complexité de tels systèmes repose sur les multiples relations nouées entre les différents composants mais aussi sur sa structure interne, ses échanges avec son environnement, sa mémoire et tous ses moyens d'évolution. La

"mémoire" du système est ici entendue comme la persistance des structures internes et la conservation des informations et données. En effet, certains systèmes, dit *adaptatifs*, sont considérés complexes car ils ont une tendance naturelle à évoluer et s'adapter à leur environnement (Holland, 1992). Leur complexité provient des modifications de leur architecture interne provoquées par leurs interactions avec l'environnement extérieur. Cette modification de leur structure altère également les interactions entre leurs composant. Ils sont donc en perpétuelle évolution. Ce type de système est particulièrement fréquent en biologie. Par exemple, ce mécanisme d'évolution est la base du fonctionnement du système immunitaire.

Nous avons vu qu'un système se caractérise par quatre propriétés : organisation, interactions, globalité et complexité. En particulier, la complexité d'un système provient à fois de son organisation et des nombreuses interactions qui permettent l'apparition de phénomènes émergents.

Les systèmes d'information sont un cas particulier où les interactions entre les entités sont des flux d'informations. Ils se distinguent par leur forte hétérogénéité : ils se composent de multiples capteurs, automates, logiciels voire humains. Chacun de ces composants dispose de caractéristiques qui lui sont propres : par exemple, la précision des capteurs, la fiabilité des automates, la flexibilité des logiciels et la capacité humaine de raisonnement. Cette multitude de composants divers engendre une forte complexité du système autant qu'une grande organisation. En effet, un ensemble aussi hétérogène est nécessairement organisé pour remplir efficacement ses fonctions. Par exemple, un capteur n'est pas chargé des prises de décision de même qu'un humain n'effectue pas les mesures. Du fait des nombreux échanges d'information et de leur forte hétérogénéité et organisation, les systèmes d'information peuvent donc être considérés comme complexe. La section suivante présente les différentes approches permettant de modéliser un système complexe.

3.1.3 Principes de modélisation des systèmes complexes

La grande majorité des modélisations d'un système complexe reposent sur une approche analytique ou une approche systémique.

L'approche analytique décompose la réalité (le système) en éléments plus simples afin de les étudier séparément. Elle se focalise sur une connaissance aussi complète que possible de chaque élément pour en déduire le comportement global de l'ensemble en agrégeant les connaissances.

Contrairement à l'approche analytique qui consiste à décrire un système le plus précisément possible via son architecture interne, la systémique se concentre sur son aspect fonctionnel en intégrant notamment la relation entre le système, pris comme un tout, et son environnement. Les concepts fondateurs de la systémique, présentée comme le *nouveau discours de la méthode*, ont été énoncés par Le Moigne (Le Moigne, 1994). Ces préceptes s'opposent à l'approche cartésienne comme en témoigne le tableau 3.1.

Analytique	Systémique
Évidence : ne jamais concevoir une chose comme étant vraie sauf à la connaître comme telle	Pertinence : toute chose se définit selon les intentions du modélisateur
Réductionnisme : diviser la difficulté en autant de parcelles nécessaire à leur compréhension	Globalisme : tout objet est une partie immergée et active d'un ensemble plus grand
Causalisme : analyser en allant du plus simple vers le plus complexe	Téléologisme : analyser les liens entre le comportement d'un objet et ses finalités
Exhaustivité : S'assurer de ne rien omettre	Agrégativité : Toute représentation d'un système est nécessairement simplificatrice

Table 3.1 – Liste des préceptes des approches analytique et systémique

L'évidence pousse à s'interroger sur la véracité des choses tandis que la pertinence propose de s'en tenir aux objectifs du modèle. Dans l'approche systémique, seules les finalités sont à prendre en compte. Contrairement au réductionnisme qui vise à isoler les éléments pour mieux les étudier et les comprendre, le globalisme repose sur une hypothèse d'ouverture des systèmes : tout objet évolue dans un environnement dont la compréhension est nécessaire à la connaissance de l'objet (Le Moigne, 1994). À l'opposé du principe de causalisme qui suppose d'étudier ou modéliser un objet ou phénomène à partir de ses causes, le téléologisme s'appuie sur la connaissance des conséquences pour guider l'étude ou la modélisation. Enfin, l'agrégativité s'oppose à l'exhaustivité : un modèle ne nécessite pas une connaissance complète mais seulement une connaissance suffisante pour représenter un objet. L'agrégativité suppose donc par principe que la connaissance est nécessairement parcellaire.

L'approche analytique se focalise sur l'étude du système lui-même tandis que

la systémique le considère comme évoluant dans un environnement. La modélisation d'un système est donc soit guidée par la définition des causes, c.-à-d. un ensemble restreint de paramètres propres et de règles d'évolutions, soit par la définition des conséquences, c.-à-d. l'impact du système sur son environnement. Dans ce dernier cas, l'état du système n'est pas mis en valeur au profit de ce qu'il produit.

Nous souhaitons modéliser la dynamique du système autant que son architecture interne. De plus, nous considérons chaque élément comme une boite noire. En conséquence, notre modèle ne considère pas le fonctionnement interne du système mais repose plutôt sur ses finalités et son actif principal : l'information. L'approche systémique est donc celle qui répond le mieux à nos contraintes de modélisation.

3.2 Modélisation d'un système d'information

La modélisation d'un système d'information est nécessaire à la construction d'une mesure de confiance dans le système global. Nous présentons ci-dessous notre modélisation d'un système d'information, de ses entités ainsi que de ses blocs fonctionnels. En particulier, nous présentons cinq types élémentaires de blocs fonctionnels. Cette typologie est basée sur la façon dont les blocs échangent de l'information. Dans ce travail, nous supposons que le temps est discret : l'évolution du système d'information est rythmée par la cadence d'une horloge. En effet, dans un système de navigation, les informations sont envoyées à une fréquence qui dépend des spécifications du capteur. Ces informations alimentent ensuite le reste du système. Nous fixons le rythme du système d'information à la fréquence du capteur le plus rapide.

3.2.1 Modélisation des entités et des blocs fonctionnels

Un système d'information est un ensemble d'entités : capteurs, automates, logiciels, sous-systèmes, etc. Chacune de ces entités contribue à diverses fonctions telles que la navigation ou la propulsion dans le cas d'un navire. Un bloc fonctionnel est une partie d'une entité qui assure une unique fonction. Par exemple, les yeux (blocs fonctionnels) collectent de l'information visuelle utilisée en premier lieu par la tête (entité) puis par le corps (vu comme un SI).

A partir de la façon dont les blocs interagissent avec l'information, nous distinguons cinq types élémentaires de blocs fonctionnels : le bloc isolé (pas d'inter-

action), le collecteur (collecte), le bloc de traitement (traite), la source (distribue) et le bloc de rétroaction (traitement itératif).

Le cas d'un bloc isolé est un cas particulier. En effet, ce bloc ne reçoit ni ne produit d'information. Il ne dispose donc d'aucune information à traiter ou stocker. Par conséquent, il n'intervient pas dans la chaîne de propagation de l'information. De ce fait, nous choisissons de ne pas l'intégrer à notre modélisation.

Soient $\mathcal{B}$ l'ensemble des blocs et $\mathcal{S}, \mathcal{C}, \mathcal{T}, \mathcal{F}$ les ensembles respectifs des sources, des collecteurs, des blocs de traitement et des blocs de rétroaction. Par définition, nous avons $\mathcal{B} = \mathcal{S} \cup \mathcal{C} \cup \mathcal{T} \cup \mathcal{F}$. Nous définissons également l'ensemble $\mathcal{I}_t$ des informations manipulées à l'instant t par le système d'information. Par extension, l'ensemble $\mathcal{I} = \bigcup_{t \geq 0} \mathcal{I}_t$ est constitué de toutes les informations manipulées par le système depuis l'instant $t = 0$. Nous modélisons un bloc fonctionnel sous la forme d'une fonction qui associe une unique information à l'ensemble des informations qu'il reçoit en entrée. En toute généralité, pour un bloc à n entrées ($n \in \mathbb{N}$) nous avons donc :

$$\forall t \quad f_B : \mathcal{I}_{t-1}^n \to \mathcal{I}_t$$

où f_B est la fonction associée au bloc B qui produit une information à l'instant t à partir des informations reçues à l'instant $t - 1$.

Le premier type de bloc élémentaire que nous considérons est la **source**. Une source est un bloc qui génère de l'information. Nous modélisons une source $S \in \mathcal{S}$ comme une fonction $f_S : \emptyset \to \mathcal{I}_t$ pour tout t positif ou nul.

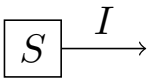


FIGURE 3.2 – Une source S distribue une information I

Une entité est complexe car elle peut émettre un nombre important d'informations à un même instant t. De plus, le nombre d'informations transmises n'est pas nécessairement stable car certaines informations sont envoyées occasionnellement (*e.g.* les alertes SAR [1] ou CPA [2] de l'AIS [3]). Nous voyons donc l'avantage de

1. Search And Rescue.

2. Closest Point of Approach, alerte avertissant la présence d'un obstacle sur la route suivie par le navire.

3. Automatic Identification System, système standardisé par l'Organisation Maritime Internationale pour la diffusion en temps réel des informations de navigation par VHF.

pouvoir modéliser une entité comme étant composée de sources, chacune produisant un unique type d'information avec ses propres caractéristiques. D'une part, ce modèle est simple : une source est spécialisée c'est-à-dire qu'elle n'envoie qu'un seul type d'information et qu'elle ne sert qu'une seule fonctionnalité. Un GPS qui envoie trois types d'informations (cap, position et vitesse) est donc une entité composée d'autant de sources que d'informations produites. D'autre part, ce modèle est flexible : il est aisé d'ajouter ou d'enlever une fonctionnalité à une entité en ajoutant ou supprimant la source correspondante. Cela se révèle intéressant en cas de défaillance par exemple d'un capteur du système de navigation. Un système de navigation fournit, entre autre, une information de position mesurée par de multiples capteurs : radar, un ou deux GPS, centrale inertielle. Du fait de la précision de la mesure, les informations fournies par le(s) GPS sont préférées à celles du radar ou de la centrale. Toutefois, en cas de défaillance du ou des GPS, le système fonctionne en *mode dégradé*, c'est-à-dire que la position qu'il transmet aux autres sous-systèmes du navire n'est plus mesurée par la source principale (ici le GPS). Notre choix de modélisation de chaque source en tant qu'émettrice d'un unique type d'information permet d'expliciter ce type de défaillance. Plus largement, cette modélisation permet de modéliser la redondance des informations au sein d'une entité.

Nous distinguons les sources des producteurs d'information. Comme nous venons de le présenter, une source est un bloc qui génère de l'information. Au contraire, un producteur désigne indifféremment tout bloc ou entité qui émet de l'information. Cette définition recouvre à la fois les sources, les blocs de traitement et toute entité émet de l'information.

Comme le montre la figure 3.3, un **collecteur** est un bloc qui reçoit des informations provenant de multiples producteurs. Par exemple, un système de visualisation est un collecteur. Il reçoit des données, les trie et les organise avant de les afficher, éventuellement sous forme de graphiques. Quelles que soient les données reçues, celles-ci ne sont cependant pas modifiées par le collecteur. Les caractéristiques intrinsèques des informations dépendent donc uniquement du bloc que l'a produite.

Contrairement à une source qui produit un seul type d'information (comme la température ou la position), un collecteur peut recevoir des données de natures différentes (*e.g.* images satellite, longitude et latitude pour visualiser la position du navire). Un bloc ne pouvant produire qu'un seul type d'information, un collecteur acquiert ses données d'entrée grâce à plusieurs producteurs différents. Nous

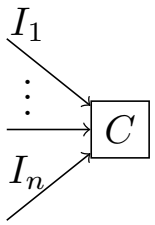


FIGURE 3.3 – Un collecteur C reçoit des informations $(I_i)_{i=1..n}$ provenant de une ou plusieurs sources

modélisons un collecteur $C \in \mathcal{C}$ par une fonction $f_C : \mathcal{I}_{t-1}^n \to \emptyset$ pour tout t positif.

Un bloc peut envoyer ou recevoir des informations mais il peut également les traiter. Un **bloc de traitement** reçoit des informations (comme un collecteur) puis les traite avant d'émettre une nouvelle information qui résulte du traitement des entrées (cf Figure 3.4). Nous modélisons donc un tel bloc par la fonction $f_T : \mathcal{I}_{t-1}^n \to \mathcal{I}_t$ pour des informations provenant de n blocs à tout instant t.

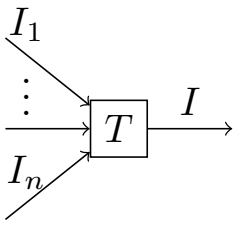


FIGURE 3.4 – Un bloc de traitement T reçoit des informations $(I_i)_{i=1..n}$ provenant de une ou plusieurs sources puis transmet le résultat de son traitement

Comme une source, un bloc de traitement alimente le système en information. Toutefois celui-ci n'effectue aucune mesure, c.-à-d. que les informations qu'il produit sont le résultat d'un traitement et non d'une observation de son environnement. Contrairement à une source, un bloc de traitement reçoit donc au moins une information provenant d'un autre bloc. Les blocs n'émettent qu'une seule information. En conséquence, ceux-ci transmettent leurs informations à de nombreuses entités en parallèle. Par exemple, sur la figure 3.5, les blocs $Pred$ et AVG sont distincts. Ces deux blocs reçoivent les mêmes informations de la part des mêmes blocs (deux GPS) mais produisent des informations différentes : vitesse et position respectivement. Le bloc AVG calcule la moyenne de ses entrées. Au contraire, le bloc $Pred$ produit une estimation de la vitesse à partir de la position courante et celle qui la précède. Dans notre modèle, ils sont donc considérés comme deux

blocs différents. À l'instar des collecteurs, les blocs de traitement reçoivent leurs informations de plusieurs blocs. Cette propriété est importante dans le cas où un bloc fusionne des informations pour améliorer leur qualité. Dans ce cas, il collecte des informations redondantes provenant de plusieurs producteurs différents.

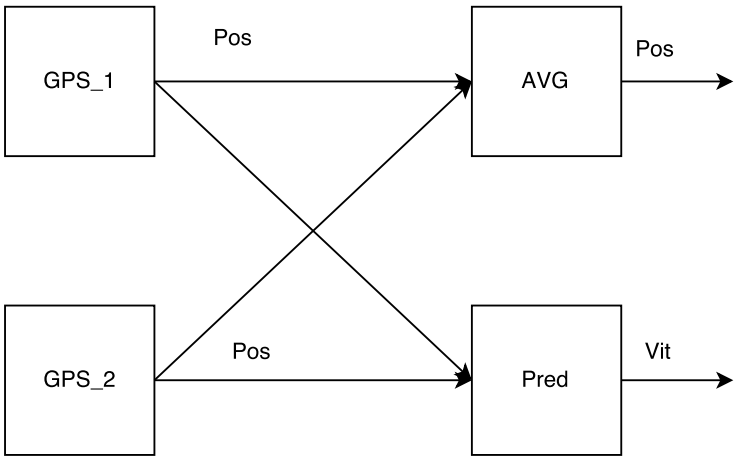


FIGURE 3.5 – Deux blocs ayant les mêmes entrées mais des sorties différentes

Enfin, le dernier type de bloc élémentaire est le **bloc de rétroaction** qui traite une donnée un certain nombre de fois (figure 3.6). Il est modélisé par une fonction f_F similaire à f_T mais avec la contrainte supplémentaire que ses informations produites à l'instant t (*i.e.* $f_F(\mathcal{I}^n_{t-1})$) sont réutilisées pour produire les informations de l'instant suivant $t+1$:

$$\exists i_0 \leq n \text{ tel que } f_F^{-1}(\mathcal{I}_{t+1}) = I_0 \times \cdots \times I_{i_0-1} \times f_F(\mathcal{I}^n_{t-1}) \times I_{i_0+1} \times \cdots \times I_n \subset \mathcal{I}^n_t$$

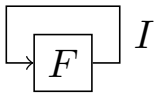


FIGURE 3.6 – Un bloc de rétroaction F traite une une même information plusieurs fois par itérations successives

Pour traiter une information plusieurs fois de suite, ce bloc a besoin de se souvenir du résultat de son calcul. Contrairement aux autres blocs qui produisent, traitent ou reçoivent de l'information à chaque instant, ce bloc mémorise et re-traite ses propres informations avant de les ré-injecter dans le système. Par exemple, une centrale inertielle calcule les différences entre une position de référence et ses propres estimations successives de la position courante.

Les **entités** d'un SI combinent les quatre types de blocs fonctionnels présentés. Par exemple, une base de données est à la fois un collecteur (elle reçoit des informations à stocker) et une source (elle envoie des données en réponse à des requêtes). Par ailleurs, il est important de noter qu'une entité est composée de bloc *fonctionnellement* distincts mais pas nécessairement *physiquement* distincts. En effet, un GPS est une entité qui émet des informations de différentes natures (position, vitesse, cap). Une telle entité est composée de trois sources (une pour chaque nature d'information) qui font parties intégrantes du même composant électronique. Au sens large, une entité est donc un ensemble organisé de blocs fonctionnels et, selon notre modèle, une entité E est un élément de l'ensemble $\mathcal{P}(\mathcal{B})$ des parties de $\mathcal{B}$. À partir de la typologie présentée, nous pouvons construire des entités complexes qui composent les systèmes d'information.

La figure 3.7 montre un exemple de système de navigation modélisé par assemblage de blocs fonctionnels. Pour simplifier nos propos, seules les informations de cap, de vitesse et de position sont représentées. Les informations se propagent dans le système depuis les sources (à gauche sur la figure) vers le système ECDIS qui agrège les informations pour que l'utilisateur du système visualise la situation du navire.

Ce système est constitué de divers blocs tels que les blocs AVG qui calculent la moyenne des informations qu'ils reçoivent en entrée ou le bloc $Pred$ qui estime la vitesse du navire à partir des positions passées. Enfin, le bloc CI est une centrale inertielle qui estime la position à partir de la position passée.

3.2.2 Relations entre les blocs

Un système d'information est composé de multiples blocs qui manipulent et s'échangent de l'information. Ces blocs sont reliés par de multiples liens qui influencent l'information qu'ils reçoivent, traitent ou émettent. Deux blocs peuvent être liés de multiples façons : dépendance d'un bloc à l'autre, distance les séparant, appartenance à une même entité voire émission d'une information de même nature. Tous ces liens, pris dans leur ensemble, sont caractéristiques du SI. Après avoir exposé les différents modèles de relations possibles, nous présentons ci-dessous les multiples relations que nous considérons dans notre modélisation d'un système d'information.

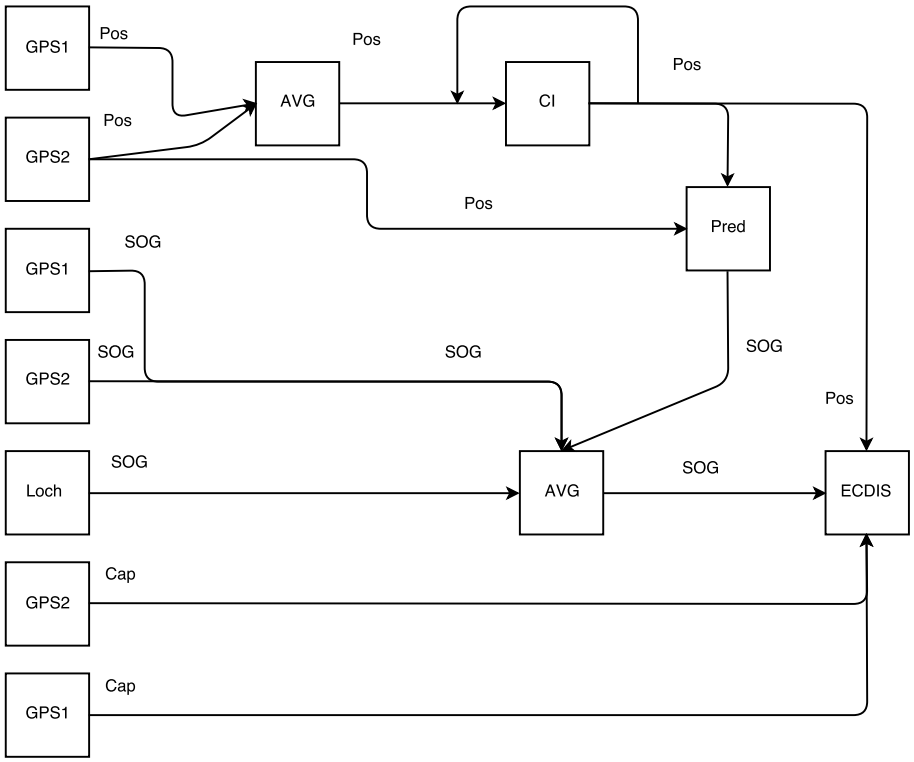


FIGURE 3.7 – Exemple de système modélisé par un ensemble de bloc

3.2.2.1 Modélisation des relations : généralités

Étudiées en sociologie, psychologie, philosophie, économie et en mathématiques, les relations permettent d'étudier le comportement d'un ensemble d'entités plutôt que de considérer celles-ci individuellement. En négligeant les caractéristiques propres à un système, La modélisation des relations sert à mettre en évidence les points communs ou les interactions entre les entités. En mathématiques comme en logique ou en informatique, une relation met en correspondance plusieurs ensembles, généralement au nombre de deux. Par souci de simplicité, tous les exemples détaillés dans cette section concernent les relations binaires. Toutefois, sauf indication contraire, les propos restent d'ordre général et concernent les relations n-aires (*i.e.* d'arité supérieure à 2). Une relation exprime une propriété partagée, ou non, par certains éléments des ensembles. Ainsi, une relation R entre les ensemble E et F est telle que $R \subset E \times F$. Les éléments de la relation sont les couples d'éléments mis en correspondance. Par exemple, l'ordre strict "$<$" met en

relation deux nombres tels que le premier est plus petit que le second. Dans ce cas, tous les nombres ne sont pas en relation : 1 est en relation avec 2 ($1 < 2$) mais pas avec 0 (1 n'est pas plus petit que 0). La relation est alors strictement plus petite que l'ensemble sur lequel elle est définie ($R = \{(x, y) \in \mathbb{N}^2 | x < y\} \neq \mathbb{N}^2$).

Lorsque deux ensembles sont mis en relation, leurs éléments partagent ou non une propriété commune (*i.e.* si R est une relation sur $E \times F$ alors pour tout $(x, y) \in E \times F$, $(x, y) \in R$ ou $(x, y) \notin R$). Toutefois, certaines propriétés peuvent être partagées jusqu'à un certain degré. Par exemple, deux personnes qui se ressemblent sans être le même individu, deux couleurs similaires sans être les mêmes (*e.g.* rouge et orange), deux pièces étant à une même température mais pas tout à fait (*e.g.* un frigo à 6 ° C et un congélateur à -12 ° C sont tous les deux froids mais pas autant), ... Pour tous ces exemples, il existe une gradation dans la relation : un enfant qui ressemble plus à sa mère qu'à son père, un orange foncé est plus proche du rouge que du jaune, la température du frigo est pus proche de celle du congélateur que de celle du four, etc. Cette gradation est représentée par un niveau d'appartenance à la relation. Soit $\mu_R : R \subset E \times F \rightarrow [0; 1]$ la fonction d'appartenance à la relation. $\mu_R(x, y)$ est le degré d'appartenance de (x, y) à la relation R (Ovchinnikov, 1991).

Les concepts évoqués ci-dessus concernent les relations n-aires. Dans le cas particulier des relations binaires (*i.e.* $n = 2$), celles-ci possèdent ou non un certain de nombre de propriétés qui les caractérisent.

Nous listons ci-dessous les différentes propriétés d'une relation $R \subset E \times F$:

- Homogénéité : R est dite homogène si $E = F$. Elle se note alors R_E, la relation R définie sur l'ensemble E. Dans le cas contraire (*i.e.* $E \neq F$) elle est dite hétérogène.
- Réflexivité : une relation homogène R_E est dite réflexive si tout élément de l'ensemble peut être mis en relation avec lui-même. ($\forall x \in E, (x, x) \in R_E$).
- Symétrie : une relation homogène R_E est dite symétrique si la propriété est partagée par les éléments, quelque soit leur ordre ($\forall x, y \in E, \quad (x, y) \in R_E \Leftrightarrow (y, x) \in R_E$).
- Antisymétrie : une relation homogène R_E est dite antisymétrique si elle n'est symétrique pour aucun élément ($\forall x, y \in E, \quad (x, y) \in R_E$ et $(y, x) \in R_E \Rightarrow \quad x = y$)
- Transitivité : une relation homogène R_E est transitive si la propriété est "transmissible" ($\forall x, y, z \in E, \quad (x, y) \in R_E$ et $(y, z) \in R_E \Rightarrow (x, z) \in$

4. Source inconnue, Wikipedia

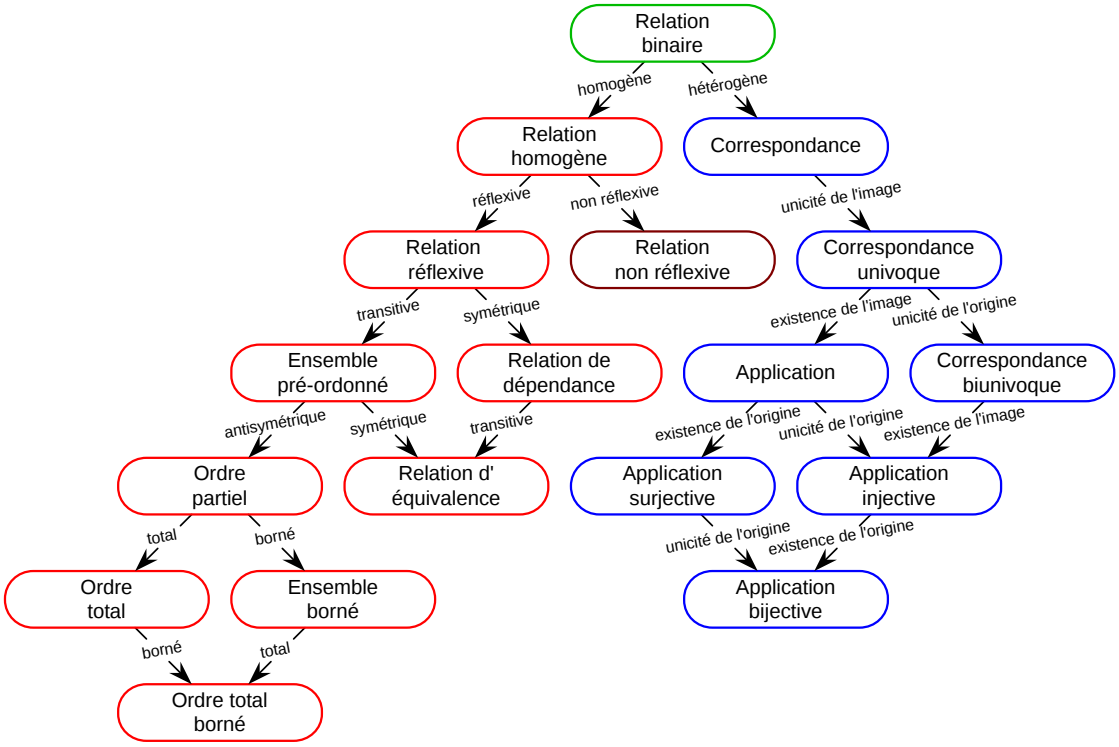


FIGURE 3.8 – Relations binaires [4]

R_E).

- Totalité : une relation homogène est totale si tous les éléments de l'ensemble de définition peuvent être mis en relation ($\forall x, y \in (x, y) \in R_E$ ou $(y, x) \in R_E$).
- Bornée : une relation partiellement ordonnée R_E est bornée sur E si E admet une borne supérieure [5] et une borne inférieure [6].
- Unicité de l'image : pour tout élément x de E, il existe au plus un élément de F avec lequel x peut être en relation ($\forall x \in E$ s'il existe $y \in F$ tel que $(x, y) \in R$ alors y est unique)
- Existence de l'image : tout élément de E est en relation avec au moins un élément de F ($\forall x \in E \quad \exists y \in F$ tel que $(x, y) \in R$).
- Unicité de l'origine : pour tout élément y de F, il existe au plus un élément de E qui est en relation avec y ($\forall y \in F$ s'il existe $x \in E$ tel que $(x, y) \in R$ alors x est unique).
- Existence de l'origine : tout élément de F est en relation avec au moins

5. La borne supérieure d'un ensemble est son plus petit majorant
6. La borne inférieure d'un ensemble est son plus grand minorant

un élément de E ($\forall y \in F \quad \exists x \in E$ tel que $(x, y) \in R$).

La figure 3.8 résume les différentes propriétés des relations binaires. Selon ses propriétés, une relation définit une structure particulière sur le ou les ensembles qu'elle relie. La structure est induite par les propriétés se trouvant sur les arêtes y menant. En particulier, les relations d'équivalence (*i.e.* réflexive, symétrique et transitive) et d'ordre (*i.e.* réflexive, antisymétrique et transitive) partiel ou total sont très utilisées en mathématiques.

Une relation binaire homogène peut se représenter sous forme de graphe. En fait, le graphe définit entièrement la relation. Les propriétés de la relation correspondent à des propriétés du graphe et inversement. Par exemple, le graphe (E, R_E) est le graphe dont les nœuds sont les éléments de l'ensemble E et ses arêtes sont les éléments de la relation binaire R_E définie sur E. Sur un tel graphe, la propriété de symétrie signifie que le graphe est non orienté.

3.2.2.2 Relations entre les blocs fonctionnels

Les relations, en particulier binaires, constituent un outil important pour modéliser les interactions au sein d'un système. En effet, un système d'information est composé de multiples relations entre ses constituants, lesquelles affectent la structure et le comportent global du SI. La Section 3.2.1 a exposé la modélisation des composants d'un SI en tant que blocs fonctionnels. Nous souhaitons maintenant présenter la modélisation des relations entre ces différents types de blocs. Nous nous intéressons aux relations qui impactent le fonctionnement des blocs et par conséquent les informations qu'ils produisent ou manipulent. Dans ce contexte, sans prétendre à l'exhaustivité, nous proposons une modélisation de quatre relations : relation de dépendance, relation topologique, relation de nature et relation d'entité. En accord avec ce que nous avons exposé dans la section précédente, nous définissons nos relations comme des sous-ensembles de $\mathcal{B} \times \mathcal{B}$. Chacune des relations considérées est donc homogène.

Lorsqu'un bloc B_1 envoie de l'information à un bloc B_2, ce dernier est dépendant du premier. B_1 et B_2 entretiennent une **relation de dépendance** dans le sens où l'information produite par B_1 (et donc reçue par B_2) impacte B_2. En effet, toute erreur ou défaillance de B_1 entrainant une altération ou une perte d'information se traduit par la réception par B_2 d'une information altérée ou non-reçue dans le cas d'une perte. Nous supposons que les mécanismes de contrôle d'intégrité sont suffisants pour protéger l'information de toute altération lors de son achemi-

nement (de B_1 vers B_2). Par exemple, dans un système de navigation, un GPS transmet des positions au système de cartographie *Electronic Chart and Display* (ECDIS) lequel permet à l'utilisateur de visualiser diverses informations parmi lesquelles ses positions successives. Si le GPS, pour quelque raison que ce soit, tombe en panne et n'émet plus d'information alors le système ECDIS ne peut plus afficher les positions (lui-même ne les mesurant pas). Ainsi, le bon fonctionnement de l'ECDIS est dépendant de celui du GPS. Dans le cas de plusieurs blocs envoyant leurs informations à une même destination, chacun n'a pas le même impact sur cette dernière : leur criticité pour le bloc destinataire varie selon l'importance que leurs informations représentent pour lui.
La relation de dépendance possède plusieurs propriétés : elle est homogène et plus particulièrement transitive sur l'ensemble des blocs. Par définition, la relation de dépendance est orientée et ne concerne donc pas tous les blocs. En effet, si les informations d'un bloc B_i sont nécessaires à un bloc B_j alors le bloc B_i dispose d'une capacité d'émission d'information. C'est-à-dire que B_i est soit une source soit un bloc de traitement. De même, le bloc B_j est capable de recevoir de l'information. B_j est donc soit un collecteur soit un bloc de traitement. Dans le cas où $i = j$, B_i est un bloc de rétro-action qui s'alimente lui-même en informations.

Les informations manipulées par les blocs sont également affectées par les **relations topologiques** entre ces derniers. En particulier, deux sources proches seront plus susceptibles de mesurer une information similaire : elles observent le même phénomène physique. Par exemple, deux anémomètres seront soumis à un même vent s'ils sont proches. Cela est d'autant moins probable s'ils sont éloignés. Dans ce cas, leurs environnements sont alors susceptibles d'être très différents. Lorsque deux sources sont proches, l'information qu'elles manipulent doit donc être similaire. Toutefois, cette proximité est dépendante de la notion de distance qui s'y rattache. En effet, deux thermomètres peuvent être proches mais dans deux pièces séparées : un dans la cuisine et l'autre dans la chambre froide. La notion de distance n'est donc pas nécessairement liée à la distance euclidienne. Elle peut également être liée à la topologie de l'environnement, à sa structure. Cette topologie se modélise sous forme de graphe dont les nœuds sont les pièces et les arêtes les passages entre ces pièces. Pour reprendre l'exemple exposé ci-dessus, la chambre froide et la cuisine sont deux nœuds reliés par une arête. Un nœud correspond donc dans ce cas à un phénomène observé (ici les températures respectives de la chambre froide et de la cuisine). Deux sources liées au même nœud observent donc le même phénomène et les informations qu'elles produisent sont donc similaires. Cette relation ne concerne que les sources. En effet, aucun autre

bloc ne se comporte différemment selon sa localisation. Ceci est dû au fait que les sources sont les seuls blocs à observer directement des phénomènes physiques. La relation topologique est une relation d'équivalence. En effet, toute source observe le même phénomène qu'elle-même (réflexivité), la symétrie découle directement de la définition et la transitivité provient quant à elle de l'unicité du phénomène observé. En effet, deux sources S_i et S_j observant un phénomène P et S_j et S_k observant un phénomène Q, alors $P = Q$ car S_j ne peut émettre qu'une seule information à chaque top d'horloge.

Outre les relations topologique et de dépendance, la **relation d'entité** est également à prendre en compte pour évaluer la confiance dans le SI. Une entité est composée d'un ensemble de blocs. Chacun de ces blocs possède ses propres caractéristiques et ses propres vulnérabilités. Cependant, la compromission de l'un de ces blocs impacte le fonctionnement de l'ensemble. Par exemple, nous modélisons un GPS par une entité composée de 3 sources qui émettent respectivement des informations de position, de vitesse et de cap. Dans le cas où l'on observe la compromission de l'une de ces sources, cela peut provenir d'une attaque ciblant cette dernière ou bien d'une attaque de plus grande ampleur ciblant le GPS lui-même. Par conséquent, la confiance de chacune des sources composant le GPS, et plus généralement des blocs composant une entité quelconque, impacte directement la confiance accordée à l'entité à laquelle elles appartiennent. Tout comme la relation topologique, la relation d'entité est réflexive et symétrique. Toutefois, elle n'est pas transitive, un même bloc pouvant appartenir simultanément à plusieurs entités distinctes. Soient B_i appartenant à une entité E_l et B_k appartenant à une entité E_m telle que $E_m \neq E_l$. Soit un bloc B_j appartenant à la fois à E_l et à E_m, alors B_j et B_i appartiennent à la même entité de même que B_j et B_k. Cependant, par hypothèse, B_i et B_k n'appartiennent pas à la même entité.

Considérons enfin la **relation de nature**. Certains blocs émettent des informations qui peuvent être de nature différente d'un bloc à l'autre. Pour reprendre l'exemple des sources du GPS, celles-ci émettent respectivement des informations de position, de vitesse et de cap. Le GPS est donc une entité composée de 3 blocs de natures différentes. Pour des soucis de fiabilité, les informations sont généralement redondées, c'est-à-dire produites simultanément par plusieurs sources. La redondance des informations permet de les comparer pour déceler des différences dues par exemple à des erreurs de mesure. Pour comparer les informations produites par les blocs, il est alors nécessaire d'en connaître la nature. En effet, deux informations de nature différente n'ont pas la même sémantique et ne sont donc

pas comparables. Cette relation concerne tous les blocs émettant de l'information : sources, blocs de traitement et blocs de rétro-action. Elle possède les propriétés d'une relation d'équivalence. En effet, tout bloc est de même nature que lui-même (réflexivité), la propriété de symétrie est naturelle et un bloc B_i de même nature qu'un bloc B_j, lui-même de nature identique à B_k implique que B_i et B_k partagent également cette nature. Dans le cas contraire, B_j devrait être de deux natures différentes, ce qui est écarté par la définition des blocs fonctionnels : la nature d'un bloc est aussi unique celle des informations qu'il émet.

Les différentes relations, leurs propriétés et les types de blocs qu'elles affectent sont résumés dans le tableau 3.2.

Chacune des relations peut être valuée, c.-à-d. être affectée d'une valeur qui peut ne pas être exclusivement numérique, qui exprime la raison de l'existence de la relation (relation de nature, topologique et d'entité) ou qui nuance le lien qui existe entre les blocs (relation de dépendance).

Type	Description	Origine	Destination	Propriétés	Valuation
Dépendance	Les informations envoyées par un bloc B_i sont nécessaires à un bloc B_j	source, bloc de traitement, bloc de rétroaction	collecteur, bloc de traitement, bloc de rétroaction	homogène, transitive	criticité
Topologique	Deux blocs observent le même phénomène	source	source	homogène, réflexive, symétrique, transitive	distance
Nature	B_i et B_j sont des blocs envoyant des informations de mêmes natures	source, bloc de traitement, bloc de rétroaction	source, bloc de traitement, bloc de rétroaction	homogène, réflexive, symétrique, transitive	nature de l'information produite (position, vitesse, etc.)
Entité	B_i et B_j font partie d'une même entité E_k	tous	tous	homogène, réflexive, symétrique	E_k

TABLE 3.2 – Tableau récapitulatif des relations

3.3 Modélisation de sources d'informations

Pour construire une mesure de confiance, nous avons besoin de modéliser l'information manipulée par le système. Une source d'information est un bloc qui observe un phénomène et le restitue au système sous différentes formes (valeur numérique, texte, image, vidéo, etc.). Les concepts développés ici sont d'ordre général : ils sont facilement transposables à tout type d'information. Toutefois, l'approche présentée est centrée sur les valeurs numériques, lesquelles sont des mesures fournies par des capteurs embarqués. Dans le cas où la source est un capteur, elle mesure une grandeur physique (vitesse, température, etc.).

Cette mesure est imparfaite et entachée d'erreurs. En effet, la mesure est dépendante des caractéristiques du capteur (précision, sensibilité, usure, etc.). Deux sources mesurant le même phénomène et ayant les mêmes caractéristiques ne rendront pas forcément compte de la réalité de la même manière en raison d'un bruit dans la mesure. Cependant, ces mesures ne seront pas très éloignées et en tous cas seront proches de la réalité à moins de la défaillance du capteur ou d'une attaque. Suivant la complexité des capteurs, des phénomènes physiques observés et des composants électroniques utilisés, il est plus ou moins difficile de quantifier cette erreur dans la mesure. Néanmoins, une solution simple consiste à résumer l'ensemble des bruits de la chaîne d'acquisition à un bruit blanc additif ; c'est-dire également distribué sur toutes les fréquences du signal. Il s'agit d'un modèle de bruit très largement utilisé en traitement du signal (Papoulis et Pillai, 1986).

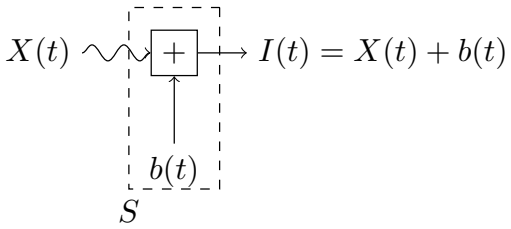


Figure 3.9 – Modélisation d'une mesure numérique via un canal gaussien

Comme illustré en figure 3.9, une source S observe le phénomène réel $X(t)$, une fonction dépendante du temps (*e.g.* température, vitesse, position), transmet sa mesure $I(t) = X(t) + b(t)$ où b est une variable aléatoire de loi normale $\mathcal{N}(\mu(t), \sigma(t))$ de moyenne $\mu(t)$ et d'écart-type $\sigma(t)$, également dépendants du temps.

Les moments statistiques du bruit b (μ et σ) peuvent varier. En effet, les

erreurs de mesures sont sujettes à l'usure des sources. Celles-ci ont une durée de vie limitée pendant laquelle elles se détériorent au fil de leurs sollicitations. Cette usure impacte négativement la précision de la mesure. Une source est donc de moins en moins précise au fur et à mesure de son fonctionnement.

Toutefois, pour certaines applications (*e.g.* industrielles), les sources ont une durée de vie très longue. Elles sont alors très fiables, c'est-à-dire résistantes aux pannes ou à l'usure. C'est par exemple le cas dans les systèmes industriels composés de multiples capteurs, automates et actionneurs soumis à des conditions parfois extrêmes. Afin de prévenir toute atteinte à la sûreté de fonctionnement de l'installation, ceux-ci ont donc une durée de vie pouvant atteindre plusieurs dizaines d'années (ANSSI, 2015). Dans ces cas-là, nous pouvons donc supposer que les moments statistiques de b sont stables, c'est-à-dire μ et σ sont constantes.

Les erreurs peuvent provenir d'un bruit dans la mesure mais également d'un biais dû à un mauvais calibrage ou à l'environnement (par ex. des vibrations). Cette erreur, qualifiée de systématique, se modélise par l'ajout d'un biais déterministe dans la mesure (c.-à-d. $\mu \neq 0$). Ce type d'erreur est cependant identifiable lors de tests préalables du capteur. Sans perte de généralité, nous supposerons donc que b est centrée (*i.e.* $\mu = 0$).

Par la suite, une source sera dite *idéale* ou *parfaite* si celle-ci renvoie telle quelle l'information observée, c'est-à-dire $X(t) = I(t)$ pour tout t (*i.e.* $\mu = 0$, $\sigma = 0$).

La figure 3.10 montre un exemple de mesures de vitesses simulées. Les courbes représentent l'évolution de la vitesse d'un navire au fil des informations transmises (*i.e.* les mesures). Sur ce graphique, une centaine de mesures ont été effectuées. Elles sont issues d'informations de vitesse fournies par le système AIS qui ont ensuite été bruitées grâce au modèle exposé dans cette section. L'AIS transmet des données de positionnement à des intervalles de temps dépendant du contexte de navigation. Parmi les informations transmises se trouvent la position, le cap, le temps, la vitesse et un identifiant unique du navire.
La mesure idéale est en trait plein tandis que les mesures bruitées sont en pointillés. Elles correspondent à des bruits de variances respectivement égales à 0.5 et 0.1. En particulier, une variance égale à 0.1 correspond à la précision d'un *loch doppler* qui est un capteur servant à mesurer la vitesse d'un navire par rapport au fond à l'aide d'ultrasons. Plus la valeur de σ devient grande et plus les mesures ont des chances de s'écarter de la valeur d'origine. Par ailleurs, les écarts sont aléatoires : les erreurs de mesures sont différentes même pour deux capteurs de même précision. Ceci est

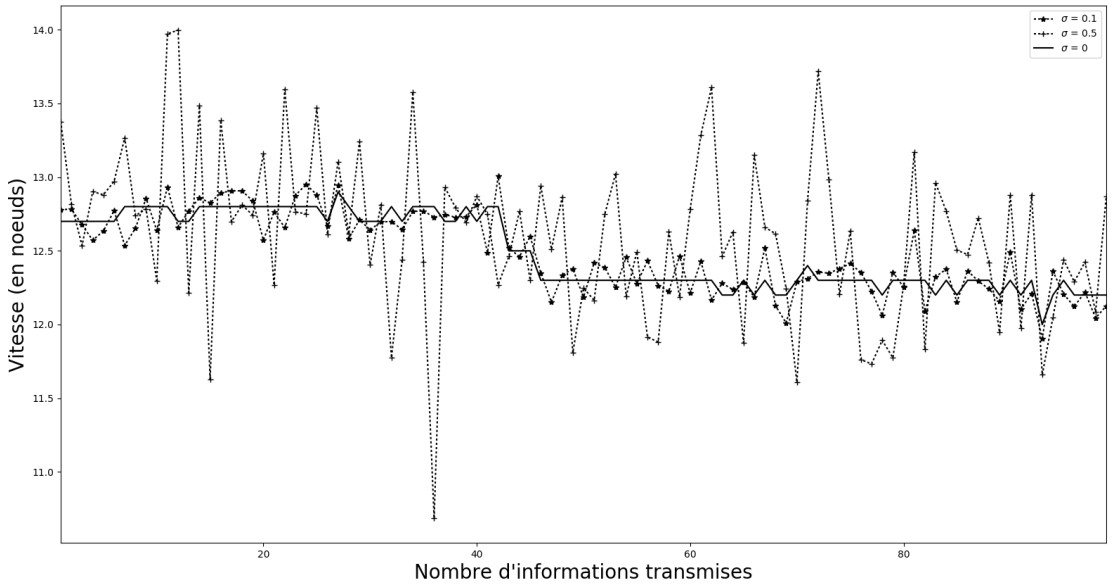


FIGURE 3.10 – Exemple de simulation de mesures de vitesse

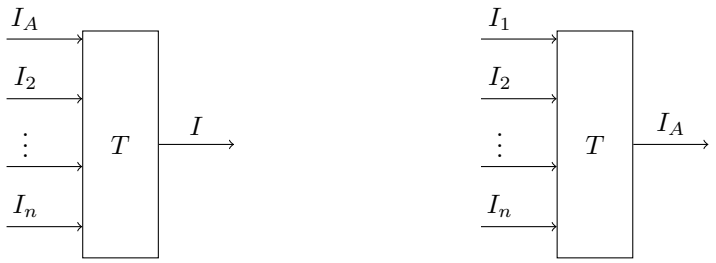
illustré par l'erreur à la 32e information transmise : les deux courbes en pointillé se retrouvent de part et d'autre de la courbe de référence. Les erreurs commises ne sont donc pas identiques : une mesure est supérieure à la mesure d'origine tandis que la seconde lui est inférieure.

Dans cette section, nous avons modélisé l'information renvoyée par des sources de type capteur, lesquels mesurent une grandeur physique. Cette modélisation spécifique à l'information numérique reflète le fonctionnement des capteurs embarqués dans les systèmes d'information. Elle est de plus nécessaire à la modélisation des attaques, lesquelles visent à corrompre le système en falsifiant l'information.

3.4 Modélisation d'attaques

Divers moyens peuvent être mis en place pour protéger le système d'information d'un navire dans sa globalité ou au niveau de chacun de ses constituants : authentification, contrôle d'accès, confidentialité des échanges ... Ces solutions ont néanmoins leurs limites dans le cas où l'un des éléments est leurré ou malveillant. En effet, lorsqu'un bloc est leurré, celui-ci reçoit en entrée une information falsifiée. Un attaquant souhaitant leurrer un bloc lui envoie des informations modifiées, c'est-à-dire qui ne reflètent pas la réalité. Par exemple, un fichier PDF malicieux,

un document office piégé, une injection de code, des données capteurs modifiées etc. Au contraire lorsqu'un élément du système est malveillant, c'est-à-dire directement contrôlé par l'attaquant, ce sont les informations qu'il émet qui sont falsifiées (par lui-même). Dans ce cas, c'est donc sa sortie qui est modifiée. Un leurre affecte également la sortie d'un bloc car celle-ci dépend des entrées du bloc. Cependant, un attaquant ne contrôle pas précisément la sortie d'un bloc leurré. Ces deux types d'attaques, leurre et malveillance d'un bloc, sont représentés sur la figure 3.11.



(a) Bloc leurré recevant une information I_A falsifiée

(b) Bloc malveillant envoyant une information I_A falsifiée

FIGURE 3.11 – Types d'attaques pouvant cibler un bloc

Dans le contexte d'un système de navigation, une attaque consiste à modifier une information de façon à ce que celle-ci ne représente plus la réalité. Cette falsification vise à perturber le fonctionnement normal du système, par exemple en influençant les décisions du pilotage automatique via les informations fournies par les capteurs. Nous modélisons dans ce travail trois attaques reposant sur la modification des informations manipulées par un SI. Ces attaques ont des paramètres communs mais également des paramètres qui leur sont spécifiques. Par exemple, toute attaque doit avoir une cible et agit pendant une certaine durée mais chacune dépend également de paramètres propres : puissance, incrément, séquence à enregistrer.

D'après le modèle présenté en section 3.3, les informations forment une suite numérique dépendante du temps. À un instant t, de multiples informations sont produites ou reçues par les blocs fonctionnels. Une attaque qui altère des informations dépend donc du bloc qui les a produites (la *cible*), du temps (*début* et *durée de l'attaque*) mais également de la transformation appliquée à l'information falsifiée. Nous présentons ci-après trois transformations distinctes.

Attaque par offset La première transformation est un *offset*. L'information $I_A(t)$ falsifiée par un attaquant A est l'information $I_B(t)$ produite par le bloc B à l'instant t augmentée d'une constante a qui est la puissance de l'attaque : $\exists t \quad I_A(t) = I_B(t) + a$ où a est une constante. L'information $I_A(t)$ est ensuite diffusée dans le système au lieu de $I_B(t)$.
La figure 3.12 montre un exemple d'attaque par offset sur une mesure de vitesse. La mesure réelle est en trait plein et la mesure falsifiée est en pointillés. L'attaque a été menée sur les informations 30 à 40 qui indiquent une vitesse plus élevée d'un nœud que la vitesse réelle. Cette attaque peut notamment servir à ralentir le navire pour l'intercepter (*e.g.* par des pirates). En effet, si la vitesse paraît plus élevée qu'elle ne l'est en réalité, les décisions basées sur cette information vont chercher à contrebalancer en ralentissant afin d'atteindre une vitesse limite. Par exemple, si le navire souhaite naviguer à une vitesse de 10 nœuds, l'augmentation artificielle de la vitesse par l'attaquant poussera le navire à réduire son allure à 9 nœuds. Par symétrie, cette attaque peut aussi pousser le navire à accélérer en réduisant sa vitesse. L'utilisation de ce type d'attaque contre des navires a déjà été utilisée (Bhatti et Humphreys, 2014).

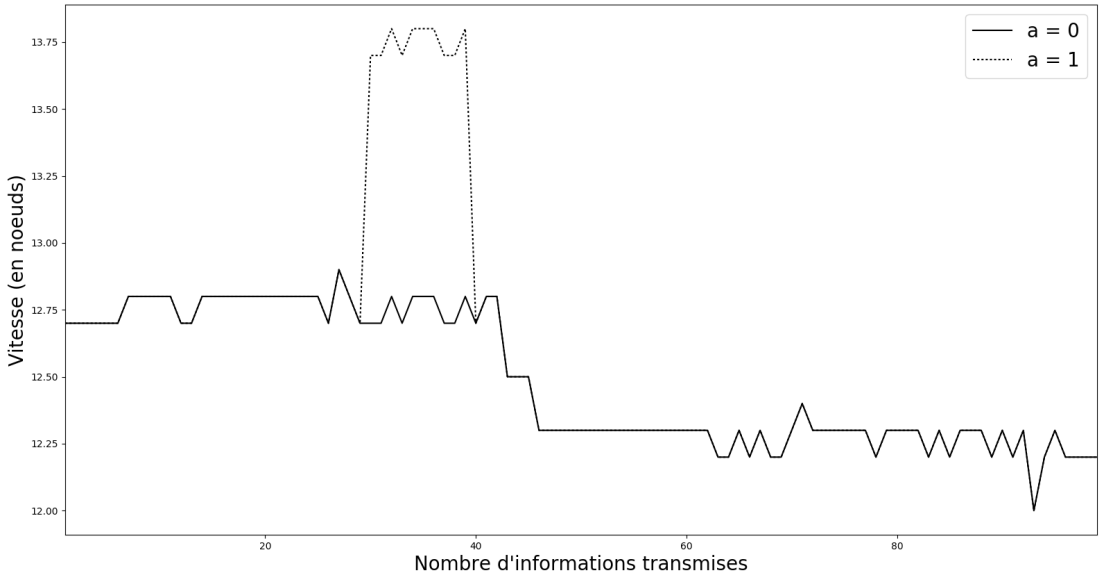


Figure 3.12 – Exemple d'une attaque par offset sur 10 informations

Attaque incrémentale La seconde transformation est un *offset incrémental*, c'est-à-dire que la puissance de l'attaque $a(t)$ augmente au cours du temps. Par

exemple, celle-ci peut être une répétition d'offset modélisée sous la forme d'une suite récurrente : $\forall t > 0 \quad a(t) = a(t-1) + a(0) = (t+1) * a(0)$. L'information $I_A(t)$ falsifiée par un attaquant A est l'information $I_B(t)$ produite par le bloc B à l'instant t augmentée d'une constante a qui est la puissance de l'attaque : $\forall t \quad I_A(t) = I_B(t) + a(t)$. L'information $I_A(t)$ est ensuite diffusée dans le système au lieu de $I_B(t)$.
La figure 3.13 présente un exemple d'attaque incrémentale. A l'instar de l'exemple d'attaque par offset, la coure en trait plein est la vitesse réelle tandis que la courbe en pointillés est celle qui est altérée. L'attaque présentée altère progressivement les informations de vitesse. Elle commence à partir de la 30e information et continue jusqu'à la 40e laquelle indique une vitesse d'un nœud supérieur à la vitesse réelle. L'intérêt de cette attaque réside dans son évolution progressive qui la rend plus difficile à déceler (Bhatti et Humphreys, 2017).

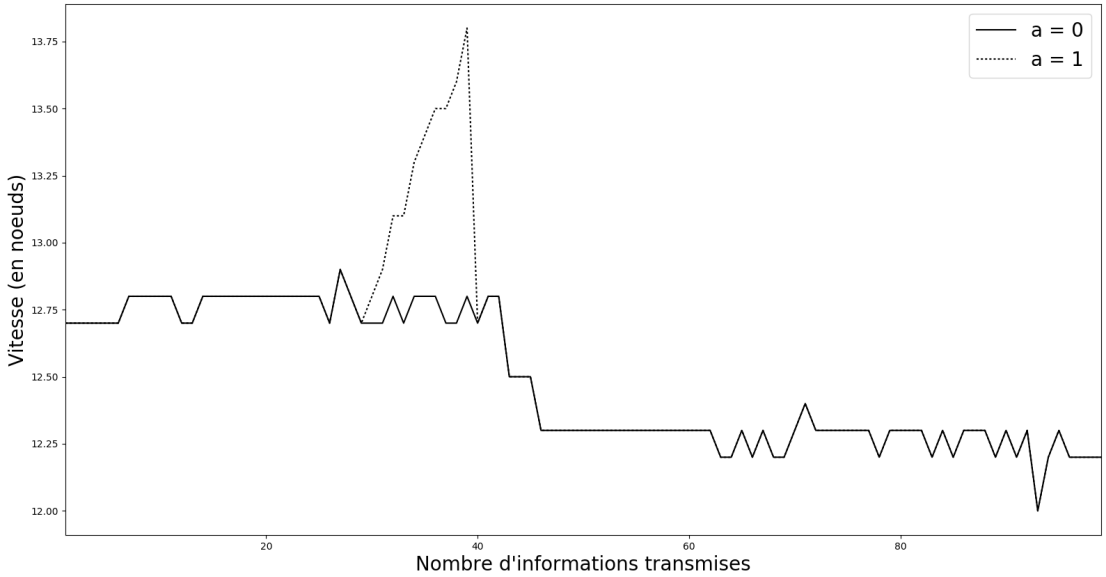


Figure 3.13 – Exemple d'une attaque incrémentale sur 10 informations

Attaque par rejeu La dernière attaque est une *attaque par rejeu* : elle consiste à remplacer les informations courantes par des informations produites précédemment par le système ou à présenter à nouveau au système ces anciennes informations, celles-ci étant valides. Cette attaque vise en général à contourner les mesures de sécurité mises en place, notamment de chiffrement, sans lever la confidentialité des échanges, c'est-à-dire sans connaître les informations secrètes du système (*e.g.* les

clés permettant de déchiffrer les messages). Par exemple, en rejouant des identifiants ou des informations certifiées fiables. Malgré leur certification, elles peuvent cependant ne plus être valides. De fait, leur rejeu représente un danger pour la sécurité du SI. Par exemple, le rejeu d'une vieille carte maritime électronique lors d'une mise à jour de l'ECDIS peut conduire à des incidents maritimes graves si des zones de danger ne sont pas signalées sur les anciennes cartes, considérées à jour par le système ciblé.
La figure 3.14 montre un exemple de rejeu d'informations. Les informations 10 à 20 sont rejouées et remplacent les informations 50 à 60. Les impacts de cette attaque sur la vitesse sont similaires à celles des attaques précédentes. Toutefois, dans ce cas, les données n'ont pas besoin d'être connues de l'attaquant, celui-ci n'ayant qu'à observer et copier les échanges d'informations.

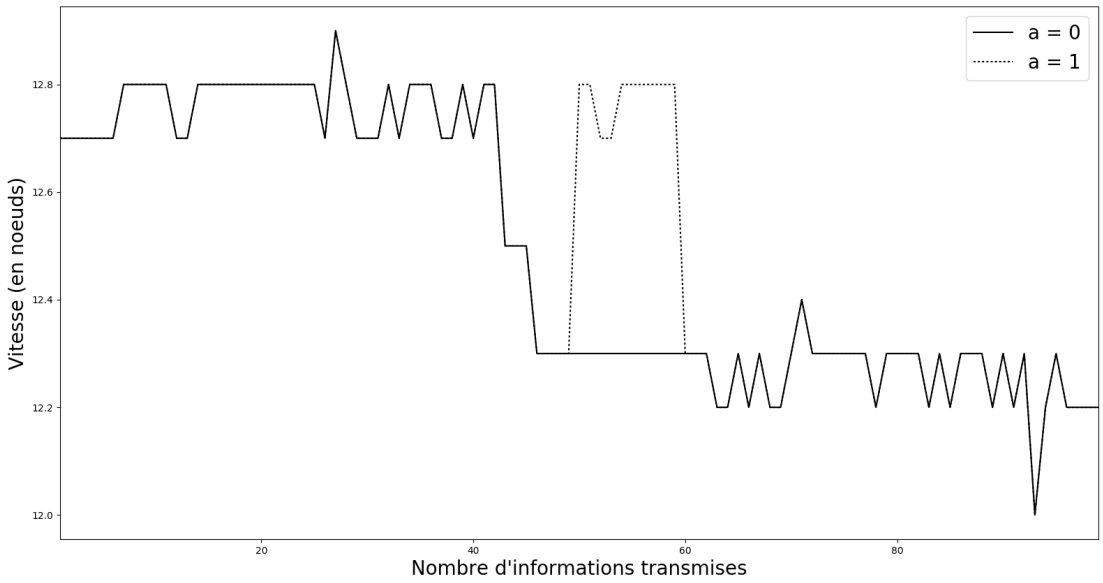


FIGURE 3.14 – Exemple de rejeu de 10 informations

Les attaques présentées dans cette section montrent diverses manipulations possibles des informations remontées par des sources de type capteur. Ces sources sont notamment présentes dans les systèmes de navigation des bateaux. Bien que simples, ces attaques visent à perturber le comportement du navire en orientant ses actions via les informations sur lesquelles sont basées les décisions prises, par exemple, par le chef de bord ou le pilote automatique. Elle ont de plus été montrées efficaces (BALDUZZI et al., 2014 ; BHATTI et HUMPHREYS, 2017).

Conclusion

Ce chapitre a présenté notre modélisation d'un système d'information ainsi que des attaques pouvant le cibler. Notre modèle de SI s'appuie sur une **approche systémique** centrée sur les fonctions et objectifs du système. Cette approche permet de focaliser notre étude sur les interactions entre les constituants du système d'information plutôt que sur les constituants eux-mêmes. Ces constituants sont vus comme des entités de complexité variable composées de blocs fonctionnels. Nous avons établis une typologie des blocs fonctionnels selon la façon dont ils interagissent avec l'information. Nous avons également identifié quatre relations entre les blocs qui nous semblent importantes pour l'évaluation d'indices de confiance dans le système.
Nous avons retenu un **modèle gaussien** de l'information, très largement utilisé en traitement du signal. En effet, l'information est aux cœur des échanges internes et externes du système. Elle peut être de diverses natures : nombre, texte, image, vidéo ... Toutefois, dans notre contexte de système de navigation, l'information numérique remontée par les capteurs est primordiale pour la bonne marche des navires.
Par ailleurs, l'information numérique, actif principal de tels systèmes, est également un vecteur d'attaque privilégié. Nous avons donc modélisé des attaques qui visent à compromettre l'intégrité du système en falsifiant les informations que celui-ci manipule, soit en altérant la mesure effectuée par un capteur (**offset** ou **attaque incrémentale**) ou en **rejouant** des informations précédemment émises par le système. À partir de l'ensemble des modèles présentés dans ce chapitre (i.e. du système, de l'information et des attaques) et des définitions de la confiance présentées dans le chapitre précédent, nous allons construire des mesures de confiance pour assurer la sécurité des systèmes d'information.

Chapitre 4

Modéliser et mesurer la confiance au sein d'un système d'information

Nous souhaitons utiliser la confiance pour assurer la sécurité des systèmes d'information à travers l'élaboration de bases d'un système de détection. Ainsi, nous proposons un système capable de détecter des attaques ciblant le SI à partir de la variabilité de la confiance dans les blocs fonctionnels qui le composent. Nous avons vu dans le chapitre 2 que la confiance est une notion complexe reposant, selon les domaines d'étude, sur de nombreux critères parmi lesquels la compétence, la sincérité, le risque et la connaissance. Dans le présent chapitre, nous choisissons d'exprimer la confiance à partir de la compétence et la sincérité. Sur la base du modèle de SI proposé au chapitre précédent, nous présentons notre modélisation de la confiance et sa mise en œuvre, sous forme de mesures, dans un système d'information.

La section 4.1 pose les caractéristiques sur lesquelles repose la confiance : la compétence et la sincérité. Nous discutons ensuite les notions de risque et de connaissance dans notre contexte d'étude qu'est la sécurisation des systèmes embarqués à bord des navires. La section 4.2 aborde l'élaboration d'une mesure de confiance dans les sources d'information, lesquelles sont les points d'entrée du système. Enfin, la section 4.3 aborde les moyens de propagation de la confiance depuis les sources jusqu'aux autres blocs fonctionnels.

4.1 Modélisation de la confiance

Dans le chapitre 2, nous avons vu que la confiance peut être évaluée sur la base de différents critères. Ceux-ci peuvent être internes ou externes à la relation de confiance.

Lorsqu'une entité fait confiance, dans une certaine mesure, à une autre, toutes deux forment une relation de confiance. Les critères servant de base à l'évaluation de la confiance peuvent se différencier selon qu'ils sont externes ou internes à la relation et, dans ce dernier cas, qu'ils sont caractéristiques de l'origine ou de la cible de la relation. Sur cette base, nous proposons d'étudier quatre critères : la compétence (caractéristique de la cible, interne à la relation), la sincérité (idem), la connaissance (caractéristique de l'origine, interne à la relation) et le risque (externe à la relation).

4.1.1 Compétence

L'essor rapide du commerce électronique et des interactions interpersonnelles via les nouveaux moyens de communication (mails, messageries instantanées, réseaux sociaux, etc.) ont posé de nouveaux défis quant à la confiance que l'on peut avoir dans ces services. Dans ce cadre, la compétence fut proposée pour gérer l'ensemble de ces nouveaux acteurs autant que les humains traditionnellement au centre des modèles. Dans le contexte des multiples services disponibles sur Internet, Grandison et Sloman, 2000 définissent la compétence comme la « capacité d'une entité à remplir les fonctions qui lui sont attribuées » (Grandison et Sloman, 2000). Ce contexte est proche du nôtre où les entités considérées sont des blocs accomplissant une unique fonction : fournir une position ou une vitesse. Par conséquent, leur compétence n'est appréciée que dans un seul domaine, lequel est lié à la nature de l'information produite.

Plus généralement, la compétence est spécifique à un domaine ou une activité particulière. Un mécanicien est compétent pour réparer des voitures mais pas nécessairement pour suggérer un restaurant ou pratiquer une opération de chirurgie. Inversement, un chirurgien est considéré compétent dans ce dernier domaine mais sans garantie dans les deux autres. La compétence du chirurgien n'est donc pas la même que celle du mécanicien, leurs domaines d'application étant distincts. Dans ce qui suit, nous ne distinguerons pas le domaine d'application de la compétence. Nous retenons la définition de Grandison et Sloman, 2000.

Telle que définie, la compétence rend compte de la capacité d'une entité à faire des erreurs. Une erreur est un manquement à la fonction attribuée. S'agissant d'une source d'information, la fonction attendue est l'émission d'informations qui sont conformes à la réalité. Il en va de même de tout autre bloc fonctionnel du SI producteur d'information. Dans ce cas, une erreur se caractérise par l'écart entre l'information et la réalité qu'elle décrit. Pour un capteur, l'écart entre la valeur réelle et la valeur mesurée est dépendante de la précision du capteur. Par exemple, un GPS qui transmet une position précise à quelques mètres près. Plus généralement, nous distinguerons deux catégories d'erreurs :

1. Les *erreur de première espèce* : ces erreurs faibles mais régulières sont intrinsèques au matériel ou aux méthodes utilisées. Il peut s'agir de la précision d'un algorithme (*e.g.* une estimation statistique) ou du matériel utilisé par un capteur (*e.g.* composants électroniques) voire par un humain (*e.g.* utilisation d'une règle ou d'un rapporteur). Ces erreurs sont maîtrisées car identifiées lors de la conception de l'outil et sont connues *a priori*.

2. Les *erreur de seconde espèce* : des erreurs fortes mais épisodiques qui ne sont pas prédictibles. Elles dépendent de l'environnement et/ou du phénomène observé. Par exemple, un obstacle qui perturbe la réception du signal par le GPS ou un objet chaud posé à proximité d'un thermomètre censé mesurer la température d'une pièce. Ces erreurs ou *anomalies* sont qualifiées de *fortes* car elles ne sont pas conformes au comportement attendu du système (Chandola et al., 2007). Elles se caractérisent donc par un écart significatif avec la réalité qui dépasse la précision du capteur ou de l'algorithme utilisé.

Dans notre contexte de systèmes d'information, nous limitons la compétence des blocs à leur capacité à ne pas commettre des erreurs de première espèce. Nous ne retenons pas les erreurs de seconde espèce du fait de leur nature fortement contextuelle et imprévisible. La compétence est réduite à une caractéristique **intrinsèque** des blocs et plus généralement propre à chaque entité du système, quelle qu'elle soit. Pour résumer, la compétence d'une entité traduit sa capacité à remplir une fonction de manière fiable et à ne pas faire d'erreurs. Elle est propre à l'entité considérée, quelle que soit sa nature (capteur, logiciel, humain).

4.1.2 Sincérité

La compétence n'est pas suffisante pour mesurer la confiance à des fins de sécurité. Si elle constitue un moyen d'évaluer la capacité d'une entité à faire des erreurs, elle ne permet pas de rendre compte des intentions de la cible. Sur la seule base de la compétence, il est donc difficile voire impossible de déterminer si une erreur est volontaire ou non. Par exemple, un capteur de température d'un moteur ou d'un processeur qui envoie une mesure erronée est-il mal réglé ou bien modifie-t-il sciemment ses mesures pour conduire à l'usure prématurée, voire la casse, du matériel ? Dans ce second cas, l'émission de fausses mesures ne permet aucune déduction sur les capacités d'acquisition du capteur.

En plus de la compétence, il apparaît nécessaire de prendre en compte les intentions des entités pour évaluer leurs interactions. Dans la section précédente, nous avons défini la compétence comme la capacité à ne pas commettre des *erreurs de première espèce*. La compétence définit alors un seuil en-dessous duquel les erreurs sont tolérées par le système (erreurs de première espèce) et au-dessus duquel celles-ci sont considérées comme significatives (erreurs de deuxième espèce). Dans ce cas, elles sont considérées comme pouvant révéler les intentions et doivent donc être traitées par un autre aspect de la confiance. Dans ce but, la bienveillance et la sincérité ont déjà été utilisées conjointement à la compétence (Lorini et Demolombe, 2008).

La bienveillance est souvent utilisée dans les modèles de confiance (McKnight et Chervany, 2000). Elle peut se définir comme une « attention envers le bien-être d'autrui et la motivation d'agir dans son intérêt » (McKnight et Chervany, 1996). Dans le cas d'un système d'information, une entité bienveillante doit donc agir dans l'intérêt du système. Cela implique que cette entité a le choix de décider ou non d'agir pour le bien du SI. Bien qu'adaptée à la nature humaine, ce n'est pas le cas pour des éléments tels que des capteurs qui ne possèdent pas de volonté propre. De plus, dans le cas particulier des sources d'information, il a été montré que la sincérité et la compétence sont suffisantes pour évaluer la confiance (Liu et Williams, 2002). En conséquence, nous proposons d'utiliser la notion de sincérité qui témoigne des intentions d'une entité qui peut être définie comme « le lien entre ce qu'une entité dit et ce qu'elle croit » (Demolombe, 2001). En particulier, la sincérité permet de représenter la malveillance, c'est-à-dire des intentions contraires au bien-être ou intérêts d'autrui.
Grâce aux progrès en micro-électronique, les capteurs intelligents sont aujourd'hui

dotés de capacités de calcul (Yick et al., 2008). Un capteur peut donc mesurer une grandeur (*e.g.* une température) et appliquer ensuite une transformation (post-traitement). Bien souvent, cette dernière sert à améliorer la qualité des mesures. Dans le cas d'une attaque, ce traitement peut également modifier les informations pour influencer le système. Lorsque le capteur améliore la qualité de ses mesures, il est considéré sincère au sens de Demolombe, 2001 : ce qu'il « dit » (la mesure traitée) est similaire à ce qu'il « pense » (la mesure brute). Dans le cas contraire, où il modifie les informations, les deux mesures ne coïncident pas et il est donc considéré non sincère. Cette définition impose toutefois d'avoir accès au fonctionnement interne du capteur pour comparer les mesures avant et après le post-traitement. Comme expliqué dans le chapitre 3, nous considérons les composants d'un système d'information comme des boîtes noires dont le fonctionnement interne est inconnu. Seule la mesure traitée est accessible dans notre modèle, ce qui pose le problème de l'évaluation de la sincérité.

La sincérité permet de prendre en compte les intentions de l'entité cible de la relation de confiance et donc, par extension, son éventuelle malveillance à l'égard du système. De plus, celle-ci est transposable à tout composant du système, quelle que soit la nature de celui-ci. Elle se démarque ainsi de concepts réservés aux relations humaines, tels que la bienveillance. Avec la compétence, celle-ci sert donc de base à notre définition de la confiance.

4.1.3 Risque et connaissance

La confiance est liée à la connaissance (Luhmann, 1979) qui peut se manifester sous différentes formes. Dans une relation, la confiance évolue au fil des interactions entre les acteurs (Mui et al., 2002), chacune apportant une certaine quantité d'information aux acteurs sur l'autre. Par accumulation, ces informations forment une connaissance qui permet de prédire avec plus ou moins de facilité les futurs échanges. La connaissance est liée au temps : la somme des interactions constitue un historique duquel il est possible d'extraire une connaissance des deux parties. Cette évolution de la connaissance introduit une dynamique de la confiance : la confiance à l'instant t dépend des instants précédents. Dans le même temps, des informations peuvent être connues à l'avance ou *a priori*, c'est-à-dire non déduites par une quelconque analyse des échanges. La cartographie du système d'information (entités et leurs relations associées), les caractéristiques des composants (plages de fonctionnement, précision, statut (en service, à l'arrêt, en

panne)) ou encore des informations externes non produites par le système (*e.g.* une analyse de risque ou une carte pour un Système d'Information Géographique) sont des exemples de telles connaissances *a priori*.

Cette prédictibilité des échanges est inhérente à la confiance mais est imparfaite et introduit alors un risque : le prochain échange n'étant pas exactement prédictible, il peut ou non tourner en la défaveur d'un des acteurs, sans que celui-ci puisse s'adapter. Le risque inclut des pertes potentielles (impact) et la probabilité d'un danger éventuel (Dufour et Pouillot, 2002). Un danger est un évènement qui est suffisant pour provoquer des pertes. Dans une relation de confiance, les pertes éventuelles sont supérieures à l'utilité de la relation pour chacun des acteurs (Deutsch, 1958), c.-à-d. que les bénéfices apportés par la relation ne suffisent pas à compenser les risques encourus. Un système d'information est sujet à de nombreux risques pouvant affecter sa sûreté de fonctionnement autant que sa sécurité. Par exemple, une voie d'eau, une surtension voire un bogue, c.-à-d. un dysfonctionnement logiciel pouvant avoir des répercussions physiques et causé par une erreur accidentelle de programmation, peuvent altérer ou détériorer le fonctionnement d'un ou plusieurs éléments du SI. De plus, un système d'information peut être la cible de différentes menaces, des dangers qui résultent d'une action intentionnelle. Du fait de leur rôle parfois prépondérant pour la sécurité physique des personnes et/ou du matériel, les systèmes d'information embarqués sur les navires constituent des cibles intéressantes pour des personnes malintentionnées (*e.g.* pirates, terroristes, hacktivistes). Les risques sont omniprésents : à chaque instant, le SI peut être atteint par une défaillance résultant d'une erreur humaine, logicielle (Kuhn et al., 2004 ; Lions, 1996) ou mécanique (*e.g.* usure d'une pièce) ou même par une attaque d'un niveau allant de générique (*i.e.* sans cible désignée *a priori*) à ciblées. Toute situation est donc risquée.

La connaissance et le risque sont deux composantes de la confiance qui ne sont cependant pas explicitement prises en compte dans les modèles. Elles se manifestent implicitement au travers de divers paramètres qui vont influencer la confiance dans les entités du SI. Ces paramètres peuvent être des coefficients ou des algorithmes. Cette confiance est évaluée par un observateur passif (notre système d'évaluation) à partir de l'analyse des informations échangées entre les différents blocs mais également des multiples connaissances dont il dispose *a priori*. Notre système évalue donc la confiance que peut avoir le système d'information dans ses composants. La sécurité du système d'évaluation, et donc la confiance qu'il peut avoir dans lui-même, ne fait pas l'objet de notre étude. Cependant, des pré-

conisations existent pour limiter les risques lors de l'utilisation de tels systèmes (Chifflier et Fontaine, 2014).

La compétence et la sincérité ne sont donc pas les seuls critères à prendre en compte dans un modèle de confiance pour la sécurité d'un système d'information : la connaissance et le risque sont également à considérer. Cependant, nous ne les prendrons pas en compte de la même manière. Comme nous le verrons, ils feront partie des éléments environnementaux.
La connaissance dont dispose le système chargé d'évaluer la confiance pourra être acquise par observation des échanges d'information entre les blocs ou bien donnée avant le démarrage du système. Ainsi, pour ce qui nous concerne, nous nous limitons à une connaissance *a priori* de l'architecture du système (les blocs fonctionnels, leurs relations et leurs caractéristiques) et à la prise en compte de sa dynamique (voir la section 4.3). Au-delà, nous ne prendrons pas en compte le risque, celui-ci étant considéré comme omniprésent et nous cherchons à le détecter. En particulier, la performance des solutions que nous proposons par la suite pourraient être améliorée en tirant parti d'analyses de risques, telles que celles qui existent sur la vulnérabilité de l'AIS (Iphar et al., 2016), et compléter le modèle du SI utilisé.

En conclusion, les mesures de confiance que nous proposons ci-après s'appuieront sur des mesures de sincérité et de compétence, qui prendront en paramètres des éléments de connaissance connus *a priori*.

4.2 Mesures de la confiance

Telle que définie précédemment, la confiance est une fonction de la compétence et de la sincérité lesquelles évoluent au cours du temps (Paglieri et al., 2014 ; Liu et Williams, 2002). Avant de rentrer dans le détail de la mesure de ces différentes caractéristiques au sein d'un système d'information, il convient de préciser deux hypothèses contextuelles importantes que nous avons faites afin de réduire la complexité du problème à traiter. Dans le contexte des systèmes d'information embarqués à bord des navires, la première considère que le matériel est en parfait état de marche : le système ne comporte pas, par défaut, d'éléments défaillants ou fonctionnant en mode dégradé. Dans un second temps, nous considérons que chaque élément du système respecte les spécifications techniques nécessaires à son bon fonctionnement ; spécifications fournies ou si besoin mises à jour avant le dé-

marrage de notre système. Plus clairement, les caractéristiques intrinsèques d'un élément du système ne changent pas au fil du temps. Les composants ne s'usent donc pas au cours du temps. Cette hypothèse a un sens du fait de la grande durée de vie des matériels (plusieurs dizaines d'années) et des fortes contraintes de sûreté de fonctionnement auxquelles ceux-ci sont soumis. C'est sous ces deux hypothèses que nous cherchons à construire des mesures de compétences, de sincérité puis de confiance.

Ces mesures sont par ailleurs établies sur la base d'une modélisation des SI que nous avons décrite dans le chapitre 3. Comme illustré en figure 4.1, nous rappelons qu'un système d'information est constitué de plusieurs blocs fonctionnels de types différents. Cette typologie est basée sur la façon dont les blocs interagissent avec l'information. Les sources observent l'environnement réel et émettent de l'information (*e.g.* mesure d'un phénomène physique), les collecteurs reçoivent de l'information, les blocs de traitement reçoivent puis émettent de l'information et enfin les blocs de rétroaction ont leur sortie connectée à leur entrée pour traiter itérativement l'information.

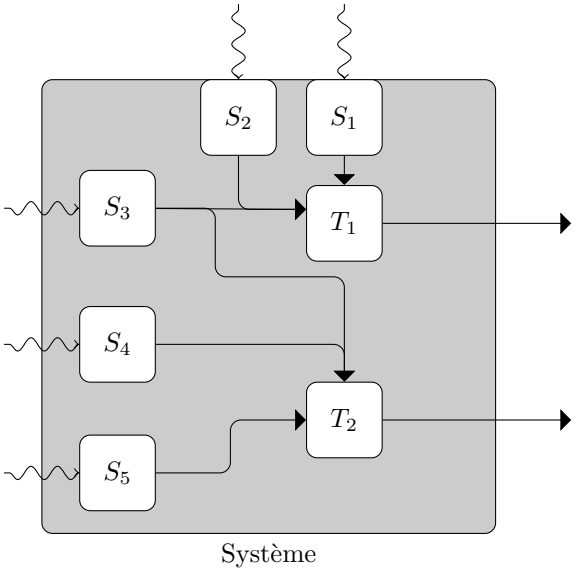


FIGURE 4.1 – Exemple de la modélisation d'un système d'information simple constitué de blocs fonctionnels

Dans ce qui suit, nous décrivons d'abord les mesures de compétence, de sincérité et de confiance pour des sources d'information. Nous traiterons en section 4.3 le cas des autres blocs fonctionnels et la problématique de la propagation de la

confiance. En effet, les sources sont les points d'entrée du système : elles perçoivent l'environnement du système et alimentent les autres blocs fonctionnels avec leurs mesures. Tout bloc du système reçoit donc, directement ou non, des informations provenant d'une ou plusieurs sources. Le niveau de confiance qui peut leur être accordé dépend donc des autres éléments du système d'information.

4.2.1 Mesurer la compétence des sources

Dans notre étude, nous nous intéressons à des sources de type « capteur » produisant des mesures numériques (des signaux monodimensionnels). La mesure n'étant pas parfaite, nous avons modélisé dans le chapitre 3 ces sources d'information à l'aide de processus aléatoires où l'imprécision de la mesure d'un signal (processus $X(t)$) est modélisée comme l'ajout d'un bruit gaussien ($b(t)$) à la réalité observée (figure 4.2). L'information produite par les capteurs est alors modélisée par une observation augmentée d'un bruit aléatoire de loi normale centrée $\mathcal{N}(0, \sigma)$.

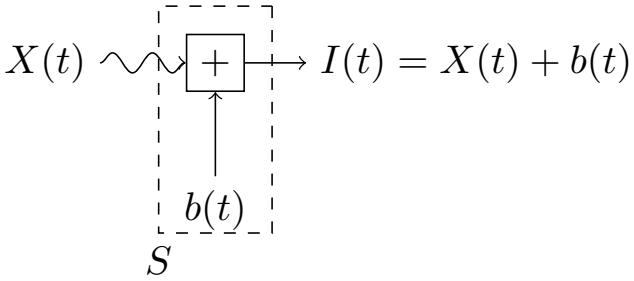


FIGURE 4.2 – Modélisation d'une mesure numérique via un canal gaussien

D'après la définition proposée en section 4.1.1, la compétence *Comp* d'une entité caractérise sa « capacité à remplir les fonctions qui lui sont attribuées ». Dans le cas d'une source, la compétence est donc dépendante de l'imprécision de sa mesure. Il s'agit d'une observation de la réalité entachée d'erreurs. En effet, la mesure est plus ou moins précise et reflète plus ou moins bien la réalité. Dans le cas d'une source idéale, la mesure est strictement égale à la valeur réelle de l'observation ($I(t) = X(t)$). À l'inverse, une imprécision élevée résulte en une mesure décorrélée de l'environnement. Ainsi, une source est jugée d'autant plus compétente si elle remplit sa fonction en fournissant une mesure au plus près de l'observation réelle. Nous venons là d'exprimer différentes contraintes sur notre mesure de compétence :

- la compétence d'une source doit être maximale lorsque l'imprécision est minimale (cas d'une source idéale : $Comp = 1$) ;
- lorsque l'imprécision est maximale, la compétence devient minimale ;
- nous postulons également que la compétence d'une source est **décroissante** selon son imprécision.
- considérant deux sources, la plus compétente des deux est celle qui est la plus précise. Sa mesure représente mieux ou est la plus fidèle à la réalité.

Sur la base de notre modèle de source d'information, où l'imprécision correspond à l'ajout d'un bruit gaussien $b(t)$ de loi $\mathcal{N}(0, \sigma)$, d'écart-type σ, nous cherchons une mesure de la compétence f_c telle que :

$$Comp = f_c(b) \overset{b \text{ est centrée}}{=} f_c(\sigma) \in [0; 1]$$

Considérant une source parfaite la compétence est maximale, *i.e.* $f_c(\sigma = 0) = 1$. A contrario, si la source est très imprécise voire indépendante de la réalité observée (*i.e.* $\sigma \to +\infty$), elle est incompétente et sa mesure de compétence telle que $\lim_{\sigma \to +\infty} f_c(\sigma) = 0$. Également, dans le cas de deux sources d'information S_i et S_j de **même nature** et observant le **même phénomène** mais avec des précisions différentes, *i.e.* $\sigma_i \neq \sigma_j$, f_c doit satisfaire la propriété suivante :

$$\forall (i, j) \in \mathbb{N}^2, i \neq j, \quad \sigma_i \leq \sigma_j \Rightarrow Comp_i \geq Comp_j$$

La fonction f_c que nous proposons d'utiliser dépend d'un seul paramètre mais peut prendre une expression plus ou moins complexe. Pour ce qui nous concerne, nous avons considéré différentes fonctions :

$$\bullet \quad f_n = \frac{1}{1 + \sigma^n} \quad \text{avec } n \in \mathbb{N}^* \tag{4.1}$$

$$\bullet \quad f_{\log} = \frac{1}{1 + \log(\sigma + 1)} \tag{4.2}$$

$$\bullet \quad f_{\exp} = e^{-\sigma} \tag{4.3}$$

Notre choix porte sur la simplicité de ces fonctions qui ont par ailleurs des comportements bien différents comme illustré en figure 4.3.

La fonction $\frac{1}{1+\log(\sigma+1)}$, du fait de sa croissance lente, atteint une valeur limite nulle lorsque σ tend vers l'infini. En pratique, considérant que $\sigma < \sigma_{max}$, pour tout matériel, cette fonction converge vers une constante $f_{\log}(\sigma_{max})$. Cette fonction définit donc un minimum de compétence non nul qui varie selon la précision des capteurs visés. Une telle propriété est désirable en présence de capteurs pouvant

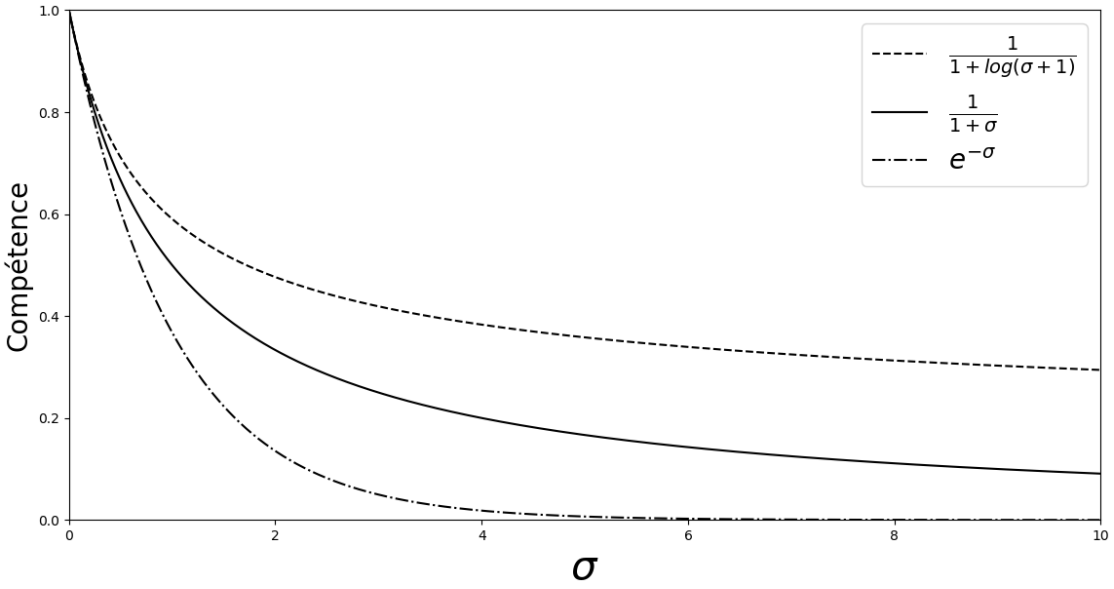


FIGURE 4.3 – Comparaison de plusieurs fonctions possibles pour une mesure compétence

fonctionner en mode dégradé : même si leur précision diminue, leur compétence reste positive. Cela signifie que les mesures des capteurs sont toujours pertinentes pour le système, mais pas au même degré que lorsqu'ils sont en parfait état de marche.

La fonction $e^{-\sigma}$ décroît rapidement vers 0 quand σ augmente. Cette fonction permet, pour des capteurs de précisions proches, d'obtenir des mesures de compétence très différentes pour de petites valeurs de σ. Il peut néanmoins y avoir un intérêt à avoir un effet de seuil. C'est ce que proposent les fonctions $\frac{1}{1+\sigma^n}$.

Plus la valeur de n est grande et plus la fonction aura un effet de seuil important comme illustré en figure 4.4. Plus clairement, pour une petite variation autour d'une valeur critique (ici $\sigma = 1$), la fonction de compétence prend des valeurs très différentes. En effet, pour $n = 10$, nous pouvons observer que $\frac{1}{1+\sigma^n} \approx 1$ pour $\sigma \leq 1$ et $\frac{1}{1+\sigma^n} \approx 0$ sinon.

Dans notre contexte, les sources ne s'usent pas et sont en parfait état de marche. De plus, les capteurs embarqués dans les systèmes de navigation sont relativement précis : une erreur moyenne de l'ordre de 10^{-5} pour la mesure d'une latitude par un GPS ou $0,1$ pour un Loch évaluant une vitesse par ultrasons. Pour des raisons de simplicité, nous choisissons d'utiliser la fonction $\frac{1}{1+\sigma}$ en tant que mesure de compétence.

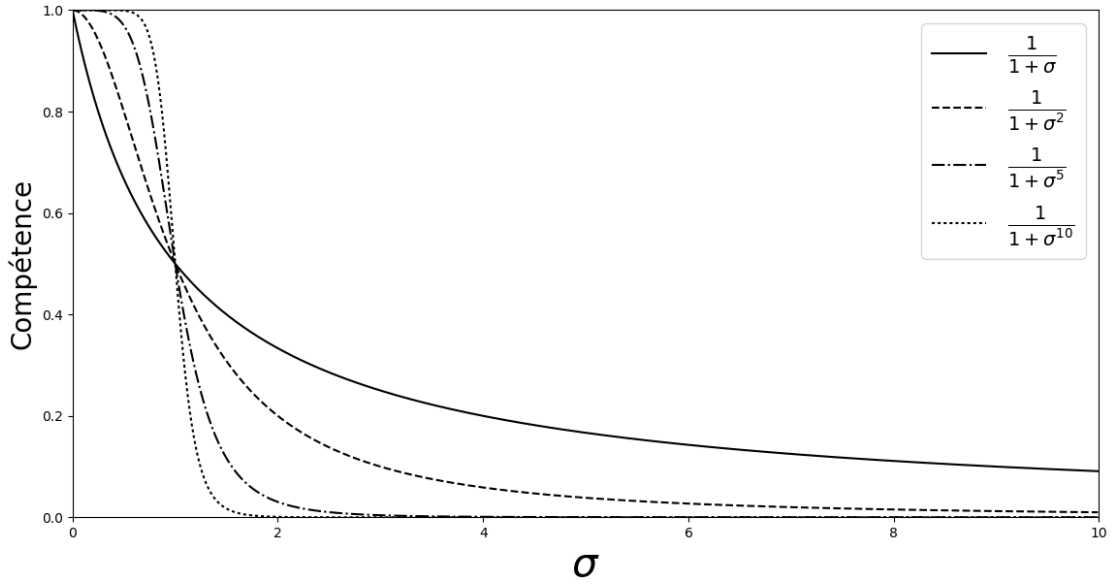


FIGURE 4.4 – Évolution des fonctions puissance selon le paramètre n

Ces fonctions de compétence peuvent évoluer avec le temps. Sur la base de notre modèle de source, il suffit dans ce cas de faire varier le paramètre de précision de la source σ avec le temps (c.-à-d. $\sigma(t)$). Cette variabilité dans la mesure peut traduire de l'usure du matériel, par exemple. Les fonctions que nous proposons restent d'actualité et peuvent être précisées en ajoutant un facteur d'échelle :

- $f_n = \dfrac{1}{1+(\frac{\sigma(t)}{a})^n}$ avec $n \in \mathbb{N}^*$
- $f_{\log} = \dfrac{1}{1+\log(\frac{\sigma(t)}{a}+1)}$
- $f_{\exp} = e^{-\frac{\sigma(t)}{a}}$

Ce facteur d'échelle a permet de positionner en fonction du besoin le point d'inflexion des fonctions $\frac{1}{1+(\frac{\sigma(t)}{a})^n}$.

4.2.2 Mesurer la sincérité des sources

Dans la section 4.1.2, nous avons défini la sincérité comme étant « le lien entre ce qu'une entité dit et ce qu'elle croit ». Sur la base de cette définition, LIU et WILLIAMS, 2002 proposent de mesurer la sincérité d'une source d'information, un humain dans leur contexte, en comparant directement les informations

qu'elle transmet avec ses « pensées ». En ce qui nous concerne, les blocs fonctionnels constituant un système d'information sont vus comme des boîtes noires dont le fonctionnement interne est inconnu. Le concept de « pensée » d'une source n'est donc pas valide [1]. Nous proposons alors de comparer les informations des différentes sources entre elles à l'instar de Paglieri et al., 2014. Ainsi, la sincérité d'une source est évaluée à partir des informations émises par les autres sources. Cependant, différents aspects doivent être considérés. Toutes les sources n'émettent pas les mêmes types d'informations. Pour un même phénomène observé, celles-ci sont de différentes natures. Par exemple, le système de navigation d'un navire se repose sur de multiples capteurs pour mesurer sa vitesse (Loch), sa position (GPS) et son cap (Gyroscope). Ces informations ne sont pas comparables entre elles : elles ne traduisent pas la même grandeur et par conséquent ne s'interprètent pas de la même façon (*e.g.* 1 degré de cap et 1 nœud de vitesse). Pour mesurer la sincérité des sources, il est donc nécessaire de se restreindre aux sources émettant des informations de **même nature**. Quelle que soit leur nature, les informations ne résultent pas nécessairement de l'observation du même phénomène. Celles-ci dépendent de la **topologie** du système (cf. section 3.2.2). Nous considérons que deux sources sont *proches* si elles observent le même phénomène.

Ainsi, considérant un ensemble de sources de même nature et proches, il s'agit d'évaluer le degré d'accord de la source i avec les autres à un instant t. Noté $p_i(t)$, ce degré se mesure en comparant les informations que celle-ci fournit avec celles émises par les autres sources. Il sera élevé si l'information émise par la source est en accord avec celle des autres. Inversement, il sera d'autant plus faible que la source sera contestée par les autres. La sincérité est donc une fonction **croissante** selon l'accord entre les sources. Une source émettant une information similaire à la majorité sera jugée plus sincère qu'une source contestée (*i.e.* en accord avec une minorité). Tel qu'exprimé, le degré d'accord est une mesure de consensus, c'est-à-dire à quel point la source est supportée par les autres sources.

Le consensus se modélise mathématiquement par une fonction de similarité qui mesure la proximité des opinions des membres d'un groupe (Chiclana et al., 2013). En ce qui nous concerne, ces opinions sont modélisées sous la forme d'informations numériques et elles s'expriment plusieurs fois au fil du temps, constituant alors des suites d'informations. Il existe de nombreuses façons de mesurer la similarité entre deux suites numériques. Les différentes mesures sont des variantes

1. Le concept de « pensée » est toutefois transposable à des éléments non humains. Un exemple basé sur des capteurs intelligents est explicité en section 4.1.2

de quelques fonctions de base qui peuvent être regroupées selon deux approches (Cha, 2007) :

- Approche vectorielle : les différentes informations sont supposées indépendantes au cours du temps. Chaque information est vue comme un point dans un espace et les mesures de similarités exploitent les relations géométriques qui relient les deux ensembles de points.
- Approche probabiliste : les deux suites d'informations résultent de processus probabilistes qui peuvent être inférés à partir des informations produites. Les mesures de similarité déterminent les ressemblances des processus d'origine.

Quelle que soit l'approche considérée, une fonction de similarité permet d'évaluer un consensus entre deux entités. Ces consensus individuels peuvent ensuite être agrégés pour mesurer le consensus entre une entité particulière et le groupe dont elle fait partie. Cette agrégation peut être vue comme le ratio entre le nombre de sources en accord avec la source i à l'instant t, et le nombre total de sources. Pour mesurer l'accord entre deux sources, une solution possible est de passer par un consensus binaire, comme proposé par Paglieri et al., 2014, où deux sources sont complètement d'accord ou en complet désaccord. Dans le contexte de ce travail, et avec le modèle de source vu en section 3.3, un consensus binaire n'est pas l'approche la plus judicieuse, car l'information correspond à des nombres réels. Du fait de précisions différentes, les mesures ne sont pas égales. Nous avons donc besoin de quantifier l'écart entre les mesures pour évaluer le consensus entre les sources. De plus, nous avons modélisé les sources d'information comme des processus gaussiens. Nous proposons alors d'utiliser une fonction de similarité, notée Sim, continue et basée sur une approche probabiliste pour mesurer le consensus prenant en compte les informations émises aux instants précédents, c'est-à-dire :

$$\forall t > 0 \quad p_i(t) = \begin{cases} 1 & i = 1 \\ \dfrac{1}{n-1} \displaystyle\sum_{\substack{j=1 \\ j \neq i}}^{n} Sim(\{Y_i(t)\}_{t>0}, \{Y_j(t)\}_{t>0}) & i \geq 2 \end{cases}$$

où n est le nombre de sources et $\{Y_i(t)\}_{t>0}$ l'ensemble des informations émises par la source i jusqu'à l'instant t. Dans le cas de plusieurs sources d'informations, la mesure $p_i(t)$ est la moyenne des similarités entre les informations envoyées par la sources et celles transmises par les autres. De manière à garantir $p_i = 1$ lorsque toutes les sources sont en accord et inversement si $p_i = 0$ lorsque la source i s'oppose à toutes les autres, la fonction de similarité utilisée ci-après correspond à

une mesure de corrélation entre les informations des différentes sources. Un autre intérêt de cette mesure est que la valeur de p_i se stabilise lorsque le nombre n de sources est grand. Dans le cas particulier d'une unique source, le consensus ne peut être mesuré à cause de l'absence d'informations supplémentaires. Par convention, nous proposons alors de poser $p_1(t) = 1$ ce qui symbolise l'accord de la source avec elle-même.

Il est également important de souligner qu'il existe un phénomène de dépendance entre la compétence et la sincérité. En effet, lorsqu'une source est incompétente, elle émet une information très imprécise qui rend difficile sa comparaison avec des informations fournies par des sources compétentes. Plus une information est imprécise et plus celle-ci sera éloignée de la réalité et, par voie de conséquence, des autres informations plus précises. Dès lors, dans le cas où la compétence de la source est faible, sa sincérité doit l'être également. Par contraposition, lorsque la compétence de la source est élevée (i.e. proche de 1), aucune conclusion ne peut être induite sur sa sincérité. Nous proposons alors de borner la mesure de sincérité d'une source par sa compétence :

$$\forall i \geq 1 \quad Sinc_i(t) = min(p_i(t), Comp_i) \ \leq \ Comp_i$$

Il en résulte une égalité directe importante entre la sincérité d'une source unique et sa compétence :

$$Sinc_1(t) = Comp_1(t) \text{ pour tout } t.$$

4.2.3 Mesurer la confiance des sources

À partir des mesures de compétence et de sincérité que nous venons d'établir, nous souhaitons construire une mesure de confiance. Comme nous l'avons vu au chapitre 2, différentes représentations de la confiance existent selon l'expressivité voulue : discrète, continue mono ou multidimensionnelle. En général, une mesure de confiance est comprise dans l'intervalle $[0; 1]$. La valeur 0 représente la méfiance totale. La valeur $\frac{1}{2}$ est ambiguë : elle peut représenter un avis partagé (autant de confiance que de méfiance) ou bien l'absence totale de connaissance. En ce qui nous concerne, nous nous intéressons à des problématiques liées à la sécurité des SI où la menace est omniprésente : chaque bloc peut compromettre l'intégrité du système. Par voie de conséquence, nous faisons le choix de considérer tout élément étranger comme suspect et non digne de confiance. En l'absence de connaissance,

notre mesure de confiance est donc égale à 0 et la valeur $\frac{1}{2}$ représente un avis partagé.

Pour obtenir une mesure de confiance $Conf_i(t)$ dans une source S_i à partir des mesures de compétence et de sincérité (i.e. $Conf_i(t) = f(Comp, Sinc(t))$), plusieurs solutions ont été définies dans (Liu et Williams, 2002). Ces mesures respectent toutes les contraintes suivantes :

- $Conf(1, 1) = 1$
- $Conf(0, 0) = 0$
- $Conf(Comp, 1) = Comp,\ Comp \in [0; 1]$
- $Conf(1, Sinc) = Sinc,\ Sinc \in [0; 1]$

Plus généralement, nous postulons qu'une mesure de confiance est **croissante** selon chacune de ses deux opérandes. Plus clairement, pour deux sources de compétences (resp. de sincérités) identiques, celle qui est la plus de confiance est celle qui est la plus sincère (resp. la plus compétente).

Les auteurs (De Cock et Da Silva, 2006 ; Liu et Williams, 2002) proposent ainsi plusieurs mesures en adéquation avec les contraintes évoquées ci-avant :

$$Conf_1(Comp, Sinc) = Comp * Sinc \tag{4.4}$$
$$Conf_2(Comp, Sinc) = min(Comp, Sinc) \tag{4.5}$$
$$Conf_3(Comp, Sinc) = max(0, Comp + Sinc - 1) \tag{4.6}$$
$$Conf_4(Comp, Sinc) = 1 - (1 - Comp)(1 - Sinc) \tag{4.7}$$

La figure 4.5 montre l'évolution de ces différentes mesures de confiance selon la sincérité de la source. Dans cet exemple, la compétence est fixée à 0.9, ce qui correspond à un un capteur, comme un GPS ou un Loch mesurant une vitesse, doté d'une précision d'environ 0.1.

Cet exemple montre qu'il existe une nette différence entre la mesure $Conf_4$ et les autres : quelle que soit la sincérité, $Conf_4$ prend des valeurs proches de 1. Notamment, $Conf_4$ est non nulle lorsque la compétence ou la sincérité de la source est nulle, propriété cependant souhaitée dans notre contexte. Cela revient à ajouter au jeu de contraintes précédent, les règles supplémentaires suivantes :

- $Conf(0, Sinc) = 0,\ Sinc \in [0; 1]$
- $Conf(Comp, 0) = 0,\ Comp \in [0; 1]$

Ces règles traduisent une absence de confiance en une source incompétente ou non sincère. En effet, une source incompétente produit des informations qui sont très éloignées de la réalité ainsi que de celles produites par les autres sources. Au-delà

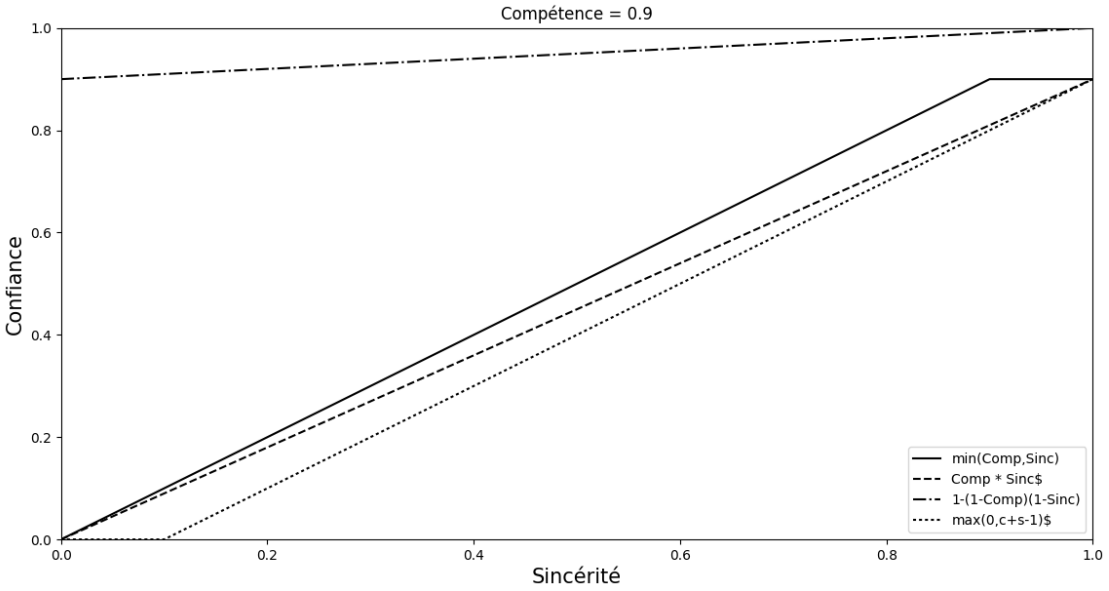


FIGURE 4.5 – Différentes fonctions de confiance pour une source de compétence $Comp = 0.9$

d'une certaine imprécision, dépendante du contexte dans lequel la source est utilisée, les informations ne sont pas considérées fiables et ne peuvent être exploitées par le système (*e.g.* une erreur de positionnement de plusieurs kilomètres). Ainsi, nous considérons ces sources comme n'étant pas de confiance. De même, une source non sincère n'est pas en accord avec les autres alors que toutes rendent compte d'une même réalité. Nous considérons donc également de telles sources comme n'étant pas de confiance.

Dans cette section, nous avons considéré plusieurs contraintes à prendre en compte lors de la construction d'une mesure de confiance. L'ensemble des propriétés désirées pour toutes nos mesures est résumé dans le tableau 4.1.

4.3 Propagation de confiance dans un système d'information

Dans la section précédente, nous avons montré comment il est possible de mesurer la confiance en une source d'information sur la base de mesures de compétence et de sincérité. Cependant, les sources ne sont pas les seuls composants

Mesures	Propriétés
Compétence	• Fonction de l'imprécision de la mesure : $Comp_i = f_c(\sigma_i)$ • $f_c(0) = 1$ • $\lim_{\sigma_i \to +\infty} f_c(\sigma_i) = 0$ • Décroissance : $\forall i \neq j, \quad \sigma_i < \sigma_j \Rightarrow Comp_i \geq Comp_j$
Sincérité	• Compétence **faible** $\Rightarrow$ On considère une sincérité **faible** • Compétence **forte** $\not\Rightarrow$ sincérité **forte** • $Sinc_i(t) = f_s(p_i(t), Comp_i)$, $p_i(t)$ une mesure du consensus entre les sources à l'instant t • Croissance par rapport au consensus : $\forall i \neq j,\ p_i(t) < p_j(t)$ et $Comp_i = Comp_j$ $\Rightarrow Sinc_i(t) \leq Sinc_j(t)$ • Croissance par rapport à la compétence : $\forall i \neq j,\ p_i(t) = p_j(t)$ et $Comp_i < Comp_j$ $\Rightarrow Sinc_i(t) \leq Sinc_j(t)$
Confiance	• $Conf = f(Comp, Sinc)$ • $f(0,0) = 0$ • $f(1,1) = 1$ • $f(Comp, 1) = Comp,\ Comp \in [0;1]$ • $f(1, Sinc) = Sinc,\ Sinc \in [0;1]$ • $f(0, Sinc) = 0$ • $f(Comp, 0) = 0$ • Croissance par rapport à la compétence : $\forall i \neq j,\ Comp_i < Comp_j$ et $Sinc_i(t) = Sinc_j(t)$ $\Rightarrow Conf_i(t) \leq Sinc_j(t)$ • Croissance par rapport à la sincérité : $\forall i \neq j,\ Comp_i = Comp_j$ et $Sinc_i(t) < Sinc_j(t)$ $\Rightarrow Conf_i(t) \leq Conf_j(t)$

TABLE 4.1 – Tableau des propriétés des mesures associées à la compétence, la sincérité et la confiance

d'un système d'information et d'autres blocs fonctionnels existent. Pour mesurer la confiance dans un bloc fonctionnel, celui-ci doit émettre de l'information. Nous rappelons que les sources sont les points d'entrée du système et qu'à ce titre, elles sont les seules à produire de l'information sans avoir besoin d'en recevoir au préalable. Par conséquent, aux premiers instants du système, la confiance ne peut être

mesurée pour les blocs fonctionnels autres que les sources d'information. La question posée est alors comment propager la confiance dans les sources aux autres blocs puis entités d'un système d'information.

Par définition, la confiance se propage d'une entité à une autre à travers un lien qui les unit. La confiance de l'entité cible est dépendante de celle de l'origine ainsi que des caractéristiques du lien. Par exemple, selon que le lien soit uni ou bidirectionnel, la propagation peut s'effectuer dans un sens ou dans les deux. Dans la section 3.2.2, nous avons défini quatre relations importantes à considérer dans notre contexte : la relation de nature, la relation topologique, la relation de dépendance et la relation d'entité. Nous avons montré que les deux premières doivent être prises en compte lors de la mesure de la sincérité des sources d'information (cf.section 4.2.2). Nous souhaitons désormais définir deux types de propagation, chacune basée sur une relation différente : la relation de dépendance et la relation d'entité. Nous qualifions respectivement ces deux propagations d'*horizontale* et de *verticale*.

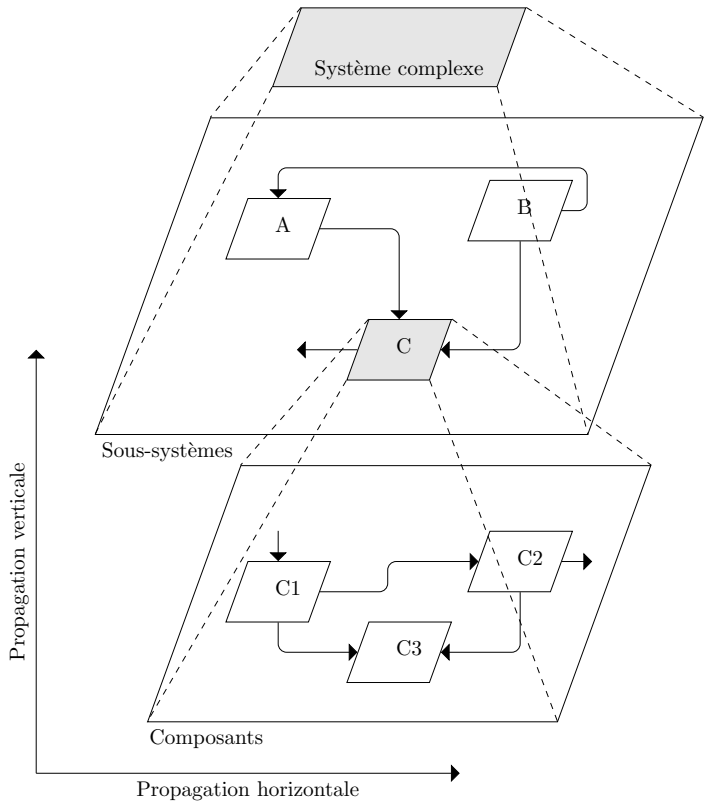


FIGURE 4.6 – Sens de propagation de la confiance au sein d'un système d'information

La figure 4.6 montre les sens de propagation de la confiance. La confiance se propage horizontalement dans une même couche du système puis verticalement, depuis la base vers le sommet.

Dans le chapitre 2, nous avons présenté deux opérateurs permettant de propager la confiance : la concaténation (ou propagation en série) et l'agrégation (ou propagation parallèle). Nous rappelons ci-dessous leurs propriétés :

Propagation parallèle (opérateur $\oplus$)	Propagation en série (opérateur $\otimes$)
• Commutativité • Associativité • La confiance propagée croît quand la plus faible des confiances d'origine augmente	• Associativité • La confiance propagée décroît quand le nombre de relations augmente

TABLE 4.2 – Propriétés des opérateurs de propagation de la confiance

Formellement, ces deux opérateurs prennent en opérandes deux mesures de confiance pour produire une nouvelle mesure de confiance propagée. Comme nous le verront par la suite, ils nous serviront de base pour construire nos propres mesures de propagation au sein de systèmes d'information.

4.3.1 Propagation horizontale

La propagation horizontale est basée sur la relation de dépendance. La confiance accordée aux blocs dépend des multiples sources et des autres blocs alimentant leurs entrées. Les sources sont les points d'entrée du système : elles perçoivent l'environnement du système et alimentent les autres blocs avec leurs mesures. Tout autre bloc du système reçoit donc des informations, au moins indirectement, provenant d'une ou plusieurs sources. Ainsi, les informations irriguent le système en se propageant d'un bloc à l'autre. De même, les confiances dans chacune des sources, déduites de l'analyse des informations produites, se propagent aux autres blocs à travers les liens qui les relient. La figure 4.7 montre les dépendances du bloc T_1 (en traits pleins).

Du point de vue du bloc qui reçoit les informations, ses entrées sont considérées **commutatives** : il n'existe pas d'ordre (séquentiel) entre elles ; toutes les entrées du bloc peuvent être considérées simultanément. La confiance ne peut se

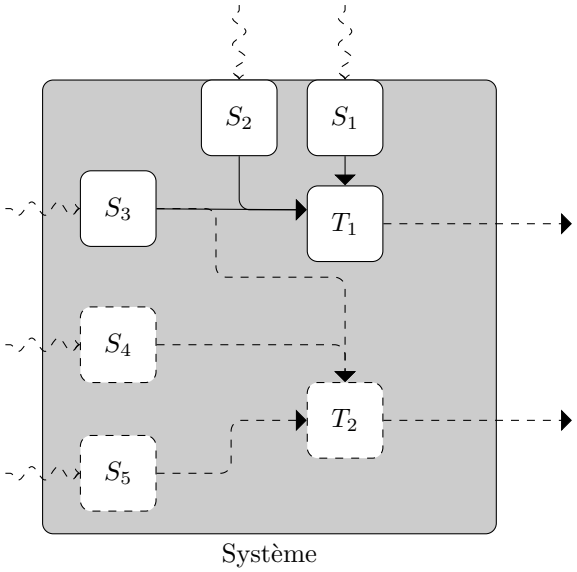


FIGURE 4.7 – Vue locale d'acquisition d'informations par un bloc

propager à un bloc qu'à partir de blocs dont les confiances sont connues, ce qui signifie *de facto* que leurs dépendances (quand elles existent) ont déjà été prises en compte. De cette façon, chaque bloc en entrée peut être considéré indépendamment des autres au moment d'évaluer la confiance du bloc destination (de la relation de dépendance).

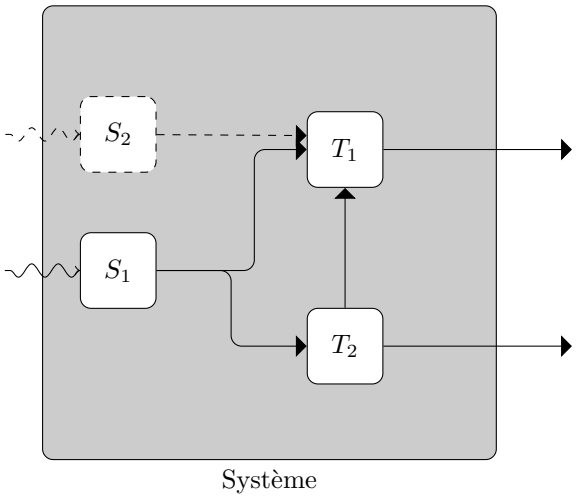


FIGURE 4.8 – Illustration de la propriété de commutativité : les confiances de la source S_1 et du bloc T_2 se propagent simultanément au bloc T_1.

Par exemple, sur la figure 4.8, la source S_1 alimente les blocs de traitement T_1 et T_2 en informations et ce dernier alimente également le bloc T_1. Lors de l'évaluation de la confiance du bloc T_1, les confiances de S_1 et de T_2 sont considérées connues. Par conséquent, la dépendance de T_2 à S_1 a déjà été prise en compte et les deux blocs (S_1 et T_2) peuvent être considérés simultanément lors de l'évaluation de la confiance de T_1. De plus, considérant un nombre m d'entrées, l'ajout d'un bloc supplémentaire dont la confiance est au moins égale à la plus faible des confiances des m premiers blocs ne diminue pas la confiance propagée dans le bloc destination. Sur la figure 4.8, l'ajout de S_2 telle que $Conf(S_2) \geq \min(Conf(S_1), Conf(T_2))$ ne diminue pas la confiance de T_1. La propagation de confiance dans le cas de la relation de dépendance est dans une configuration **parallèle** : les entrées sont considérées commutatives et la confiance du bloc destination dépend de l'ensemble des confiances des blocs en entrée.

Dans un système composé de n blocs, la confiance dans un bloc B_i recevant des informations de la part de m blocs $\left(B_{i_j}\right)_{j=1..m}$, $i_j \in \{1, \ldots, n\}$, dépend de leurs criticités $c_{i_j,i}$. La criticité d'une relation entre deux blocs peut être vue comme une mesure de l'importance ou de la contribution d'une entrée par rapport aux autres dans le traitement effectué par le bloc. La criticité porte donc une information sur le lien entre des blocs et non pas sur la contribution individuelle d'un bloc dans le fonctionnement du système. Pour cette raison, les $c_{i_j,i}$ sont normalisés, c'est-à-dire tels que $\sum_{j=1}^{m} c_{i_j,i} = 1$. Par convention et en l'absence de connaissance sur les spécificités du traitement des blocs, nous supposerons que toutes les entrées d'un bloc sont de même importance, c.-à-d. $c_{i_j,i} = \frac{1}{m}\ \forall j \in \{1, \ldots, m\}$.

À partir de ces définitions, nous proposons de mesurer la confiance dans un bloc B_i recevant des informations de la part de m blocs $\left(B_{i_j}\right)_{j=1..m}$ comme suit :

$$Conf_i(t) = \bigoplus_{B_{i_j}} Conf_{B_{i_j}}(t-1) \tag{4.8}$$

$$= \frac{1}{m} \sum_{j=1}^{m} c_{i_j,i} \times Conf_{i_j}(t-1) \tag{4.9}$$

où $Conf_i(t)$ est la confiance dans le bloc B_i à l'instant t et $c_{i_j,i}$ est la criticité du bloc B_{i_j} vis-à-vis du bloc B_i. Telle qu'exprimée, la confiance est dépendante du temps et se propage sur les liens à chaque top d'horloge. Par conséquent, tant qu'un bloc n'a pas reçu d'information de la part de chacune de ses entrées, sa confiance ne peut être calculée. Nous la fixons cependant à 0 ce qui implique que tout bloc qui n'a ni reçu ni produit de l'information est considéré comme n'étant

pas de confiance.

En considérant des valeurs de criticités de même valeur en entrée d'un bloc, l'impact d'une source sur la confiance en un bloc est relatif au nombre de blocs fonctionnels qui les sépare. Plus ce nombre est grand, ou plus cette distance est forte, moins la source a d'influence. En effet, une source reliée directement à un bloc ayant m entrées a une criticité de $\frac{1}{m}$ tandis qu'une source alimentant un bloc à m entrée via un intermédiaire ayant l entrées aura une criticité (pour le bloc en bout de chaîne) de $\frac{1}{m \times l}$. Ainsi, plus un bloc est enfoui dans le système et moins les sources auront d'impact sur lui. Implicitement, nous avons donc défini un opérateur de concaténation $\otimes$ comme une multiplication : la criticité d'un bloc B_i pour un bloc B_j est le produit de toutes les criticités des blocs permettant de relier B_i à B_j. Ce phénomène est représenté sur la figure 4.9. Les différentes criticités des blocs sont notées sur les arêtes. La flèche en pointillés exprime la relation indirecte entre la source S_2 et le bloc T_2. Pour ce dernier, la source S_2 a un impact équivalent à une criticité de valeur $\frac{1}{4}$. Au contraire, la source S_3 a une criticité deux fois plus élevée due à sa proximité avec le bloc T_2.

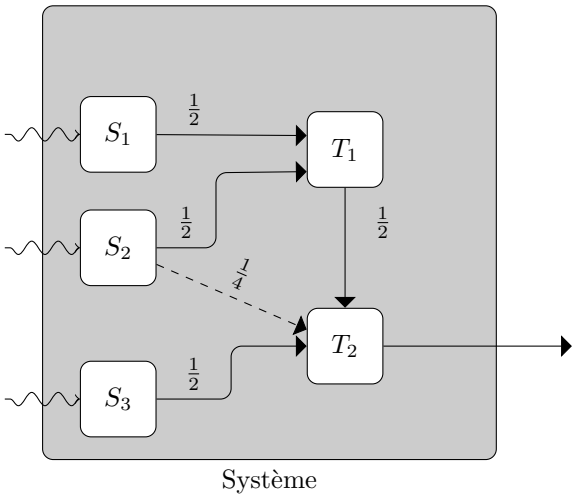


FIGURE 4.9 – Impact des informations transmises par une source au travers d'une chaîne

4.3.2 Propagation verticale

Au contraire de la propagation horizontale qui se propage entre les blocs, la propagation verticale permet d'évaluer la confiance dans une entité complexe à partir des confiances dans les blocs qui la constitue (cf. figure 4.6). Par exemple,

la confiance dans le système d'information d'un navire dépend des confiances dans les sous-systèmes de navigation et de propulsion qui dépendent elles-mêmes des confiances dans leurs capteurs, automates et logiciels qui les composent. Cette propagation est qualifiée de verticale car elle répand la confiance depuis les blocs vers un niveau macroscopique où le (sous-)système est perçu comme une entité unique.

Du point de vue de la relation d'entité, il n'existe pas d'ordre entre les blocs. En effet, telle que définie au chapitre 3, une entité est un ensemble de blocs. Cet ensemble ne possède pas de hiérarchie entre ses composants : un bloc n'appartient pas plus qu'un autre. Un opérateur de propagation basé sur la relation d'entité possède donc la propriété de **commutativité**. Par ailleurs, cet opérateur possède également la propriété de **croissance** : parmi tous les blocs constitutifs de l'entité, l'un d'entre eux a une confiance qui est minimale, c.-à-d. plus faible que celles des autres. Si la confiance de ce bloc augmente alors la confiance globale de l'entité fait de même.

Tout comme la propagation horizontale, nous proposons d'utiliser un opérateur d'agrégation pour propager la confiance vers l'entité cible. Soient $\mathcal{B}$ l'ensemble des blocs du système et $\mathcal{E} \subset \mathcal{P}(\mathcal{B})$ l'ensemble des entités ; une entité étant un ensemble de blocs par définition (cf. section 3.2). La confiance $Conf_E(t)$ dans une entité $E = \{B_1, \ldots, B_m\} \in \mathcal{E}$ à l'instant t se définit alors telle que

$$Conf_E(t) = \bigoplus_{B \in E} Conf_B(t) \tag{4.10}$$

$$= \frac{1}{m} \sum_{i=1}^{m} Conf_{B_i}(t) \tag{4.11}$$

De la même manière que pour la mesure de propagation horizontale, nous définissons cet opérateur comme une moyenne des confiances des blocs, cette mesure étant un candidat naturel pour l'opérateur d'agrégation (Jøsang et al., 2006b ; Richardson et al., 2003 ; Wang et Singh, 2006 ; Ries, 2007). Toutefois, dans ce cas, nous n'affectons pas de poids aux confiances des différents blocs, les blocs étant indistinguables au regard de la relation d'appartenance à une entité.

Il est à noter que, telle que définie, la confiance d'une entité ne peut être calculée que lorsque la confiance de chacun de ses éléments existe. Par conséquent, la confiance d'une entité ne peut être calculée qu'à partir de l'instant $t_{\min,E}$ où la confiance de chaque élément composant l'entité E existe. Avant cet instant, nous fixons la confiance de l'entité à 0.

4.4 Discussion

Nous avons proposé dans ce chapitre un ensemble de mesures permettant d'évaluer la confiance au sein d'un système d'information. Contrairement aux approches présentées au chapitre 2, notre proposition couvre l'ensemble des problématiques liées à la confiance (sa définition, sa mesure puis sa propagation), sans se limiter à un aspect particulier. La solution retenue, quoique simple, constitue donc davantage un socle destiné à évoluer qu'une solution définitive. Elle présente en particulier deux limites.

Nous avons choisi de limiter la confiance à la compétence et la sincérité des sources. Bien que nous considérons le risque comme étant omniprésent, l'ensemble de nos propositions pourraient tirer parti d'analyses des risques et menaces pouvant cibler un système d'information particulier. En effet, nous avons défini les criticités des différents blocs comme étant identiques, en l'absence de toute connaissance *a priori* concernant leur importance dans le traitement effectué par le récipiendaire des informations. L'identification des risques encourus par chaque bloc pourrait permettre une évaluation plus précise de leurs criticités respectives. En effet, une telle analyse révèle les biens considérés essentiels pour le bon fonctionnement du système ainsi que leurs vulnérabilités. Ces deux éléments (biens essentiels et vulnérabilités) sont deux facteurs pouvant influencer les criticités des blocs.

Notre modèle ne prend pas en compte les menaces internes au SI. À l'instar des informations, la confiance se propage depuis les sources vers l'ensemble des blocs fonctionnels. La confiance des blocs ne résulte pas uniquement de leurs entrées mais également de leurs sorties. En analysant les informations produites par les blocs fonctionnels au cœur du système d'information, les mesures pourraient davantage prendre en compte les spécificités des blocs fonctionnels (*e.g.* les traitements qu'ils réalisent).
De plus, considérer les informations en sortie des blocs situés au cœur du SI, permettrait d'adapter les répercussions d'une attaque sur les confiances des blocs. En effet, les variations des niveaux de confiance s'atténuent au fil des propagations successives. Au contraire, l'impact réel d'une altération des informations traitées et manipulées par les blocs est difficile à évaluer car il dépend des traitements effectués par les blocs. En particulier, un bloc qui produit une information complètement différente de ses entrées (*e.g.* un cap à partir d'une suite de positions) produit des effets non quantifiables sans connaissance de son fonctionnement interne. Une telle connaissance serait souhaitable pour une adéquation plus précise entre la confiance

du bloc et l'altération de ses informations. Afin de maintenir un niveau minimal de propagation de confiance dans un bloc supposé sain (c.-à-d. garantir qu'il sera impacté dans une certaine proportion par une attaque), des tests pourraient être menés afin d'évaluer, en fonction du minimum recherché, la profondeur maximale requise pour le système, c.-à-d. déterminer combien de blocs peut traverser une information émise par une source. Cette profondeur s'apprécie en fonction des différentes criticités mais également du nombre de relations entre les blocs. Plus ces relations sont nombreuses, plus les criticités seront faibles et moins la confiance se propage.

La mesure de sincérité, telle que nous l'avons définie, est une fonction dépendante du consensus entre les différentes sources d'information. Les informations se propagent d'un bloc à l'autre à chaque top d'horloge. Dans le cas général, deux blocs produisant des informations de même nature ne sont donc pas nécessairement synchronisés. Par exemple, une source produit des informations à partir de l'instant $t = 0$ tandis qu'un bloc de traitement produit des informations à partir d'un instant $t \geq 1$, celui-ci devant avoir été alimenté au préalable en informations. Pour comparer les informations, il faut donc tenir compte des décalages entre chaque bloc du système.

Conclusion

Pour évaluer la confiance dans un système d'information, nous avons proposé une définition basée sur les critères propres à la cible de la confiance : compétence et sincérité. Ces deux critères permettent de prendre en compte tant la capacité d'une source d'information à faire des erreurs que ses intentions. Les sources sont les points d'entrée du système : tout autre bloc fonctionnel reçoit des informations produites directement par les sources ou qui résultent d'un ou plusieurs traitements de leurs informations. Notre méthodologie d'évaluation de la confiance dans le système global s'appuie donc sur la mesure de la confiance dans les sources puis sa propagation selon deux axes. Dans un premier temps, nous avons proposé des mesures de compétence, de sincérité et de confiance. Tandis qu'une mesure de compétence est basée sur l'imprécision des mesures des capteurs, la sincérité d'une source est mesurée à partir de son degré d'accord avec les autres sources de même nature et observant le même phénomène. La confiance est ensuite déduite de ces deux mesures. Dans un second temps, nous avons présenté deux mesures de propagation de confiance : horizontale et verticale. Ces mesures permettent,

à partir de la mesure de confiance dans les sources d'information, d'évaluer les confiances dans les divers blocs constitutifs du système (propagation horizontale) puis celles dans les entités plus complexes composées de plusieurs blocs jusqu'au système lui-même.

Nous avons présenté dans ce chapitre notre proposition théorique. Dans le chapitre suivant, nous montrerons leur mise en pratique au travers de divers scénarios.

Chapitre 5

Simulations et expérimentations

Dans les chapitres précédents, nous avons proposé une modélisation de système d'information ainsi que des mesures de compétence, de sincérité et de confiance mais également de propagation de cette dernière au sein du système. Nous souhaitons désormais valider l'utilisation de ces mesures dans un cadre de détection d'attaques menaçant un système d'information. Pour cela, nous avons organisé ce dernier chapitre en trois parties.

Dans un premier temps, nous présentons la conception d'un simulateur permettant de gérer des scénarios d'attaques et intégrant les différentes métriques exposées au chapitre 4. Après avoir exposé l'aspect fonctionnel du logiciel, nous présentons les outils utilisés pour son implémentation. Dans un second temps, nous exposons divers scénarios considérés pour tester les différentes mesures puis les résultats que nous en avons tiré. Enfin, nous concluons par une discussion autour des résultats obtenus dans chacun des scénarios testés.

5.1 Conception d'un simulateur

Pour tester les mesures de confiance proposées au chapitre précédent, nous nous intéressons aux systèmes d'information des navires et en particulier à leur sous-système de navigation. Celui-ci est chargé d'assurer les déplacements du navire en collectant puis restituant des informations sur son environnement. Les déplacements peuvent être décidés soit automatiquement, via un programme de pilotage automatique, soit manuellement par le commandant de bord. L'ensemble des informations collectées par ce sous-système, dites « informations de naviga-

tion », est ensuite réutilisé non seulement par le système lui-même mais également par de nombreux autres sous-systèmes (*e.g.* système d'armes pour les navires militaires). Les systèmes de navigation assurent donc un rôle particulièrement critique dans la conduite du navire et c'est la raison pour laquelle nous les avons choisis comme support de notre étude.

Toutefois, pour des raisons pratiques, nous avons besoin de tester le fonctionnement d'un système de navigation dans différentes configurations (*e.g.* composants et attaques). Afin d'évaluer l'efficacité des mesures que nous avons proposées et de valider leur adéquation avec notre objectif de détection, nous avons conçu un simulateur permettant de concevoir divers scénarios et d'évaluer la confiance via les métriques du chapitre précédent.

5.1.1 Conception d'un système d'information de scénarisation de données et d'évaluation de la confiance

Nous présentons dans cette section la méthodologie suivie par notre simulateur pour scénariser des données réelles puis évaluer la confiance sur la base des données produites.

5.1.1.1 Méthodologie de scénarisation

Nous avons élaboré une méthodologie de scénarisation permettant de tester divers scénarios d'attaques. À partir d'un jeu de données réelles, nous souhaitons produire plusieurs **jeux de données scénarisées**, c'est-à-dire représentatifs de la mise en œuvre d'une, ou plusieurs, attaque(s) sur un système. Un scénario décrit donc l'attaque d'un système d'information. Il se compose de **données réelles**, d'un **modèle de système** ainsi que d'un ensemble d'**attaques**.

Tout d'abord, les données réelles sont la base du scénario : elles représentent sa réalité, c'est-à-dire l'information vraie qui doit être mesurée par les sources. Elles constituent une description de l'environnement du système. En effet, dans le cas d'un système de navigation, les informations qu'il manipule sont principalement produites sous la forme de mesures de capteurs. Elle ne sont pas générées en interne par le système comme le sont des adresses IP privées par exemple. En conséquence, la description d'un système ne suffit pas à définir un scénario : la description de l'environnement est également indispensable. Dans le cadre d'une simulation, cette description n'a toutefois pas besoin d'être exhaustive : seules les

informations réutilisées par le système sont nécessaires. Par exemple, si le système dispose de capteurs mesurant la température de l'eau ou la vitesse du vent, ces informations doivent faire partie du jeu initial afin de pouvoir être injectées dans le système. Au contraire, les informations inutilisées par le système sont superflues (*e.g.* la pression des pneus du vélo du capitaine).

Le second élément nécessaire à la mise en place d'un scénario est le modèle du système. En effet, il n'est pas toujours possible de procéder à l'extraction des données d'un système de navigation. La modélisation du système et des traitements qu'il applique à l'information est indispensable à la production d'un jeu de données reflétant son comportement.

Enfin, une ou plusieurs attaques peuvent altérer les informations, avant ou au cours de leur propagation dans le système. La description de ces attaques est donc essentielle dans la composition du scénario. Plus précisément, nous définissons une attaque à partir de 5 paramètres :

1. Cible : le bloc qui verra ses informations altérées.
2. Début de l'attaque : instant à partir duquel les informations sont falsifiées.
3. Durée de l'attaque
4. Type de l'attaque : offset, incrémentale ou rejeu.
5. Paramètre(s) spécifique(s) de l'attaque : pour une attaque par offset ou une attaque incrémentale, ce paramètre fixe la modification maximale des informations. Pour une attaque par rejeu, il faut préciser l'instant à partir duquel les informations doivent être enregistrées ainsi que la durée de la séquence mémorisée. Dans le cas où la séquence rejouée est plus courte que la durée de l'attaque, le rejeu se répète.

La procédure de scénarisation des données est présentée en figure 5.1. Dans l'encadré grisé se trouvent tous les éléments servant à la production de données scénarisées. La production de données réelles est assurée dans notre cas par le Système d'Identification Automatique (AIS). Ces données servent à la définition de scénarios, lesquels sont ensuite exécutés par un programme. L'exécution d'un scénario consiste à lire ses paramètres (modèle du système, attaques et données) et à produire le jeu de données correspondant. Les données, une fois scénarisées, peuvent être analysées pour mesurer puis propager la confiance dans le système.

Un programme de gestion de scénarios se décompose donc en deux parties :

- Stockage : Les différents paramètres disponibles (jeux de données, modèles de différents systèmes, attaques) sont stockés avant leur assemblage. Les

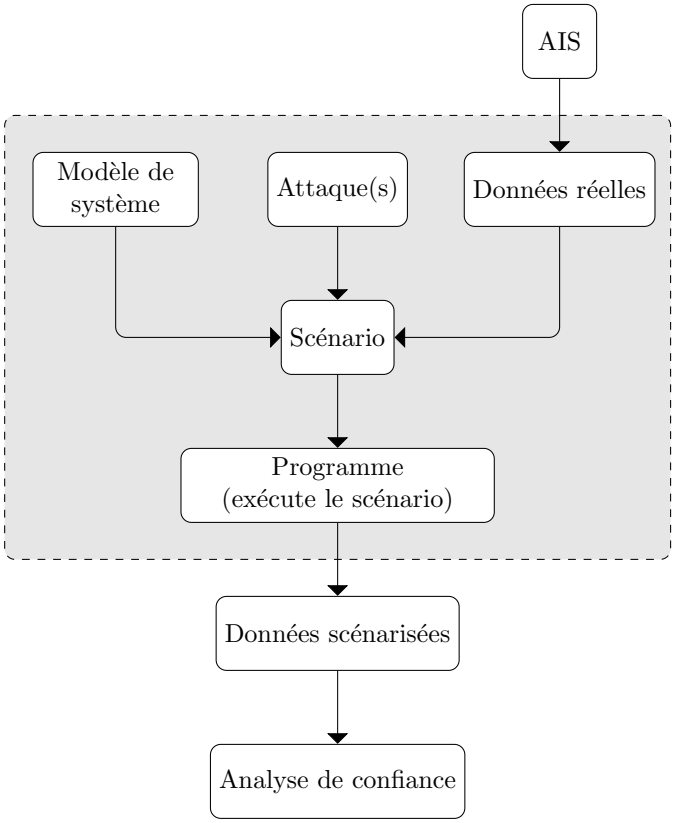


FIGURE 5.1 – Méthodologie de scénarisation des données

scénarios sont stockés sous la forme de pointeurs ou index vers les paramètres correspondant pour éviter la réplication des données d'origine : deux scénarios d'attaques différentes ciblant un même système ne nécessitent qu'une seule description de celui-ci. De plus, afin de pouvoir rejouer un scénario à l'identique et de séparer la gestion de scénarios de l'analyse de confiance, les données scénarisées sont également mémorisées.

- Exécution du scénario : pour générer un nouveau jeu de données scénarisées, le programme lit les paramètres du scénario à exécuter, va chercher les données nécessaires (*i.e.* données réelles, description du système et des attaques) et les traite en conséquence. Le résultat est conservé pour une analyse (de confiance) post-mortem.

La figure 5.2 décrit la chaine de traitement permettant la production d'une analyse de la confiance. Depuis des fichiers au format CSV, les données sont insérées dans une base de données. Les attaques et les modèles de SI mis à disposition sont insérés à la main. Nous nous sommes en effet limité à un nombre restreint

de systèmes et d'attaques, ce qui ne requiert pas l'automatisation du processus étant donné la forte contribution nécessaire de l'utilisateur pour la conception du scénario (modélisation du système, choix des paramètres d'attaques). Faciliter la gestion des attaques et des scénarios demanderait un investissement qui dépasse le cadre de notre travail.

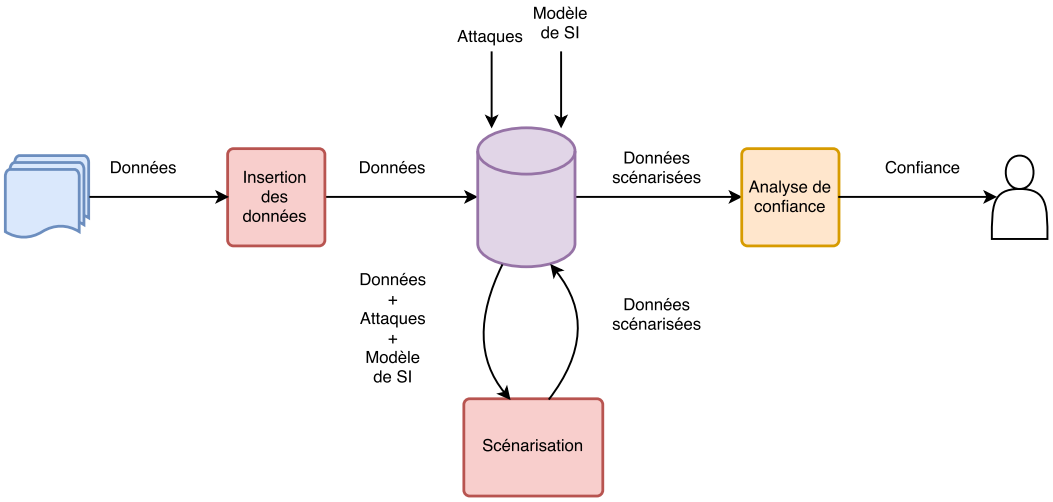


FIGURE 5.2 – Chaîne de traitement des données pour produire des indices de confiance

Dès lors que l'ensemble des informations servant à la conception de scénarios est mis à disposition dans une base, le processus de scénarisation les propose à l'utilisateur qui peut les combiner selon ses envies. Une fois ces éléments choisis, les requêtes adéquates permettent de récupérer les différentes informations (*i.e.* données, attaque(s) et modèle) et de scénariser les données. Enfin, les données scénarisées peuvent être soumises à une analyse de confiance, c'est-à-dire l'utilisation des différentes mesures présentées au chapitre 4. Finalement, les indices de confiance (*i.e.* les résultats des mesures) produits sont mis à la disposition de l'utilisateur.

5.1.1.2 Évaluation de la confiance

Une fois les données scénarisées, elle peuvent être analysées pour procéder à l'évaluation de la confiance. Cette évaluation est réalisée en trois phases.

Tout d'abord, le programme d'analyse doit demander à l'utilisateur de choisir

le nom du scénario qui devra être joué. À l'aide d'une requête, le programme va interroger la base et charger le scénario en mémoire. L'évaluation de confiance peut alors commencer en analysant les données produites par les sources. Les sources sont facilement identifiables sans connaissance préalable du système. En effet, dans notre contexte, le système est considéré vierge de toute information à son démarrage. Par conséquent, à l'instant $t = 0$, seules les sources peuvent émettre de l'information. Tout autre bloc fonctionnel doit en effet recevoir de l'information avant d'en émettre. Les données lues sont stockées sous la forme d'une matrice et l'évaluation se réalise en les comparant ligne par ligne, chaque ligne représentant un pas de temps. Cette phase évalue la confiance dans les sources à chaque pas de temps. Une fois cette confiance mesurée, elle peut être propagée dans les autres blocs.

La deuxième phase est la phase de propagation horizontale. À partir de la confiance dans les sources, il devient possible de mesurer la confiance dans les autres blocs par agrégation telle que définie au chapitre 4. Pour cela, le programme cherche dans quels blocs la confiance peut être propagée. Plus clairement, il détermine quels sont les blocs qui ont pour antécédents des sources, celles-ci étant les seuls blocs disposant d'un indice de confiance connu à l'instant $t = 0$. Il itère ainsi pour chaque pas de temps. A la fin de cette phase, chaque bloc dispose d'un indice de confiance pour chaque pas de temps où il interagit avec une information.

Durant la dernière phase, la confiance se propage verticalement. Les entités sont définies dans la description du système. Le programme interroge donc la base pour connaître le système sur lequel est basé le scénario courant. Dès lors qu'il dispose de la liste des entités et de leur composition, il peut agréger les confiances des blocs concernés, leurs confiances étant connues.

5.1.2 Implémentation

Nous avons développé un logiciel de gestion de scénarios pour automatiser la production de jeux de données simulant un système d'information (attaqué ou non) à partir de données réelles. Ce logiciel est composé de deux parties : la première sert à stocker les différentes informations relatives aux scénarios et la seconde est chargée d'exécuter un scénario choisi pour produire les données correspondantes. Nous détaillons ci-dessous nos choix d'implémentation.

Pour stocker les différentes informations servant à l'élaboration des scénarios, nous nous sommes tournés vers une base de donnée *PostgreSQL* pour ses capacités

de manipulation des données spatiales. Dans une perspective de visualisation des données, celle-ci nous apparaît en effet importante au regard de la prédominance de l'information géographique dans les études liées à la navigation des navires.

Dans une base de données, nous avons stocké les différents éléments qui constituent nos scénarios. Les jeux de données réelles contiennent des informations de navigation telles que la vitesse ou le cap du navire. Les informations contenues dans un jeu sont dépendantes de celles émises par l'AIS du navire étudié. En ce qui nous concerne, celles-ci seront détaillées dans la section 5.2.1.1. Les informations disponibles étant variables et les jeux indépendants entre eux (ils ne représentent pas nécessairement un même navire), chaque jeu est stocké dans une table différente. Toutes les tables sont rassemblées dans un même schéma.

Les attaques sont toutes décrites dans une même table dont les champs sont ceux que nous avons décrits dans la section précédente : cible, début, durée, type et paramètre spécifique de l'attaque (cf. section 5.1.1.1).

Pour stocker le modèle d'un système d'information, nous avons séparé les éléments en deux parties : d'un côté l'ensemble des blocs et leurs propriétés respectives, de l'autre chaque système comme un ensemble de relations. Concernant les blocs fonctionnels, ceux-ci sont listés dans une unique table qui leur associe un identifiant numérique unique, un nom (équivalent textuel de l'identifiant numérique) permettant à un opérateur humain de les reconnaître, leur précision ainsi que leur type qui est la nature de l'information qu'ils produisent, celle-ci pouvant être nulle pour les récepteurs. Les blocs étant centralisés dans une unique table, un système particulier se décrit sur la base de ses relations. Pour modéliser un système, nous listons toutes ses relations répertoriées selon 6 critères : un identifiant numérique unique, le type de la relation (cf. section 3.2.2), son sens (uni ou bidirectionnelle), son origine, sa destination ainsi qu'une valuation (*e.g.* la criticité du bloc origine). La structure des tables décrivant des blocs et un système de navigation est présentée en figure 5.3.

Pour exécuter les scénarios et opérer une analyse de la confiance des données, nous avons utilisé le langage Python. Celui-ci dispose de nombreuses bibliothèques logicielles permettant de manipuler les données et permet, de plus, de s'abstraire des problématiques liées à la gestion de la mémoire. Parmi les modules à disposition, nous nous sommes appuyé sur *psycopg2* pour interagir avec la base de données, en particulier pour les phases de scénarisation, le module *pandas* pour manipuler les données et procéder aux évaluations de confiance et le module *matplotlib* pour produire les graphiques. Chacun de ces modules est utilisé dans sa

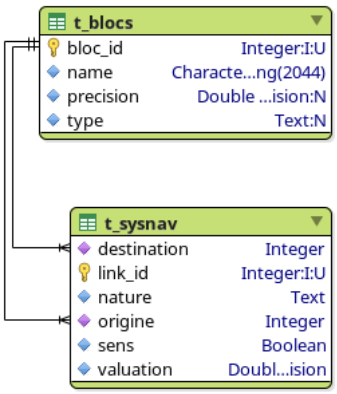


FIGURE 5.3 – Diagramme de structure

dernière version stable.

Au lancement du programme de scénarisation, l'utilisateur peut choisir soit d'intégrer des données dans la base à partir d'un fichier soit de concevoir un nouveau scénario.

Une fois les données insérées dans la base, l'utilisateur peut créer un scénario. Il doit pour cela lui choisir un nom qui servira à le référencer dans la base de données. Ensuite, il peut choisir les éléments qu'il souhaite assembler pour constituer son scénario parmi la liste de ceux déjà disponibles dans la base. Un exemple est donné en figure 5.4.

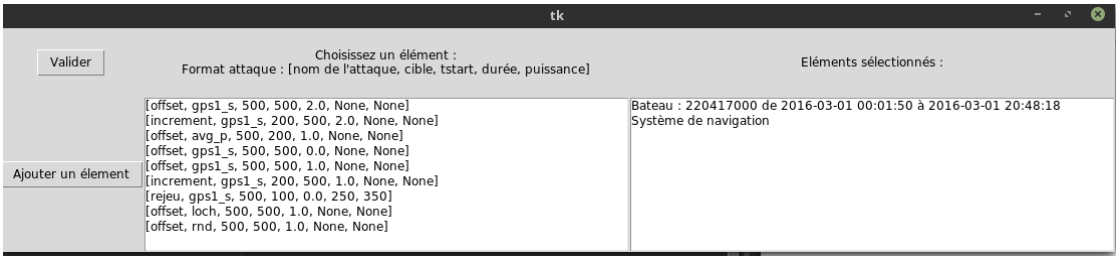


FIGURE 5.4 – Écran de construction du scénario : choix de l'attaque

5.2 Expérimentations

Afin d'expérimenter nos différents modèles, nous nous sommes appuyés sur différents scénarios. Nous détaillons dans cette section les scénarios particuliers

ayant servi de base à nos expérimentations avant de présenter nos résultats.

5.2.1 Scénarios

Nous avons construits des scénarios à partir de jeux de données réelles, de systèmes d'informations modélisés et d'attaques. Nous explicitons ci-après les différents éléments considérés dans notre contexte.

5.2.1.1 Données utilisées

Les scénarios sur lesquels nous avons basé nos expérimentations sont construits à partir de données originales obtenues via l'AIS. Ces données concernent un navire de type cargo, choisi pour sa vitesse stable. En effet, l'AIS envoie des messages à des intervalles de temps différents selon la vitesse du navire. Plus la vitesse est élevée et plus la fréquence d'émission des messages l'est également, ceci afin de limiter les collisions. Ainsi, un navire ayant une vitesse stable envoie donc des messages à une fréquence régulière ce qui correspond à notre modèle.

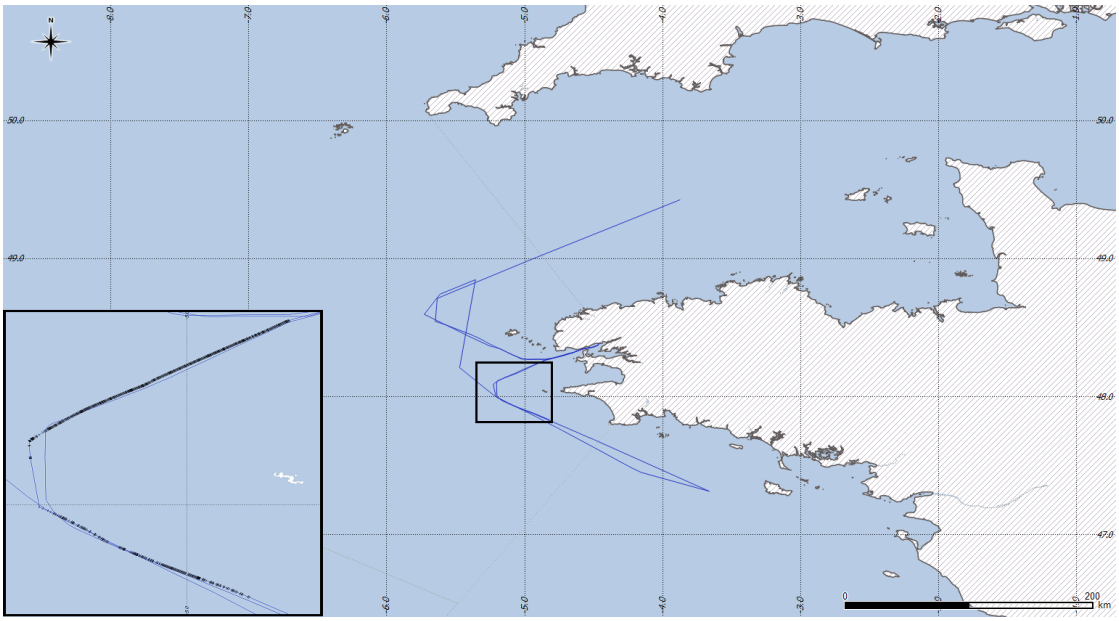


FIGURE 5.5 – Trajectoire et positions du navire étudié

Nous avons sélectionnés des données de navigation du porte-conteneur iden-

tifié par le *Maritime Mobile Service Identity* (MMSI) 636092340 dans la nuit du 21 au 22 mars 2015 lors de son passage, au large de Brest (voir figure 5.5). Parmi les informations disponibles se trouvent le cap, la vitesse, les coordonnées (latitude et longitude) géographiques, le temps, le taux de virage et la route du navire (voir figure 5.6).

	mmsi	speed	cog	heading	st_x	st_y	navigational_status	rot	time
1	636092340	15	302	144	-4.81885194778442	47.8292617797852	Under Way	-15	2015-03-21 22:38:53
2	636092340	12.7	304	303	-4.91112804412842	47.8690719604492	Under Way	8	2015-03-21 23:00:14
3	636092340	12.7	301	302	-4.91262006759644	47.869686126709	Under Way	-12	2015-03-21 23:00:33
4	636092340	12.7	301	303	-4.91329145431519	47.8699531555176	Under Way	11	2015-03-21 23:00:43
5	636092340	12.7	301	302	-4.92085027694702	47.8730506896973	Under Way	-11	2015-03-21 23:02:24
6	636092340	12.7	301	302	-4.92152833938599	47.8733139038086	Under Way	-8	2015-03-21 23:02:34
7	636092340	12.7	301	302	-4.92227840423584	47.8736228942871	Under Way	13	2015-03-21 23:02:43
8	636092340	12.8	301	302	-4.92386674880981	47.8742561340332	Under Way	-9	2015-03-21 23:03:04
9	636092340	12.8	301	302	-4.92454195022583	47.8745307922363	Under Way	9	2015-03-21 23:03:13
10	636092340	12.8	301	302	-4.9283766746521	47.8761138916016	Under Way	15	2015-03-21 23:04:04

FIGURE 5.6 – Extrait des informations contenues dans la base de données

Cette trajectoire, représentative de celles observées au large de Brest, est extraite d'un ensemble de données contenant 9623 enregistrements qui couvrent la période allant du 21 mars 2015 à 22 heures 38 minutes et 53 secondes au 29 du même mois à 7 heures 38 minutes et 45 secondes. Sur cette période, le navire avance à une vitesse moyenne de 12 nœuds pour une vitesse maximale de 18 nœuds et un minimum de 0 lorsqu'il est au mouillage.

5.2.1.2 Systèmes étudiés

Pour répondre à des besoins différents, nous présentons dans cette section deux catégories distinctes de scénarios, chacune basée sur un système différent. Nous nous limitons à ces deux systèmes mais les résultats sont transposables à tout autre système d'information. Notre choix de séparation des divers scénarios exposés dans la section suivante repose sur les éléments ciblés par nos attaquants fictifs. Dans les scénarios de la première catégorie, la cible est toujours la même afin de tester ses réactions face à des attaques différentes (voir section 3.4). Au contraire, les scénarios de la seconde catégorie reproduisent une même attaque sur divers composants du système étudié pour comparer les impacts des attaques en fonction du choix de la cible. Les systèmes choisis dans ces deux catégories de scénarios sont explicités ci-après.

Scénarios à cible constante La première catégorie de scénarios est composée de trois entités productrices d'informations : deux GPS et un Loch Doppler (voir figure 5.7). Ces producteurs d'informations peuvent se retrouver embarqués sur des navires tels que des navires de croisière par exemple. Dans notre contexte expérimental, les deux GPS sont situés respectivement à l'avant et à l'arrière du navire et le Loch Doppler en son milieu. Si un GPS est constitué de trois sources donnant trois types distincts d'informations : la position, la vitesse et le cap, un Loch Doppler ne comporte qu'une seule source : la vitesse. En effet, un Loch Doppler mesure la vitesse du navire par rapport au fond en utilisant un signal ultrasonore. Ces trois sources envoient leurs informations au système de cartographie électronique *Electronic Chart and Display* (ECDIS) qui les restitue visuellement à un opérateur humain.

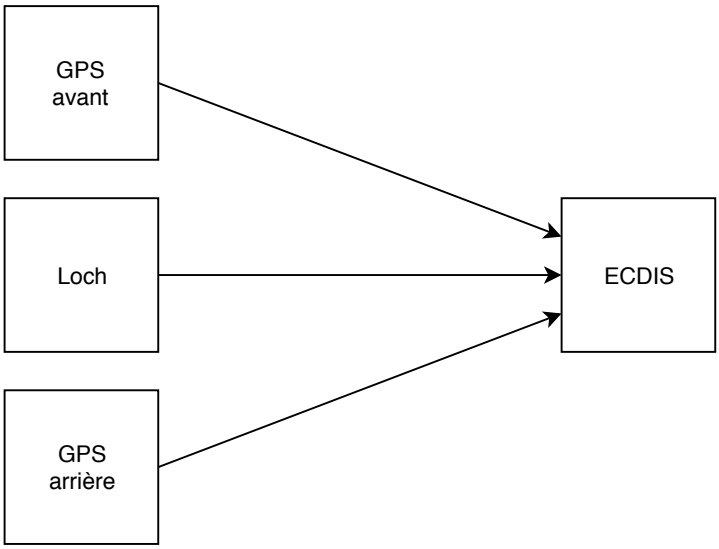


FIGURE 5.7 – Modèle de système utilisant 3 sources de vitesse employé dans les simulations

Pour les besoins de l'expérience, nous avons choisi de fixer le nombre de sources à 3. En effet, pour être mesurée, la sincérité nécessite de la redondance, c'est-à-dire de disposer de plusieurs sources émettant une même information. De plus, les navires ne disposent pas souvent, en pratique, de beaucoup de capteurs émettant une même information. Cependant, sur des bateaux d'une certaine taille (tels que des navires de croisière), cela est plus fréquent. Simuler 3 sources constitue donc un bon compromis : un nombre de capteurs qui est réaliste et qui permet

de tester les mesures élaborées dans cet article en exploitant la redondance des informations. Pour simuler les 3 sources, un bruit gaussien centré a été ajouté suivant le modèle décrit au chapitre 3.

Scénarios à cible variable La seconde catégorie de scénarios concerne ceux basés sur un exemple de système de navigation. Ce système, plus complet, permettra d'expérimenter les mesures de propagation. Il se compose de plusieurs blocs intermédiaires (de traitement ou de rétroaction) ainsi que d'un récepteur. Pour les blocs de traitement, nous avons choisi d'utiliser des blocs AVG qui calculent la moyenne de leurs entrées ainsi qu'un bloc RND qui renvoie aléatoirement une de ses entrées. Son comportement étant inconnu *a priori*, ce bloc nous permet d'expérimenter notre hypothèse de boîte noire tout en introduisant un élément non déterministe dans le système. Nous avons choisi d'intégrer dans notre système une centrale inertielle ; le bloc CI sur la figure 5.8. Une centrale inertielle évalue la position courante en calculant le différentiel par rapport à une position de référence qui est la dernière position mesurée. Elle dispose donc d'une mémoire modélisée sous la forme d'une rétroaction. La mesure de position par une centrale inertielle n'est pas très précise, les erreurs de calcul se cumulant à chaque itération. Par conséquent, afin de garantir une précision maximale, sa position de référence est remplacée par la position moyenne des deux GPS, lesquels ont une précision bien plus élevée. Ainsi, en cas de défaillance des GPS, la centrale inertielle calcule la position courante à partir de la dernière position mesurée par les GPS. Ce procédé de recalage constant de la centrale à partir des informations fournies par les GPS permet de limiter l'accumulation d'erreurs dans la mesure.

Sur la figure 5.8, les informations circulent via les arêtes. Les types des informations sont précisés par des acronymes : Pos (la position du navire), SOG (*Speed Over Ground*, la vitesse du navire), Cap (le cap suivi par le navire). À l'instar de la première catégorie de scénarios, l'ensemble des informations est réceptionné par un système ECDIS.

Modèle de malveillance Dans chacun des scénarios, quelque soit sa catégorie, nous souhaitons simuler une attaque qui falsifie la vitesse du navire. En effet, un attaquant peut vouloir envoyer de fausses informations de vitesse pour ralentir le navire (*e.g.* pour faciliter son interception par des pirates) ou bien le faire accélérer (*e.g.* surconsommation, usure prématurée du moteur ou de la ligne d'arbre).

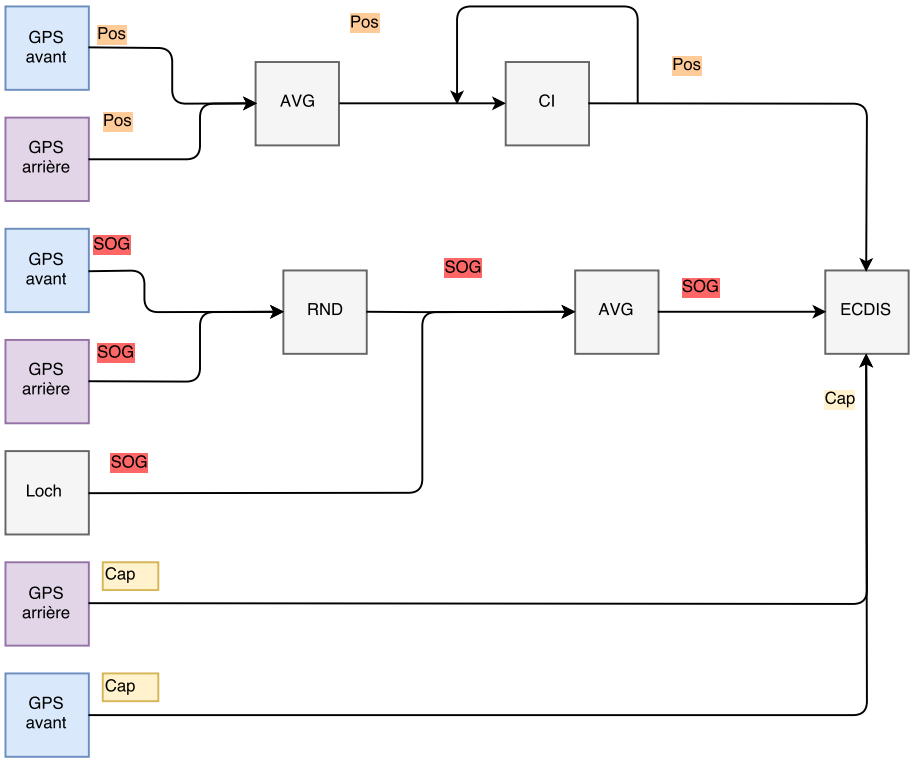


FIGURE 5.8 – Modèle de système de navigation employé dans les simulations

5.2.2 Résultats

Nous exposons ci-dessous plusieurs scénarios mettant en scène les différentes attaques évoquées au chapitre 4 et ciblant les systèmes présentés dans la section précédente. La première série de scénarios montre qu'il est possible d'utiliser la confiance comme outil de détection. Ensuite, la seconde série montre les conséquences sur la propagation au sein du système. Chaque scénario présenté montre l'évolution de différentes mesures de confiance. Pour des raisons de lisibilité, les mesures de sincérité et de compétence ne sont pas représentées. La compétence étant considérée constante au cours du scénario, la variabilité de la confiance dépend seulement de la variabilité de la sincérité.

L'ensemble des scénarios présentés dans cette section vise à valider les fonctions choisies ainsi que l'ensemble de notre démarche. Pour cela, nous avons testé différentes configurations récapitulées par le tableau 5.1. Ces scénarios peuvent

se décomposer en deux catégories : les scénarios à cible constante et ceux à cible variable.

Cible(s) / Attaque(s)	3 sources	Système de navigation	
		Source	Composant interne
Offset	5.9	5.13, 5.14, 5.16	5.15
Incrémentale	5.10		
Rejeu	5.11		

TABLE 5.1 – Tableau récapitulatif des scénarios testés

Les scénarios à cible constante (figures 5.9, 5.10 et 5.11) visent à étudier le comportement des mesures de confiance en fonction du type d'attaque. Au contraire, les scénarios à cible variable (figures 5.13, 5.14, 5.15 et 5.16) consistent à fixer une attaque et étudier différentes alternatives de choix de cible.

5.2.2.1 Scénarios d'attaques à cible constante

La figure 5.9 montre le comportement des différentes mesures de confiance en simulant les trois sources de vitesse des producteurs à partir des données de vitesse produites par l'AIS. Les trois bruits gaussiens sont de variance identique, tirée des spécifications des constructeurs. Le bruit d'une source $i \in \{1, 2, 3\}$ est de variance $\sigma_i = 0.1$ ce qui donne une mesure de compétence $Comp_i = \frac{1}{1+\sigma_i} \approx 0.91$. Pour simuler une attaque de leurrage, le GPS avant émet de fausses informations à partir de l'instant $t = 500$; instant à partir duquel la vitesse transmise est supérieure de 1 nœud à la vitesse réelle.

La première ligne de courbes en figure 5.9 montre la vitesse telle que perçue par chaque source avec une précision d'environ 0.1 nœuds (spécifications constructeur). En particulier, la première courbe met en évidence la falsification des informations transmises par le GPS avant.
La dernière ligne montre l'évolution de différentes mesures de la confiance de chacune des sources au fil du temps. Nous constatons l'impact de l'attaque du GPS avant au niveau de chacune des mesures de confiance des trois sources. Dès que ce dernier indique une vitesse différente de celle mesurée par l'autre GPS et le Loch Doppler, la mesure de sa confiance tend à la baisse et de manière plus forte que les mesures de confiance des deux autres sources. D'autre part, à partir de la 1000^{e} information reçue, l'attaque prend fin et la confiance augmente. Les mesures ont

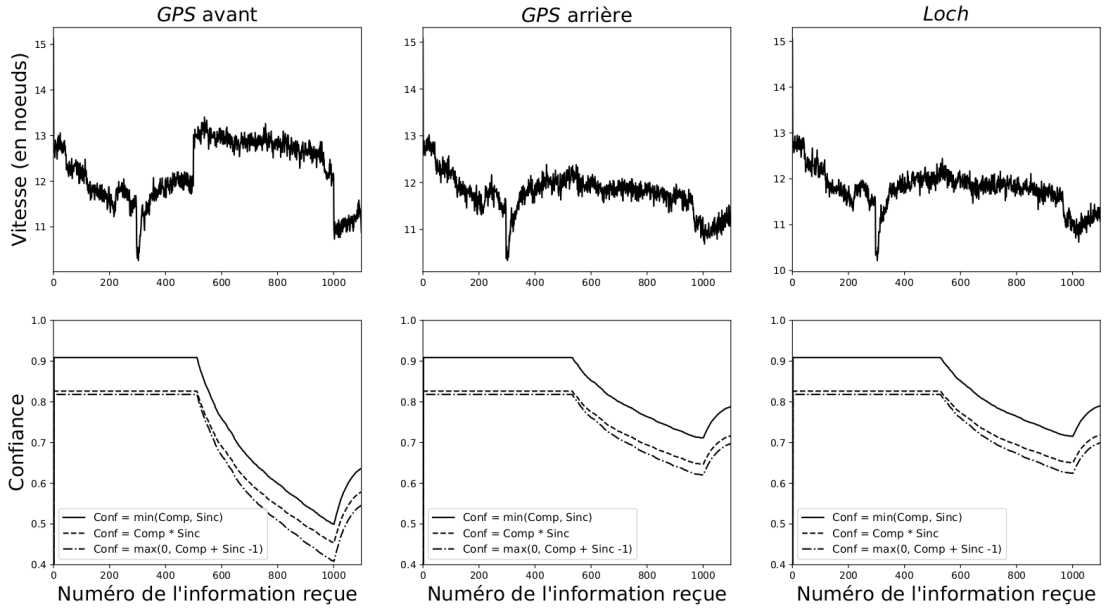


FIGURE 5.9 – Évolution de trois mesures de confiance en présence de trois sources de vitesse sur un navire

donc réagi à l'attaque de manière appropriée : une baisse de confiance est enregistrée lorsque le système est attaqué et un retour à la normale lorsqu'elle prend fin.

Comme Bhatti et Humphreys l'ont démontré, il est possible d'avoir une falsification incrémentale de l'information (BHATTI et HUMPHREYS, 2017), ceci afin de masquer l'attaque. Sur la figure 5.10, nous avons rejoué le scénario en modifiant le comportement de l'attaquant. Le but étant de tester le comportement des mesures de compétence, de sincérité et de confiance, selon différents niveaux de complexité d'attaques. Une nouvelle attaque de leurrage est simulée, le GPS avant émet de fausses informations à partir de l'instant $t = 200$. À partir de cet instant, la vitesse transmise par le GPS avant augmente progressivement jusqu'à être supérieure de strictement 1 nœud à la vitesse réelle à l'instant $t = 1000$.

Quelque soit la mesure employée, la confiance du GPS avant est la plus faible. L'attaque n'est cependant pas détectée dès le moment où elle survient. En effet, les informations sont falsifiées à partir de l'instant $t = 200$ mais la confiance commence à varier à partir de l'instant $t = 537$. En falsifiant les informations progressivement, sans changement important, il est donc possible de modifier et de contrôler la vitesse du navire. Afin de réussir et donc ne pas être détectée, cette

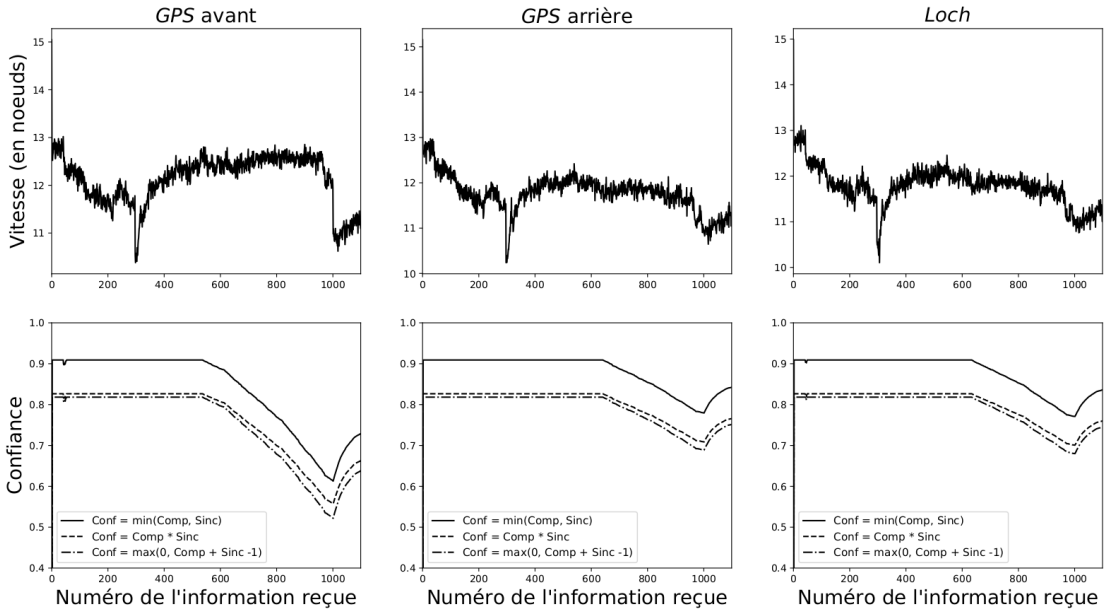


FIGURE 5.10 – Évolution de trois mesures de confiance en présence d'une falsification incrémentale des informations d'une source parmi trois

attaque nécessite néanmoins d'être exécutée plus longtemps qu'une attaque par offset.

D'autres types d'attaques utilisant la manipulation d'informations peuvent affecter un système d'information. Par exemple, les *attaques par rejeu* consistent pour un attaquant à répéter une information déjà émise par une source. Ce type d'attaque a généralement pour but de corrompre le SI, sans nécessiter de lever la confidentialité des échanges. Les informations peuvent en effet être chiffrées et donc incompréhensibles pour un attaquant. Cependant, celui-ci peut les enregistrer (sous forme chiffrée) et les ré-émettre ultérieurement dans le but de perturber le bon fonctionnement du système.

Dans le cas d'un navire, une telle attaque peut répéter une séquence ayant pour but de stopper le navire ou bien de le faire accélérer lors d'un accostage. Dans ce dernier cas, le navire nécessite de manœuvrer à faible vitesse. Rejouer une séquence d'informations, enregistrées lors d'une phase d'accélération (*e.g.* au moment de quitter un port), conduit donc à un comportement qui est dangereux dans une telle situation.

Sur la figure 5.11, le Loch Doppler ré-émet une séquence d'informations (entre $t = 250$ et $t = 350$) à partir de l'instant $t = 500$. Cette séquence comprend un

ralentissement brutal du navire. Comme pour la première attaque présentée, la confiance de la source attaquée diminue au moment de la malversation.

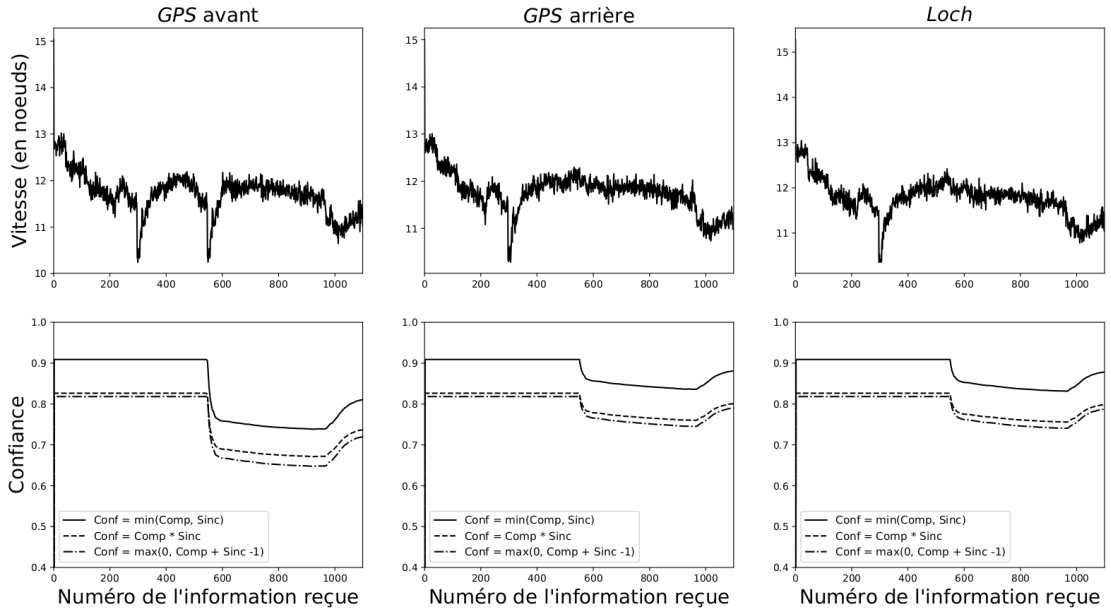


FIGURE 5.11 – Évolution des mesures de confiance en présence d'un rejeu des informations d'une source parmi trois

Enfin, nous proposons d'étudier le comportement du système lorsque deux sources sont attaquées. Nous avons repris l'attaque par offset présentée plus haut (cf. figure 5.9) mais ciblant cette fois-ci le Loch Doppler et une falsification incrémentale ciblant le GPS situé à l'avant du navire. Les résultats sont présentés sur la figure 5.12.

Le GPS est le premier affecté à partir de l'instant $t = 200$ et ce jusqu'à $t = 700$. À partir de $t = 500$ c'est le Loch Doppler qui est attaqué à son tour. Sa vitesse est alors augmentée de 1 nœud au-dessus de sa vitesse réelle. Avant l'attaque du Loch, la confiance du GPS avant est la seule à varier. Elle diminue à partir de l'instant $t = 420$. Elle est détectée plus vite que l'attaque incrémentale montrée précédemment car elle dure moins longtemps et donc les augmentations successives de la vitesse sont plus importantes. Lorsque les informations du Loch sont altérées, les confiances de chacune des sources décroissent. Entre les instants $t = 500$ et $t = 700$, la confiance du GPS avant décroît plus lentement que cette même confiance à partir de $t = 700$. En effet, les deux attaques mises en œuvre ciblent le même objectif : une augmentation de la vitesse de 1 nœud au-dessus

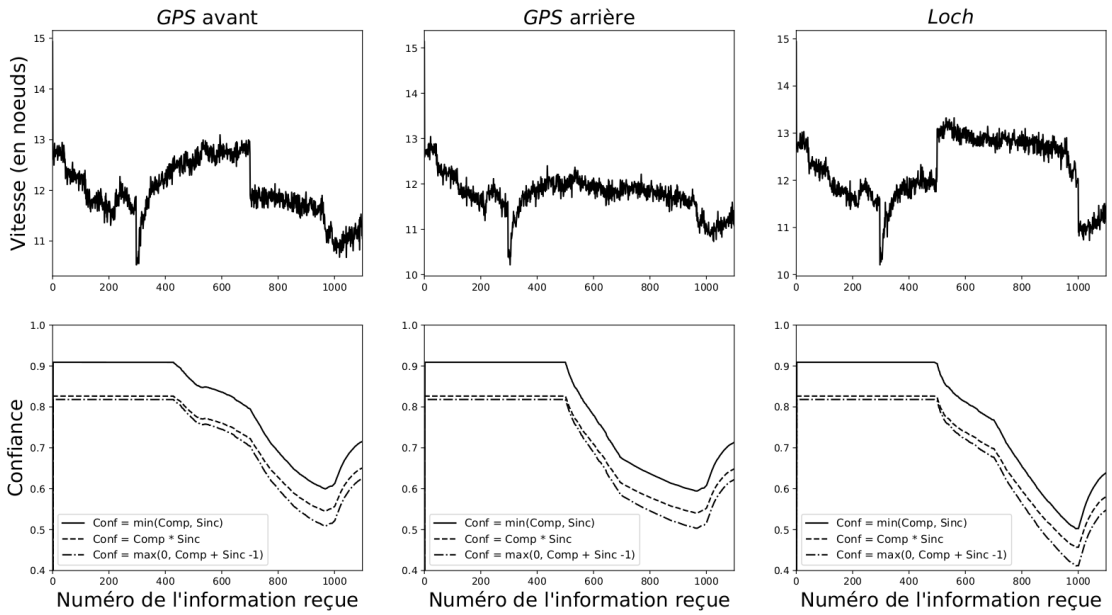


FIGURE 5.12 – Évolution des mesures de confiance en présence de deux attaques sur deux sources différentes

de la vitesse réelle. Pendant l'intervalle de temps où les informations des deux sources sont falsifiées, leurs informations sont donc davantage similaires qu'elles ne le sont avec celle du GPS arrière, lequel est sain. Nous observons par ailleurs le phénomène inverse pour la confiance de ce dernier : sa confiance décroît plus lentement à partir de $t = 700$. En fait, entre les instants $t = 535$ et $t = 700$, la confiance de la source saine (le GPS arrière) est plus faible que celles des deux autres sources attaquées. Ainsi des attaques différentes ayant un objectif commun peuvent diminuer la confiance d'une source saine. En effet, dans notre scénario à trois sources, la source saine est en minorité lorsque les deux autres sont attaquées, celles-ci étant au moins partiellement en accord, du fait de leur objectif commun.

De ces premières expérimentations, nous pouvons déjà conclure que quels que soient les scénarios, les mesures de confiance testées réagissent de la même façon. Afin de ne pas surcharger les graphiques, nous n'utiliserons donc qu'une seule mesure pour la seconde série de tests : la multiplication. Celle-ci est en effet la mesure la plus couramment rencontrée à travers la littérature.

5.2.2.2 Scénarios d'attaque à cibles variables

La figure 5.13 montre les multiples informations de vitesse manipulées par le système de navigation (présenté en section 5.2.1) ainsi que la confiance dans chacun des blocs impliqués. À partir de $t = 500$, la vitesse émise par le GPS avant a subi une augmentation de 2 nœuds par rapport à la vitesse réelle (première courbe de la première ligne). Nous avons ici doublé la puissance de l'attaque pour mieux en observer la propagation. En effet, la modification se répercute sur les informations fournies par les blocs RND et AVG comme le montrent les deux dernières courbes. La forme particulière de ces courbes provient du bloc RND : les informations qu'il émet sont altérées ou pas selon l'entrée choisie, ce qui provoque les oscillations. Le même phénomène se produit pour le bloc AVG.

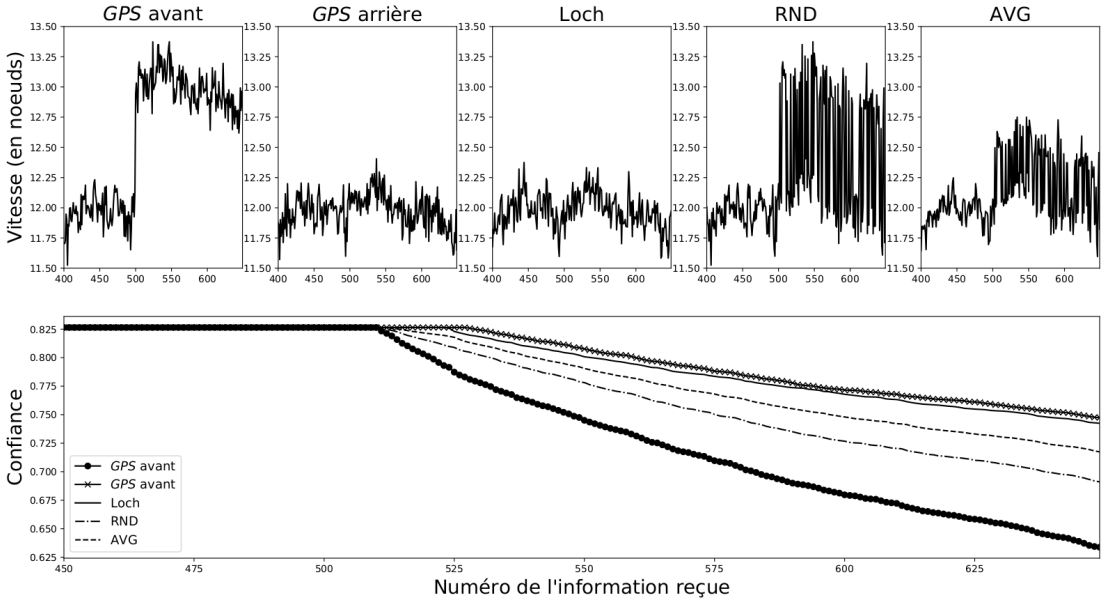


FIGURE 5.13 – Propagation horizontale en présence d'une attaque sur un GPS

La seconde ligne comporte un unique graphique représentant les mesures de confiance pour chaque bloc. La confiance dans le GPS avant, la cible de l'attaque, est la plus faible. Les confiances dans les autres blocs sont ensuite affectées selon l'impact de l'attaque. Le bloc RND reçoit ses informations directement du GPS avant, sa confiance est la plus faible après celle de ce dernier. Le bloc AVG qui reçoit ses informations du bloc RND est ensuite affecté de la même façon. Les deux autres sources (le *Loch* et le GPS arrière) ont les confiances les plus élevées puisqu'elles ne sont pas affectées par l'attaque.

Le deuxième scénario expérimenté augmente, comme le précédent, la vitesse de 2 nœuds par rapport à la vitesse réelle à partir de l'instant $t = 500$. En revanche, la cible de l'attaque est cette fois-ci le *LochDoppler*. Le *Loch* est relié directement au bloc AVG. La confiance dans le bloc AVG doit donc être inférieure dans ce scénario par rapport à celle dans le scénario précédent. De plus, l'attaque modifie significativement le comportement de ce bloc. Au contraire du scénario précédent où il recevait deux informations correctes de temps en temps, selon ce que lui envoyait le bloc RND, le bloc AVG reçoit dans ce cas deux informations dont l'une est systématiquement falsifiée (lorsque l'attaque a cours). L'attaque n'a donc pas le même impact selon la cible visée comme le montrent les 5e courbes des figures 5.13 et 5.14.

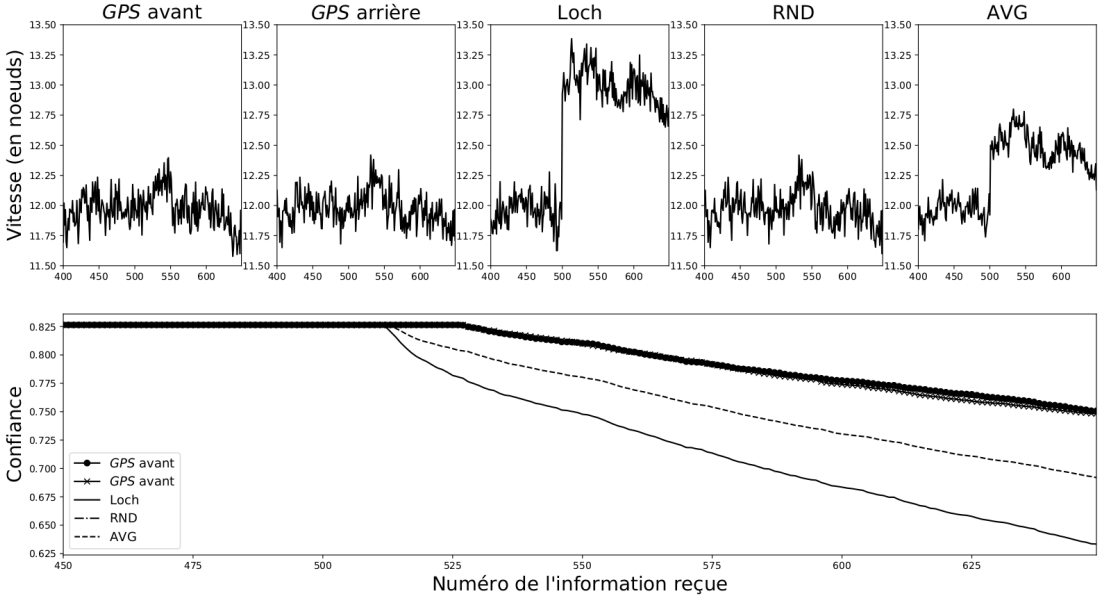


FIGURE 5.14 – Propagation horizontale en présence d'une attaque sur un Loch

La figure 5.14 montre les évolutions de la vitesse dans le cas d'une attaque visant le Loch Doppler ainsi que les confiances associées aux blocs considérés. Comme le montrent les courbes du graphique en deuxième ligne, la confiance dans le loch diminue, à l'instar des exemples précédents, plus rapidement que celles des autres. La confiance du bloc AVG est ensuite affectée de la même façon. Les fonctions réagissent comme dans l'exemple précédent. Toutefois, la confiance dans le bloc AVG, bien qu'elle soit affectée dans les deux cas (i.e. attaque du GPS avant et attaque du *Loch*), n'a pas la même valeur : $0,67$ à $t = 650$ contre $0,7$ au même instant dans la configuration précédente. Du fait de la proximité des deux

blocs, l'attaque s'est répercutée plus fortement sur le bloc AVG.

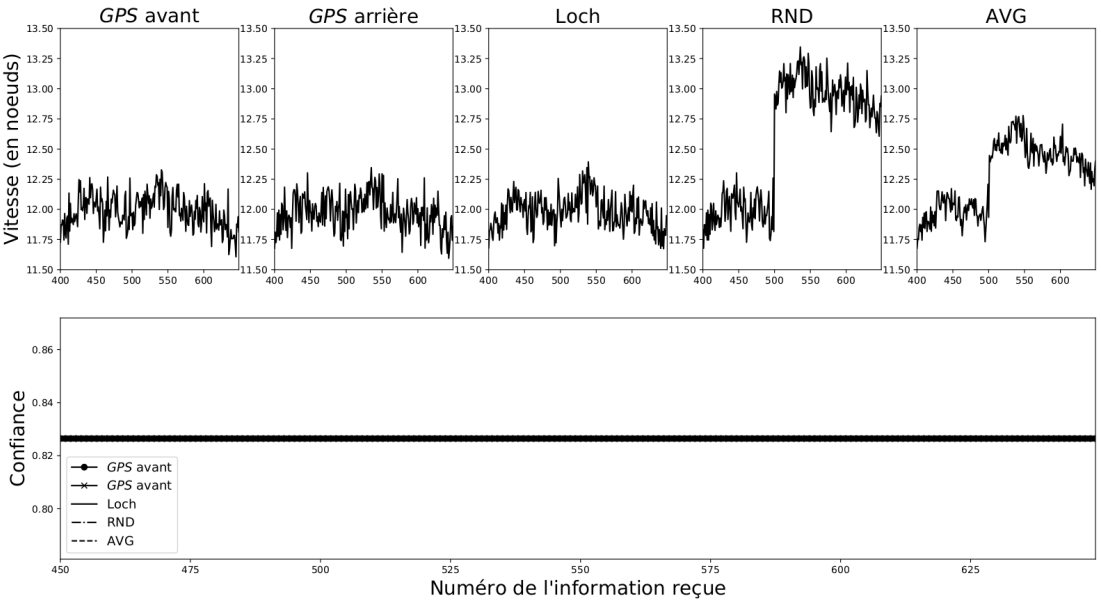


FIGURE 5.15 – Propagation horizontale en présence d'une attaque sur composant interne du SI

Le troisième scénario illustre le cas d'un bloc ciblé par une attaque et non une source. Contrairement aux scénarios présentés précédemment, la confiance ne varie pas dans ce cas comme illustré en figure **??**. En effet, seule les confiances des sources sont mesurées, les autres étant propagées. De ce fait, une modification, en interne, de l'information n'est pas détectée par notre analyse.

La figure 5.16 montre les différentes confiances des sources constitutives d'un GPS dont l'une est attaquée. Ici, la source de vitesse subit une attaque par offset (augmentation de la vitesse de 1 nœud au-dessus de la valeur réelle). Les résultats sont similaires à une propagation horizontale dont une entrée est attaquée : la confiance du GPS est inférieure à celle de ses entrées saines mais supérieure à celle du bloc ciblé.

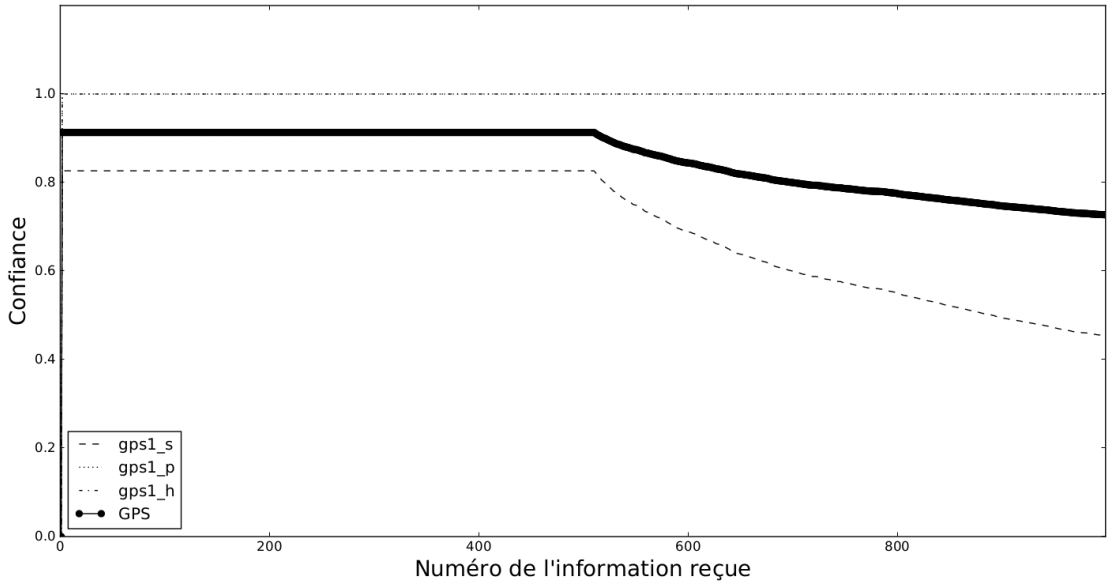


FIGURE 5.16 – Propagation verticale de confiance dans un GPS

5.3 Discussion

Dans la section précédente, nous avons étudié et présenté les réactions des diverses mesures choisies. Les résultats obtenus valident notre approche pour plusieurs raisons que nous explicitons ci-après. Tout d'abord, nous avons constaté que les fonctions testées réagissent bien à une attaque : la confiance mesurée est plus faible lorsque la source est attaquée. Une modification de l'information est donc détectable par nos mesures.

D'autre part, la confiance est d'autant plus faible que l'attaque est puissante. Ceci est également dans la logique d'un processus de détection : moins l'attaquant cherche à se dissimuler et plus il est facile d'observer ses manœuvres. Cependant, pour améliorer la détection des attaques incrémentales, une piste envisageable est de favoriser les évènements récents ou significatifs. Par exemple, la similarité entre les informations tend à augmenter lors d'une attaque incrémentale. Plus l'attaque dure dans le temps et plus l'écart absolu entre les mesures de la source attaquée et celles des autres sera important. Par conséquent, c'est à l'instant courant que l'écart est le plus grand et donc qu'il est le plus significatif pour détecter une attaque.

La mesure est résiliente : lorsque le nombre de sources augmente, la confiance dans les autres sources (celles qui ne sont pas attaquées) est moins affectée. Ceci est une conséquence de notre hypothèse selon laquelle la majorité a raison. Nous considérons cette hypothèse pertinente car dans le cas où un attaquant contrôle la majorité des informations reçues par le système, ce dernier (le système) est à la merci du premier (l'attaquant). Nous avons constaté en particulier cette situation dans le cas où deux sources parmi trois sont soumises à des attaques partageant un objectif commun. Dans cette situation, notre hypothèse se révèle insuffisante. Pour adopter ou rejeter cette hypothèse, nous suggérons de nous baser sur une analyse des risques encourus par chaque bloc pour évaluer la plausibilité d'un tel scénario. Par ailleurs, en présence d'un nombre important de sources, seule la source attaquée voit sa confiance diminuée. En contrepartie, un système composé d'une unique source est d'autant plus vulnérable à une attaque, sa confiance ne dépendant que d'un seul composant : lui-même. Ce défaut pourrait être pallié en considérant les spécifications ou l'environnement particulier de la source elle-même.

Enfin, la propagation ne considère pas les cas de menace interne : la confiance se propage depuis les sources jusqu'aux autres blocs fonctionnels. La confiance de ces derniers ne s'évalue donc qu'en fonction de leurs entrées et ne tient pas compte

de leur sortie ou de leurs spécifications internes, c.-à-d. relatives aux traitements qu'ils appliquent à l'information. La combinaison de ces paramètres supplémentaires (sorties, caractéristiques intrinsèques du bloc) à leurs entrées pourrait permettre de détecter ce type particulier de menace.

Conclusion

Le travail présenté dans ce manuscrit se situe à la frontière entre les domaines de la sécurité, des systèmes d'information et de l'étude de la confiance. L'objectif de ce travail était de montrer comment la confiance peut servir à sécuriser les systèmes d'information en posant les bases d'un système de détection. Pour répondre à cette problématique, nous avons élaboré différents modèles fondés sur l'étude des systèmes navals.

Apports

Sur la base de la théorie des systèmes complexes, nous avons proposé un modèle de système d'information en tant qu'ensemble de blocs fonctionnels inter-reliés qui produisent, traitent et reçoivent des informations. Parmi ces blocs fonctionnels, les sources d'information constituent un cas à part : elles alimentent directement ou indirectement en informations tous les autres blocs du système. Notre modèle de système d'information repose sur l'hypothèse que les blocs fonctionnels sont des boîtes noires dont les mécanismes internes sont supposés inconnus. En pratique, les composants du système sont élaborés puis fournis pas des sous-traitants qui restreignent l'accès à leurs architectures internes en usant de divers moyens (*e.g.* protection physique, obfuscation du code, rupture du maintien en conditions opérationnelles). Ne pas considérer le fonctionnement interne des composants du système d'information facilite l'intégration de notre outil de détection en limitant la connaissance *a priori* nécessaire à son bon fonctionnement.

Les systèmes navals se caractérisent par une grande hétérogénéité : de nombreux systèmes nécessitant des mesures de sécurité particulières mais également un facteur humain très présent. Nous en avons tenu compte dans l'élaboration

de notre modèle qui peut s'adapter à des outils automatisés autant qu'à des personnes. En effet, la confiance, la compétence, la sincérité sont des notions qui concernent originellement des humains mais qui peuvent être transposées au sein de systèmes d'information. A partir de ces notions, nous avons donc construit des mesures permettant d'évaluer la confiance dans chacun des producteurs d'information au sein d'un SI. Le modèle que nous proposons s'adapte non seulement aux diverses natures des composants mais également à la taille du système. D'une part la propagation de la confiance permet d'évaluer la confiance dans chacun des éléments du système d'information : à l'instar de l'information qui irrigue le SI, la confiance mesurée se propage à toutes les entités quelque soit leur complexité. D'autre part, pour évaluer la confiance dans le système de navigation, la connaissance du système de propulsion est superflue. Leurs confiances peuvent donc être mesurées séparément.

Afin de tester les modèles proposés, nous avons conçu un simulateur permettant l'élaboration, le stockage ainsi que la mise en œuvre de scénarios d'attaques ciblant des systèmes d'information.

Perspectives

Modélisation du système d'information

Nous avons proposé des mesures de confiance et montré que celles-ci peuvent servir à la détection de falsifications d'informations numériques. Un système d'information manipule cependant d'autres types de données (texte, image, son, vidéo ...) qui peuvent également être falsifiées. L'extension de nos mesures à ces types d'information permettrait de détecter leurs altérations. Notre approche est en effet transposable à des informations non numériques : nous avons construit nos mesures en définissant des contraintes qui ne se limitent pas à un type de donnée particulier.

Nous avons adopté une approche systémique pour modéliser un système d'information. Cette approche qui se focalise sur les relations entre les différentes entités du SI (*e.g.* leurs entrées et sorties) plus que sur les entités elles-mêmes apparaît suffisante dans notre contexte. Toutefois, une approche analytique prenant en compte le fonctionnement interne des composants du système pourrait enrichir notre proposition, notamment la modélisation des blocs et leur sensibilité à leur

environnement.

Nous avons considéré les erreurs de mesures des sources d'information au sein d'un modèle probabiliste. Ce modèle a été choisi car il limite la connaissance spécifique des composants et s'adapte donc à l'hétérogénéité des systèmes navals. Nous suggérons cependant que cette approche pourrait bénéficier des outils d'apprentissage automatisé, lesquels, sur la base de spécifications plus complètes, pourraient permettre de construire (automatiquement) un modèle mieux adapté à chaque composant.

Modélisation de la confiance

Dans notre modèle de confiance, nous considérons que les différents blocs fonctionnels ont des criticités équivalentes pour un bloc donné. Cela n'est toutefois pas toujours le plus approprié. Par exemple, un bloc recevant des informations de position de la part de deux GPS distincts dont l'un utilise des mécanismes de protection (*e.g.* chiffrement, contrôle d'intégrité) ne considérera pas leurs informations de la même manière. Le GPS protégé fournit dans ce cas des informations plus fiables, une attaque le ciblant étant moins probable, et est considéré plus critique pour le bloc destinataire des informations, sa compromission rendant plus vraisemblable celle des blocs moins bien protégés. Une analyse des risques et menaces pouvant cibler les composants du système d'information permettrait de pondérer en conséquence les criticités de chaque bloc et partant, de rendre plus pertinente la propagation de la confiance.

La criticité dépend des risques et ceux-ci sont variables selon les situations. Un navire en mer par beau temps est exposé à des risques différents d'un navire qui entre dans un port embrumé. Ainsi la définition de *profils* selon quelques critères clés (*e.g.* à la mer ou environnement portuaire, météo) permettrait d'adapter les criticités aux risques auxquels est soumis le navire.

Simulateur et systèmes navals

Le passage d'un système d'information réel à son modèle composé de blocs fonctionnels est réalisé « à la main ». Dans le chapitre 5, nous nous sommes en effet restreints à des modèles simples mais représentatifs du fonctionnement des systèmes de navigation. Toutefois, pour des systèmes de taille importante, il apparait souhaitable de disposer d'un moyen automatisé capable de produire le modèle

correspondant à la description d'un système. Cette description est généralement réalisée via des outils tels que UML (Engels et al., 2000), SysML (Huang et al., 2007) ou Capella (Voirin et al., 2016). Utilisés en amont, lors d'une phase de spécification, ceux-ci permettent de décrire précisément les fonctionnalités des différents composants du SI, leurs caractéristiques ainsi que leur rôle au sein du système global. La description alors obtenue est structurée ce qui lui permet d'être exploitée pour produire une modélisation du système telle que celle que nous proposons dans ce travail.

Notre système de détection ne caractérise pas les attaques subies par le système. En effet, la confiance décroît indifféremment selon le type d'attaque perpétrée. Afin de réduire le délai d'intervention des personnes chargées de la sécurité, l'identification des attaques serait souhaitable. Par ailleurs, afin d'apporter une réponse proportionnée à la menace détectée, que celle-ci soit avérée ou non, la visualisation des mesures de confiance est également importante. Des exemples de mise en œuvre de représentations visuelles de données d'un système d'information à des fins de sécurité ont déjà été proposées (Varga et al., 2017).

Plus spécifiquement, concernant les systèmes navals, la mise en œuvre des modèles proposés dans des situations réelles puis leur extension à des sous-systèmes autres que celui servant à la navigation (*e.g.* propulsion, système d'arme) constituent des perspectives importantes pour l'intégration de notre solution aux sein des systèmes navals.

Bibliographie

Sinclair, Sara et Sean W Smith (2008). « Preventative directions for insider threat mitigation via access control ». In : *Insider Attack and Cyber Security*. Springer, p. 165-194.

Adomavicius, Gediminas et al. (2011). *Recommender Systems Handbook*. Sous la dir. de Francesco Ricci, Lior Rokach, Bracha Shapira et Paul B. Kantor. Springer.

Massa, Paolo et Paolo Avesani (2007a). « Trust-aware Recommender Systems ». In : *RecSys07*.

Atzori, Luigi, Antonio Iera et Giacomo Morabito (2010). « The internet of things : A survey ». In : *Computer networks* 54.15, p. 2787-2805.

Choo, Kim-Kwang Raymond (2011). « The cyber threat landscape : Challenges and future research directions ». In : *Computers & Security* 30.8, p. 719-731.

ANSSI (juin 2017). « Retour technique de l'incident de TV5Monde ». In : *Symposium sur la Sécurité des Technologies de l'Information et des Communications*.

Mohurle, Savita et Manisha Patil (2017). « A brief study of wannacry threat : Ransomware attack 2017 ». In : *International Journal of Advanced Research in Computer Science* 8.5.

Langner, Ralph (2011). « Stuxnet : Dissecting a cyberwarfare weapon ». In : *IEEE Security & Privacy* 9.3, p. 49-51.

Untersinger, Martin (2017). « Oups, vos Éléctions ont Été piratées ! Vraiment ? » In : *Symposium sur la Sécurité des Technologies de l'information et des Communications*.

Disterer, Georg (avr. 2013). « ISO/IEC 27000, 27001 and 27002 for information security management ». In : *Journal of Information Security* 4.2, p. 92-100.

Kocher, Paul, Joshua Jaffe et Benjamin Jun (1999). « Differential power analysis ». In : *Advances in cryptology—CRYPTO'99*.

Genkin, Daniel, Adi Shamir et Eran Tromer (2014). « RSA key extraction via low-bandwidth acoustic cryptanalysis ». In : *International Cryptology Conference*, p. 444-461.

Yitbarek, Salessawi Ferede, Misiker Tadesse Aga, Reetuparna Das et Todd Austin (fév. 2017). « Cold Boot Attacks are Still Hot : Security Analysis of Memory Scramblers in Modern Processors ». In : *2017 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, p. 313-324.

Juels, Ari (2006). « RFID security and privacy : A research survey ». In : *IEEE journal on selected areas in communications* 24.2, p. 381-394.

Shiu, Yi-Sheng, Shih Yu Chang, Hsiao-Chun Wu, Scott C-H Huang et Hsiao-Hwa Chen (2011). « Physical layer security in wireless networks : A tutorial ». In : *IEEE wireless Communications* 18.2.

Halfond, William G, Jeremy Viegas et Alessandro Orso (2006). « A classification of SQL-injection attacks and countermeasures ». In : *Proceedings of the IEEE International Symposium on Secure Software Engineering*. T. 1.

Ehrenkranz, Toby et Jun Li (2009). « On the state of IP spoofing defense ». In : *ACM Transactions on Internet Technology (TOIT)* 9.2.

Srivastava, A, BB Gupta, A Tyagi, Anupama Sharma et Anupama Mishra (2011). « A recent survey on DDoS attacks and defense mechanisms ». In : *Advances in Parallel Distributed Computing*, p. 570-580.

Boneh, Dan (1999). « Twenty years of attacks on the RSA cryptosystem ». In : *Notices of the AMS* 46.2, p. 203-213.

BIMCO, CLIA, International Chamber of Shipping, INTERCARGO et INTERTANKO (jan. 2016). *The Guidelines on Cyber Security onboard Ships*.

Yadav, Tarun et Arvind Mallari Rao (2015). « Technical aspects of cyber kill chain ». In : *International Symposium on Security in Computing and Communication*, p. 438-452.

Cimpean, Dan, Johan Meire, Vincent Bouckaert, Stijn Vande Casteele, Aurore Pelle et Luc Hellebooge (2011). *Analysis of Cyber Security aspects in the maritime sector*. Rapp. tech. ENISA.

Bothur, Dennis, Guanglou Zheng et Craig Valli (2017). « A critical analysis of security vulnerabilities and countermeasures in a smart ship system ». In : *Proceedings of the 15th Australian Information Security Management Conference*.

ANSSI (2015). *Cybersecurity for Industrial Control Systems*. Rapp. tech. Agence Nationale pour la Sécurité des Systèmes d'information.

Schmidt, Desmond, Kenneth Radke, Seyit Camtepe, Ernest Foo et Michał Ren (2016). « A survey and analysis of the gnss spoofing threat and countermeasures ». In : *ACM Computing Surveys (CSUR)* 48.4, p. 1-64.

Balduzzi, Marco, Alessandro Pasta et Kyle Wilhoit (déc. 2014). « A Security Evaluation of Automated Identification System ». In : *Proceedings of the 30th Annual Computer Security Applications Conference*. ACM, p. 436-445.

Fitton, Oliver, Daniel Prince, Basil Germond et Mark Lacy (2015). *The Future of Maritime Cyber Security*. Rapp. tech. Lancaster university.

Tam, Kimberly et Kevin Jones (2018). « Cyber-Risk Assessment for Autonomous Ships ». In : *International Conference On Cyber Security (Cyber Security 2018)*.

Dacier, Marc (1994). « Vers une évaluation quantitative de la sécurité informatique ». Thèse de doct. Institut National Polytechnique de Toulouse (INPT).

Arduin, Pierre-Emmanuel (2018). *Insider Threats*. Sous la dir. d'ISTE Editions. T. 10. Information systems, web and pervasive computing. ISTE et John Wiley & Sons.

National Institute of Standards and Technology (fév. 2014). *Framework for Improving Critical Infrastructure Cybersecurity*.

Mayer, Nicolas (2009). « Model-based management of information system security risk ». Thèse de doct. University of Namur.

Polatidis, Nikolaos, Michalis Pavlidis et Haralambos Mouratidis (2018). « Cyber-attack path discovery in a dynamic supply chain maritime risk management system ». In : *Computer Standards & Interfaces* 56, p. 74-82.

Kröner, Ulrich et Franc Dimc (2010). « Hardening of civilian GNSS trackers ». In : *Proceedings of the 3rd GNSS Vulnerabilities and Solutions Conference*.

Goudossis, Athanassios et Sokratis K Katsikas (2018). « Towards a secure automatic identification system (AIS) ». In : *Journal of Marine Science and Technology*, p. 1-14.

Diallo, David et Mathieu Feuillet (2014). « Détection d'intrusion dans les systèmes industriels : Suricata et le cas de Modbus ». In : *C&ESAR 2014*.

Garcia-Teodoro, Pedro, J Diaz-Verdejo, Gabriel Maciá-Fernández et Enrique Vázquez (2009). « Anomaly-based network intrusion detection : Techniques, systems and challenges ». In : *Computers & Security* 28.1-2, p. 18-28.

Chandola, Varun, Arindam Banerjee et Vipin Kumar (2007). *Anomaly Detection : A Survey*. Rapp. tech. University of Minnesota.

Sekar, R, Ajay Gupta, James Frullo, Tushar Shanbhag, Abhishek Tiwari, Henglin Yang et Sheng Zhou (2002). « Specification-based anomaly detection : a new approach for detecting network intrusions ». In : *Proceedings of the 9th ACM conference on Computer and communications security*. ACM, p. 265-274.

Ristic, Branko, Barbara La Scala, Mark Morelande et Neil Gordon (2008). « Statistical analysis of motion patterns in AIS data : Anomaly detection and motion prediction ». In : *Proceedings of the 11th international conference on Information fusion*. IEEE, p. 1-7.

Psiaki, Mark L, Brady W O'Hanlon, Jahshan A Bhatti, Daniel P Shepard et Todd E Humphreys (2013). « GPS spoofing detection via dual-receiver correlation of military signals ». In : *IEEE Transactions on Aerospace and Electronic Systems* 49.4, p. 2250-2267.

Sultan, Bastien, Fabien Dagnat et Caroline Fontaine (2017). « A Methodology to Assess Vulnerabilities and Countermeasures Impact on the Missions of a Naval System ». In : *Computer Security*. Springer, p. 63-76.

Lorenz, E. H. (1988). « Neither friends nor strangers : informal networks of subcontracting in French industry ». In : p. 194-210.

Sabel, C. F. (1990). « Studied trust : building new forms of co-operation in a volatile economy ». In : *Industrial Districts and Local Economic Regeneration*.

Lorini, Emiliano et Robert Demolombe (2008). « From Binary Trust to Graded Trust in Information Sources : A Logical Perspective ». In : *LNAI 5396*, p. 205-225.

Marsh, Stephen Paul (1994b). « Formalising trust as a Computational Concept ». Thèse de doct. Department of Computer Science et Mathematics, University of Stirling, p. 184.

Tucker, Albert W. (1983). « The Mathematics of Tucker : A Sampler ». In : *The Two-Year College Mathematics Journal* 14.3, p. 228-232.

Axelrod, Robert M. (1984). *The Evolution of Cooperation*. Basic Books.

Mayer, Roger C. et H. Davis James (1995). « An Integrative Model of trust ». In : *The Academy of Management Review* 20.3, p. 709-704.

Williams, Bernard (2000). « Formal Structures and Social Reality ». In : *Trust : Making and Breaking Cooperative Relations*.

Demolombe, Robert (2001). « To trust information sources : a proposal for a modal logical framework ». In : *Trust and deception in virtual societies*. Springer, p. 111-124.

— (2004). « Reasoning about trust : A formal logical framework ». In : *Trust Management*. Springer, p. 291-303.

Lorini, Emiliano et Robert Demolombe (2009). « From Trust in Information Sources to Trust in Communication Systems : An Analysis in Modal Logic ». In : *LNAI 5605*.

Paglieri, Fabio, Cristiano Castelfranchi, Célia da Costa Pereira, Rino Falcone, Andrea Tettamanzi et Serena Villata (2014). « Trusting the messenger because of the message : feedback dynamics from information quality to source evaluation ». In : *Computational and Mathematical Organization Theory* 20.2, p. 176-194.

Gambetta, Diego (1988). « Can we trust trust ? » In : *Trust : Making and Breaking Cooperative Relations*, p. 213-237.

Blomqvist, Kirsimarja (1997). « The many faces of trust ». In : *Scandinavian journal of management* 13.3, p. 271-286.

Teacy, W. T. Luke, Jigar Patel, Nicholas R. Jennings et Michael Luck (2006). « TRAVOS : Trust and Reputation in the context of inaccurate information sources ». In : *Autonomous Agents and Multi-Agent Systems* 12.2, p. 183-198.

Deutsch, Morton (1958). « Trust and suspicion ». In : *Journal of conflict resolution*, p. 265-279.

Luhmann, Niklas (1979). *Trust and Power*. U.M.I., p. 208.

Lewis, J. David et Andrew Weigert (1985). « Trust as a Social Reality ». In : *Social Forces* 63.4, p. 967-985.

Castelfranchi, Cristiano et Rino Falcone (2000). « Trust Is Much More than Subjective Probability : Mental Components and Sources of Trust ». In : *Proceedings of the 33rd Hawaii International Conference on System Sciences*. IEEE.

Luhmann, Niklas (2000). « Familiarity, Confidence, Trust : Problems and Alternatives ». In : *Trust : Making and Breaking Cooperative Relations*.

McKnight, D. Harrison et Norman L. Chervany (2000). « What is Trust ? A Conceptual Analysis and an Interdisciplinary Model ». In : *Americas Conference on Information Systems*, p. 827-833.

Jøsang, Audun et Stephane Lo Presti (2004). « Analysing the relationship between risk and trust ». In : *Trust Management*. Springer, p. 135-145.

Castelfranchi, Cristiano et Rino Falcone (1998). « Principles of trust for MAS : Cognitive anatomy, social importance, and quantification ». In : *Proceedings of the International Conference on Multi Agent Systems*. IEEE, p. 72-79.

Grandison, Tyrone et Morris Sloman (2000). « A survey of trust in internet applications ». In : *Communications Surveys & Tutorials, IEEE* 3.4, p. 2-16.

Cofta, Piotr (2004). « Computing Recommendations to Trust ». In : *Trust Management*. Springer, p. 340-346.

Bizer, Christian et Radoslaw Oldakowski (2004). « Using Context- and Content-Based Trust Policies on the Semantic Web ». In : *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*. ACM, p. 228-229.

Avesani, Paolo, Paolo Massa et Roberto Tiella (2005). « A Trust-enhanced Recommender System application : Moleskiing ». In : *Proceedings of the 2005 ACM Symposium on Applied computing*. ACM, p. 1589-1593.

Massa, Paolo et Paolo Avesani (2007b). « Trust metrics on controversial users : balancing between tyranny of the majority and the echo chambers ». In : *International Journal on Semantic Web and Information Systems* 3.1, p. 39-64.

Abdul-Rahman, Alfarez et Stephen Hailes (1997). « Using Recommendation for managing trust in distributed systems ». In : *Proceedings of IEEE International Conference on Communication*.

Yu, Bin et Munindar P. Singh (2002). « An evidential model of distributed reputation management ». In : *Proceedings of the first international joint conference on Autonomous agents and multiagent systems : part 1*. ACM, p. 294-301.

Kamvar, Sepandar D., Mario T. Schlosser et Hector Garcia-Molina (2003). « The EigenTrust Algorithm for Reputation Management in P2P Networks ». In : *Proceedings of the 12th International Conference on World Wide Web*, p. 640-651.

Grishchenko, Victor S. (2004). « A fuzzy model for context-dependent reputation ». In : *Trust, Security and Reputation Workshop at ISWC 2004*.

Falcone, Rino, Giovanni Pezzulo et Cristiano Castelfranchi (2003). « A Fuzzy Approach to a Belief-Based Trust Computation ». In : *Trust, reputation and security : theories and practice*. Springer, p. 73-86.

McKnight, D. Harrison et Norman L. Chervany (1996). *The Meanings of Trust*. Rapp. tech. University of Minnesota.

— (2001a). « Conceptual Trust : A Typologiy and E-Commerce Customer Relationships Model ». In : *Proceedings of the 34th Hawaii International Conference on System Sciences*.

Mui, Lik, Mojdeh Mohtashemi et Ari Halberstadt (2002). « A Computational Model of Trust and Reputation ». In : *Proceedings of the 35th Hawaii International Conference on System Sciences*.

Fullam, Karen K. et K. Suzanne Barber (2007). « Dynamically Learning Sources of Trust Information : Experience vs Reputation ». In : *Proceedings of the 6th international joint conference on Autonomous Agents and Multi-Agent Systems*.

Lin, Kwei-Jay, Haiyin Lu, Tao Yu et Chia-en Tai (2005). « A Reputation and Trust Management Broker Framework for Web Applications ». In : *Proceedings of the 2005 IEEE International Conference on e-Technology, e-Commerce and e-Service*. IEEE, p. 262-269.

Artz, Donovan et Yolanda Gil (2007). « A survey of trust in computer science and the Semantic Web ». In : *Web Semantics : Science, Services and Agents on the World Wide Web* 5.2, p. 58-71.

Jøsang, Audun, Roslan Ismail et Colin Boyd (2007). « A Survey of Trust and Reputation Systems for Online Service Provision ». In : *Decision Support Systems* 43.2, p. 618-644.

Jøsang, Audun, Ross Hayward et Simon Pope (2006b). « Trust Network Analysis with Subjective Logic ». In : *Proceedings of the 29th Australasian Computer Science Conference*. T. 48. Australian Computer Society, Inc., p. 85-94.

Wooldridge, Michael J (2000). *Reasoning about rational agents*. Sous la dir. de MIT press. MIT press.

Marsh, Stephen (1994a). « Trust in Distributed Artificial Intelligence ». In : *Artificial Social System*. Springer.

Lamport, Leslie, Robert Shostak et Marshall Pease (1982). « The Byzantine Generals Problem ». In : *ACM Transactions on Programming Languages and Systems (TOPLAS)* 4.3, p. 382-401.

Choi, Sung et David Zage (2012). « Addressing Insider Threat using "Where You Are" as Fourth Factor Authentication ». In : *International Carnahan Conference on Security Technology*, p. 147-153.

Abdul-Rahman, Alfarez (1997). « The PGP trust model ». In : *The Journal of Electronic Commerce*. T. 10. 3, p. 27-31.

Denning, Dorothy Elizabeth (1982). *Cryptography and Data Security*. Addison-Wesley Longman Publishing Co., Inc.

Blaze, Matt, Joan Feigenbaum et Martin Strauss (1998). « Compliance Checking in the PolicyMaker Trust Management System ». In : *Financial Cryptography*. Springer, p. 254-274.

Blaze, Matt, Joan Feigenbaum, John Ioannidis et Angelos D. Keromytis (1999). « The Role of Trust Management in Distributed Systems Security ». In : *Secure Internet Programming*. Springer, p. 185-210.

Massa, Paolo et Paolo Avesani (2009). « Trust Metrics in Recommender Systems ». In : *Computing with Social Trust*. Springer, p. 259-285.

Khaleghi, Bahador, Alaa Khamis et Fakhreddine O. Karray (2011). « Multisensor data fusion : A review of the state-of-the-art ». In : *Information Fusion* 14.1, p. 28-44.

Demolombe, Robert (2011). « Transitivity and Propagation of Trust in Information Sources : An Analysis in Modal Logic ». In : *Computational Logic in Multi-Agent Systems*. Springer, p. 13-28.

Jøsang, Audun (1999). « An Algebra for Assessing Trust in Certification Chains ». In : *Network and Distributed Security Symposium*. T. 99. 6.

— (2001). « A logic for uncertain probabilities ». In : *International Journal of Uncertainty, Fuzziness and Knowledge-based Systems* 9.3, p. 279-311.

Jøsang, Audun, Stephen Marsh et Simon Pope (2006a). « Exploring Different Types of Trust Propagation ». In : *Trust Management*. Springer, p. 179-192.

Alhadad, Nagham, Yann Busnel, Patricia Serrano-Alvarado et Philippe Lamarre (2014). « Trust Evaluation of a System for an Activity with Subjective Logic ». In : *Trust, Privacy, and Security in Digital Business*. Springer, p. 48-59.

Lewicki, Roy J., J. McAllister Daniel et Robert J. Bies (1998). « Trust and Distrust : new relationships and realities ». In : *The Academy of Management Review* 23.3, p. 438-458.

McKnight, D. Harrison et Norman L. Chervany (2001b). « Trust and Distrust Definitions : One Bite at a Time ». In : *Trust in Cyber-societies*. Springer, p. 27-54.

Da Costa Pereira, Célia (2009). « Distrust is not always the Complement of Trust (Position Paper) ». In : *Normative Multi-Agent Systems*.

De Cock, Martine et Paulo Pinheiro Da Silva (2006). « A many valued representation and propagation of trust and distrust ». In : *Fuzzy Logic and Applications*. Springer, p. 114-120.

Guha, R., Ravi Kumar, Prabhakar Raghavan et Andrew Tomkins (2004). « Propagation of Trust and Distrust ». In : *Proceedings of the 13th international conference on World Wide Web*, p. 403-412.

Deschrijver, Glad et Etienne E Kerre (2003). « On the relationship between some extensions of fuzzy set theory ». In : *Fuzzy sets and systems* 133.2, p. 227-235.

Yan, Zheng, Peng Zhang et Teemupekka Virtanen (2003). « Trust Evaluation Based Security Solution in Ad Hoc Networks ». In : *Proceedings of the Seventh Nordic Workshop on Secure IT Systems*. T. 14.

Das, Anupam et Mohammad Mahfuzul Islam (2012). « SecuredTrust : a dynamic trust computation model for secured communication in multiagent systems ». In : *IEEE Transactions on Dependable and Secure Computing* 9.2, p. 261-274.

Jøsang, Audun, Magdalena Ivanovska et Tim Muller (juil. 2015). « Trust Revision for Conflicting Sources ». In : *Proceedings of the 18th International Conference on Information Fusion (FUSION 2015)*, p. 550-557.

Matt, Paul-Amaury, Maxime Morge et Francesca Toni (2010). « Combining statistics and arguments to compute trust ». In : *Proceedings of 9th International Conference on Autonomous Agents and Multiagent Systems*, p. 209-216.

Dung, Phan Minh (1993). « On the acceptability of arguments and its fundamental role in nonmonotonic reasoning and logic programming ». In : *International Joint Conferences on Artificial Intelligence*, p. 852-857.

Stranders, Ruben, Mathijs de Weerdt et Cees Witteveen (2008). « Fuzzy Argumentation for Trust ». In : *Computational Logic in Multi-Agent Systems*. Springer, p. 214-230.

Parsons, Simon, Yuqing Tang, Elizabeth Sklar, Peter McBurney et Kai Cai (2011). « Argumentation-based reasoning in agents with varying degrees of trust ». In : *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems*.

Villata, Serena, Guido Boella, Dov M. Gabbay et Leendert van der Torre (2011). « Arguing about the Trustworthiness of the Information Sources ». In : *Symbolic and quantitative approaches to reasoning with uncertainty*. Springer.

Da Costa Pereira, Célia, Andrea. B. TeTettamanzi et Serena Villata (2011). « Changing One's Mind : Erase or Rewind ? Possibilistic Belief Revision with Fuzzy Argumentation Based on Trust ». In : *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*. T. 1, p. 164-171.

Villata, Serena, Guido Boella, Dov M. Gabbay et Leendert van der Torre (2013). « A socio-cognitive model of trust using argumentation theory ». In : *International Journal of Approximate Reasoning* 54.4, p. 541-559.

Capra, Licia et Mirco Musolesi (2006). « Autonomic Trust Prediction for Pervasive Systems ». In : *20th International Conference on Advanced Information Networking and Applications*. T. 2. IEEE, p. 48-59.

Sun, Yan Lindsay, Zhu Han, Wei Yu et KJ Ray Liu (2006). « A Trust Evaluation Framework in Distributed Networks : Vulnerability Analysis and Defense Against Attacks ». In : *INFOCOM*, p. 1-13.

Wang, Yonghong et Munindar P Singh (2007). « Formal Trust Model for Multiagent Systems ». In : *International Joint Conference on Artificial Intelligence*, p. 1551-1556.

Esfandiari, Babak et Sanjay Chandrasekharan (mai 2001). « On How Agents Make Friends : Mechanisms for Trust Acquisition ». In : *4th workshop on deception, fraud and trust in societies*. T. 222.

Richardson, Matthew, Rakesh Agrawal et Pedro Domingos (2003). « Trust management for the semantic web ». In : *The Semantic Web-ISWC 2003*. Springer.

Wang, Yonghong et Munindar P Singh (2006). « Trust representation and aggregation in a distributed agent system ». In : *AAAI*. T. 6, p. 1425-1430.

Christianson, Bruce et William Harbison (1996). « Why isn't trust transitive ? » In : *Security protocols*. Springer, p. 171-176.

Seigneur, Jean-Marc (2006). *AmbiTrust ? Immutable and Context-Aware Trust Fusion*. Rapp. tech. University of Geneva.

Ismail, Roslan et Audun Jøsang (2002). « The Beta reputation system ». In : *Bled Proceedings* 41.

Ruohomaa, Sini (2004). *Trust management Survey*. Rapp. tech. University of Helsinki.

De Courcy, Richard (1992). « Les systèmes d'information en réadaptation ». In : *Québec, Réseau International CIDIH et facteurs environnementaux* 1.5, p. 7-10.

Landry, R et M Santerre (1999). *Méthodes de simulation en science politique*. Rapp. tech. Université Laval, Québec, Canada.

Ray, Cyril (2003). « ATLAS, une plate-forme pour la modélisation et la simulation de systèmes désagrégés ». Thèse de doct. Université Rennes 1.

FORRESTER, Jay Wright et Patrick SYLVESTRE-BARON (1984). *Principes des systèmes*. Sous la dir. de Presses universitaires de LYON. Presses universitaires de Lyon.

DUBOZ, Raphael (mar. 2004). « Intégration de modèles hétérogènes pour la modélisation et la simulation de systèmes complexes ». Thèse de doct. Ecole doctorale de l'Université du Littoral.

MANSON, Steven M. (2001). « Simplifying complexity : a review of complexity theory ». In : *Geoforum* 32.3, p. 405-414.

KOLMOGOROV, Andrei N (1965). « Three Approaches to the quantitative definition of information ». In : *Problems of information transmission* 1.1, p. 1-7.

BENNETT, Charles (1988). « Logical Depth and Physical Complexity ». In : sous la dir. de Rolf HERKEN. Oxford University Press. Chap. 2, p. 227-257.

DELAHAYE, Jean-Paul (2013). « Mesurer la Complexité des objets numériques ». In : *Bulletin de la société informatique de France*. 1, p. 35-53.

HOLLAND, John (1992). « Complex Adaptative Systems ». In : *Daedalus* 121.1, p. 17-30.

LE MOIGNE, Jean-Louis (1994). *La théorie du système général : théorie de la modélisation*. Sous la dir. de Presses Universitaires de FRANCE. Presses Universitaires de France.

OVCHINNIKOV, Sergei (1991). « Similarity relations, fuzzy partitions, and fuzzy orderings ». In : *Fuzzy Sets and Systems* 40.1, p. 107-126.

PAPOULIS, Athanasios et S Unnikrishna PILLAI (1986). *Probability, random variables, and stochastic processes*. Sous la dir. d'International EDITION. McGraw Hill, New York.

BHATTI, Jahshan et Todd E. HUMPHREYS (2014). « Covert Control of Surface Vessels via Counterfeit Civil GPS Signals ».

— (2017). « Hostile control of ships via false GPS signals : Demonstration and detection ». In : *Navigation* 64.1, p. 51-66.

LIU, Wei et Mary-Anne WILLIAMS (2002). « Trustworthiness of information sources and information pedigree ». In : *Intelligent Agents VIII*. Springer, p. 290-306.

YICK, Jennifer, Biswanath MUKHERJEE et Dipak GHOSAL (2008). « Wireless sensor network survey ». In : *Computer networks* 52.12, p. 2292-2330.

DUFOUR, Barbara et Régis POUILLOT (2002). « Approche qualitative du risque ». In : *Epidémiologie et santé animale* 41.

Kuhn, D Richard, Dolores R Wallace et Albert M Gallo (2004). « Software fault interactions and implications for software testing ». In : *IEEE transactions on software engineering* 30.6, p. 418-421.

Lions, Jacques-Louis et al. (1996). *Ariane 5 flight 501 failure*. Rapp. tech. European Space Agency.

Chifflier, Pierre et Arnaud Fontaine (2014). « Architecture système sécurisée de sonde IDS réseau ». In : *Computer & Electronics Security Applications Rendez-vous (C&ESAR)*.

Iphar, Clément, Aldo Napoli, Cyril Ray, Erwan Alincourt et David Brosset (2016). « Risk Analysis of falsified Automatic Identification System for the improvement of maritime traffic safety ». In : *ESREL 2016*. Taylor & Francis, p. 606-613.

Chiclana, Francisco, JM Tapia GarcíA, Maria Jose del Moral et Enrique Herrera-Viedma (2013). « A statistical comparative study of different similarity measures of consensus in group decision making ». In : *Information Sciences* 221, p. 110-123.

Cha, Sung-Hyuk (2007). « Comprehensive survey on distance/similarity measures between probability density functions ». In : *International Journal of Mathematical models nad Methods in applied sciences* 1.4, p. 300-307.

Ries, Sebastian (2007). « Certain trust : a trust model for users and agents ». In : *Proceedings of the 2007 ACM symposium on Applied computing*, p. 1599-1604.

Engels, Gregor, Reiko Heckel et Stefan Sauer (2000). « UML—A Universal Modeling Language ? » In : *International Conference on Application and Theory of Petri Nets*. Springer, p. 24-38.

Huang, Edward, Randeep Ramamurthy et Leon F McGinnis (2007). « System and simulation modeling using SysML ». In : *Proceedings of the 39th conference on Winter simulation*. IEEE, p. 796-803.

Voirin, Jean-Luc, Stéphane Bonnet, Daniel Exertier et Véronique Normand (2016). « Simplifying (and enriching) SysML to perform functional analysis and model instances ». In : *INCOSE International Symposium*. T. 26. 1. Wiley Online Library, p. 253-268.

Varga, Margaret, Carsten Winkelholz et Susan Träber-Burdin (2017). *The Application of Visual Analytics to Cyber Security*.

Publications

Costé, Benjamin, Cyril Ray et Gouenou Coatrieux (2018). « Trust Assessment for the Security of Information Systems ». In : t. 8. Springer, p. 1-23.

— (jan. 2017b). « Modèle et mesures de confiance pour la sécurité des systèmes d'information ». In : t. 22. 1. Hermès-Lavoisier, p. 19-41.

— (jan. 2017a). « Mesure de la confiance dans les systèmes d'information : application aux données de navires ». In : *17ème Journées Francophones Extraction et Gestion des Connaissances (EGC 2017)*. T. E-33. RNTI. Éditions RNTI, p. 117-128.

— (juin 2016). « Évaluation de la confiance dans un environnement multi-sources ». In : *Informatique des Organisations et Systèmes d'Information et de Décision (INFORSID 2016), Atelier Sécurité des systèmes d'information : technologies et personnes*, p. 1-7.

Brosset, David, Camille Cavelier, Benjamin Costé, Yvon Kermarrec, Joffrey Lartigaud et Pedro Merino Laso (2017). « Cr@ ck3n : A cyber alerts visualization object ». In : *International Conference On Cyber Situational Awareness, Data Analytics And Assessment (Cyber SA 2017)*. Sous la dir. d'IEEE. IEEE, p. 1-2.

Becmeur, Thomas, Xavier Boudvin, David Brosset, Gaël Héno, Benjamin Costé, Yvon Kermarrec et Pedro Merino Laso (2017). « Generating data sets as inputs of reference for cyber security issues and industrial control systems ». In : *Proceedings of the 11th International Conference on Research Challenges in Information Science (RCIS 2017)*. IEEE, p. 453-454.

Résumé

Dans le domaine maritime, la maitrise de la navigation et de la conduite d'un navire sont deux aspects essentiels pour la bonne marche et la sécurité du navire, des personnels et la préservation de l'environnement maritime. Or, les navires modernes embarquent de plus en plus de technologies informatisées, connectées et automatisées pour gérer ces fonctions primordiales. Ces technologies (capteurs, actionneurs, automates, logiciels) qui constituent le système d'information (SI) d'un navire peuvent cependant être leurrées ou corrompues par un tiers, remettant ainsi en cause la confiance qui leur est accordée.

Dans ce contexte, une nouvelle approche de détection des falsifications des informations fondée sur l'évaluation de la confiance dans les composants du SI est proposée. Du fait de leur complexité, les systèmes d'information des navires peuvent être considérés comme des ensembles de blocs fonctionnels inter-reliés qui produisent, traitent et reçoivent des informations. La confiance d'un bloc fonctionnel producteur d'information est évaluée au travers de sa capacité, divisée en deux composantes (compétence et sincérité), à rendre compte de la situation réelle du navire. Elle se propage ensuite, à l'instar de l'information, aux autres entités du système quelle que soit leur complexité.

Différents scénarios ont été expérimentés grâce à l'élaboration d'un simulateur. La variabilité de la confiance face à des altérations volontaires d'informations numériques permet de déduire la survenue d'une attaque ainsi que sa cible sous certaines conditions. Sans se restreindre aux systèmes navals, l'approche proposée permet de s'adapter à une grande variété de situations incluant le facteur humain.

Les travaux de cette thèse ont été soutenus et co-financés par la région Bretagne ainsi que la Chaire de Cyber Défense des Systèmes Navals impliquant l'École Navale, IMT Atlantique, Naval Group et Thales.

Titre : Détection contextuelle de cyberattaques par gestion de confiance à bord d'un navire

Mots clés : confiance, sécurité des systèmes d'information, systèmes d'information

Résumé : Dans le domaine maritime, la maitrise de la navigation et de la conduite d'un navire sont deux aspects essentiels pour la bonne marche et la sécurité du navire, des personnels et la préservation de l'environnement maritime. Or, les navires modernes embarquent de plus en plus de technologies informatisées, connectées et automatisées pour gérer ces fonctions primordiales. Ces technologies (capteurs, actionneurs, automates, logiciels) qui constituent le système d'information (SI) d'un navire peuvent cependant être leurrées ou corrompues par un tiers, remettant ainsi en cause la confiance qui leur est accordée.
Dans ce contexte, une nouvelle approche de détection des falsifications des informations fondée sur l'évaluation de la confiance dans les composants du SI est proposée. Du fait de leur complexité, les systèmes d'information des navires peuvent être considérés comme des ensembles de blocs fonctionnels inter-reliés qui produisent, traitent et reçoivent des informations. La confiance d'un bloc fonctionnel producteur d'information est évaluée au travers de sa capacité, divisée en deux composantes (compétence et sincérité), à rendre compte de la situation réelle du navire. Elle se propage ensuite, à l'instar de l'information, aux autres entités du système, quelle que soit leur complexité.
Différents scénarios ont été expérimentés grâce à l'élaboration d'un simulateur. La variabilité de la confiance face à des altérations volontaires d'informations numériques permet de déduire la survenue d'une attaque ainsi que sa cible sous certaines conditions. Sans se restreindre aux systèmes navals, l'approche proposée permet de s'adapter à une grande variété de situations incluant le facteur humain.
Les travaux de cette thèse ont été soutenus et co-financés par la région Bretagne ainsi que la Chaire de Cyber Défense des Systèmes Navals impliquant l'Ecole Navale, IMT Atlantique, Naval Group et Thales.

Title : Trust management for contextual cyberattacks detection on board ship

Keywords : trust, security, information systems

Abstract : Navigation and ship's management are two essential aspects for the security of the ship itself and people on board as much as the maritime environment protection. Modern ships ensure these functions by increasingly embedding connected and automated technologies such as sensors, actuators, programmable logic controllers and pieces of software. However, the security of this objects as well as the trust in the information they produce cannot be guaranteed: they can be deceived or under the control of a malicious third party.
In this context, a novel approach of data falsification detection is proposed. It is based on trust assessment of information system components which can be seen as inter-related functional blocks producing, processing and receiving pieces of information. The trust one can have in production blocks, called information sources, is assessed through its ability to report real situation of the ship. Trust is then propagated to the remainder part of the system.
A simulator was made thanks to which we experiment several scenarios including intentional modification of numerical data. In these cases and under some conditions, the variability of trust give us the ability to identify the attack occurrence as much as its target. Our proposition is not restricted to naval information systems and can be employed in various situations even with human factor.