République Algérienne démocratique et populaire

Ministère de l'enseignement supérieur et de la recherche scientifique

MEMOIRE

Présenté

A l'université MOULOUD MAMMERI de TIZI-OUZOU

Faculté Génie Electrique et Informatique

Pour l'obtention du diplôme de MASTER en Informatique

Option : **Réseaux, mobilité et systèmes embarqués**

Par

WASSILA HOCEINI

Thème

Un nouveau système de confiance pour les Réseaux de Capteurs sans fils

Encadré par

Mr Saïd TALBI

- Septembre 2014 -

Remerciements

Je remercie le bon DIEU qui m'a donné la volonté et le courage pour la réalisation de ce modeste travail.

Ma profonde gratitude et mes chaleureux remerciements à mon enseignant et encadreur Mr Saïd TALBI pour son accompagnement précieux depuis ma troisième année licence. Je le remercie d'avoir proposé et dirigé ce travail, avec une qualité d'encadrement remarquable. J'ai grandement apprécié ses compétences, son énergie, sa motivation et le suivi régulier de l'avancement de mon travail. Je tiens à le remercier pour sa patience, sa patience d'ange. Que cette page soit le parfait témoignage de ma gratitude envers cet enseignant.

Mes sincères remerciements à tous mes enseignants qui m'ont transmis les bases de l'informatique tout au long de ces deux années de master.

Un merci particulier à mon enseignant Mr Mhammed DAOUI pour son encouragement, sa disponibilité, son orientation, et ses efforts fournis pour mon avancement. Je le remercie pour la qualité de ses cours, pour ses animations superbes et surtout pour le nombre important de séances de travail supplémentaires. Je lui dois la grande part de ma formation dans le domaine des réseaux et des systèmes embarqués. Merci Mr DAOUI, c'était un grand honneur de travailler avec vous.

J'exprime également ma gratitude à Mr Mustapha LALAM pour ses conseils précieux, ses remarques pertinentes et ses encouragements pour le bon déroulement de notre formation.

J'adresse mes remerciements à un excellent et ancien enseignant de l'UMMTO, Mr Mohammed OUAMRANE, pour son encouragement, ses conseils et son soutien. Je le remercie d'avoir été toujours à mes cotés.

Mes remerciements s'adressent aussi aux messieurs le président et les membres de jury d'avoir accepté d'examiner mon travail.

Je remercie ma famille pour leurs soutien et accompagnement tout au long de ce travail. Particulièrement la personne avec laquelle je passe la grande partie de mon temps, Mon Père qui me transmit chaque jour son savoir et son expérience. Merci Papa.

Un grand merci à mes amis avec qui j'ai passé des meilleurs moments au sein de l'UMMTO.

Que ceux qui se sentent oubliés trouvent ici ma profonde gratitude pour leur concours dans l'accomplissement de ce travail.

Abstract

***W**ireless **S**ensor **N**etwork is a fast growing and exciting research area that has attracted considerable research attention in the recent past. The creation of large-scale sensor networks interconnecting several hundred to a few thousand sensors nodes opens up several technical challenges and immense application possibilities. The wireless and resource-constraint nature of a sensor network makes it an ideal medium for attackers to do any kinds of vicious things. Research on security in WSNs has also advanced, showing cryptography mechanisms, intrusion detection systems, and efficient routing protocols. However, using those traditional techniques to eliminate insider attacks is not possible. In order to filter out compromised nodes from sensor networks, some trust-based systems have recently been modeled. The resource efficiency and dependability of a trust system are the most fundamental requirements for any wireless sensor network (WSN). In this document, we proposed a **T**rust-based **C**lustering and **R**outing scheme for WSNs, which employ clustering algorithms. This approach can effectively reduce energy consumption. Moreover, a self-organization method is defined for trust Cluster-heads election.*
Theory as well as simulation results shows that TCR consume less energy compared with the current typical trust systems for WSNs.

***Key words: Reputation, Trust management, Energy, Wireless Sensor Network**.*

Résumé

*Les **R**éseaux de **C**apteurs sans **F**ils est un domaine émergent qui a attiré l'attention de beaucoup de domaines de recherches dans le passé récent. La création d'un réseau de capteurs à large échelle interconnectant plusieurs centaines à quelques milliers de nœuds senseurs donne lieu à différents challenges et beaucoup de possibilité d'applications. Les contraintes et la nature sans fil d'un réseau de capteurs lui rend un medium exposé aux attaqueurs pour exécuter des faits vicieux.*
Les recherches dans la sécurité des RCSFs sont aussi avancées, des mécanismes de cryptographie, des systèmes de détection d'intrusions, et des protocoles de routage efficients. Néanmoins, utiliser ces techniques traditionnelles pour éliminer les attaques internes n'est pas évident. Afin de filtrer les nœuds compromis des nœuds du réseau, quelques systèmes basés confiance ont été modélisés récemment. L'efficacité des ressources et la fiabilité d'un système de confiance sont les exigences les plus fondamentales pour n'importe quel réseau de capteurs.
Dans ce document, nous avons proposé un schéma de confiance, Groupement et Routage basé Confiance pour les réseaux de capteurs sans fils, qui emploie les algorithmes de clustering. Cette approche peut réduire considérablement la consommation d'énergie. De plus, une méthode d'auto-organisation est définie pour une élection sécurisée des cluster-heads.
La théorie et les résultats de simulation montrent que TCR consomme moins d'énergie comparée aux systèmes de confiance actuels pour les RCSFs.

***Mots clés : Réputation, management de confiance, Energie, Réseau de Capteurs sans fils**.*

Table des matières

Introduction générale 6

1. Chapitre 1. Réseaux de capteurs sans fils 8

1.1 Introduction 10
1.2 Architecture 10
1.3 Anatomie d'un micro-capteur 11
1.3.1 Sonde ou unité de captage 11
1.3.2 Unité de traitement 12
1.3.3 Unité de transmission 12
1.3.4 Unité de contrôle d'énergie 12
1.4. Plateformes existantes 12
1.5. Communication 13
1.5.1 Pile protocolaire 13
1.5.1.1 Couche physique 14
1.5.1.2 Couche lien de données 14
1.5.1.3 Couche réseau 14
1.5.1.4 Couche transport 14
1.5.1.5 Couche application 14
1.5.1.6 Plan de gestion d'énergie 15
1.5.1.7 Plan de gestion de la mobilité 15
1.5.1.8 Plan de gestion de tâches 15
1.5.2 Standards offerts 15
1.6 .Applications des WSN 16
1.6.1 Environnement 16
1.6.2 Phénomènes naturelles 16
1.6.3 Agriculture 16
1.6.4 Industrie 17
1.6.5 Surveillance des humains 17
1.6.6 Monitoring Militaire 18
1.6.7 Sécurité routière 18
1.6.8 La maison intelligente 18
1.7. Caractéristique et Challenges 19
1.7.1 Durée de vie 19
1.7.2 Energie 19
1.7.3 Coût d'un capteur 19
1.7.4 Environnement 19

1.7.5 Architecture tierce (energy, size and price) 19
1.7.6 Routage 20
1.7.7 Organisation en clusters 20
1.7.8 Sécurité 20
1.8. Conclusion 22
2. Chapitre 2. Sécurité dans les réseaux de capteurs 20
2.1 Introduction 24
2.2 Menaces 24
2.2.1 Classification des attaques par nature et origine 24
2.2.1.1 Attaques passives VS attaques actives 25
2.2.1.2 Attaques externes VS Attaques internes 25
2.2.2 Les attaques les plus connus 25
2.2.2.1 Subversion of a Node (Subversion d'un nœud) 25
2.2.2.2 Traffic Analysis (Analyse du Trafic ou collecte active d'informations) 25
2.2.2.3 Denial of Service (Déni de Service) 26
2.2.2.4 jamming (Brouillage radio) 26
2.2.2.5 Manipulation de trafic 26
2.2.2.6 Spoofing de l'identité 27
2.2.2.7 Faux routage 27
2.2.2.8 Black Hole 27
2.2.2.9 Grey hole (trou gris) 28
2.2.2.10 SinkHole (trou de la base) 28
2.2.2.11 Clock Skewing 28
2.2.2.12 Distorsion de l'agrégation de données 28
2.3 Sécurité d'amorçage (bootsrapping) dans un réseau de capteurs 29
2.4 Architectures de sécurité existantes 29
2.4.1 µTESLA (micro time efficient streaming loss-tolerant authentication) 29
2.5 Critères de sécurité 30
2.5.1 Confidentialité 30
2.5.2 Intégrité 30
2.5.3 Fraicheur 30
2.5.4 Disponibilité 30
2.5.5 Auto-organisation 31
2.5.6 Authentification 31
2.5.7 Responsabilité (non-répudiation) 31
2.6 Mécanismes de défense 32
2.6.1 Chiffrement 32
.2 Stéganographie (écriture couverte) 32

2.6.2 Notions d'authentification 33
2.6.3 Signature numérique 33
2.6.4 Fonction de hachage 33
2.6.5 Partitionnement des données 33
2.6.6 Détection d'intrusion 33
2.6.7 Certificats électroniques 34
2.6.8 Indice de confiance et réputation 34
2.7 Discussion 35
2.8 Conclusion 36
3. Chapitre 3. Confiance dans les réseaux de capteurs 33
3.1 Introduction 39
3.2 Confiance 40
3.3 Réseaux de capteurs et confiance 40
3.4 Confiance vs Réputation 41
3.5 Réseaux de capteurs et réputation 41
3.6 Modèles de réputation 42
3.6.1 Réputation subjective 42
3.6.2 Réputation indirecte 42
3.6.3 Réputation fonctionnelle 43
3.7 Intérêt d'un système de réputation 43
3.8 Score de réputation 44
3.9 Conception d'un système de confiance et réputation 44
3.9.1 Architecture 44
3.9.2 Evaluation de confiance 44
3.9.2.1 Métriques d'évaluation 45
3.10 Attaques ciblant les systèmes de réputation 45
3.11 Systèmes de gestion de confiance dans les réseaux ad hoc 46
3.12 Systèmes de gestion de confiance dans les réseaux de capteurs 47
3.12.1 PLUS (Parameterized and Localized trUst management Scheme for WSNs) 47
3.12.2 RFSN (Reputation-based Framework for Sensor Networks) 48
3.12.3 GTMS (Group-Based Trust Management Scheme for Clustered Wireless Sensor Networks) 48
3.12.4 GCP (Generic Communication Protocol) 50
3.12.5 2-ACKT (Two-Way Acknowledgment-Based Trust Framework for Wireless Sensor Networks) 51
3.12.6 TSRF (A Trust-Aware Secure Routing Framework in Wireless Sensor Networks) 51
3.12.7 A Secure Trust Establishment Scheme for Wireless Sensor Networks 52
3.12.8 LDTS (A Lightweight and Dependable Trust System for Clustered Wireless Sensor Networks) 53
3.13 Présentation 55

a. Outil de conception....62
b. Tests et évaluations....63
3.14 Conclusion....72
Conclusion générale....65
Références....67

Liste des figures

Figure 1 **Architecture générale d'un réseau de capteurs sans fils** 10
Figure 2. Architecture d'un micro-capteur 11
Figure 3. Pile protocolaire des réseaux de capteurs 13
Figure 4. Les capteurs ZigBeef[28] 17
Figure 5. L'attaque BlackHole [90] 27
Figure 6. Distorsion d'agrégation de données [90] 28
Figure 7. Cryptographie d'un texte [107] 32
Figure 8. Le format d'un paquet GCP 50
Figure 9. Modélisation d'un round dans TCR 61
Figure 10. Modèle de simulation du protocole TCR 62
Figure 11. Energie consommée dans chacun des schémas TCR, RFSN et PLUS pour les scénarios de recommandation 66
Figure 12. Taux moyen des paquets perdus 67
Figure 13. Comparaison en le taux de perte de paquets avec TCR et sans TCR 69
Figure 14. Taux de perte de paquets avec 12,5 % d'attaques et 37,5% de nœuds égoïstes 70
Figure 15. WhitList et BlackList dans un cluster de 8 membres (simulation 1) 71
Figure 16.WhitList et BlackList dans un cluster de 8 membres (simulation 2) 72

Liste des tableaux

Tableau 1. Paquets du protocole PLUS 47
Tableau 2. Paquets de RFSN 48
Tableau 3. Les paquets de TCR 61
Tableau 4. Les paires de recommandations entre les Cluster-heads 65
Tableau 5. Paramètres de simulation du scénario 1 65
Tableau 6.Paramètres de simulation du scénario 2 67
Tableau 7. Paramètres de simulation pour le scénario 4 71

Introduction générale

Le vingtième et unième siècle a marqué une évolution exponentielle dans les technologies de l'information émergentes, une démocratisation dans les communications sans fils et une haute intégration de circuits avec une meilleure performance et fonctionnalité à un coût de plus en plus réduit. L'un des facteurs majeurs de la production de ces circuits complexes à haute intégration est l'avantage de la loi de Moore (Moore's Law)*1*. Le développement de la mécanique traditionnelle a atteint un pallier au milieu du vingtième siècle qui ne constitue pas une source d'innovations à moins de coupler cette mécanique aux technologies de fabrication électronique qu'on appelle MEMS (MicroElectroMechanical Systems) ou systèmes embarqués.

Les premiers systèmes embarqués sont apparus en 1971 avec l'apparition d'Intel 4004. L'Intel 4004 *2* développé en 1971, le premier microprocesseur, était le premier circuit intégré incorporant tous les éléments d'un ordinateur dans un seul boîtier: unité de calcul, mémoire, contrôle des entrées/sorties. Alors qu'il fallait auparavant plusieurs circuits intégrés différents, chacun dédié à une tâche particulière, un seul microprocesseur pouvait assurer autant de travaux différents que possible. Très rapidement, des objets quotidiens tels que fours à micro-ondes, télévisions et automobiles à moteur à injection électronique ne tardèrent pas à être équipés de microprocesseurs. Ce sont alors les débuts de l'informatique embarquée.

Un système embarqué est un système électronique et informatique autonome ne possédant pas des entrées sorties standard comme un clavier ou un écran d'ordinateur. On le définit aussi généralement par le fait qu'il n'est pas visible en tant que tel, mais est intégré dans un équipement doté d'une autre fonction, on dit aussi que le système est enfoui, ce qui traduit plus fidèlement le terme anglais Embedded. Voici quelques systèmes informatiques embarqués que nous utilisons quotidiennement: gestion de l'ascenseur, auto radio, calculateur d'airbag, distributeur de boissons, routeur Internet, téléphone mobile, distributeur de billets, console de jeux et capteurs embarqués. Ces capteurs embarqués minuscules offrent une possibilité de surveillance de l'environnement de déploiement, captage des données sous forme de valeurs numériques et leur communication à une station de base. Ils sont autonomes et disposent pour cela d'une réserve énergétique, dont le renouvellement peut s'avérer impossible, ce qui limite leur durée de vie.

1 « Le nombre de transistors intégrés dans une puce doublera tous les deux ans » c'est la loi de Moore Gordon, l'un des fondateurs d'Intel.

2 1968. Création d'Intel, 1971. Premier processeur, le 4004(2300 transistors).

Pour permettre un bon échange, un traitement de données et une communication à la volée, ces capteurs sont reliés en réseau avec différentes techniques d'auto-organisations. Par conséquent, une variété de topologies peuvent exister pour former un réseau de surveillance temps réel appelé réseau de capteurs sans fils (*WSN, Wireless Sensor Network*). Un WSN peut atteindre une échelle de dizaine à plusieurs centaines de nœuds interconnectés. Cette densité permet de couvrir les pertes de nœuds qui peuvent survenir lors du déploiement du réseau ou au cours de son fonctionnement, qui sont due soit à un problème lié au hardware du capteur ou à des facteurs de l'environnement. Cependant, ces pertes ne constituent pas un problème majeur, vu que ces nœuds capteurs sont généralement dotés d'une résistance aux changements environnementaux et d'un mécanisme de tolérance aux fautes. Habituellement, la position des capteurs n'est pas arrangée ou prédéterminée. Le déploiement d'une manière aléatoire vise plus souvent les terrains accidentés ou se fait pour les opérations de désastre. Pour assurer cette possibilité de déploiement les protocoles et algorithmes doivent disposer de capacité d'auto-organisation. Comme il a était précédemment argumenté, un WSN peut être organisé selon plusieurs hiérarchies, visant à obtenir une meilleure coopération entre les nœuds capteurs. De telle manière que les nœuds n'envoient pas de données brutes aux nœuds responsables de la fusion, ils utilisent leurs propres possibilités de traitement local et transmettent seulement les données traitées ou partiellement traitées.

Une large variété d'applications peut être assurée grâce à l'apport émergent des WSN, la détection et la surveillance des désastres, le contrôle de l'environnement, le bâtiment intelligent, la médecine et la santé qui promettent un accompagnement et une surveillance totale des malades et des personnes âgés. Toutefois, les WSN sont assujettis à des contraintes fortes et de nature multiples, énergétiques et calculatoires, entre autre, ce qui limite les services fournis par les nœuds capteurs. Des mesures sont prises pour une moindre consommation en énergie lors du routage, agrégation des données et d'autres traitements accomplis par les capteurs, et des stratégies sont élaborées pour balancer les tâches entre les nœuds capteurs et permettre un équilibre énergétique et une durée de vie plus importante du réseau.

Un défi majeur qui influe directement sur la bonne marche du réseau est sa possibilité d'accomplir ses tâches en toute sécurité, sans la perturbation d'un agent externe malveillant. Les capteurs doivent être équipés de mécanismes de robustesse pour pallier aux problèmes de trafic injecté par les intrus, les empêcher d'analyser le réseau et de se servir de ses failles et vulnérabilités. La robustesse peut être en terme de sûreté pour permettre aux capteurs d'être en mesure de résister aux mauvais fonctionnements (pannes, erreurs) et aux aléas de l'environnement, ou en terme de sécurité pour envisager des techniques de résistance aux attaques et aux actes de malveillance. Beaucoup d'outils intelligents ont été proposés pour

permettre une bonne réflexivité (capacité d'analyser et d'auto-diagnostiquer son état) et adaptabilité (capacité d'adapter, auto-organiser et planifier son comportement en fonction d'objectifs de robustesse et de performance) d'un capteur.

Dans ce document, nous présenterons une étude sur la sécurité des réseaux de capteurs, ses critères, les attaques sur les différentes couches, nous mettrons l'accent sur la confiance dans les réseaux de capteurs et nous terminerons par la présentation de notre contribution. Ce document est organisé comme suit, nous consacrerons le premier chapitre aux réseaux de capteurs sans fils, dans le deuxième chapitre nous ferons une étude détaillée sur la sécurité dans les réseaux de capteurs, nous enchainerons, dans le troisième chapitre, avec la confiance dans les RCSF et nous présenterons notre contribution avec tests et évaluations, nous concluons en résumant ce qui a été fait et nous terminerons par des perspectives pour la poursuite de nos travaux.

Chapitre 1

Réseaux de capteurs

Abstract

Wireless Sensor Networks *consist of spatially distributed autonomous sensors to monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants and to cooperatively pass their data through the network to a main location.*

They are composed of huge number of sensor nodes, so it can be used as an effective tool for gathering data in various situations. They are used in many applications, in military, ecological, ocean and wildlife monitoring, manufacturing machinery performance monitoring and health related areas. However WSNs suffer from many constraints, including limited energy resources, limited memory and the use of insecure wireless communication channels. In this section we present an overview of Wireless Sensor Networks, we outline the characteristics, constraints, standards and specifications. We also point out the open researches issues and important challenges in WSNs.

Keywords: *Wireless Sensor Network, Sensor Network Applications, Routing, Standards, WSN Challenges.*

1.1 Introduction

Un réseau de capteur est formé d'un large nombre de nœuds capteurs minuscules, avec des batteries intégrées, qui sont configurés pour détecter des événements spéciaux, collecter les informations qui y sont relatives et les communiquer à la station de base via un routage multi sauts et à travers un medium sans fil. Captage, traitement et communication sont les trois éléments clé combinés dans un seul capteur donnant lieu à un vaste nombre d'applications. Les réseaux de capteurs fournissent un large éventail d'opportunités, mais en même temps posent de formidables défis, comme par exemple le fait que l'énergie est alarmante et souvent non renouvelable, la mémoire couvrant une faible étendue et d'autres contraintes qui exigent une soigneuse gestion de ressource et des techniques de communication pertinentes pour une meilleure longévité.

1.2 Architecture

Un réseau de capteurs est une infrastructure composé de nœuds capteurs capable de capturer, mesurer et communiquer des données relatives aux champs de déploiement, donnant ainsi à l'administrateur l'habilité d'instrumenter, observer et réagir aux événements et phénomènes dans un environnement spécifique. L'administrateur peut être un civile, un gouvernement ou une entité industrielle ou commerciale. L'environnement peut être le monde physique, un système biologique ou une structure technologique. Bien que dans différents champs et environnement, tous les réseaux de capteurs sont supposés avoir une même architecture générale. Dans la figure suivante est présentée l'architecture générale d'un réseau de capteur.

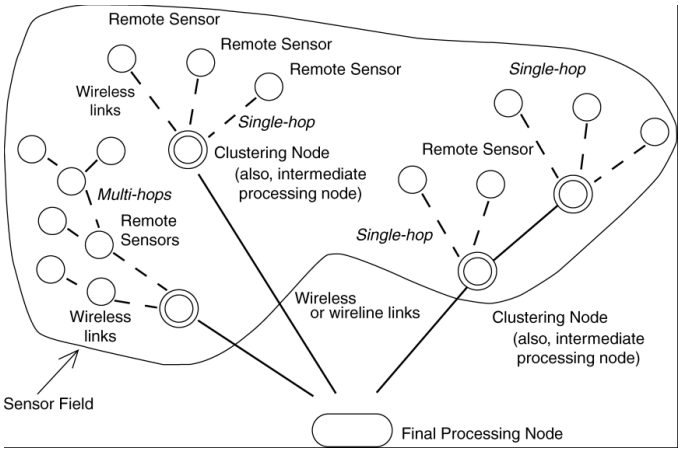

Figure 1 **Architecture générale d'un réseau de capteurs sans fils**

Les capteurs sans disséminés à l'intérieur du champ du captage (*Sensor Field*) ou tout prés du phénomène à mesurer. Ils s'auto-organisent en clusters pour collaborer à l'accomplissement d'une tâche issue de l'utilisateur. La position des nœuds n'est pas toujours prédéfinie. Une fois les applications de captage génèrent une large quantité de données, ces

données seront sujettes à la fusion et l'agrégation avant d'être relier au nœud responsable (*Sink Node*), pour un gain de temps et d'énergie. Les nœuds senseurs utilisent leur capacité de traitement pour envoyer des données partiellement traitées. Le nœud sink servit comme une passerelle à l'extérieur du réseau utilisant un autre réseau qui peut être filaire ou sans fils, comme internet par exemple. Pour envoyer les données au nœud sink, un nœud peut utiliser un seul saut (Single-hop) ou plusieurs sauts (Multi-hop) dépendamment de la position du nœud par rapport au sink. Un réseau de capteur avec de telles capacités peut fournir à l'utilisateur final des résultats avec une intelligence et une compréhension parfaite de l'environnement.

1.3 Anatomie d'un micro-capteur

Un nœud capteur est principalement composé de quatre unités de base : la sonde ou l'unité de captage, l'unité de traitement, l'unité de transmission, et l'unité de contrôle d'énergie. Il peut contenir également, suivant son domaine d'application, des modules supplémentaires tels qu'un système de localisation (GPS), ou bien un système générateur d'énergie (cellule solaire ou photovoltaïque). On rencontre aussi des micro-capteurs, un peu plus lumineux, dotés d'un système mobilisateur chargé de déplacer le micro-capteur en cas de nécessité.

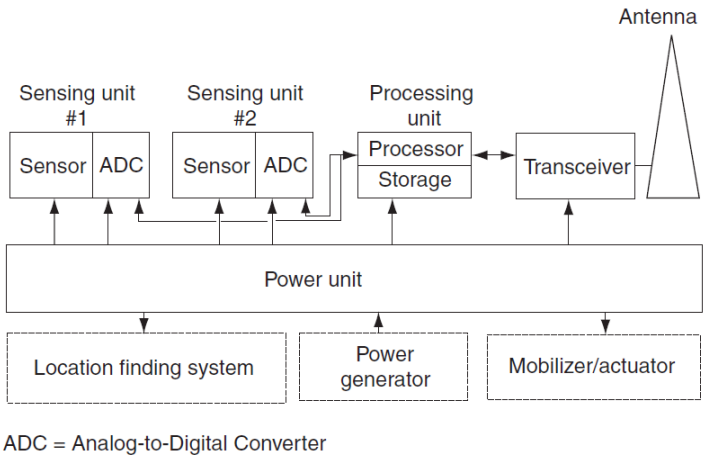

Figure 2. Architecture d'un micro-capteur

1.3.1 Sonde ou unité de captage

La sonde, c'est à dire le capteur proprement dit, est un dispositif transformant l'état d'une grandeur physique observée en une grandeur utilisable par exemple, une tension électrique, une intensité lumineuse ou encore une température.

1.3.2 Unité de traitement

Elle comprend une MCU avec un processeur intégré contenant des programmes, mémoires, timers, ports entré/sortie configurables, convertisseur analogique-Numérique (ADC) et d'autres périphériques. Elle fonctionne à l'aide d'un système d'exploitation spécialement conçu pour les micro-capteurs (TinyOS par exemple).

1.3.3 Unité de transmission

Elle est composé d'un émetteur récepteur (tranceiver) radio et d'une antenne. Elle effectue toutes les émissions et réceptions des données sur un medium sans fil.

1.3.4 Unité de contrôle d'énergie

Un micro-capteur est muni d'une ressource énergétique (généralement une batterie). Étant donné sa petite taille, cette ressource énergétique est limitée et généralement non-remplaçable. Elle doit répartir l'énergie disponible aux autres modules, de manière optimale en réduisant les dépenses inutiles et en mettant en veille les composants inactifs [1].

1.4. Plateformes existantes

Des nœuds pour les réseaux de capteurs sans fils ont été conçus par plusieurs groupes de recherches durant les dernières décennies. Suite aux avancements dans les technologies de communication et les traitements low-power, les plateformes ont été améliorées significativement avec le temps.

Puisque la présence de toutes les exigences des WSNs dans une seule plateforme s'avère impossible, les recherches sur les plateformes ont été divisées en deux branches :

Les plateformes à haute performance (High-performance Platform) et les plateformes à basse énergie (low-power Platform).

Les plateformes à haute performance sont typiquement développées pour les recherches complexes de traitement et fusion de données dans un nœud capteur. Ces plateformes utilisent des MCUs à haute performance qui ont des dizaines en MIPS de performance de traitement et des milliers en kilooctets de programmes et mémoires données.

Exemple de plateforme à haute performance : Piconode (Reason and Rabaey 2004) [2], µAMPS (Min et al. 2002) [3], et Stargate(Crossbow Technology 2004)[4].

Les plateformes à énergie basse ont pour objectif de maximiser la durée de vie et minimiser la taille physique d'un capteur. Celles-là sont obtenues en minimisant la complexité hardware et la consommation énergétique. Les travaux les plus connus parmi low-power plateformes des WSN ont été conduits à l'université de Californie, Berkeley. Ils ont créé un

nombre de plateformes nommées motes à travers le projet SmartDust (Warneke et al. 2001)[5]. En addition, d'autres projets de recherche ont développés des plateformes WSN intensives. Medusa MK-2(Savvides et Srivastava 2002) [6] est une plateforme sortie en 2002, par l'université de Californie et Los Angeles [7, 8].

1.5. Communication

Le modèle de couches Open System Interconnections (OSI, Stallings 2004) [9] est le plus utilisé. Il comprend sept couches : Application, présentation, session, transport, réseau (network), liaison de données (data link), et la couche physique. A cause des contraintes des applications des réseaux de capteurs qui diffèrent significativement des applications bureau, les WSNs n'utilisent pas toutes les couches définis dans le modèle OSI. La pile protocolaire des WSNs est présentée par 5 couches comme montré dans la figure 3. Le plus notable, la couche session est souvent non utilisée et les protocoles transport de contrôle de flux end-to-end sont rarement utilisés.

Dans les WSNs, les couches les plus essentielles sont la couche physique, le protocole MAC dans liaison de données (DLL, Data Link Layer) et le protocole de routage dans la couche réseau. Dans ce qui suit, nous présentons les cinq couches de la pile protocolaire en détail.

1.5.1 Pile protocolaire

La pile de protocoles utilisée pour un nœud puits (sink) et un nœud senseur est présentée ci-dessus. Cette pile de protocoles combine avec promptitude énergie et routage. La pile est composée de : Couche physique – Couche lien de données – Couche réseau – Couche de transport – Couche application – Plan de gestion de l'énergie – Plan de gestion de la mobilité – Plan de gestion de la tâche.

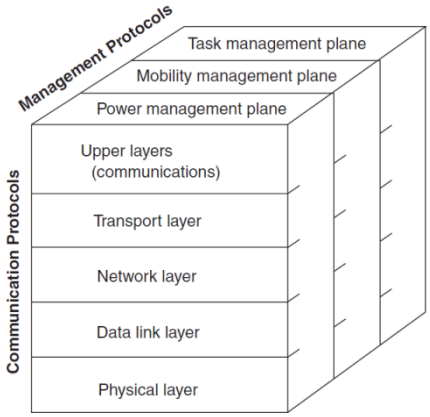

Figure 3. Pile protocolaire des réseaux de capteurs

1.5.1.1 Couche physique

La couche physique est responsable de la sélection de fréquence, de la génération de la fréquence de porteuse, de la détection du signal, de la modulation et du cryptage de la donnée [10,11].

1.5.1.2 Couche lien de données

Elle est responsable du multiplexage des flots de données, de la détection de la trame de données, du MAC et du contrôle d'erreurs. Elle assure de manière fiable les connexions point-à-point et point-multipoint dans un réseau de communication [12, 13].

1.5.1.3 Couche réseau

Les nœuds senseurs sont répartis de manière dense dans un champ soit près d'un phénomène ou à l'intérieur de ce phénomène à observer. De spéciaux algorithmes de routage sans fil multi-sauts entre les nœuds et le puits (sink) sont nécessaires. Différents schémas ont été proposés pour les réseaux de senseurs se rapportant à la couche réseau :

☐Small Minimum Energy Communication Network [14],
☐Flooding [15],
☐Gossiping [16],
☐Sensor Protocols for Information via Negotiation [15],
☐Low Energy Adaptive Clustering Hierarchy, LEACH [17,18],
☐Directed Diffusion [19],
☐SMECN [20],
☐SPIN [15],
☐SAR [21].

Ces protocoles sont à améliorer en prenant en considération les grands changements topologiques et la forte scalabilité.

1.5.1.4 Couche transport

Cette couche est nécessaire pour les systèmes planifiés à être accédés via Internet ou d'autres réseaux externes. Il n'y a pas de proposition de recherche dans la littérature pour les réseaux senseurs.

1.5.1.5 Couche application

Plusieurs domaines d'applications pour réseaux de senseurs ont été définis et proposés mais les protocoles de la couche application pour les réseaux de senseurs demeurent une région inexplorée. Trois protocoles de couche application possibles sont à examiner :

☐SMP : Senseur Management Protocol [22]
☐TADAP : Task Assignment and Data Advertisement Protocol,
☐SQDDP : Sensor Query and Data Dissemination Protocol.

Ces protocoles de la couche application demeurent des thèmes ouverts à la recherche.

1.5.1.6 Plan de gestion d'énergie

Il montre comment un nœud senseur doit utiliser son énergie. Par exemple, le senseur met son Emetteur/Récepteur en mode Off après avoir reçu un message d'un de ses voisins. Cela permet d'éviter de recevoir des messages dupliqués. En plus, quand le niveau d'énergie d'un nœud senseur est bas, le senseur avise ses voisins par un broadcast que son niveau d'énergie est bas et donc il ne peut participer aux messages de routage. Le reste de l'énergie est réservé à la détection.

1.5.1.7 Plan de gestion de la mobilité

Il détecte et enregistre le mouvement des nœuds senseurs ainsi le routage de retour à l'utilisateur est toujours maintenu et les nœuds senseurs peuvent garder la trace de leurs voisins senseurs. En connaissant leurs voisinages, les nœuds senseurs peuvent balancer leur énergie et la tâche d'usage.

1.5.1.8 Plan de gestion de tâches

Il balance et ordonnance les tâches de détection pour une région spécifique. Ce ne sont pas tous les nœuds de cette région qui sont nécessaires à la tâche de détection en même temps. Comme résultat, certains nœuds senseurs exécutent cette tâche plus que les autres en fonction de leurs niveaux d'énergie.

Ces plans de gestion sont nécessaires pour que les nœuds puissent travailler ensembles pour une meilleure efficacité énergétique, pour router la donnée dans un réseau de senseurs et partager les ressources entre les nœuds senseurs. Sans ces plans de gestion, chaque nœud senseur travaillera de manière individuelle. D'un point de vue d'un réseau de senseurs, il est plus efficace aux nœuds senseurs de collaborer pour pouvoir prolonger la durée de vie du réseau.

1.5.2 Standards offerts

Le but des ingénieurs WSNs est de développer des standards de communication sans fils qui garantit une basse consommation énergétique, la sécurité et la fiabilité. Parmi les standards les plus aptes à être exploités dans les réseaux de capteurs sans fil se retrouvent la double pile protocolaire Bluetooth / ZigBee. D'un point de vue matériel, de nouvelles techniques vont influencer considérablement l'avenir des réseaux de capteurs. UWB (Ultra Wide Band) est un très bon exemple. Cette technique de transmission permettra d'atteindre des niveaux de consommation extrêmement bas grâce à sa simplicité au niveau matériel.

1.6 .Applications des WSN

Un grand terrain d'applications est possible avec les réseaux de capteurs sans fils. Le monitoring environnemental, applications militaires, éducation des enfants, la microchirurgie et l'agriculture ne sont que peu d'exemples d'applications.

1.6.1 Environnement

Grâce aux efforts joints de l'université de Californie, Berkeley et l'université d'Atlantique, le monitoring environnemental est réalisé à côté de Maine à Great Duck Island en installant un réseau équipé des motes de Berkeley [23]. Ce n'est qu'un début pour élargir cette idée dans le projet Pods Project à l'université d'Hawaii [24], dont les données environnementales (température, lumière, vent et l'humidité relative à l'environnement) sont collectées par des capteurs relatifs aux climat. La majeure préoccupation des chercheurs dans ce sens est le camouflage des capteurs pour les rendre invisibles aux touristes curieux.

Dans le projet Princeton's Zebranet [25], un réseau de capteur dynamique a été créé en attachant de spéciaux colliers équipés avec un système GPS low-power aux cous des zèbres pour collecter leurs mouvements et comportements. Depuis, ces réseaux sont désignés à opérer dans un environnement infrastructure, un échange Peer-to-Peer d'informations est utilisé pour produire des bases de données redondantes, cela était possible car les chercheurs ont recours à un nombre limité de zèbres.

1.6.2 Phénomènes naturelles

Les réseaux de capteurs peuvent être utilisés pour le monitoring des phénomènes naturels dont la présence humaine est déconseillée, comme les ouragans et les feux de forets. Les efforts unis entre l'université de Harvard, l'université de New Hampshire et l'université de Carolina du nord ont mené un projet de déploiement d'un réseau de capteurs pour le monitoring des éruptions du volcan Tungurahua, un volcan au centre de l'Ecuador. Un réseau des motes de Berkeley à collecté les infrasons[3] durant les éruptions et les données étaient transmises à travers un lien sans fil à une station de base situant plus de 9Km à l'observatoire volcanique [26].

1.6.3 Agriculture

Le vignoble sans fil d'Intel [27] est un exemple d'utilisation de l'informatique ubiquité pour l'agriculture. Dans cette application, on n'attend pas du réseau que la collecte et l'interprétation des données, mais aussi d'utiliser les données pour former des décisions

[3] Vibration acoustique de fréquence inférieure à 15 hertz, inaudible pour l'oreille humaine

visant à détecter la présence des parasites et permettant l'utilisation d'un type approprié d'insecticide.

Un autre projet au Etats-Unis, les éleveurs de bovins disposent de troupeaux de grande taille dont la supervision manuelle nécessite de grands efforts. L'application ZigBeef [28] vient pour faciliter cette tâche. Les bovins sont identifiés et équipés (à l'oreille) de capteurs Zigbeef comportant plusieurs informations. Avec cette application l'éleveur bénéficie de deux types de services : l'identification des bovins qui sont dans un pâturage et la supervision de leur état de santé.

Figure 4. Les capteurs ZigBeef[28]

1.6.4 Industrie

Un grand apport des réseaux de capteurs dans la supervision des pipelines, le pétrole et le gaz jouent un rôle important dans notre vie quotidienne ce qui impose la sécurisation des pipelines qui assurent le transport de ces produits. Leur longueur impressionnante, leur accès souvent difficile et le risque qu'ils représentent rendent cette tâche critique [29].

PipeNet [29,30] a été développé en collaboration entre Imperial College (université) à Londres, Intel Research et le MIT pour superviser et sécuriser les pipelines. PipeNet assure deux fonctions : la mesure de la pression et du pH de l'eau et des liquides et la supervision du niveau de l'eau dans le système des égouts.

1.6.5 Surveillance des humains

Les réseaux de capteurs peuvent être aussi utilisés pour le monitoring du comportement humain. Dans le Smart Kindergarten à UCLA [31], des jouets avec capteurs intégrés, connectés en réseau sans fil et d'autres objets de classe supervisent l'apprentissage des enfants et permettent un monitoring discret par l'enseignant.

Les recherches médicales peuvent grandement bénéficier des réseaux de capteurs : le monitoring des signes vitaux et la reconnaissance des accédants sont les applications les plus naturelles [32,33].

1.6.6 Monitoring Militaire

Une abondance d'applications militaires, un exemple intriguant les mines de DARPA [34], une auto-organisation de capteurs où une communication Peer-to-Peer entre les mines est utilisée pour répondre aux attaques et répartir les mines dans l'ordre de compliquer la progression des troupes de l'ennemi [35].

1.6.7 Sécurité routière

Dans le projet de contrôle de voitures CORTEX (Sihavaran et al., 2004) [36], le système implémenté sélectionne automatiquement la route optimale selon le temps désiré pour atteindre la destination, distance, l'état courant ou prévu de la circulation, les conditions du temps(métio), et toute autre information qui pourra être nécessaire.

Le projet Safe Traffic (Svensoon, 2005) [37] vise l'implémentation d'une infrastructure de communication intelligente. Ce système de communication doit fournir les informations nécessaires sur tous les véhicules, personnes et les objets localisés prés de la route pour une circulation saine.

1.6.8 La maison intelligente

Smart Home, une merveilleuse idée pour l'automatisation d'une maison est l'utilisation de l'habileté de tourner la lumière en On et Off automatiquement, monitorer le sommeil d'un bébé sans être dans la chambre, et avoir une tasse fraiche d'un café chaud dans la cuisine pour son petit déjeuner (CRUISE, 2006) [38].

Afin d'atteindre les objectifs de déploiement de ces applications et garantir leurs bon fonctionnement, les concepteurs doivent faire face à plusieurs contraintes. Dans l'application Zigbeef par exemple, la durée de vie moyenne d'une bête est de l'ordre de huit ans. Les capteurs doivent donc fonctionner sans interruption pendant toute cette durée. D'autre part, pour les applications PipeNet et soins à domiciles, les capteurs doivent avoir en plus une connectivité constante et stable. Les liens radio doivent être alors robustes. En effet, plusieurs contraintes régissent le fonctionnement des réseaux de capteurs. Un bref aperçu des contraintes et challenges fera l'objet de ce qui suit.

1.7. Caractéristique et Challenges

Dans un réseau ad hoc, les nœuds sans fils s'auto-organisent en un réseau sans infrastructure avec une topologie dynamique. Les réseaux de capteurs partagent cette caractéristique, mais montrent beaucoup de caractéristiques qui les distinguent des autres types de réseaux sans fils.

1.7.1 Durée de vie

La durée de vie est extrêmement critique pour beaucoup d'applications et le facteur primaire qui la limite est l'énergie. Bien que, il est assumé que l'énergie de transmission prend la part du lion de la consommation globale, la sonde, traitement des signaux et d'autres opérations hardwares en mode stand-by ont leur consommation consistante [39,40].

1.7.2 Energie

La durée de vie d'un réseau est liée directement à la consommation énergétique du nœud capteur. Dans beaucoup de cas le capteur sans fil a une source limitée (<500mA, 1.2 V), donc les pertes énergétiques inutiles sont à éviter.

1.7.3 Coût d'un capteur

Le cout d'un capteur est critique pour le cout total de déploiement d'un réseau de capteur. Clairement, le cout d'un capteur doit être pris que les autres métriques intervenant dans le fonctionnement d'un réseau de capteur. Les capteurs basés Bluetooth coutent environ 10$.

1.7.4 Environnement

Les réseaux de capteurs sont prévus à opérer dans des endroits inaccessibles par l'homme, et donc sans surveillance. Dans ce type d'environnements, un défi est d'apporter des mécanismes de management du réseau fiables. A côté de ça, la grande échelle de nœuds engendre un déploiement dense dans la proximité ou l'intérieur de l'environnement à observer.

1.7.5 Architecture tierce (energy, size and price)

Bien que la loi de Moore prédit que le hardware d'un senseur doit devenir de plus en plus minuscule, cheaper, et plus puissant, le couplage de ces trois critères reste un défi. Malgré que les métriques « fast » (rapide) et small (petit) évoluent, mais il faut toujours trouver un compromis : les nœuds ont besoin d'être rapide (faster) ou plus efficient en terme d'énergie, petit (smaller) ou capable, cheaper ou plus durable. Le choix d'une seule plateforme hardware devra faire un compromis. Les plateformes légères les plus connues actuellement sont les motes Berkeley basées TinyOS[41,42].

1.7.6 Routage

Plusieurs familles de protocoles de routage ont été développées pour les réseaux de capteurs. Entre ces familles on retrouve les protocoles dont le but est de minimiser la consommation énergétique moyenne, en utilisant par exemple des algorithmes d'acheminement des paquets, et qui prennent en compte le critère de performance et de consommation électrique.

Par exemple (Energy aware routing for low energy ad hoc sensor networks) [43], où Shah et Rabaey proposent d'utiliser une fonction de probabilité dépendant de la consommation énergétique qu'exige chaque route pour sélectionner les chemins favorisant l'allongement de la durée de vie du réseau.

Une autre famille regroupe pour sa part les protocoles dont le but est de limiter les flux applicatifs transitant sur le réseau. En minimisant ainsi les flux, le protocole permet d'économiser de l'énergie. Cette famille intègre les protocoles de type « data-centric » comme SPIN [44], Directed Diffusion [45,46], Rumor Routing [47], COUGAR [48] et ACQUIRE [49]. Elle comprend aussi les protocoles de type « Network flow et QoS-aware protocols » comme l'algorithme « Maximum Lifetime Data Aggregation (MLDA) » [50], CMLDA [51], le protocole Minimum cost forwarding [52], Sequential assignment routing (SAR) [53] et SPEED [54].

Il y a aussi les protocoles de routage géographique où les informations de localisation sont nécessaires soit pour orienter les paquets vers une destination bien précise ou bien pour prendre en compte les distances inter-nœuds afin de mieux estimer l'énergie consommée. Parmi les protocoles de routage géographique, on peut citer de manière non exhaustive : Minimum Energy Communication Network (MECN) [55], SMECN [56], Geographic Adaptive Fidelity (GAF) [57], Geographic and Energy-Aware Routing (GEAR) [58] et Greedy Perimeter Stateless Routing (GPSR) [59].

1.7.7 Organisation en clusters

L'organisation des réseaux de capteurs sans fil en clusters est une forme d'approche de « Control Topology » qui par le biais de l'agrégation de nœuds en clusters, permet de réduire la complexité des algorithmes de routage, d'optimiser les ressources réseaux, de faciliter l'agrégation en faisant gérer, localement, certaines fonctionnalités par un chef de clusters [60].

1.7.8 Sécurité

Dans les réseaux de capteurs, beaucoup de gageures sont à relever pour assurer la sécurité des nœuds senseurs et des données générées. Par exemple, le fait d'embarquer des nœuds dans un environnement présente des problèmes de : la sécurité physique, l'intégrité de données, les communications risquées, ...etc. Cela peut faire de la sécurité des WSNs significativement différente de celle des réseaux conventionnels. Les attaqueurs peuvent

modifier le hardware d'un nœud, en le remplaçant par un homologue malicieux, ou injecter dans des nœuds idiots des requêtes qui ne reflètent pas l'environnement à surveiller. Sécuriser un large réseau en prenant en compte le maximum de critères de sécurité, en assurant l'exécution de tous les contrôle relatives à la sécurité envisagée par tous les nœuds du réseau, et en détectant toutes les attaques qui peuvent survenir est un défi difficile à relever. Du fait que la sécurité ne peut pas être garantie en sécurisant quelques nœuds du réseau.

Les ressources limitées dans les nœuds senseurs minuscules peuvent poser problème. Elaborer de nombreux schémas de cryptage est peu réaliste, vu les ressources intensives, énergie, mémoire, et le temps nécessaire pour envoyer des données brutes (données cryptées, c-à-d, elles n'auront recours ni à l'agrégation, ni au traitement). De plus, la protection contre l'écoute indiscrète (eavesdropping) est particulièrement importante dans les WSNs. Traditionnellement cela signifie un cryptage de bout-en-bout. Ce qui veut dire que la donnée est cryptée, dés qu'elle est créée, transmise à travers le réseau, et finalement reçue par un nœud sécurisé, où les clés de décryptage peuvent être enregistrées sans risques ou exposition.

Malheureusement, une énergie limite qui actionne les besoins de traitements des senseurs, par conséquent, il ne faut pas confondre les schémas de sécurité des WSNs avec ceux traditionnels. Les nœuds à l'intérieur du réseau ne peuvent pas performer les applications de traitement des algorithmes des données cryptées.

Décrypter les données dans chaque nœud, signifie que les clés de décryptage sont stockées au niveau de chaque nœud, malheureusement, le nœud lui-même est exposé, et il ne peut pas assumer d'être en dehors de l'atteinte de l'attaqueur. Dans beaucoup de technologies d'information, les réseaux de capteurs soulèvent plusieurs questions à propos de la sécurité des nœuds senseurs. Certains aspects de sécurité ont été progressivement érodés, cela est dû à une variété de forces : par exemple, les engins de recherches sur Internet qui semblent omnisciences. Les réseaux de capteurs, similaire aux autres réseaux, est une technologie qui peut être utilisée pour enrichir et améliorer notre vie, ou tourner d'une manière invasive. Autant que les WSNs deviennent plus répondus, ils deviendront un point important à considérer dans le débat continu entre information et vie privée.

1.8. Conclusion

La miniaturisation avancée des appareils électroniques low-power et low-cost a mené une progression dans les recherches concernant les réseaux à large échelle, sans fils, de capteurs et actionneurs low-power. Une aptitude de captage provenant d'un ensemble de capteurs légers révolutionne le monde qui nous entoure. Les capteurs nous préviennent de contaminants invisibles dans l'air qu'on respire ou dans l'eau qu'on boit, détectent un tremblement de terre prés à se produire, et beaucoup d'applications qui étaient avant irréalisables sont devenues possibles. Les WSNs peuvent ouvrir le chemin vers de nouvelles générations de scientifiques pour des phénomènes qui n'étaient jamais avant observables, permettant une compréhension à l'environnement. Les réseaux de capteurs seront éventuellement intégrés dans nos maisons, notre milieu de travail, …. d'une manière qu'on ne peut pas imaginer aujourd'hui. Peut être, un jour, un capteur électronique sera utilisé naturellement comme tout autre objet, et sera pour nous un sens inné. Que le temps qui racontera.

Chapitre 2

Sécurité des RCSFs

Abstract

Security *is a major concern in WSNs. Since the wireless links are susceptible to various attacks against the network. All security approaches require a certain amount of resources for the implementation, including data memory, code space, and energy to power the sensor. However, due to inherent resource and computing constraints, security in sensor networks poses different challenges than traditional networks security. It is difficult to directly employ the existing security approaches to the area of wireless sensor network: currently these resources are very limited in a tiny wireless sensor. Therefore, to develop useful security mechanisms while borrowing the ideas from the current security techniques, it is necessary to know and understand these constraints first. In this section, we identify the vulnerabilities associated with the operational paradigms currently employed by Wireless Sensor Networks. A survey of current WSN security research is presented. First we outline the constraints, security requirements, and attacks with their corresponding countermeasures in WSNs. We then present a holistic view of security issues. Along the way we highlight the advantages and disadvantages of various WSN security protocols and further compare and evaluate these protocols based on different approaches. We also point out the open research issues in each sub-area and conclude with possible future research directions on security in WSNs.*

Keywords: *Security, WSNs, attacks, vulnerabilities.*

2.1 Introduction

Les WSNs ont attiré beaucoup d'attention suite à leur grand potentiel d'être utilisés dans plusieurs applications. Comparant aux réseaux basés infrastructures existants, les WSNs peuvent travailler dans n'importe quel environnement, spécialement quand les connexions filaires sont impossibles. Bien que beaucoup de propositions ont été reporté concernant la sécurité dans les réseaux de capteurs, mais ça reste un point critique. Les exigences de sécurité dans les réseaux de capteurs imposent des contraintes couteuses vu les ressources limitées d'un nœud capteur. Dans ce contexte, beaucoup de chercheurs ont entamé des techniques permettant de maximiser les capacités de traitement, conserver de l'énergie des senseurs, et aussi de sécuriser le réseau contre les attaques. Tous les aspects d'un WSN doivent être examiner, y compris un routage efficient et sécurisé[61,62,63,64], agrégation de données[65,66,67,68,69,70], la formation de groupes[71,72,73], ainsi de suite.

Le but de ce chapitre est de présenter le contexte général des mécanismes de sécurité pour les WSNs. Les recherches dans ce domaine, qui sont minimal, seront présentées. Les menaces contre les WSNs seront identifiées. Un état de l'art sur la sécurité dans les réseaux de capteurs sera détaillé, avec une présentation des récentes idées, enfin pour terminer des recommandations seront proposées.

2.2 Menaces

Beaucoup d'attaques et vulnérabilités menacent les WSNs, y compris les fautes de conception et les dommages non prévus des facteurs environnementaux. Entre les designs proposés pour les WSNs, la sécurité est l'une des plus importants aspects qui doit prendre beaucoup d'attention, si on considère les opportunités des applications énormes. Cette section conduit les lecteurs vers ce domaine vaste en présentant une étude des différentes attaques potentielles dans les WSNs. Pour éclaircir cette présentation, nous classons d'abord les attaques en se basant sur leurs natures et origine. Par la suite on présentera les attaques les plus connues. On détaillera les mécanismes et effets de ces attaques et on termine avec quelques contremesures potentielles.

2.2.1 Classification des attaques par nature et origine

Dans les réseaux de capteurs, un attaquant peut effectuer une variété d'attaques n'ayant pas forcément le même objectif ou motivations. Ainsi le choix d'une stratégie de sécurité doit se baser sur une modélisation de l'attaque, ceci afin d'éviter un déploiement excessif de moyens de protection conduisant à des solutions irréalistes. Selon (Yong, et al., 2006), les attaques sur les réseaux de capteurs peuvent être classifiées dans les catégories suivantes :

2.2.1.1 Attaques passives VS attaques actives

Les attaques passives "eavesdropping" se limitent à l'écoute et l'analyse du trafic échangé. Ce type d'attaque est plus facile à réaliser (il suffit de posséder un récepteur adéquat) et il est difficile de le détecter puisque l'attaquant n'apporte aucune modification sur les informations échangées. L'intention de l'attaquant peut être la connaissance des informations confidentielles ou bien la connaissance des nœuds importants dans le réseau (chef de groupe, "cluster head"). En analysant les informations de routage, l'attaquant va se préparer à mener ultérieurement une action précise.

Dans les attaques actives, un attaquant tente de supprimer ou modifier les messages transmis sur le réseau. Il peut aussi injecter son propre trafic ou rejouer d'anciens messages pour perturber le fonctionnement du réseau ou provoquer un déni de service.

2.2.1.2 Attaques externes VS Attaques internes

Dans le cas de l'attaque externe, le nœud attaquant n'est pas autorisé à participer dans le réseau de capteurs. Des techniques de cryptographie et d'authentification protègent l'accès au réseau à ce type d'attaquant. Cependant ce dernier peut uniquement déclencher des attaques passives tels que l'écoute clandestine, le brouillage radio, ou l'attaque par rejoue. L'attaque interne est considérée comme la plus *dangereuse* du point de vue sécurité. Puisque l'attaquant qui capture un nœud, peut lire sa mémoire et avoir accès à son matériel cryptographique et par conséquent peut s'authentifier comme un nœud légitime et émettre des messages aléatoires erronés sans qu'il soit identifié comme intrus, puisqu'il utilise des clés valides. Les méthodes cryptographiques s'avèrent donc inefficace pour ce genre d'attaque. Il est donc nécessaire d'utiliser d'autres méthodes complémentaires telles que les systèmes de monitoring et les systèmes de réputation.

2.2.2 Les attaques les plus connus

2.2.2.1 Subversion of a Node (Subversion d'un nœud)

Si un nœud senseur est détecté, il peut être tempéré, interrogé électroniquement ou compromis. Une fois compromis, le nœud senseur peut divulguer son matériel de la clé cryptographique et accéder aux hauts niveaux de communication et la fonctionnalité du senseur peut devenir valable à l'attaqueur.

2.2.2.2 Traffic Analysis (Analyse du Trafic ou collecte active d'informations)

Bien que les communications puissent être cryptées, une analyse des scénarios de communication et d'activités des senseurs peut dévoiler assez d'informations pour permettre à l'adversaire de bloquer le réseau ou arrêter sa mission. Les informations d'adressage et de routage transmises en clair souvent contribuent à l'analyse du trafic.

L'analyse du trafic est une question qui a attirée occasionnellement l'attention des auteurs [74,75,76]. L'analyse du trafic est le terme utilisé pour le processus de déduction d'informations à propos des communications d'un réseau cible crypté. Malgré qu'on ne peut pas lire le contenu des messages cryptés, l'analyste examine l'extérieur, la station à laquelle on envoie, à qui les messages sont envoyés ? Et éventuellement les scénarios d'activité.

Les WSNs sont spécialement vulnérables pour telles types d'attaques, du fait que la transmission sans fils est la méthode dominante utilisée par les senseurs pour échanger les données. Durant la transmission, les signaux sans fils sont diffusés dans l'air, et ainsi ils sont accessibles au public. La capacité d'écoute dépend de la puissance des antennes. Cette écoute est un comportement passif, de telles attaques sont rarement détectables [77, 78,79].

2.2.2.3 Denial of Service (Déni de Service)

L'attaque déni de service dans un WSN peut prendre plusieurs formes et elle concerne pratiquement toutes les couches. Une telle attaque peut être un jamming du lien radio, un épuisement de ressources, ou mal-router les données. Karlof et Wagner [80] identifient les attaques DoS incluant: "Black Hole", "Resource Exhaustion", "Sinkholes", "Induced Routing Loops", "Wormholes", et "Flooding" qui sont dirigées contre les protocoles de routage employés dans les WSNs.

2.2.2.4 jamming (Brouillage radio)

L'attaque Jamming perturbe la disponibilité du media de transmission. L'approche est d'introduire des interférences intenses pour occuper le canal et priver les senseurs de la chance de communiquer. Avec un appareil de jamming, un adversaire peut perturber un réseau entier en déployant un nombre suffisant de tels appareils. Le problème de telles attaques est que les appareils de jamming ont le risque d'être identifiés par les senseurs, le fait de détecter un haut niveau de bruit.

2.2.2.5 Manipulation de trafic

La communication sans fils dans les WSNs (et les autres réseaux sans fils) peut être manipulée aisément au niveau de la couche MAC. Les attaqueurs peuvent transmettre des parquets au moment où les senseurs légitimes font de même pour causer des collisions excessives. La contention[4] augmentée diminue la qualité du signal et la disponibilité du réseau et elle réduit dramatiquement la robustesse du réseau en toutes ses parties [81, 82]. D'ailleurs, dans les schémas MAC largement utilisés où les transmissions sont soigneusement coordonnées, les attaqueurs peuvent participer à l'usage du canal et désobéir agressivement aux règles de coordination [83, 84,85].

4 Mode de fonctionnement d'un canal sur lequel plusieurs entités peuvent émettre en même temps. Il faut alors prévoir un système de gestion des collisions.

2.2.2.6 Spoofing de l'identité

MAC identity spoofing est une autre attaque dans la couche MAC [86]. Due à la nature des communications sans fils de diffuser les paquets, l'identité MAC (comme exemple : l'adresse MAC ou un certificat d'un senseur) est ouvert à tous les voisins, y inclus les attaqueurs. Sans une propre protection, un attaqueur peut falsifier une identité et prétendre d'être une entité du réseau. Une attaque typique à MAC identity spoofing dans l'attaque Sybil [87,88], dans lequel un attaqueur présente illégalement de multiples identités MAC. Pour accéder au réseau, un attaqueur peut envoyer des paquets en utilisant l'adresse d'un senseur légitime. Il peut aussi diffuser des messages en utilisant l'adresse de la station de base ou celle du point d'agrégation pour obtenir un privilège non autorisé. Si réussi, tous le réseau se paralyse.

Les attaques spoofing sont souvent la base des attaques cross-layer qui peuvent causer des problèmes sérieux. Par exemple, l'attaque Sybil [87,88] peut exposer des informations légitimes à l'adversaire ou fournir des fausses informations pour le routage pour le lancement des attaques faux routage.

2.2.2.7 Faux routage

Comme son nom l'indique, les attaques faux routage [89] sont lancées en enfonçant des informations fausses du routage. Il ya trois approches différentes d'enfoncement [89] :

- Overflowing routing tables (Déborder les tables de routage)
- Poisoning routing tables (Empoisonner les tables de routage)
- Poisoning routing caches (Empoisonner les caches de routage)

2.2.2.8 Black Hole

L'attaque trou noir est l'une des attaques les plus simples du routage dans les WSNs. Dans cette attaque, l'adversaire avale tous les messages qu'il reçoit, tout à fait comme un trou noir qui absorbe toute chose qu'on lui passe. En refusant d'acheminer tous les messages reçus, l'attaqueur affecte toute la circulation du trafic qui le traverse.

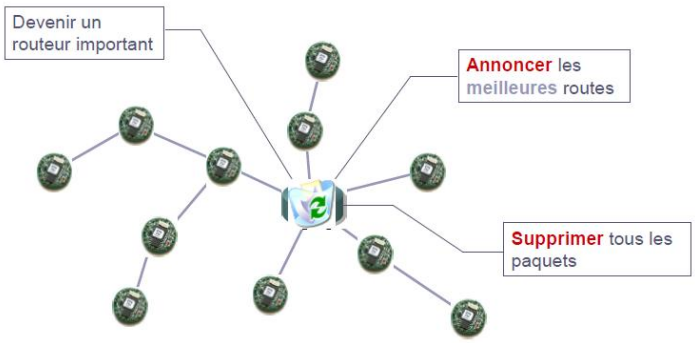

Figure 5. L'attaque BlackHole [90]

2.2.2.9 Grey hole (trou gris)

Une variante de l'attaque précédente est appelée trou gris, dans laquelle seuls certains types de paquets sont ignorés par le nœud malicieux. Par exemple, les paquets de données ne sont pas retransmis alors que les paquets de routage le sont.

2.2.2.10 SinkHole (trou de la base)

L'attaque Sinkhole est plus complexe [80] comparé avec Balck hole. Donnant quelques informations sur le protocole en utilisation, l'attaqueur essaye d'attirer le trafic d'une région particulière vers lui. Par exemple, l'attaqueur peut annoncer un meilleur chemin faux en diffusant des données fausses sur une puissance attractive, bande passante, ou route de haute qualité pour une région particulière.

2.2.2.11 Clock Skewing

La cible de cette attaque sont les senseurs qui ont besoin des opérations de synchronisation [91,92,93]. En disséminant des fausses informations du timing, l'attaque acte pour désynchroniser les senseurs (i.e. présente mal leurs clocks). Par exemple, dans IEEE 802.11 (qui peut être appliqué aux WSNs), les nœuds doivent être synchronisés avec le point d'accès. Les paquets beacon sont diffusés par le point d'accès périodiquement. Ces paquets contiennent les informations du timing qui doivent être utilisées par les nœuds pour l'ajustement de la clock. Les attaqueurs peuvent envoyer de faux paquets avec des fausses informations du timing [92, 94].

2.2.2.12 Distorsion de l'agrégation de données

Une fois les données sont collectées, les senseurs souvent les renvoient à la station de base pour les traitements. Les attaqueurs peuvent modifier les données à agréger malicieusement (Corruption de messages), et produisent les résultats finaux altérés d'agrégation pour les stations de base.

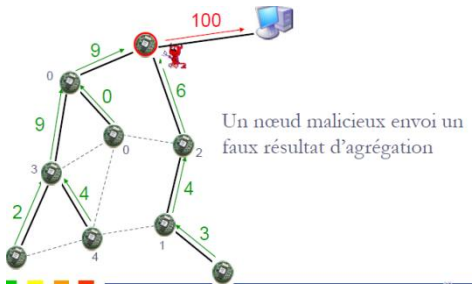

Figure 6. Distorsion d'agrégation de données [90]

2.3 Sécurité d'amorçage (bootsrapping) dans un réseau de capteurs

Un schéma d'amorçage pour un réseau de capteur doit satisfaire les exigences suivantes :

- les nœuds déployés doivent être capables d'établir une communication nœud-à-nœud sécurisée.
- Les nœuds additionnels légitimes déployés dans un temps futur peuvent former des connexions sécurisées avec les nœuds déjà déployés.
- Les nœuds non autorisés ne doivent pas être capables d'entrer au réseau, injecter des paquets ou se faire passer pour d'autres nœuds.
- Ce schéma doit fonctionner sans une connaissance préalable d'un senseur de ses futurs voisins dans le réseau.
- Les exigences de compétition et de stockage de ce schéma doivent être dans la limite des capacités d'un senseur.
- Ce schéma doit être robuste aux attaques DoS de l'extérieur du réseau.

2.4 Architectures de sécurité existantes

Les protocoles de sécurité pour les WSNs (SPINS project) [95] consiste en deux principales applications : un protocole de cryptage pour les motes SmartDust nommé Secure Network Encryption Protocol (SNEP) et un protocole d'authentification nommé micro-Timed, efficient et temps réel, connu sous le nom de µTESLA (Timed Efficient, Streaming, Loss-tolerant Authentication Protocol). Chaque nœud senseur partage une seule clé master avec la SB. Les autres clés requises par SNEP sont dérivées de cette clé master.

Le protocole SNEP s'intéresse à la protection des communications entre un capteur et une station de base ou entre deux nœuds capteurs dans le réseau.

2.4.1 µTESLA (micro time efficient streaming loss-tolerant authentication)

Le protocole µTESLA s'appuie sur le protocole TESLA développé pour les réseaux ad hoc et s'adapte aux faibles ressources des capteurs. Il assure l'authentification des paquets émis par la station de base en diffusion sur le RCSF [96].

2.5 Critères de sécurité

2.5.1 Confidentialité

La confidentialité des données est le problème le plus posé dans les WSNs. Tout réseau devra adresser ce problème en premier. Dans les réseaux de capteurs, la confidentialité se rapporte aux exigences suivantes [91,97] :

- Un réseau de capteurs ne doit laisser les senseurs divulguer les lectures à leurs voisins.
- Dans beaucoup d'applications les nœuds communiquent des données secrètes, ainsi, il est très important d'établir un canal sécurisé dans un WSN.
- Les informations publiques des senseurs comme les identifiants et les clés publiques, doivent aussi être cryptées pour quelques ampleurs afin de se protéger contre l'attaque analyse du trafic.

2.5.2 Intégrité

Avec l'implémentation de la confidentialité, un adversaire peut être incapable de voler les informations. Cependant, cela ne signifie pas que les données sont saines. L'adversaire peut changer les données, de sorte que le réseau sera déstabilisé. Par exemple, un nœud malicieux peut ajouter quelques fragments ou manipuler les données dans un paquet. Le nouveau paquet modifié peut être ensuite envoyé au récepteur original. La perte ou dommages des données peut se produire sans la présence de nœuds malicieux, comme conséquence à un environnement dur. Ainsi, l'intégrité des données assure que les données reçues ne sont pas altérées en transit.

2.5.3 Fraicheur

Même si la confidentialité et l'intégrité des données sont assurées, on a aussi besoin d'assurer la fraicheur de chaque message. Informellement, la fraicheur de données suggère que les données sont récentes, et assure qu'aucun ancien message n'a été relayé.

2.5.4 Disponibilité

Les exigences de sécurité n'affectent pas seulement les opérations du réseau, mais aussi elles sont très importantes pour maintenir la disponibilité de tout le réseau.

Ce requis permet non-seulement de sécuriser le système mais rend aussi celui-ci tolérant aux fautes. Ainsi les ressources doivent rester disponibles jusqu'à ce que la faute soit réparée [98]. Certain messages doivent rester circonscrit à un moment ou à un endroit défini, pour ne pas induire en erreur les senseurs si l'information n'est plus pertinente [99]. Plusieurs applications nécessitent une réponse rapide de la part des capteurs ou du réseau ad-hoc, car le délai rendra le message obsolète. Ce qui peut causer des situations désastreuses. Il est donc primordial que les ressources soient disponibles en temps et lieu opportuns [100].

2.5.5 Auto-organisation

Un réseau de capteurs est typiquement un réseau ad hoc, qui exige que chaque senseur soit indépendant et assez flexible pour s'auto-organiser et s'auto-réparer selon différentes situations. Il n'ya pas d'infrastructure fixe disponible pour le besoin du management d'un réseau de senseurs. Cette caractéristique de base apporte un grand challenge à la sécurité des réseaux de capteurs sans fils. Par exemple, la mobilité de tout le réseau empêche l'idée de pré-installation de clés partagées entre la station de base et les senseurs [103]. Plusieurs schémas de pré-distribution aléatoires de clés ont été proposés dans le contexte des techniques symétriques de cryptage [101,102,103,104]. Dans le contexte d'application des techniques de cryptographie à clé publique dans les réseaux de senseurs, un mécanisme efficient pour la distribution d'une clé publique est nécessaire.

De la même manière que les réseaux de senseurs doivent s'auto-organiser pour supporter le routage multi-sauts, ils doivent aussi s'auto-organiser pour conduire le management de clé et développer des relations de confiance dans le réseau. Si l'auto-organisation est manquée dans un WSN, les dommages résultants d'une attaque ou même d'un environnement hasardeux peuvent être désastreux.

2.5.6 Authentification

Un adversaire n'est pas limité à modifier les paquets de données. Il peut changer les flots de paquets de tout le réseau en injectant des paquets additionnels. Donc, le récepteur a besoin de s'assurer que les données utilisées dans n'importe quelle décision proviennent d'une source originale. D'autres parts, à la construction du réseau de senseurs, l'authentification est nécessaire pour les tâches administratives (ex : la reprogrammation du réseau ou son contrôle). De ce qui précède, on peut constater que l'authentification des messages est importante pour beaucoup d'applications dans les WSNs. Informellement, l'authentification des données permet au récepteur de vérifier que les données réellement sont envoyées par un émetteur affirmé. Dans le cas de deux parties de communication, l'authentification des données peut être achevé à travers un mécanisme purement symétrique : l'émetteur et le récepteur partagent une clé secrète pour calculer le code d'authentification de message (*Message Authentication Code (MAC)*) de toutes les données communiquées.

Adrian Perring et al. proposent un système de distribution d'une chaine de clé pour leur protocole de secure broadcast μTESLA [95].

2.5.7 Responsabilité (non-répudiation)

Ce requis permet d'empêcher une entité de nier d'avoir participer à une communication. Il permet de protéger le système contre le déni d'un nœud qui indique n'avoir pas participé à une communication alors qu'il l'a fait. La non-répudiation permet donc au récepteur de prouver qu'il a reçu le message d'un tiers de communication. Ainsi, pour chaque message reçu, l'émetteur peut être clairement identifié [105]. Le but général de la non-répudiation est

de collecter, de maintenir et de rendre disponibles toutes les évidences à propos d'un évènement ou d'une action, afin de résoudre des disputes à propos d'une occurrence ou non d'une action. La non-répudiation dépend donc de l'authentification. Le système peut ainsi identifier l'auteur d'un message malveillant [106].

2.6 Mécanismes de défense

2.6.1 Chiffrement

Le chiffrement consiste à transformer une donnée (texte, message, …) afin de la rendre incompréhensible par une personne autre que celui qui a crée le message et celui qui en est le destinataire. La fonction permettant de retrouver le texte clair a partir du texte chiffré porte le nom de déchiffrement.

2.6.1.1 Cryptographie (écriture cachée)

Le mot cryptographie provient du grec Kryptus (caché) et graphein (écrire). C'est la technique visant à protéger un échange d'informations par un codage du message. Cette technique est pratiquée par des cryptographes. La cryptographie traditionnelle est l'étude des méthodes permettant de transmettre des données de manière confidentielle afin de protéger un message, on applique sur ce dernier une transformation qui le rend incompréhensible. Il donne à partir d'un texte en clair (plaintext) un texte chiffré ou un cryptogramme (ciphertext). Inversement, le déchiffrement est l'action qui permet de reconstruire le texte en clair à partir du texte chiffré [107,108].

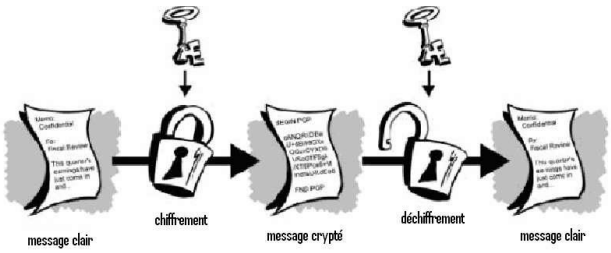

Figure 7. Cryptographie d'un texte [107]

2.6.1.2 Stéganographie (écriture couverte)

L'objectif principal de la stéganographie est de cacher ou d'intégrer un message, soit dans un autre message ou dans un ensemble de données multimédia (image, son, etc). Cependant, cette technique nécessite des ressources de traitement considérables et il est difficile de l'intégrer dans les RCSFs en raison de leurs contraintes.

2.6.2 Notions d'authentification

L'authentification désigne le processus qui permet de valider l'identité d'un utilisateur ou d'un équipement (client, serveur, routeur, …etc.) [109].

L'authentification dans les réseaux de capteurs se fait par ajout d'une chaine de bits appelée MAC (Message Authentication Code) au message à transmettre. Un code d'authentification de message (MAC) est une donnée utilisée pour authentifier un message. Un algorithme de calcul du MAC accepte en entrée une clé secrète et le message à authentifier, et en sortie le MAC (parfois connu sous le nom de *tag).*

2.6.3 Signature numérique

Une signature numérique est un condensé de message crypté qui est joint à un document. Une telle signature peut être utilisée pour confirmer l'identité de l'émetteur et l'intégrité des données. Elle combine l'utilisation du cryptage à clé publique et d'une fonction de hachage.

L'invention de la cryptographie à clé publique a rendu possible l'utilisation de la signature numérique. Cette dernière ressemble à une signature manuscrite en ce sens qu'elle offre une preuve sur l'identité de l'expéditeur du message (authentification). Pour signer le message il suffit en effet de lui appliquer une fonction mathématique (appelée fonction de hachage) qui produit un résumé (code hache) du message. Le résumé obtenu est propre à chaque message, à l'image d'une empreinte digitale.

2.6.4 Fonction de hachage

Permet d'obtenir un condensé (appelé aussi haché) d'un texte, c'est-à-dire une suite de caractères assez courte représentant le texte qu'il condense.

La fonction de hachage doit être telle qu'elle associe un et un seul « hach » à un texte en clair (cela signifie que la moindre modification du document entraîne la modification de son hach). Les fonctions de hachage les plus connues sont MD5 (Message Digest 5) et SHA-1 (deuxième version de la norme américaine Secure Hash Algorithme) [110].

2.6.5 Partitionnement des données

[111] et [112] offrent une solution pour empêcher la récupération d'information dans les réseaux de capteurs sans fil par le partitionnement des données. Comme son nom l'indique le but est de découper l'information en plusieurs parties. Si un capteur cherche à envoyer une information, celui-ci va la découper en plusieurs paquets de taille fixe. Chaque paquet sera ensuite envoyé sur des chemins différents, c'est à dire qu'elles ne passeront pas par la même route et donc les mêmes nœuds.

2.6.6 Détection d'intrusion

La détection d'intrusion peut être définie comme la détection automatique d'une attaque et la génération d'une alarme pour rapporter qu'une intrusion a eu lieu ou est en cours.

2.6.7 Certificats électroniques

Un certificat est un élément d'information qui prouve l'identité du propriétaire d'une clé publique. Les certificats sont signés et transmis de façon sécurisée par un tiers de confiance appelé autorité de certification (Certificate Authority, ou CA). L'autorité de certification est chargée de délivrer les certificats, de leur assigner une date de validité, ainsi que de révoquer éventuellement des certificats avant cette date en cas de compromission de la clé (ou du propriétaire) [113].

2.6.8 Indice de confiance et réputation

Une solution proposée par [114], [115], [116], [117], [118] et [119] consiste à utiliser les mécanismes de confiance et de réputation que l'on peut trouver dans les réseaux pair à pair [120], les réseaux de communauté ou bien encore dans les sites marchands comme Ebay. Dans les réseaux de capteurs sans fil, il est difficile de savoir, au vu du nombre de nœuds, quel nœud peut être un nœud malicieux. Pour le détecter et conserver l'intégrité du réseau, chaque nœud du réseau va surveiller ses nœuds voisins et leurs actions au cours du temps. En fonction des actions réalisées par ses nœuds voisins, un nœud va augmenter une note de l'indice de confiance de ces nœuds, basée sur sa réputation. Si un nœud ne répond jamais à une requête, son indice de confiance va diminuer, de la même manière que si ce nœud retransmet toujours correctement l'information qu'on lui a relayé, son indice de confiance va augmenter [121].

A l'aide de ces indices de confiance, un nœud va alors choisir le routage le plus adapté pour transmettre son information. Contrairement à des protocoles classiques de routage où le nœud chercherait le chemin le plus rapide en nombre de sauts ou de distance géographique, il va choisir ici de transmettre son information via les nœuds avec les indices de confiance les plus élevés, en d'autre terme, la route qui lui semble la plus sûre. Ces techniques permettent d'éliminer le routage traditionnel, qui peut faire passer l'information par des nœuds potentiellement dangereux.

Les solutions basées sur l'indice de confiance sont peu coûteuses en termes d'énergie et permettent, selon le type de sécurité voulu, de ne pas avoir recours à la cryptographie. Cependant pour des réseaux qui demandent une sécurité maximale, elles ne sont pas toujours adaptées. Ainsi un nœud malicieux qui enregistrerait des informations sur le réseau et, par ailleurs, se comporterait de manière normale, est difficilement détectable.

2.7 Discussion

Le défit des concepteurs des techniques de sécurité, est de mettre en œuvre des solutions « lights » qui prennent en considération les contraintes des micro-capteurs. Pour trouver un compromis réaliste entre le surcoût du calcul, le délai et la crédibilité des données (trustworthiness), la sélection de méthodes cryptographiques appropriées est fondamentale. Par conséquent, divers services cryptographiques sont requis pour certaines applications et l'utilisation commune des algorithmes à clé symétriques tels que : AES et MAC n'ont pas uniquement imposé des problèmes tels que la gestion et la protection des clés, mais peuvent être en même temps bien plus coûteux.

La cryptographie des courbes elliptiques (ECC) a émergé comme un crypto-système à clé publique *attractif* pour les réseaux de capteurs sans fil. En comparaison aux Crypto-systèmes traditionnels comme RSA, ECC offre une sécurité équivalente avec des clés à petites tailles, ce qui a comme conséquence d'accélérer les calculs, de minimiser la consommation d'énergie, de sauvegarder la mémoire et la bande passante.

Malgré la diversité et l'efficacité prouvée des solutions basées sur la cryptographie, la majorité des solutions supposent que les nœuds capteurs sont dignes de confiance et rapportant des données de manière correcte. Cependant dans la pratique, les capteurs sont déployés dans des environnements ouverts sans surveillance, et donc sont exposé aux attaques physiques. Lorsqu'un nœud est compromis, l'attaquant peut injecter des données erronées dans le réseau. Cependant, nous déduisons que la vue conventionnelle de la sécurité basée sur la cryptographie seule est *insuffisante* à cause des caractéristiques spécifiques et des nouveaux comportements malveillants rencontrés dans des réseaux ouverts. Quoique, la cryptographie peut assurer l'intégrité, la confidentialité et l'authentification, elle échoue face aux attaques internes (*insider attacks*).

En effet dans une attaque d'égoïsme, un nœud malicieux peut agir en attaque, une fois qu'il est authentifié par un certificat valide. Ce type d'attaque a été renvoyé à des attaques byzantines et concernent les cas où les nœuds authentifiés peuvent ne pas être honnêtes pour leur faire confiance. Cela nécessite une démarche complémentaire pour faire face aux attaques internes. Une prise de décision suivant quelques métriques, avant de relayer l'information à un voisin quelconque, sera le remède aux conflits de sécurité apparus dans ce type d'attaques internes. Cette décision est en effet un choix du prochain saut suivant le degré de confiance accordé à un certain voisin et visant un chemin sûr. Ce degré de confiance est relatif à l'expérience du nœud source sur la réputation de ses voisins et à une analyse des

risques encourus en prenant un certain chemin. L'idée principale est d'évaluer le trait prévisible d'une autre entité et d'établir le niveau de confiance qui lui est porté, c'est-à-dire paraît-il digne de confiance ou non ? Est-il honnête dans les réponses aux requêtes?
Une analyse des comportements d'une entité qui feront d'elle digne de confiance ou non, le choix du prochain saut suivant sa réputation globale et une étude détaillée sur les systèmes de confiance et réputation feront l'objet du prochain chapitre.

2.8 Conclusion

Dans ce chapitre, une étude est faite sur les attaques potentielles existantes dans les réseaux de capteurs sans fils. Ces attaques ont été classifiées selon leur origine et leur nature. Des contremesures contre les attaques les plus connues, des solutions potentielles et les travaux qui ont eu un succès dans chaque paradigme ont été présentés.
Malgré que nous ayons discutés les attaques séparément dans ce chapitre, ces attaques en effet sont souvent lancées en combinaison. Cette combinaison peut être en cross-layer dans laquelle plusieurs attaques dans différentes couches sont lancées d'une manière collaborative. Par exemple, l'attaque Sybil (dans les couches MAC et réseau) fournit le spoofing d'identité à l'adversaire pour lancer une attaque Wormhole (dans la couche réseau). La combinaison peut être aussi intra-layer dans laquelle plusieurs attaques dans la même couche se lancent simultanément. Par exemple, dans la couche réseau, une attaque Wormhole peut être lancée pour attirer le trafic à un nœud compromis qui est une attaque Sinkhole. Une telle combinaison complique la situation de la sécurité d'un WSN et exige des recherches supplémentaires sur les contremesures appropriées. D'ailleurs, un même type d'attaque peut être présent dans plusieurs couches, malgré qu'elles utilisent des techniques différentes. Par exemple, les attaques DoS existent dans la couche physique, MAC et réseau. Les attaques Sybil existent dans les deux couches MAC et réseau. On note aussi que non seulement les mêmes types d'attaque peuvent se trouver dans différentes couches, mais aussi les mêmes types de contremesures peuvent apparaitre dans plusieurs couches. Par exemple, les technique de détection de mauvaise conduite peuvent être appliquées presque pour toutes les couches.

Chapitre 3

Confiance et réputation

Abstract

Trust *and reputation are two very useful tools that are used to facilitate decision making in diverse fields. Reputation is the opinion of one entity about another. In an absolute context, it is the trustworthiness of an entity. Trust, on the other hand, is the expectation of one entity about the actions of another. For over three decades, formal studies have been done on how reputation and trust can affect decision making abilities in uncertain conditions. Only recently has trust and reputation been adapted to wireless communication networks. Trust is a multidimensional entity which, if effectively modeled, can resolve many problems in wireless communication networks.*

Nodes in WSNs can make reputation and trust guided decisions. This not only provides WSNs with the capability of informed decision making, but also provides them security in the face of insider attacks where cryptographic security gives way.

In this chapter we provide an overview of current researches on reputation systems in wireless sensor networks. Several proposals are reviewed and compared according to their reputation system. Based on our observations we propose a new reputation system for protecting data integrity in wireless sensor networks. In the second part of our research, we defined a set of tests and evaluated our reputation system with respect to overall network performance, energy consumption and resistance against presence of selfish and malicious nodes. We provide complete simulation results and recommendations for further research.

Keywords *Wireless sensor network, reputation system, trust, routing, security, integrity.*

PARTIE I

ETAT DE L'ART

3.1 Introduction

Au-delà des approches traditionnelles utilisant les techniques cryptographiques classiques pour assurer les propriétés de sécurité les plus importantes (confidentialité, intégrité, authentification et non-répudiation), des mécanismes d'établissement de la confiance et réputation ont été proposés afin de créer et de gérer la confiance entre les entités actives d'un système informatique. En effet, les systèmes de confiance et réputation permettent, à chaque entité participant à un protocole donné, de mesurer la fiabilité d'une autre entité avant de décider d'interagir ou d'entrer en communication avec elle. Ils représentent donc un moyen d'inciter ces entités à un bon comportement et à toujours offrir des services ou des ressources de qualité. Dans le cas des systèmes de réputation, l'idée de base est de permettre aux entités de s'évaluer entre elles, par exemple à la fin de chaque transaction. Ensuite d'utiliser des agrégats de ces évaluations (recommandations) afin de déduire la note de réputation d'une entité donnée. Dans le cas des systèmes de confiance, l'idée de base est d'analyser et de combiner des réseaux et des chemins de relations de confiance afin de déduire des mesures de confiance subjectives permettant d'évaluer la fiabilité d'une entité donnée. Les systèmes de confiance et de réputation ont de nombreuses applications dans les environnements distribués tels que les réseaux ad hoc, les réseaux de capteurs, les réseaux pair-à-pair et bien sûr dans le cadre des réseaux classiques (par exemple pour les applications de commerce électronique telles que les ventes aux enchères).

Le système avec lequel un système découvre et utilise la réputation pour former la confiance, et utilise la confiance pour mesurer un comportement est connu sous le nom de système basé confiance (Trust based system). Ce chapitre est dédié pour donner une explication complète de ce qu'on appelle un système basé confiance.

Ce chapitre est scindé en deux parties, dans la première nous présentons l'intérêt de faire fonctionner un système basé confiance dans différentes domaines d'application, nous passerons par la suite à étudier ces systèmes dans les réseaux ad hoc et les réseaux de capteurs et le niveau de sécurité et d'intégrité de données que peut promettre ce type de système. Nous présenterons à la fin de cette partie les premiers travaux réalisés pour les réseaux ad hoc et les réseaux de capteurs. La deuxième partie est consacrée pour la présentation de notre contribution, ainsi les tests et résultats de la simulation de notre système de confiance. Des perspectives seront données à la fin de ce chapitre, et on termine avec une conclusion et un résumé de ce qui a été fait.

3.2 Confiance

Dans cette sous section, nous allons discuter l'établissement de la confiance qui est nécessaire pour que des nœuds capteurs puissent collaborer à l'accomplissement des tâches du réseau. En effet, les nœuds pourraient être enclins à adopter des comportements égoïstes ou malveillants s'il n'y avait pas de protocoles les incitants à coopérer.

Il n'existe pas de définition univoque au mot *confiance*. Cependant, il apparaît une multitude de définitions qui varient selon la méthodologie utilisée pour observer le phénomène. Quotidiennement, des mécanismes de confiance sont employés par les humains pour favoriser les relations sociales, amicales, familiales, etc. Certains d'entre nous sont prêts à accepter un risque dans des situations où on a seulement une connaissance partielle de la réalité. Dès lors qu'il est nécessaire d'agir dans une situation indéterminée, de réagir aux actions entreprises par d'autres ou simplement de prendre une décision vis-à-vis d'un choix complexe. La confiance permet d'avancer raisonnablement dans une voie à partir de connaissances partielles et parcellaires. Contrairement à l'adage qui énonce que *les amis de mes amis sont mes amis*, la confiance n'est pas forcément une notion transitive : $A \rightarrow B \rightarrow C$ ne signifie pas *obligatoirement* que $A \rightarrow C$, mais est donc plutôt une notion *pseudo-transitive* [122,123].

Les niveaux de confiance peuvent être calculés à partir de l'effort qu'un nœud est disposé à dépenser pour un autre nœud. Cet effort peut être en termes de consommation de sa batterie, de paquets expédiés ou de ceux qu'il rejette, ou n'importe quel autre critère qui aide à établir un niveau de confiance mutuel. La capacité des mobiles de communiquer sans fil permet de les utiliser pour collaborer ou coopérer avec d'autres mobiles afin de se proposer mutuellement des services ou de partager des ressources. Par exemple, plusieurs mobiles autonomes peuvent se réunir pour assurer le partage d'une ressource tel le routage ou de permettre l'accès à de nouveaux services.

3.3 Réseaux de capteurs et confiance

Dans un réseau de capteurs le but d'établissement d'un mécanisme de confiance est que les nœuds participant effectivement au réseau ne se contentent pas de l'exploiter, voire essayent de le nuire. Deux comportements nuisibles à qui s'oppose un mécanisme de confiance : l'égoïsme et la malveillance.

Un nœud deviendra égoïste dans le but de préserver ses ressources (en bande passante et en énergie). Par conséquent il pourra par exemple ne plus remplir son rôle de routeur ou se contenter de router les paquets moins couteux (petits paquets). Un nœud malveillant ira plus loin, il pourra par exemple s'employer à faire baisser la réputation des nœuds corrects. La notion de nœuds malveillant se rapproche plus de la notion attaquant présenté dans le chapitre précédent. Pour mieux expliquer la notion d'égoïsme, prenant un exemple des

réseaux peer-to-peer : un égoïste est celui qui se contente de télécharger sans rien partager. Et un malveillant est celui qui tente d'injecter dans le réseau des virus.
Il existe trois avantages majeurs liés à l'utilisation d'un système d'évaluation de confiance dans un réseau distribué. Premièrement, le fait de disposer d'une méthode d'évaluation de la confiance offre une incitation à un bon comportement des nœuds. Deuxièmement, l'évaluation de la confiance offre aux nœuds la possibilité de prédire le comportement futur des autres nœuds. Cette prédiction permet d'aider le nœud dans sa prise de décision. En d'autres termes, elle permet aux nœuds honnêtes d'éviter d'interagir avec les nœuds les moins fiables, ce qui réduit la participation des nœuds malveillants aux opérations du réseau. Troisièmement, le résultat du processus d'évaluation de la confiance peut être utilisé directement dans la détection des nœuds malfaisants et égoïstes dans le réseau et à la mise en place de sanctions.

3.4 Confiance vs Réputation

La notion de réputation est étroitement liée à celle de la confiance, mais néanmoins, il existe des différences claires et importantes entre ces deux notions. La confiance est une mesure personnelle et subjective qui s'appuie sur une variété de faits, dont certains peuvent avoir plus de poids que d'autres. Typiquement, l'expérience ou la connaissance personnelle. Tandis que la réputation est une mesure collective de la crédibilité (fiabilité) basée sur les notations faites par les membres d'une communauté.
Les systèmes de réputation produisent un score public (réputation), reflète la vision de l'ensemble de la communauté de la crédibilité d'une entité. Ils utilisent les mécanismes d'agrégation et de pondération.
Les systèmes de confiance produisent un score qui reflète la vision subjective des parties concernées (consultées) de la crédibilité d'une entité.

3.5 Réseaux de capteurs et réputation

Les systèmes de réputation peuvent être appliqués dans les réseaux de capteurs mobiles afin de doter les nœuds de moyens pour se protéger contre d'autres nœuds qui s'avèreraient malicieux. Dans cette partie, nous présentons comment se fait l'intégration de systèmes de réputation dans les réseaux de capteurs.

Les réseaux de capteurs sont composés de participants ayant minimalement les mêmes caractéristiques en termes de ressources. Ce sont des nœuds qui communiquent de manière décentralisée à travers un réseau sans-fil. Pour ces réseaux, l'on peut considérer aussi bien les communications à plusieurs sauts que les communications directes entre nœuds voisins [124]. Ces réseaux ont pour particularité de dépendre fortement les uns des autres. De ce fait,

il est important de s'assurer de l'effectivité des tâches effectuées par chaque participant du réseau en évitant des comportements égoïstes par les nœuds senseurs. Ou encore des attaques potentielles du fonctionnement global ou d'une tâche particulière. Pour mener à bien ces tâches, chaque entité du réseau observe le comportement de ses voisins en utilisant la technique de réputation. Le bon fonctionnement des réseaux de capteurs repose sur la confiance entre les différents nœuds du réseau. Hors, cette relation de confiance n'est pas toujours acquise dans un WSN. Dans les conditions normales, une simple authentification devrait être suffisante pour assurer le bon fonctionnement du réseau. Malheureusement, il se peut que certains nœuds aient des intentions malicieuses. Un système où règne une confiance aveugle entre les nœuds n'existe que dans les réseaux propriétaires. Par exemple les réseaux militaires, ou certains réseaux d'entreprises [125]. Le bon fonctionnement des réseaux auto-organisés dépend grandement du bon comportement de tous les acteurs de ce réseau. Car chaque acteur doit effectuer des tâches précises pour que tous les paquets circulent dans le réseau de façon fluide.

3.6 Modèles de réputation

Pietro et Refik [125] ont présenté le protocole CORE (Collaborative Reputation Mechanism to enforce node cooperation in Mobile Ad Hoc Networks) dans lequel ils définissent trois types de réputation: la réputation subjective qui traduit les informations collectées de façon locale par un nœud, la réputation indirecte qui prend en compte les observations réalisées par d'autres nœuds du voisinage et la réputation fonctionnelle qui s'intéresse à une fonction précise du système.

3.6.1 Réputation subjective

Le terme réputation subjective est utilisé pour parler de la réputation calculée localement par le nœud hôte. On parle alors d'observation subjective. Une réputation subjective à un temps t par un nœud S_i est calculée en utilisant la moyenne pondérée des facteurs d'observation en donnant plus d'importance aux observations passées. Dans le modèle CORE, plus de pertinence est donnée aux observations passées à cause d'une éventuelle inconsistance des récentes observations. Par exemple dans le cadre des réseaux VANETs, cette hypothèse n'est plus valable car les nœuds se déplacent à une vitesse très élevée et l'analyse se fait sur des données récentes.

3.6.2 Réputation indirecte

Dans la partie précédente, il était question d'une réputation subjective, prenant en compte seulement les observations collectées par le nœud lui-même. Mais dans la réalité, les choses ne sont pas aussi simples. Dans certaines situations, l'on considère les observations collectées par les nœuds voisins. La réputation indirecte est donc constituée des observations faites par

le nœud hôte et celles faites par d'autres nœuds (nœuds voisins préalablement considérés comme honnêtes). Quelques précautions doivent être prises lors de la collection des informations pour une réputation indirecte. Par exemple, l'on pourrait considérer seulement les notations positives pour éviter que des dénis de service perpétrés par des nœuds malicieux ne soient effectifs et contribuent à une note négative des nœuds légitimes et honnêtes. Une prise en compte du fait, que des nœuds malicieux se concertent afin de noter positivement d'autres nœuds négatifs, est à faire. Sinon, ceci contribuerait à avoir des cas de faux négatifs.

3.6.3 Réputation fonctionnelle

Le terme de réputation fonctionnelle est utilisé pour représenter le cas où les réputations subjective et indirecte sont calculées en respectant une fonction différente *f*. Ce type de réputation donne la possibilité de calculer une valeur globale de réputation d'un sujet en prenant en compte des critères d'observation et d'évaluations différentes.

3.7 Intérêt d'un système de réputation

De par sa capacité à être exploité dans les domaines divers, le concept de réputation présente un intérêt primordial pour l'analyse des comportements des entités dans un environnement donné. Dans les sciences sociales, la réputation permet l'étude de comportement des êtres humains dans un milieu social [126]. En économie, la réputation sert plutôt à l'analyse et à la prédiction des tendances économiques en tenant compte des réalités présentes et passées. En informatique, le concept de réputation est utilisé aussi bien dans le domaine de l'intelligence artificielle, du commerce électronique ou encore dans les réseaux auto-organisés (Réseau Ad-hoc, Réseaux Mobiles, Réseaux de capteurs, etc.) [127]. Pour ce dernier cas, il permet de garantir la fiabilité des nœuds en communication. Il sert aussi de mesure le comportement des différents nœuds quant à leur collaboration dans la bonne marche du réseau. Principalement un système de réputation a deux objectifs : Permettre aux nœuds de trouver les meilleurs partenaires de communication et donner à ceux-ci une raison de coopérer (par exemple pour le routage des informations).

Dans [128], [129] et [130], les auteurs présentent le but des systèmes de réputation dans les réseaux auto-organisés :

- Permettre aux nœuds de faire la distinction entre les nœuds honnêtes et les nœuds malicieux dans le réseau.
- Encourager les nœuds du réseau à coopérer les uns avec les autres.
- Décourager les nœuds malicieux à participer aux activités du réseau.

- Permettre aux systèmes de réputation de gérer tous les types de mauvais comportement des nœuds du réseau.
- Minimiser les dommages causés par une attaque perpétrée par un nœud.

3.8 Score de réputation

Abul-Ranman et Hailes [131] proposent une méthode permettant de calculer le score de réputation. En effet ils proposent un système qui divise le score de réputation en intervalle. Ainsi, le degré de confiance d'un agent pour un autre peut prendre quatre valeurs: très digne de confiance, digne de confiance, peu digne de confiance et absolument pas digne de confiance. Les témoignages provenant d'autres agents sont considérés avec un poids. Ce système a pour principal inconvénient le problème d'initialisation. Car un nouveau venu dans le réseau ne sait pas forcément à quel agent il doit se fier.

3.9 Conception d'un système de confiance et réputation

3.9.1 Architecture

L'architecture d'un système de confiance détermine la manière dont les nœuds du système s'échangent leurs évaluations. Les deux principaux types d'architecture existants sont les architectures centralisées et les architectures décentralisées.

Les systèmes de confiance et de réputation centralisés s'appuient sur une tierce partie de confiance qui s'occupe d'une part de collecter les évaluations des participants et d'autre part d'inférer sur la base des évaluations collectées, la réputation de chaque participant au système. Les notes de réputation sont ensuite rendues publiques. Il existe des systèmes où les architectures de confiance et de réputation décentralisées sont plus appropriées qu'une architecture centralisée reposant sur une unique autorité de confiance. C'est notamment le cas des réseaux ad hoc et les réseaux de capteurs. Dans ces architectures, les nœuds s'auto-évaluent, s'échangent leurs évaluations pair-à-pair, sauvegardent les évaluations reçues et dérivent localement les notes de confiance.

3.9.2 Evaluation de confiance

Interpréter la confiance comme une mesure d'évaluation de la fiabilité d'une entité a permis l'émergence de nombreuses métriques. La confiance est évaluée suivant différentes approches. Ces approches offrent une variété de méthodes de dérivation. On peut trouver de simples modèles basés sur des calculs de probabilité basiques : par exemple dans [132, 133, 134,135], leur système de réputation est basé sur la loi de probabilité binomiale qui est souvent utilisée pour représenter la distribution de probabilité a posteriori d'un événement binaire. D'autres méthodes plus générales que cette dernière, basées sur des lois de

probabilité multinomiales, ont été proposées. La méthode la plus connue est celle basée sur les systèmes bayésiens [136, 137, 138].

3.9.2.1 Métriques d'évaluation

Quelques métriques à prendre en considération sont les suivantes :

3.9.2.1.1 Propre expérience d'un nœud

L'expérience d'un nœud avec un voisin. Par exemple, combien de réponses sur les requêtes de route qui sont correctement reçues de ce voisin, combien de paquets acheminés par ce voisin, combien de paquets reçus de ce voisin.

3.9.2.1.2 Observations

Un nœud peut mettre son transceiver (émetteur/récepteur) en mode dissolu et écouter les communications sortantes de ses voisins, dans sa portée. Ce moyen vérifie si un voisin réagit correctement aux requêtes, par exemple, s'il relaye vraiment les paquets qu'il doit relayer.

3.9.2.1.3 Notifications de voisins

Un système de réputation peut utiliser les informations partagées entre les nœuds. Un nœud peut échanger les informations de réputation avec ses voisins directes [139].

3.10 Attaques ciblant les systèmes de réputation

Les systèmes de réputation sont sensibles aux attaques d'agents malveillants qui tentent par tous les moyens possibles de tirer profit des vulnérabilités de ces systèmes. Il existe dans la littérature de nombreux travaux qui adressent les questions liées à ce sujet [88, 140, 141, 142, 143].

Plusieurs attaquants peuvent également former une collusion. C'est-à-dire qu'ils mettent en commun leurs ressources et connaissances afin d'obtenir encore plus d'informations sur un autre agent ou de modifier la réputation d'un fournisseur de service. L'attaque par blanchiment de réputation, est une attaque qui permet à un agent de réinitialiser son score de réputation lorsqu'il le juge trop faible. Un attaquant peut aussi vouloir filtrer l'ensemble des témoignages concernant un fournisseur pour augmenter la proportion de témoignages favorables et ainsi augmenter son score de réputation. Si un attaquant veut modifier la réputation d'un nœud, en termes d'amélioration ou de diminution, il peut utiliser la technique du bourrage d'urne, qui consiste à émettre de multiples témoignages fallacieux en bien ou en mal à l'endroit de l'agent visé [144]. Un attaquant peut vouloir médire sur un nœud en apportant des témoignages de mauvaise qualité. Ces attaques par médisance peuvent être amplifiées si l'attaquant peut se créer de nombreuses identités, par exemple grâce à une attaque Sybil réussie ou via une collusion. Enfin, un attaquant peut également

essayer de réfuter une transaction où le nœud s'est bien comporté pour éviter d'avoir à émettre un témoignage positif [145, 146, 147,148].

Dans [149], D. Fraga, Z .Bankovic et J. M. Moya classifient les attaques ciblant les TRS (Trust and Reputation System) en trois types :

1. Les attaques contre la collecte d'informations de réputation et de confiance.
2. Les attaques contre le calcul de confiance et réputation.
3. Les attaques contre la dissémination de la confiance et réputation.

3.11 Systèmes de gestion de confiance dans les réseaux ad hoc

Les premiers protocoles proposés utilisant le mode promiscuous étaient pour les réseaux ad hoc qui sont ceux de Marti et al. [150]. Dans leurs travaux, les auteurs proposent un protocole basé sur le protocole de routage DSR (Routage à Source Dynamique, protocole de routage pour les réseaux ad hoc) [151]. Leur schéma utilise deux composants : le chien de garde (Watchdog) et l'évaluateur de chemins (Pathrater).

D'autres protocoles, beaucoup plus élaborés, ont été proposés. Bouchegger et Le Boudec ont proposé un système de renforcement de la coopération distribué et collaboratif dénommé CONFIDANT (COoperation of Nodes, Fairness In Dynamic Ad-hoc NeTworks) [152] appliqué au protocole DSR. L'objectif de CONFIDANT est d'exclure du réseau tout nœud égoïste, que ce soit au niveau du processus d'acheminement des données ou bien au niveau du processus de découverte du voisinage.

Un autre protocole semblable en de nombreux points à CONFIDANT a été proposé par Michiardi et Molva [153]. Le protocole s'appelle CORE (COllaboratif REputation mechanism) et il est basé lui aussi sur le protocole de routage DSR. La différence essentielle avec CONFIDANT réside dans la manière dont CORE calcule les valeurs de réputation.

WATCHMAN (An Overlay Distributed AAA Architecture for Mobile Ad hoc Networks), cette solution [154] se base sur une architecture AAA (Authentication Authorization, Accounting) distribuée et hiérarchique appelée WATCHMAN.

Pour conclure cette étude, nous pouvons remarquer que les propositions que nous avons décrites jusqu'ici se basent exclusivement sur le comportement des nœuds. La limite de ce type de systèmes est qu'ils ne prennent pas en considération le contenu des paquets échangés entre les nœuds du réseau. Ainsi, un nœud peut avoir une bonne réputation, car il a fait preuve d'un bon comportement (il a par exemple routé convenablement les messages de ses voisins), mais les informations qu'il envoie peuvent être fausses (en particulier, celles relatives au routage).

Dans ce contexte, nous pouvons citer comme exemple ces deux travaux très intéressants « An effective intrusion detection approach for OLSR manet protocol » [155] et « Property based intrusion detection to secure OLSR » [156]. Dans ce qui suit nous présentons les travaux les plus récents portants sur le management de confiance (Trust Mangement) dans les réseaux de capteurs sans fils.

3.12 Systèmes de gestion de confiance dans les réseaux de capteurs

3.12.1 PLUS (Parameterized and Localized trUst management Scheme for WSNs)

Z. Yao et al. (Oct. 2006) [157], ont proposés le protocole PLUS pour les réseaux de capteurs sans fils. Les auteurs adoptent une approche de localisation distribuée et la confiance est calculée en se basant sur les deux types d'observations directes et indirectes. Avec ces valeurs de confiance, les nœuds sont classés en quatre catégories :

1. Distrust ou untrustworthy (Non fiable).
2. Minimal ou low trust (faible)
3. Average (Moyen)
4. Good ou trustworthy (fiable).

A chaque fois qu'un nœud a besoin d'une recommandation à propos d'un autre nœud, il diffusera une requête (EReq) à ses voisins. Ce paquet contient l'identité du nœud à évaluer. En réponse, tous les nœuds (à part celui concerné par l'évaluation) renvoient un paquet réponse (ERep) au nœud demandeur. Une fois toutes les réponses sont reçues, le nœud récepteur calculera la valeur de confiance finale. Si le nœud évaluateur juge le nœud évalué avec un comportement malveillant, alors le nœud évaluateur diffusera un paquet d'échange d'informations (EInf) à ses voisins. Ce paquet contient les informations à propos de l'identité du nœud jugé malhonnête et le code d'erreur. Avec cette politique de confiance, les nœuds voisins échangent leurs opinions avec des paquets exchangeAck (EAck) dans le cas où ils sont d'accord avec l'expéditeur, autrement les voisins vont replayer avec des paquets exchangeArgue (EArg). La description des paquets du protocole PLUS est présenté dans la table 1.

Type	Payload	Size of payload
EReq	ID of evaluating node (2 bytes)	2 bytes
ERep	ID of evaluating node(2 bytes), trust value(4 bytes)	6 bytes
EInf	ID of evaluating node(2 bytes), Error code(2 bytes)	4 bytes
EAck	ID of evaluating node (2 bytes)	2 bytes
EArg	ID of evaluating node (2 bytes), trust value(4 bytes)	6 bytes

Tableau 1. Paquets du protocole PLUS

3.12.2 RFSN (Reputation-based Framework for Sensor Networks)

S. Ganeriwal et M. B. Srivastava (2008) [158], [159] ont proposés le protocole RFSN, où chaque nœud maintient une valeur de réputation pour les nœuds du voisinage. En se basant sur la réputation des nœuds, les valeurs de confiance sont calculées. Avec ces valeurs de confiance, les nœuds sont classés en deux catégories :

- Trusted(coopératif).
- Un-Trusted(non coopératif).

A chaque fois qu'un nœud a besoin d'une recommandation des autres nœuds, il envoie une requête (Req) aux voisins jugés trusted. La requête contient l'identité du nœud à évaluer. En réponse, les nœuds trusted renvoient un Reply (Rep). Le paquet Reply contient l'identité du nœud évalué et sa valeur de confiance. La description des paquets de RFSN est représentée dans la table 2.

Type	Payload	Size of payload
Req	ID of evaluating node (2 bytes)	2 bytes
Rep	ID of evaluating node(2 bytes), trust value(4 bytes)	6 bytes

Tableau 2. Paquets de RFSN

3.12.3 GTMS (Group-Based Trust Management Scheme for Clustered Wireless Sensor Networks)

Shaikh R.A. et. al. (Novembre 2009) [160] ont proposés un schéma hiérarchique, léger de gestion de confiance pour les réseaux de capteurs sans fils (GTMS).

Dans un cluster, chaque senseur calcule individuellement les valeurs de confiance de tous les autres nœuds, en se basant sur des observations directes ou indirectes.

Les observations directes représentent le nombre d'interactions réussies et non réussies, et les observations indirectes représentent les recommandations des entités dignes de confiance (trusted) à propos d'un nœud particulier. Ici, les interactions signifient la coopération entre deux nœuds. Par exemple, un émetteur considèrera la réussite d'une interaction, si cet émetteur reçoit un signe (assurance) qui explique que le paquet est reçu avec succès par le voisin, et ce voisin a relayé le paquet vers la destination d'une manière non altérée. Ainsi :

- Le premier requis : une réception réussie, qui est achevé par la réception d'un acquittement (ACK) link layer (utilisant par exemple, le standard IEEE.802.11)

- Le second requis : l'acheminement du paquet, qui est achevé par un acquittement passif (PACK), en écoutant les transmissions du prochain saut qui est dans la portée radio.

Dans GTMS les valeurs de confiance classent les nœuds en trois catégories :

1. Trusted (digne de confiance)
2. Un-Trusted (non digne de confiance)
3. Un-Certain (Uncertain)

De la même manière, chaque cluster maintient une valeur de confiance des autres clusters. Le schéma GTMS est composé de quatre paires de paquets de requêtes et réponses :

Paire 1 : utilisée pour une recommandation paire. A chaque fois qu'un nœud x a besoin d'une recommandation de la part du nœud y à propos de z, il envoie une requête (iTReq) d'une taille de 2 octets au nœud y. En réponse le nœud y envoie un paquet réponse (iTRep) d'une taille de 3 octets au nœud x. Le paquet iTRep inclut la valeur de confiance du nœud z.

Paire 2: utilisée pour la transmission du vecteur de confiance des nœuds vers les cluster-heads(CH). Après un intervalle de temps, le CHj diffuse une requête (iVReq) à l'intérieur du groupe. En réponse, tous les nœuds appartenant au cluster j renvoient un paquet réponse (iVRep).

Paire 3 : utilisée par un CH pour obtenir des recommandations par une station de base (SB). A chaque fois qu'un CH j a besoin d'une recommandation de la part de la SB à propos d'un autre CH k, il envoie une requête (oTReq) à la SB. En réponse, la SB envoie un paquet réponse (oTRep) au CHj qui contient la valeur de confiance du CHk. La taille de ce paquet est 3 octets.

Paire 4 : utilisée pour le transfert des vecteurs de confiance d'un CH à la SB. Après un intervalle de temps périodique, la SB diffuse une requête (oVReq) à tous les CHs du réseau. En réponse, tous les CHs renvoient un paquet réponse (oVRep) de taille de 1+3v octets, où :

- v<=|G|.
- v représente la longueur du vecteur de confiance.
- |G| représente le nombre total de clusters.

GTMS utilise une approche hybride de gestion de confiance, qui réduit le cout des évaluations de la confiance. Dans [161] R.A. Shaikh et al. présentent une étude comparative (An Extended Energy Consumption Analysis of Reputation-based Trust Management Schemes of Wireless Sensor Networks) entre les protocolee GTMS, RFSN et PLUS. Cette étude montre que, dans un scénario de recommandation, GTMS consomme moins d'énergie comparé avec les schémas PLUS et RFSN. De plus GTMS est moins couteux en mémoire par rapport aux deux schémas présentés ci-dessus (PLUS et RFSN). La mémoire consommée au niveau des CHs en utilisant le schéma GTMS, dépendra du nombre des clusters dans le réseau. Au fur et à mesure que le nombre de clusters augmente, la consommation requise en

mémoire augmente linéairement au niveau des CHs. Par exemple, si le réseau consiste en 100 clusters avec une moyenne de 20 nœuds par cluster, alors en employant GTMS, un CH consomme 2, 832 octets de mémoire. Cela montre que GTMS est adéquat aux réseaux à large échelle.

On a aussi prouvé, que GTMS est tolérant aux intrusions et fournit une protection contre les nœuds malicieux, égoïstes, et ceux défectueux.

Dans beaucoup d'applications, les identités des senseurs doivent être cachées pour faire face au problème de l'anonymat. Ainsi, le challenge de GTMS est comment établir et maintenir la confiance entre les nœuds communicants dans un environnement à identités anonymes.

3.12.4 GCP (Generic Communication Protocol)

Afin de calculer la consommation d'énergie, on doit d'abord avoir des informations à propos de nombre des bits transmis et reçus durant la phase d'évaluation de confiance, dans les différents nœuds. Le nombre de bits est calculé en utilisant des protocoles de communication spécifiques. Pour cela, R.A.Shaikh, Y.K.Lee et S.Lee proposent dans [162], un protocole de communication générique (GCP), utilisé pour transmettre les valeurs de confiance entre les différents nœuds. Le format des paquets de GCP est montré dans la figure 8. Dans laquelle IDsrc représente l'identité du nœud source, qui consiste en 2 octets [96], [163]. IDdest est l'identité du nœud destination. IDnexthop est l'identité du prochain saut.

Seq# représente le nombre séquentiel du paquet. ProtID représente l'identité du protocole de gestion de confiance, ex : RFSN, PLUS, etc. Le champ type identifie le type du paquet, ex : requête, réponse. Le champ Payload est réservé pour la taille des variables, et contient les données spécifique au type (mentionné dans le champ Type) et celles du protocole, comme ex : la valeur de confiance, l'identité du nœud évalué (IDeval), etc. MAC est le code d'authentification du message (Message Authentification Code) utilisé pour vérifier l'authenticité et l'intégrité des paquets. La taille du champ MAC est 4 octets.

ID_{src}	ID_{dst}	$ID_{nexthop}$	Seq#	ProID	Type	Payload	MAC
2 bytes	2 bytes	2 bytes	2 bytes	1 byte	1 byte	variable	4 bytes

Figure 8. Le format d'un paquet GCP

Dans [162], les auteurs ont fait une étude des protocoles GTMS, PLUS et RFSN en employant le protocole de communication des valeurs de confiance GCP. Les résultats montrent une amélioration significative de la consommation énergétique des différents schémas de gestion de confiance dans différents scénarios.

3.12.5 2-ACKT (Two-Way Acknowledgment-Based Trust Framework for Wireless Sensor Networks)

Dans [164], X. Anita, J. M.L.Manickam et M. A. Bhagyaveni (Avril 2013) proposent 2-ACKT, un nouveau schéma pour la gestion de confiance dans un WSN. Le protocole 2-ACKT calcule la confiance directe en utilisant l'acquittement de la couche liaison et l'acquittement à deux sauts d'un voisin. Le modèle 2-ACKT se base sur les acquittements de la couche liaison, et exploite aussi la nature dense du réseau pour tirer les valeurs de confiance des voisins. Au départ, chaque nœud fait confiance à l'autre pour l'acheminement des données. Le composant du monitoring du capteur dans un schéma basé confiance doit assurer : (1) Un voisin a reçu le paquet avec succès, et (2) Il l'a relayé honnêtement à ses voisins en suivant les instructions du protocole.

Dans 2-ACKT, l'occurrence de l'événement (1) sera assurée en utilisant l'acquittement de la couche liaison (Link Layer Acknowledgment, LLACK) généré avec le protocole MAC IEEE 802.15.4. Dans ce standard, l'émetteur garde les paquets dans son cache jusqu'à ce qu'il reçoit un LLACK de la part du récepteur. Quand le récepteur reçoit le paquet avec succès, il renvoie un LLACK à l'émetteur du paquet. Si l'émetteur ne reçoit pas une trame LLACK dans un temps prédéfini, donc il doit retransmettre ce paquet.

L'occurrence de l'événement (2) sera assurée en unicastant un acquittement à deux sauts au nœud source. 2-ACKT améliore les schémas de confiance qui se base sur des mécanismes multi-sauts en termes de durée de vie du réseau, réduction du trafic et les besoins en mémoire.

3.12.6 TSRF (A Trust-Aware Secure Routing Framework in Wireless Sensor Networks)

Dans [165], Junqi Duan et al. (Janvier 2014) proposent le protocole TSRF Léger avec de hautes habilités de résistance aux différentes attaques. Ce schéma combine des métriques de confiance et de QoS pour le routage afin de présenter un algorithme de routage optimisé. Dans TSRF, quand un nœud source v0 se prépare à envoyer un paquet à une destination, nœud v11 par exemple, le nœud v0 initialise le processus de dérivation des valeurs de confiance et envoie une requête trust à ses voisins. Un nœud qui reçoit une requête trust doit d'abord vérifier s'il a déjà reçu la même requête. Dans ce dernier cas, la requête doit être immédiatement ignorée. Sinon, le nœud diffusera cette requête à tous ses voisins. Une fois la requête de route arrive à sa destination, le nœud destination renvoie un Reply de route au nœud source via la route sélectionnée en inverse.

Après l'obtention des recommandations fournies par les voisins du nœud évalué, le nœud évaluateur v0 calcule la valeur de confiance en combinant la confiance directe et indirecte. Si un nœud intermédiaire digne de confiance, qui reçoit cette requête de route, a la route

optimale au nœud destination, il renvoie un Reply de route au nœud source. Ainsi, le nœud source peut trouver la route optimale au nœud destination.

Finalement, le nœud source peut envoyer les paquets de données au nœud destination à travers la route optimale. Les nœuds égoïstes qui ne participent pas dans le mécanisme de recommandation pour sauvegarder leurs batteries, leurs valeurs de confiance seront dégradées dans ce modèle de confiance et enfin ils seront expulsés du réseau.

3.12.7 A Secure Trust Establishment Scheme for Wireless Sensor Networks

Dans [166], F.Ishmanov, S.W.Kim et S.Y.Nam proposent (January 2014), une nouvelle méthode d'estimation de confiance, robuste contre les attaques on-off et persiste aux comportements malicieux. De plus, afin d'agréger les recommandations d'une manière sécurisée, F.Ishmanov et al. proposent une solution en utilisant le schéma *modified one-step M-estimator (MOSE)* [168,169]. La nouveauté du schéma proposé parait dans la combinaison des mauvaises conduites passées avec l'état courant d'une manière compréhensive. Spécifiquement, on a introduit un composant de mauvais comportements agrégés dans l'estimation de la confiance, qui aide à détecter une attaque on-off et les mauvais comportements persistants. Afin de déterminer l'état courant du nœud, en emploie les valeurs de confiance précédentes et les composants de mauvais comportements actuels. Ces composants sont combinés pour obtenir une valeur de confiance robuste.

Ce mécanisme assure qu'un nœud peut détecter les comportements des autres nœuds dans sa portée de communication, c.-à-d., un nœud peut écouter les transmissions de ses voisins, et avec ce moyen, il peut détecter si un voisin achemine ses paquets ou non. De plus, la valeur de confiance est estimée pour chaque intervalle de temps par chaque nœud en se basant sur les résultats de détection dans l'intervalle de temps spécifié. Les analyses théoriques et les évaluations montrent que ce schéma est plus performant que les autres schémas de confiance en termes de détection des attaques on-off et les mauvaises conduites persistantes. Les résultats d'évaluation prouvent que ce schéma permet une haute détection des comportements malicieux et les attaques on-off par rapport aux protocoles GTMS et ReTrust. De plus, les recommandations peuvent être agrégées d'une manière sécurisée en utilisant le schéma proposé, et ce quand le pourcentage des recommandations malhonnêtes est au-dessous de 40%.

3.12.8 LDTS (A Lightweight and Dependable Trust System for Clustered Wireless Sensor Networks)

Dans [170], Xiaoyong Li et al. (juin 2013) proposent un nouveau système léger et fiable de confiance (LDTS) qui emploie un algorithme de clustering. Dans un cluster, la confiance indirecte d'un senseur est évaluée par son CH. Ainsi chaque senseur n'a pas besoin de maintenir un feedback des autres senseurs. LDTS facilite la prise de décision de confiance en se basant sur un schéma léger. Ce schéma réduit les risques et améliore l'efficacité du système à la résolution des problèmes de calculs de la confiance quand des évidences directes sont insuffisantes. Ce schéma se base sur deux sources d'information l'observation directe (Direct Trust Degree, DTD) et le feedback indirect (Indirect Trust Degree, ITD). Le degré de confiance d'un CH est évalué par deux sources d'informations : CH-to-CH direct et BS-to-CH. Durant la communication CH-to-CH, le CH maintient un enregistrement des interactions passées d'un autre CH. Ainsi, la valeur de confiance directe peut être calculée selon le nombre des interactions réussies et celles échouées. La SB demande périodiquement aux CHs les ratings de confiance de leurs voisins. Après l'obtention de ces ratings des CHs, la SB les agrège pour former une valeur effective d'un ITD.

Les évaluations montrent que LDTS exige moins de cout de communication que deux autres systèmes de confiance notables, GTMS et ATRM. Notons que LDTS est hautement adéquat aux WSNs à large échelle, avec peu ou beaucoup de clusters. Les résultats montrent que LDTS a un PSDR (*packet successful delivery ratio)* meilleur que GTMS. Notons aussi que LDTS demeure fiable dans un environnement hautement malhonnête. Avec 30 % de CMs malhonnêtes, LDTS montre des résultats meilleurs que GTMS.

PARTIE II

NOTRE CONTRIBUTION

TCR (Trust-based Clustering & Routing)

3.13 Présentation

TCR est un schéma léger de confiance qui emploie une topologie hiérarchique et évalue la fiabilité des nœuds entrant en communication dans le réseau à base de :

- Les acquittements du sink.
- Les recommandations des voisins dignes de confiance.

Ce protocole garantit un routage sécurisé grâce à la technique employée de recherche de meilleures routes. Le nouveau dans ce protocole est la mesure globale de la confiance pour un groupe de nœuds qui forme une route. Dans les schémas de confiance présentés jusqu'à présent, on se contente de vérifier qu'un voisin a acheminé le paquet qu'on lui a envoyé pour le qualifier digne de confiance, sans prendre en considération le paquet à qui il a été relayé. Sachant que ce voisin qui est vraiment fiable peut avoir des voisins égoïstes qui peuvent diminuer de la qualité de la route prise pour joindre le sink. La sélection d'un prochain saut de confiance n'est pas suffisant pour s'assurer que le paquet de donnée sera reçu par la destination. La construction des chemins sûrs est le principe de ce protocole. Considérons dans un chemin, un nœud S qui a un voisin V fiable, si à un moment donné le nœud V relaye le paquet reçu de S à un voisin A *non fiable* alors la fiabilité du chemin $S \rightarrow V$ sera diminuée, ainsi la valeur de confiance de S sera décrémentée. Un nœud prendra la responsabilité de son choix du prochain saut. En d'autres termes, un nœud qui achemine ses paquets à un voisin égoïste verra son degré de confiance diminué chez son voisin source du paquet reçu.

Le protocole TCR fonctionne comme suit :

1. Stratégie du routage

La communication dans le réseau

Le routage dans le réseau suit une stratégie hiérarchique. Une fois les cluster-heads (CHs) sont élus, et les clusters sont formés. Les membres d'un cluster envoient les données captées à leur chef (CH). La confiance du routage concerne les chefs de groupes. La méthode suivie pour le regroupement en clusters sera détaillée dans la section suivante.

Nous assumons dans ce qui suit que le sink ait une force de signal qui peut atteindre tous le réseau. L'évaluation du degré de confiance dans un tel schéma se déroule comme suit :

1. Calcul des interactions réussies

Chaque CH évalue la confiance directe de ses voisins CHs en se basant sur les acquittements reçus du nœud Sink. Pour chaque paquet de donnée créé, un CH associe un numéro séquentiel sur 2 octet et une destination directe (le prochain saut) dans son cache, le numéro séquentiel du paquet de donnée est initialisé à chaque round, et cela pour réduire la taille allouée au numéro séquentiel. Quand le CH envoie un paquet au sink via un nœud voisin, il envoie le paquet avec son numéro séquentiel et son identifiant (l'ID de la source). A l'initiation d'une communication, celle-là est considérée réussie. Dans un enregistrement du CH, deux valeurs sont associées pour chaque voisin:

$NbInt$: Le nombre d'interactions effectuées, et $GoodInt$: le nombre d'interactions réussies.

A la réception d'un paquet de donnée le sink parcourt son cache et compare le numéro séquentiel du dernier paquet reçu du CH source avec celui reçu actuellement, si le numéro séquentiel courant ne présente pas le suivant du dernier numéro séquentiel. Le sink envoie le numéro séquentiel du premier paquet perdu au CH concerné (le numéro séquentiel du paquet attendu), qui fait aussi un acquittement pour les paquets qui précédent le paquet perdu. Ce processus sera exécuté si le numéro séquentiel du paquet reçu est plus grand du numéro séquentiel du dernier paquet au moins de 3. Dans le cas où le numéro séquentiel du paquet reçu est plus grand de 2 du numéro du dernier paquet reçu, le sink lance un timer d'attente, afin de vérifier si le paquet considéré comme perdu n'a pas pris un chemin plus long. Si le timer expire et le paquet n'apparaît pas alors il est considéré perdu. Périodiquement, un acquittement est reçu par tous les CHs de la part du sink, même s'il n'ya pas eu une perte de paquets. Cet acquittement avise un CH du numéro du paquet attendu et acquitte les paquets précédents.

A la réception d'un acquittement de la part du sink, dans le cas d'une perte, le CH vérifie dans son cache la destination directe à qui il a relayé le paquet perdu, une fois trouvée, il décrémente le nombre d'interactions réussies.

De cette manière un senseur n'a pas besoin d'écouter ses voisins pour vérifier s'ils relayent les paquets envoyés, ce qui réduit le cout de communication et élimine les cas des faux négatifs et des faux positifs.

2. Envoie de recommandations à ses voisins

Chaque round est divisé en sous périodes, après chaque sous période, chaque CH diffuse à ses voisins CHs le vecteur de recommandations sur n*2 octets qui porte les identifiants des voisins jugés non fiables. n est le nombre de nœuds jugés non fiables. A la réception d'un vecteur de recommandations, le nombre de bonnes recommandations est incrémenté pour les nœuds ne figurant pas dans le vecteur.

A partir du moment où les valeurs de confiance seront établies, les recommandations des voisins non fiables ne sera pas prise en considération. Ce moyen empêche les attaques du type déni de service qui peuvent se lancer en diffusant des fausses informations sur la fiabilité d'un nœud. Et permet ainsi une diminution considérable de la consommation énergétique, le fait d'exécuter le processus d'envoie de recommandations d'une manière périodique, sans faire appel aux requêtes de recommandations. De plus, la taille des paquets recommandations portant les nœuds non fiables est considérée plus petite en assumant que le nombre de nœuds jugés fiables dépasse ceux jugés égoïstes. Cet échange est moins couteux en énergie comparé avec les autres schémas présentés dans la littérature. Prenons par exemple le protocole PLUS (section 3.12.1), pour évaluer un seul voisin, une paire de paquets sera échangée, une requête EReq sur 2 octets pour demander une recommandation sur un seul nœud senseur et une réponse ERep sur 6 octets pour répondre à la requête.

3. Agrégation des valeurs de confiance

Par la suite la valeur de confiance d'un voisin sera calculée comme montré dans la formule (1) :

$$T = \frac{\alpha}{2}\left(\frac{GoodInt}{NbInt} + \frac{GoodRec}{NbRec}\right) \ldots\ldots\ldots\ldots\ldots(1)$$

Où:

- T: (Trust) la valeur de confiance.
- $GoodInt$: le nombre d'interactions réussies.
- $NbInt$: le nombre d'interactions réalisées avec ce nœud.
- $GoodRec$: le nombre de recommandations « bonnes ».
- $NbRec$: le nombre de recommandations (le nombre de fois où j'ai reçu le vecteur de recommandations).

- α : Chaque nœud a une valeur pour ce coefficient, il est égal à 1 si aucun voisin ne déclare le nœud en question comme nœud « Interdit ». A chaque réception d'une recommandation « Interdit », le coefficient α se diminue.

$$\alpha = \frac{\kappa}{\text{NbNeighbor}} \quad \text{.....(2)}$$

- κ est le nombre de voisins ne déclarant pas le nœud en question comme « Interdit ».
- NbNeighbor est le nombre total de voisins.

La division par 2 se fait pour avoir une valeur de confiance toujours bornée entre 0 et 1.

4. Niveaux de confiance

L'une des caractéristiques de TCR est qu'une fois les valeurs de confiance sont mises à jour, la prise de décision deviendra un processus très simple. Chaque CH sépare ses voisins dans trois groupes selon les exigences suivantes :

- Pour une valeur de confiance supérieure ou égale à 0.5, le voisin est considéré « BON ».
- Pour une valeur de confiance entre 0.5 et 0.25, le nœud est considéré « MOYEN ».
- Pour une valeur de confiance inférieure à 0.25, le nœud est classé dans la catégorie des attaques. Son identifiant sera diffusé dans le groupe des voisins et aucune interaction n'est possible entre ce nœud et le nœud qui l'a détecté. Ce qui réduit les risques et augmente l'efficacité du système.

En recevant un message déclarant un nœud comme attaque, le coefficient α sera décrémenté, comme montré dans la formule (2).

2. Clustering

La méthode du clustering se fait d'une manière simple. Le but au départ est d'élire le nœud qui a le plus de voisins comme chef de groupe. A partir du deuxième round, l'intégrité des données transmises est prise en considération pour la sélection des cluster-heads. Une liste noire sera diffusée dans le groupe à la fin de chaque round de la part du chef du groupe, et cela pour interdire l'élection des nœuds figurant dans la liste comme CH pour le prochain round.

1. Diffusion des messages Hello

Au départ tous les nœuds sont considérés dignes de confiance, ce qui autorise la candidature de tous les nœuds à devenir cluster-heads. Les nœuds s'envoient des messages Hello pour la phase d'initialisation. Chaque nœud calcule à base des Hello reçus le nombre de ses voisins. A la fin de la période des Hello, chaque nœud se candidate à devenir CH et envoie à ses voisins une invitation pour joindre son groupe. Le paquet invitation contient l'identifiant du nœud et son degré. Après d'expiration du temps alloué aux échanges d'invitations, chaque nœud consulte le degré de ses voisins (nombre de voisins) dans les paquets reçus.

Un nœud choisit son chef selon l'algorithme 1 suivant :

Algorithme 1 :

Si (mon degré est le plus haut degré) alors

- Attendre l'arrivé des messages JOIN de mes voisins ;

Sinon (Si j'ai le plus haut degré qui est égal au degré de l'un de mes voisins) alors

// Celui qui a l'identifiant le plus petit sera sélectionné comme CH

Si (Mon identifiant est le plus petit) alors

- Attendre l'arrivé des messages JOIN de mes voisins ;

Sinon

- Envoyer un JOIN à ce nœud qui a l'identifiant plus petit ;

Sinon

- Envoyer un JOIN au nœud qui a le plus haut degré ;

2. Mesure de confiance au niveau du groupe

Les CMs (Cluster Membres) envoient les données captées au CH. Un CH associe à chacun de ses membres une valeur qui correspond au degré d'intégrité des données reçues de la part de ce CM. Cette valeur appelée « `InegrityDegree` » vaut 1 au départ, et elle est diminuée de γ pour chaque donnée reçue qui prend une valeur au-delà de l'intervalle souhaité. Nous assumons que l'intervalle des valeurs qui peuvent être captées dépend de la localisation du nœud senseur.

Le paquet de donnée contient l'identifiant du CM et la donnée. A la réception du paquet le CH consulte la donnée et vérifie si elle est dans l'intervalle des valeurs souhaitées. Si ce n'est pas le cas, le degré d'intégrité de la donnée qui correspond à `InegrityDegree` du nœud source est diminué de γ.

- Une valeur est affectée à γ durant la simulation.

3. Niveaux de confiance au niveau du groupe

A la fin d'un round chaque CH diffuse une liste noire qui contient l'ensemble des nœuds du groupe interdit à devenir CH.

L'évaluation de l'ensemble des membres du groupe se fait de la manière suivante :

- Si `InegrityDegree` <0.5, le membre rentre dans la liste noire.
- Sinon il est dans la liste blanche.

De cette manière un nœud saura s'il figure dans la liste noire, et saura ainsi qu'il ne peut pas se candidater à devenir CH pour le prochain round.

4. Clustering à partir du deuxième round

A la fin d'un round si l'énergie résiduelle d'un CM passe au-dessous de la valeur du seuil d'énergie requis pour devenir CH, le CM ne se candidate pas à devenir CH pour le prochain round. Une manière simple de s'auto-isoler de la candidature à devenir CH, si le niveau d'énergie est insuffisant.

Une fois la liste noire soit diffusée, les nœuds ne figurant pas dans la liste et ayant le niveau d'énergie requis pour devenir CH, diffusent leurs invitations à joindre leurs groupes. Le processus du clustering est similaire au précédent, cependant à partir de ce round, les nœuds vérifient l'identité des paquets invitations reçus s'ils ne parviennent pas d'un nœud figurant dans la liste noire. La métrique `InegrityDegree` de chaque nœud sera initialisé par le nouveau CH et les résultats de la liste noire du nouveau round correspondent aux activités des nœuds durant le round courant. Cette technique éliminera les dommages qui peuvent être causé si un CH non fiable à été élu, durant un round.

Ce protocole considère tout comportement sortant de la norme, un comportement égoïste si la valeur de confiance d'un nœud ne passe pas au dessous d'un seuil. Un nœud égoïste pourra améliorer son score et joindre le groupe des nœuds fiables si son degré de confiance dépasse le seuil fixé auparavant. Dans le cas contraire, si le degré de confiance d'un senseur est inférieur au seuil, le nœud est considéré comme un nœud attaquant ou un nœud défectueux et il ne pourra pas améliorer son score.

Les paquets de TCR

Type	Paquet	Taille
BkLPacket (CH→CM)	V (nombre de nœuds de la liste noire), l'ID du nœud (2 octet).	2 octet x v (La taille du vecteur)
DataPacket (CM→CH, CH→Sink)	La donnée (1 octet), SeqNum (2 octet), la source (2 octets)	5 octets
RecPacket (CH→CH)	N (nombre de nœuds jugés non fiables), l'ID du nœud (2 octet).	2 octet x N (La taille du vecteur)
Ack (Sink→CH)	SeqNum (2 octet), ID destination (2 octets)	4 octet
ADV_CH (CM→CM)	Nombre de voisins (4 bits), l'ID du CM (2 octets)	2 octets + 4 bits
Hello (CM→CM)	L'ID du nœud (2 octets)	2 octets
JOIN	L'ID du nœud (2 octets)	2 octets

Tableau 3. Les paquets de TCR

BkLPacket : Black_List_Packet : le paquet contenant la liste noire.

DataPacket : Le paquet de donnée.

SeqNum : Numéro séquentiel du paquet.

RecPacket : Le paquet recommandation.

Hello : paquet hello d'initialisation.

JOIN : paquet JOIN, pour joindre le groupe d'un CH.

5. Modélisation d'un round dans TCR

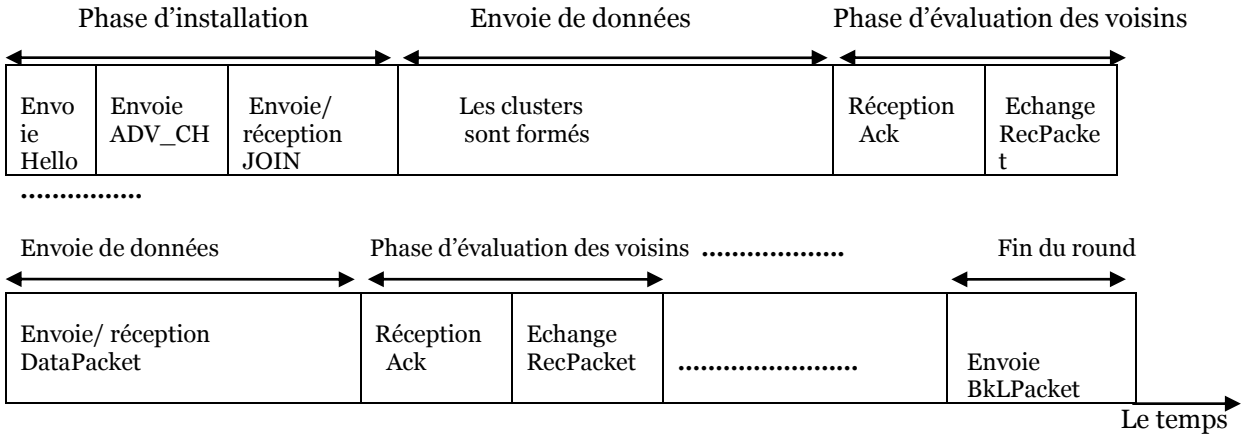

Figure 9. Modélisation d'un round dans TCR

Conception

a. Outil de conception

L'outil utilisé pour l'implémentation de notre système de confiance est OMNET++, un environnement de simulation Open source à évènements discrets, orienté objet et basé sur C++ [Omnet, 07]. Son développement a commencé en 1992 par Andras Vargas [Vargas, 07] à l'université de Budapest.

Actuellement, Ce simulateur est utilisé par des dizaines d'université pour la validation de nouveaux matériels et logiciels, ainsi que pour l'analyse des performances et l'évaluation des protocoles de communication. OMNET++ est devenu rapidement une plateforme de simulation populaire que ce soit pour la communauté des scientifiques ou des industriels.
Un modèle de simulation OMNET++ consiste en un ensemble d'entités appelées « Modules » qui communiquent entre elles en échangeant des « Messages » qui seront émis à travers des « Connections » et des «gates » (portes). Grâce à son architecture modulaire, OMNET++ offre une possibilité d'étendre le simulateur en implémentant un nouveau modèle spécifique aux réseaux de capteurs.
L'architecture du modèle sur lequel on a implémenté notre système est la suivante :

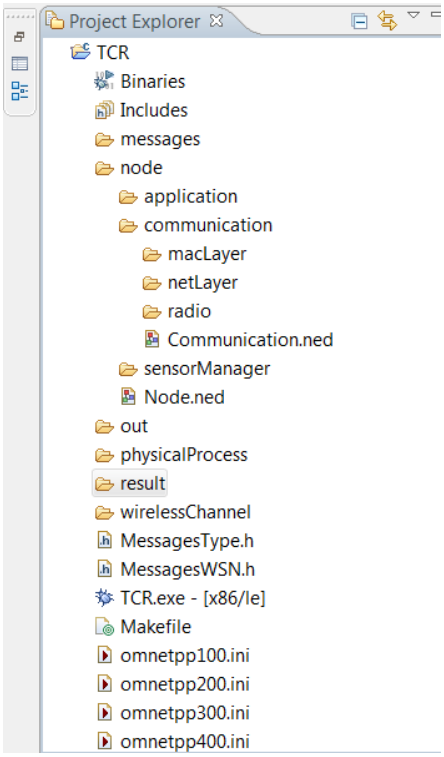

Figure 10. Modèle de simulation du protocole TCR

b. Tests et évaluations

Pour évaluer les performances de notre protocole TCR, une étude comparative entre notre protocole TCR et deux autres protocole PLUS et RFSN présentés ci-dessus fera l'objet de ce qui suit.

Pour l'analyse de la consommation énergétique, nous assumons en premier ordre un modèle radio, dans lequel l'énergie dépensée pour transférer un paquet de k bits à une distance d, et recevoir un paquet, est suggéré par H.O. Tan et I.Korpeoglu dans [171] est :

$$E_{Tx}(k,d)=k*E_{elec}+k*d^2*E_{amp} \quad \text{.....(3)}$$

$$E_{Rx}(k)=k*E_{elec}$$

Ici, E_{elec} est l'énergie dissipé par le module radio dans l'ordre de faire fonctionner le circuit d'envoie ou de réception et il est égal à 50Nj/bit. E_{amp} est l'amplificateur de transmission.

Scenario 1 : Les recommandations entre les Cluster-heads

Dans le cas de PLUS et RFSN, quand un nœud senseur a besoin d'une recommandation à propos des autres nœuds, il envoie une requête à ses voisins. Dans le cas de TCR, un CH reçoit périodiquement de ses voisins un vecteur de recommandation qui précise les nœuds égoïstes.

Dans le cas de TCR, les recommandations sont envoyées périodiquement, la requête demande de recommandation n'existe pas dans ce schéma. L'énergie consommée par l'émetteur du paquet recommandation est :

$$E= E_{Tx}(16*m, d) \quad \text{.....(4)}$$

L'énergie consommée par le récepteur du paquet recommandation :

$$E= E_{Rx}(16*m) \times Nb \quad \text{.....(5)}$$

- m est la taille du vecteur recommandation.
- Nb représente le nombre de voisins.
- 16 représente la taille du champ réservé à l'identité d'un nœud.

Dans le cas de RFSN, l'énergie consommée par l'émetteur de la requête de recommandation est :

$$E = n \times [E_{Tx}(16, d) + E_{Rx}(48)] \quad \text{.....(6)}$$

Où :

- n représente le nombre de nœuds digne de confiance dans un cluster.
- 16 et 48 représentent respectivement la taille du paquet requête et du paquet recommandation du schéma RFSN.

Aussi, dans RFSN l'énergie consommée l'émetteur du paquet réponse est :

$$E = E_{Tx}(48, d) + E_{Rx}(16)$$
$$E = 16 * E_{elec} + 48\,(E_{elec} + d^2 * E_{amp}) \quad \text{.....(7)}$$

Dans le cas du protocole PLUS, l'énergie consommée par l'émetteur de la requête de recommandation est :

$$E = E_{Tx}(16, d) + (n-2)E_{Rx}(48)$$

$$E = 16(E_{elec} + d^2 * E_{amp}) + (48\,E_{elec}) \quad \text{.....(8)}$$

Où :

- n est le nombre des Cluster-heads.

Pour l'énergie consommée par l'émetteur du paquet réponse est :

$$E = E_{Tx}(48, d) + E_{Rx}(16)$$

$$E = 48(E_{elec} + d^2 * E_{amp}) + (16\,E_{elec}) \quad \text{.....(9)}$$

- 16 et 48 représentent respectivement la taille en bits du paquet requête et réponse du schéma PLUS.

Le résumé de la consommation énergétique durant les recommandations entre les Cluster-heads est présenté dans le tableau .4. Où n représente le nombre total des cluster-heads voisins. m est la taille du vecteur recommandation dans TCR.

	TCR	RFSN	PLUS
Nombre de requêtes envoyés	0	t≤n-2 pour évaluer un seul voisin	1 pour évaluer un seul voisin
Nombre de recommandations reçus	1 pour évaluer tous les voisins	t≤n-2 pour évaluer un seul voisin	n-2 pour évaluer un seul voisin
Taille de la requête	/	16 bits	16 bits
Taille de la réponse	m*16 bits	48 bits	48 bits
L'énergie consommée à la demande	E_{Tx} (m*16, d)	n × [E_{Tx} (16, d) + E_{Rx} (48)]	E_{Tx} (16, d) + $(n-2)E_{Rx}$ (48)
L'énergie consommée à la réponse	E_{Rx} (m*16)	E_{Tx} (48, d) + E_{Rx} (16)	E_{Tx} (48, d) + E_{Rx} (16)

Tableau 4. Les paires de recommandations entre les Cluster-heads

Afin de comparer la consommation énergétique durant les scénarios de recommandations entre les Cluster-heads. Nous assumons avoir les paramètres de simulation suivants :

Paramètre	**Valeur**
Surface de simulation	**100*100*50**
Localisation de la SB	**(0,0,0)**
Nombre de nœuds	**100, 200, 300**
Nombre des SB	**1**
Période d'un Round	**10 s**
Temps de simulation	**200 s**
Nombre de CHs égoïstes	**3, 5, 7**

Tableau 5. Paramètres de simulation du scénario 1

La figure.11 montre clairement que TCR consomme moins d'énergie comparé avec les schémas RFSN et PLUS. Et cela parce que dans TCR, on ne fait pas appel aux requêtes pour demander des recommandations de ses voisins. L'envoie des paquets recommandations se fait périodiquement. De plus, le paquet recommandation ne contient que les identifiants des nœuds jugés digne de confiance. Tandis que, dans RFSN et PLUS les cluster-heads envoient des requêtes de recommandation à chaque fois qu'un CH a besoin d'une recommandation à propos d'un voisin. Cette figure montre aussi que le schéma PLUS consomme moins d'énergie, du moment que le paquet requête est diffusé à tous ses voisins cluster-heads. Tandis que, dans le schéma RFSN, le paquet requête est envoyé en unicast à tous les voisins cluster-heads dignes de confiance.

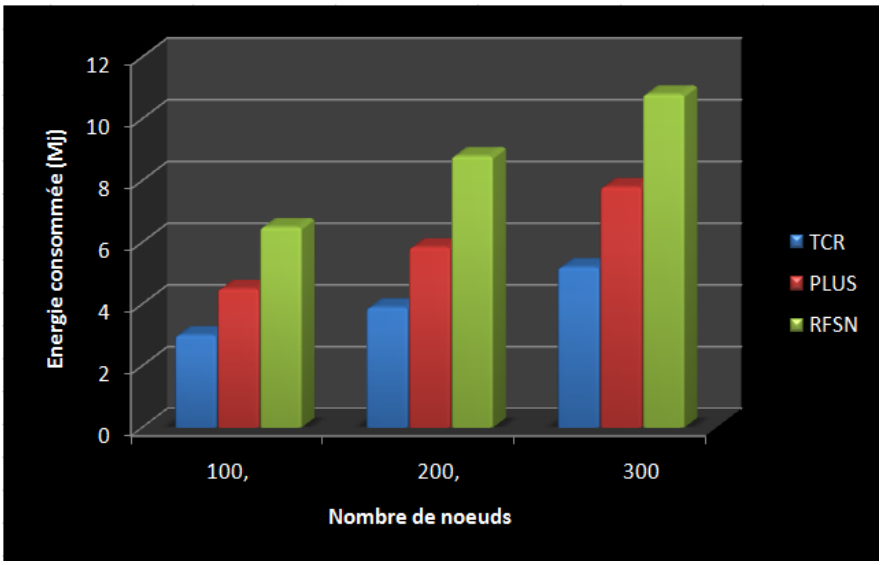

Figure 11. Energie consommée dans chacun des schémas TCR, RFSN et PLUS pour les scénarios de recommandation

Scenario 2: Taux de perte de paquets à la présence des nœuds égoïstes

Dans un scénario de 100 nœuds, on a configuré 3 CHs qui peuvent se comporter comme nœuds égoïstes. Pour s'approcher de la réalité, les nœuds sélectionnés ne vont pas agir directement comme des nœuds égoïstes, on a configuré le déroulement de la communication selon l'algorithme 2 suivant pour ces trois CHs :

Algorithme 2

- Je tire un nombre entre 0 et 1 ;

Si le nombre tiré est supérieur à 0,5 alors

- Egoïste=faux ;
- j'achemine le paquet ;

Sinon// Je suis égoïste

- Egoïste=vrai ;
- je tire une deuxième fois un nombre au hasard entre 0 et 1
  Si le nombre tiré est supérieur à 0,5 alors
  - j'achemine le paquet ;

  Sinon //Le nombre est inférieur à 0,5
  - Je supprime le paquet ;

  Fin Si

Fin Si

Les paramètres de simulation sont les suivants :

Paramètre	Valeur
Surface de simulation	**100*100*50**
Localisation de la SB	**(0,0,0)**
Nombre de nœuds	**100**
Nombre de clusters	**8**
Nombre des SB	**1**
Période d'un Round	**10 s**
Temps de simulation	**200 s**
Taux maximum des CHs égoïstes	**37,5%**

Tableau 6.Paramètres de simulation du scénario 2

La figure.12 suivante compare le taux moyen des paquets perdus en appliquant TCR, et dans le cas où ce mécanisme n'est pas appliqué.

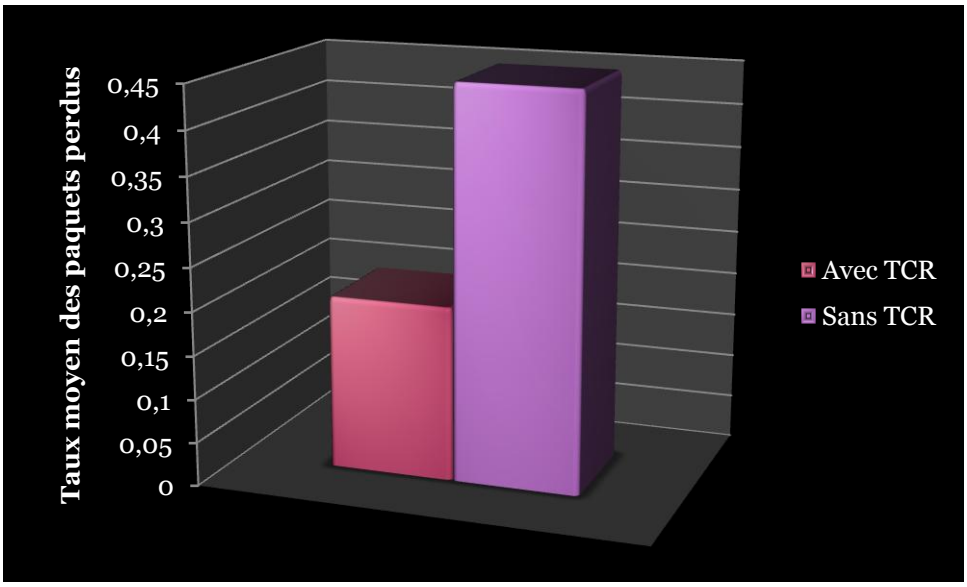

Figure 12. Taux moyen des paquets perdus

Le taux moyen de paquets perdus est calculé par rapport au nombre total de paquets envoyés via le réseau.

Les résultats présentés dans cette figure sont obtenus, pour une taille de réseau de 100 nœuds. Ces résultats sont obtenus à partir de la moyenne des résultats de 20 simulations différentes pour le même scénario. La différence apparaît clairement dans le cas d'utilisation de TCR. Les notes de confiance attribuées aux nœuds routeurs diminuent les interactions avec les nœuds qui ont des notes faibles de confiance, ce qui diminue considérablement le taux de paquets perdus.

Scenario 3: Taux de perte de paquets à la présence des nœuds égoïstes et des attaques BlackHôle

Afin de s'approcher des scénarios réels, avec 100 nœuds, on a configuré 3 CHs qui peuvent se comporter comme nœuds égoïstes ou attaque, c-à-d on ne connait pas le nombre d'attaque ou de nœuds égoïstes. A l'initialisation, chacun des CHs sélectionnés tire un nombre au hasard, si le nombre tiré est supérieur à 0,5 le nœud se comportera comme nœud égoïste, s'il est inférieur à 0,5 le nœud se comportera comme attaque BlackHôle durant tous le temps de la simulation. Comme décrit dans l'algorithme 3.

Algorithme **3**

A l'initialisation ;

Je tire un nombre entre 0 et 1 ;

Si le nombre tiré est supérieur à 0,5 alors

- Egoïste=vrai ;

Sinon

- BlackHôle=vrai ;

Fin Si

A la réception d'un paquet par l'un de ces CHs, la procédure de traitement sera comme dans l'algorithme 4 suivant.

Algorithme **4**

Si (Egoïste =vrai) alors

- je tire une deuxième fois un nombre au hasard entre 0 et 1
  Si (le nombre tiré est supérieur à 0,5 alors
  - j'achemine le paquet ;

  Sinon
  - Je supprime le paquet ;

  Fin Sinon

Sinon Si (BlackHôle=vrai) alors

- Je supprime le paquet ;

Fin Si

Dans une simulation aucune attaque n'a été détectée. Néanmoins, un taux considérable de perte de paquets est survenu à cause des nœuds égoïstes présents dans le groupe des CHs.

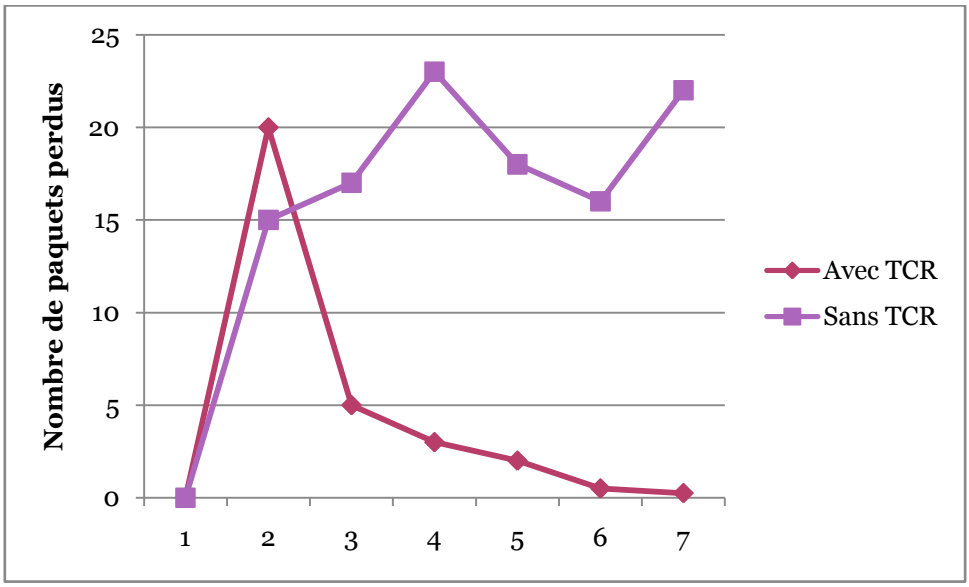

Figure 13. Comparaison en le taux de perte de paquets avec TCR et sans TCR

Les simulations ont montré aussi qu'un important taux de perte de paquets est survenu durant les premiers rounds dans le cas de l'application de TCR. Cela peut être expliqué, par le fait de considérer tous les voisins dignes de confiance au déploiement du réseau, et à force de désobéir à la tâche d'acheminement des paquets la communication avec les nœuds qui se comportent d'égoïsme s'affaiblie. Ce qui affaiblie en parallèle le taux de perte de paquets. Donc, on aura une diminution dans la perte des paquets après les premiers rounds.

Dans une autre simulation une attaque a été détectée par 3 CHs, durant les rounds 6 et 7. Ce qui veut dire aussi, que les deux autres nœuds présentent des nœuds égoïstes. La figure.14 représente le taux de perte de paquets à la présence d'une attaque BlackHôle et de deux nœuds égoïstes dans un groupe de 8 clusters. Ce qui veut dire 37,5% des CHs sont égoïstes. Et 12,5% des CHs se comportent comme des attaques internes.

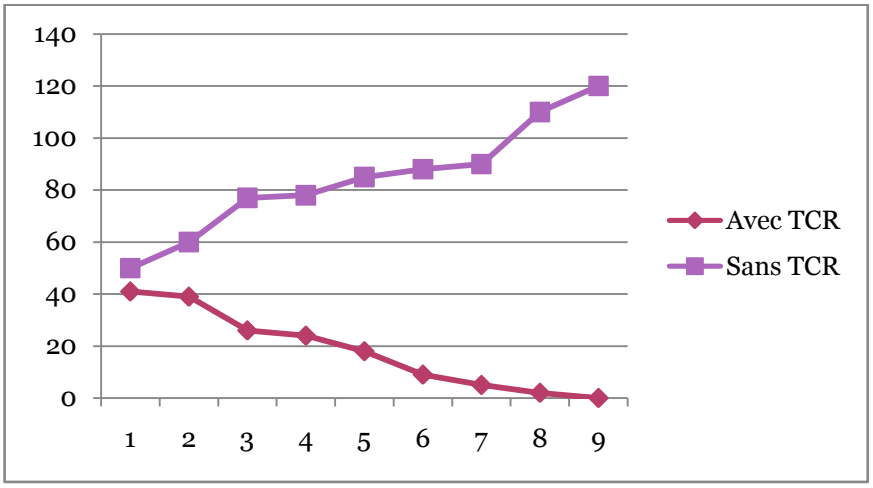

Figure 14. Taux de perte de paquets avec 12,5 % d'attaques et 37,5% de nœuds égoïstes

Comme présenté dans la figure.14, avec TCR un taux important de perte de paquets au départ, puis on voit une diminution considérable de cette perte au fil du temps. Cela paraît mieux à partir du round 6 là où les CHs commencent de détecter l'attaque. Par contre sans utiliser TCR, un important taux de perte de paquets qui évolue avec le temps, suite à l'absence des contrôles sur la réception des paquets envoyés et le degré de confiance des voisins.

Dans les deux cas de simulations présentés ci-dessus, les nœuds se comportant d'égoïsme et les attaques agissent seulement sur le routage des paquets de donnée (acheminer ou supprimer si nœud égoïste, ou supprimer si attaque). Par contre, ils ne touchent pas au processus de recommandation pour nuire à leurs voisins.

Scenario 4: Détection des nœuds malicieux à l'intérieur d'un cluster

Le processus de génération des paquets de données

Dans TCR, une fois un paquet est créé, un nœud considéré légitime tire aléatoirement une valeur dans l'intervalle des valeurs souhaitées à être captées. A la réception de ce paquet, un CH vérifie si la donnée est dans l'intervalle des valeurs acceptées. Pour tester le mécanisme proposé qui vérifie l'intégrité des données envoyées par les membres, on a étendu l'intervalle des valeurs acceptées, et on a configuré 3 membres d'un cluster, d'une taille de 8 éléments, qui tirent des valeurs de cet intervalle. Avec cette manière, on ne saura pas réellement si un de ces trois membres envoient une donnée erronée, ou correcte. Le choix de ce scénario est fait pour s'approcher un peu du monde réel.

Les paramètres de simulation sont les suivant :

Paramètre	Valeur
Nombre de membres	**8**
Nombre de clusters	**1**
Nombre de round	**1**
Temps de simulation	**10 s**
Valeur de γ	**0,125**

Tableau 7. Paramètres de simulation pour le scénario 4

Dans une simulation deux nœuds de ceux sélectionnés à agir anormalement sont classés dans la liste noire. Dans une autre simulation les trois membres choisis sont classés dans la liste noire.

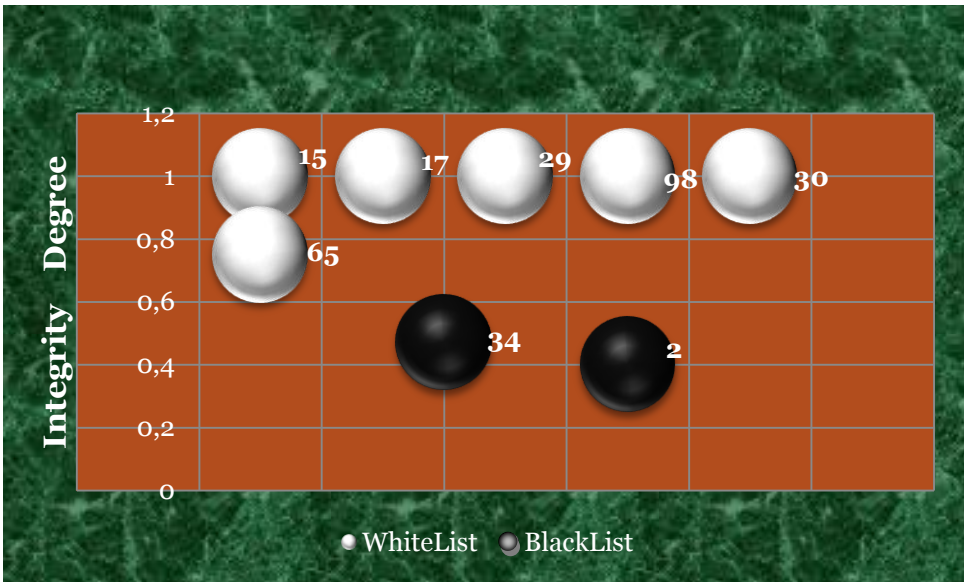

Figure 15. WhitList et BlackList dans un cluster de 8 membres (simulation 1)

Cette figure montre l'évaluation d'un CH à ses membres selon la métrique présentée auparavant pour la vérification de la plage des données reçus de ses membres. IntegrityDegree des membres 15,17,29,98 et 30 est toujours dans 1. Ce groupe de nœuds compose la liste blanche. Pour les trois membres sélectionnés dans ce scénario à envoyer des donnée dans une plage étendu que celle prévu, deux sont classés dans la liste noire (les nœuds 34 et 2) et un autre le nœud 65 est resté dans la liste blanche avec une valeur de 0,75 pour IntegrityDegree. Dans les résultats de la figure.16 d'une autre simulation tous les nœuds sélectionnés à agir anormalement dans ce scénario sont mis dans la liste noire. Cette différence entre les deux simulations est due au tirage aléatoire des valeurs de données envoyées. Ce qui veut dire que si un nœud tombe plusieurs fois dans des valeurs non

acceptées, il sera mit dans la liste noire. Tandis que, si les données envoyées sortent rarement de la plage acceptée, le membre reste dans la liste blanche.

La valeur attribuée à γ, qui rentre dans la formule d'évaluation de la métrique IntegrityDegree est choisit après plusieurs simulations, pour ne pas condamner les nœuds légitimes qui envoient des données, parfois qui sortent de la plage des valeurs acceptées.

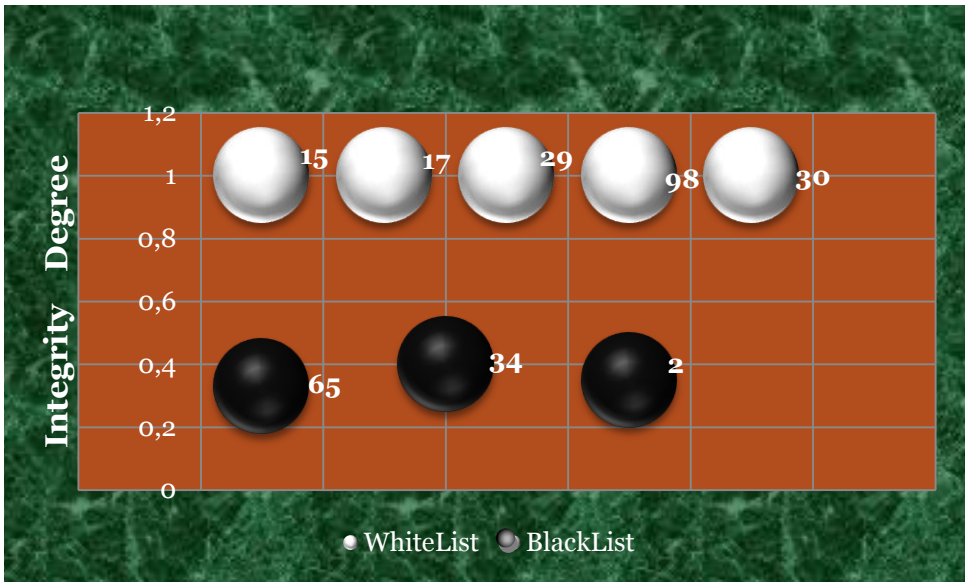

Figure 16.WhitList et BlackList dans un cluster de 8 membres (simulation 2)

3.14 Conclusion

Avec l'émergence des WSNs, le besoin des mécanismes de confiance présente un point primordiale sans lequel les données reçues n'orant aucun sens. Beaucoup de schémas de confiance ont été proposés jusqu'à présent, mais ces mécanismes restent à améliorer en prenant en compte les points non considérés et les avancements enregistrés dans le domaine des réseaux de capteurs.

Dans ce chapitre, nous avons proposé un protocole distribué de gestion de confiance des capteurs dans un réseau à topologie hiérarchique. Le but principal de TCR est la minimisation de l'énergie consommée lors des échanges de recommandations. Une méthode légère d'évaluation des voisins est proposée dans ce protocole afin de réduire les calculs et les traitements relatifs à l'agrégation des données de confiance.

Conclusion générale

Depuis le début de ce travail, nous avons essayé de proposer une solution light, moins couteuse en énergie et qui ne demande pas trop de calculs pour évaluer la confiance du voisinage d'un nœud capteur. Nous avons commencé notre travail par une étude théorique, au départ pour tous les aspects de sécurité existants, l'étude des méthodes les plus appliquées dans les réseaux de capteurs en prenant en considération les contraintes qui régissent ces systèmes. Nous avons concentré notre étude par la suite sur la confiance dans les réseaux de capteurs, les schémas de confiance existants et les travaux proposés qui ont connu un succès comme GTMS par exemple. L'étude des différents systèmes de confiance, nous a permet de souligner les facteurs majeurs de consommation d'énergie dans de tels systèmes, prenons par exemple l'écoute passive que peut appliquer un capteur pour savoir si ses paquets sont relayés par ses voisins, ce qui prévit le capteur de la mise en veille en laissant son antenne toujours active. Ainsi, le trafic circulant pour échanger les recommandations entre les voisins doit être réduit au maximum pour réserver la grande partie d'énergie au captage et à la transmission des données importantes. De plus, la fonction d'agrégation des valeurs de confiance ne doit pas abuser le microcontrôleur avec trop de traitements et calculs. Tous ces points et d'autres ont été considérés pour arriver enfin à une solution simple et efficace, qui peut remédier à plusieurs limites rencontrées dans de tels systèmes. Comme il a été démontré par les résultats de simulation, TCR est efficient en énergie, et fournit une protection contre les nœuds malicieux, égoïstes et ceux défectueux.

Néanmoins, chaque solution a ses limites, du fait qu'aucune proposition ne peut pallier à tous les problèmes de sécurité et de consommation d'énergie existants. Par exemple dans TCR, le modèle de vérification d'intégrité des messages au niveau d'un cluster n'est pas proche aux modèles utilisés dans la réalité. Pour quelques applications on ne peut pas connaitre l'intervalle des valeurs qui peuvent être captées. En outre l'évaluation se fait pour un seul round, ce qui augmente le risque si une attaque interne se comporte normalement durant un round, et exécute des faits malicieux durant un autre. Ce modèle peut être appliqué, par contre dans des applications simples comme les mesures de température. Une autre limite qui paraît aux yeux dans ce travail, est que si le sink est attaqué tous le système de confiance sera paralysé, du fait qu'il est le premier élément dans le système qui affirme les interactions réussies.

Comme perspectives, on propose d'intégrer l'énergie dans la fonction de calcul des niveaux de confiance et d'améliorer le modèle de vérification de l'intégrité des données captées. Ce

qu'on pense d'intégrer dans notre système est le fait qu'un nœud qui ne participe pas dans le processus des recommandations, sa valeur de confiance sera diminuée. De plus, on envisage d'étudier la robustesse de notre système face à d'autres attaques, celles les plus connues surtout.

Dans ce travail, on n'a pas mené une étude comparative sur le plan perte de paquets avec d'autres schémas de confiance, pour l'instant on s'est contenté d'une étude comparative sur le plan énergétique. Pour ce qui est de la résistance aux différentes attaques et l'évaluation de ce schéma en comparaison avec les autres proposés dans la littérature ça reste un point à étudier dans un prochain temps.

Références

[1]: Mauri Kuorilehto, Mikko Kohvakka, Jukka Suhonen, Panu H¨am¨al¨ainen, Marko H¨annik¨ainen, and Timo D. H¨am¨al¨ainen “Ultra-Low Energy Wireless Sensor Networks in Practice Theory, Realization and Deployment », *Tampere University of Technology, Finland*.

[2]: Reason JM and Rabaey JM 2004 A study of energy consumption and reliability in a multi-hop sensor network. *ACM SIGMOBILE Mobile Computing and Communications Review* **8**(1), 84–97.

[3]: Min R, Bhardwaj M, Cho SH, Ickes N, Shih E, Sinha A, Wang A and Chandrakasan A 2002 Energycentric enabling technologies for wireless sensor networks. *IEEE Wireless Communications* **9**(4), 28–39.

[4]: Crossbow Technology 2004a Micaz wireless measurement system. Available at: xbow.com/Products/Product_pdf_files/Wireless_pdf/6020-0060-01_A_MICAz.pdf.

[5]: Warneke B, Last M, Leibowitz B, and Pister KSJ 2001 Smart dust: Communicating with a cubicmillimeter computer. *Computer* **34**(1), 43–51.

[6]: Savvides A and Srivastava MB 2002 A distributed computation platform for wireless embedded sensing. *Proc. IEEE Int'l Conf. on Computer Design: VLSI in Computers and Processors (ICCD'02)*, pp. 220–225, Freiburg, Germany.

[7]: Silicon Labs. http ://www.silabs.com/Pages/default.aspx.

[8]: Microchip Wireless Devices. http ://www.microchip.com/.

[9]: Stallings W 2004 *Data and Computer Communications* 7 edn. Prentice-Hall.

[10]: A. Cerpa, D. Estrin, ASCENT: adaptive self-configuring sensor networks topologies, UCLA Computer Science Department Technical Report UCLA/CSDTR-01-0009, May 2001.

[11]: N. Bulusu, D. Estrin, L. Girod, J. Heidemann, Scalable coordination for wireless sensor networks: self-configuring localization systems, International Symposium on Communication Theory and Applications (ISCTA 2001), Ambleside, UK, July 2001.

[12]: B.G. Celler et al., An instrumentation system for the remote monitoring of changes in functional health status of the elderly, International Conference IEEE-EMBS, New York, 1994, pp. 908–909.

[13]: C. Chen, I. Elgorriaga, C.McConaghy, Low-power direct sequence spread-spectrum modem architecture for distributed wireless sensor networks, ISLPED'01, Huntington Beach, California, August 2001.

[14]: R.J. Cramer, M.Z. Win, R.A. Scholtz, Impulse radio multipath characteristics and diversity reception, IEEE International Conference on Communications ICC'98 Vol. 3 (1998), pp. 1650–1654.

[15]: R. Colwell, Testimony of Dr. Rita Colwell, Director, National Science Foundation, Before the Basic Research Subcommitte, House Science Committe, Hearing on Remote Sensing as a Research and Management Tool, September 1998.

[16]: DSN Team, Multilateration Poster, SensIT Workshop, St. Petersburg, FL, April 2001.[17]: W. Heinzelman, A. Chandrakasan and H. Balakrishnan, “Energy-efficient Communication Protocol for Wireless Microsensor Networks”, *Proceeding of the 33rd Hawaii International Conference on System Sciences*, January 2000.

[18]: W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “An application specific protocol architecture for wireless micro sensor networks”, *IEEE Transaction on Wireless Networking*, vol. 1, no. 4, pp. 660-670, Oct. 2002.

[19]: P. Bauer, M. Sichitiu, R. Istepanian, K. Premaratne, The mobile patient: wireless distributed sensor networks for patient monitoring and care, Proceedings 2000 IEEE EMBS International Conference on Information Technology Applications in Biomedicine, 2000, pp. 17-21.

[20]: J.M. Cramer, R.A. Scholtz, M.Z. Win, On the analysis of UWB communication channels, IEEE MILCOM'99, 1999, pp. 1191–1195.

[21]: C. Chen, I. Elgorriaga, C.McConaghy, Low-power direct sequence spread-spectrum modem architecture for distributed wireless sensor networks, ISLPED'01, Huntington Beach, California, August 2001.

[22]: G.D. Abowd, J.P.G. Sterbenz, Final report on the interagency workshop on research issues for smart environments, IEEE Personal Communications (October 2000) 36–40.

[23] : A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson, "Wireless sensor networks for habitat monitoring," in *First ACM Workshop on Wireless Sensor Networks and Applications*, Atlanta, GA, Sept. 2002.

[24]: E. Biagioni and K. Bridges, "The application of remote sensor technology to assist the recovery of rare and endangered species," *International Journal of High Performance Computing Applications*, vol. 16, pp. 315–324, Aug. 2002.

[25]: P. Juang, H. Oki, Y. Wang, M. Martonosi, L. Peh, and D. Rubenstein, "Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with ZebraNet," in *Proceedings of the 10th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS-X)*, San Jose, CA, Oct. 2002.

[26]: G. Werner-Allen, J. Johnson, M. Ruiz, J. Lees, and M. Welsh, "Monitoring volcanic eruptions with a wireless sensor network," in *Proceedings of the Second European Workshop on Wireless Sensor Networks (EWSN'05)*, Jan. 2005.

[27]: J. Burrell, T. Brooke, and R. Beckwith, "Vineyard computing: Sensor networks in agricultural production," *IEEE Pervasive Computing*, vol. 3, no. 1, pp. 38–45, 2004.

[28]: "Zigbeef llc," 2011 (accessed September 22, 2011). [Online].Available http://inknowvation.com/node/16096.

[29]: I. Stoianov, L. Nachman, S. Madden, and T. Tokmouline, "Pipeneta wireless sensor network for pipeline monitoring," in *Proceedings of the 6th international conference on Information processing in sensor networks*, ser. IPSN '07. New York, NY, USA : ACM, 2007, pp. 264–273. [Online].Available : http://doi.acm.org/10.1145/1236360.1236396.

[30]: W. Dargie and C. Poellabauer, *Fundamentals of Wireless Sensor Networks: Theory and Practice*, ser. Wireless Communications and Mobile Computing. John Wiley & Sons, 2010. [Online] . Available: http://books.google.com/books?id=8c6k0EVr6rMC.

[31]: M. Srivastava, R. Muntz, and M. Potkonjak, "Smart kindergarten: Sensor- based wireless networks for smart developmental problem-solving enviroments," in *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking (MobiCom'01)*, Rome, Italy, pp. 132–138, 2001.

[32]: T. Fulford-Jones, D. Malan, M. Welsh, and S. Moulton, "CodeBlue: An ad hoc sensor network infrastructure for emergency medical care," in *International Workshop on Wearable and Implantable Body Sensor Networks*, London, UK, 2004.

[33]: D. Myung, B. Duncan, D. Malan, M. Welsh, M. Gaynor, and S. Moulton, "Vital dust: Wireless sensors and a sensor network for realtime patient monitoring," in *8th Annual New England Regional Trauma Conference*, Burlington, MA, 2002.

[34]: "Self-healing Mines" http://www.darpa.mil/ato/programs/SHM/.

[35]: M. Maroti, G. Simon, A. Ledeczi, and J. Sztipanovits, "Shooter localization in urban terrain," *IEEE Computer*, vol. 37, pp. 60–61, Aug. 2004.

[36]: Sihavaran, T., Blair, G., Friday, A., Wu, M., Limon, H. D., Okanda, P. & Sorensen, C. F. Cooperating sentient vehicles for next generation automobiles. In *Proceeding of MobiSys 2004, 1st ACM Workshop on Applications of Mobile Embedded Systems* , June 2004.

[37]: Svensoon, A. (2005). Executive Summary – The Safe Traffi c Project, February. Swaszek , P. & Willett , P. (1995). Parley as an approach to distributed detection . *IEEE Transactions on Aerospace and Electronic Systems* , **31** (1) , pp. 447 – 457 .

[38]: CRUISE, WP112, D112.2: Update on WSN applications, their requirements, applicationspecifi c WSN issues and evaluation metrics, December 2006, website: http://www. istcruise.eu.

[39]: A. Goldsmith and S. Wicker, "Design challenges for energy-constrained ad hoc wireless networks," *IEEE Wireless Communications Magazine*, vol. 9, pp. 8–27, Aug. 2002.

[40] : L. Yuan and G. Qu, "Energy-efficient Design of Distributed Sensor Networks," in *Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems*, M. Ilyas and I. Mahgoub, eds., Boca Raton, FL, pp. 38.1–38.19, CRC Press, 2004.

[41] :Jason Hill, Robert Szewczyk, Alec Woo, Seth Hollar, David Culler, and Kristofer Pister. System architecture directions for networked sensors. In *Proceedings of the Ninth International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS-IX),* pages 93–104, Cambridge, MA, USA, November 2000,ACM.

[42] :J.M. Kahn, R.H. Katz, and K.S.J. Pister. Next century challenges: mobile networking for Smart Dust. In *Proceedings of the fifth annual ACM/IEEE international conference on Mobile computing and networking,* pages 271–278, 1999.

[43] : R. Shah and J. Rabaey. Energy aware routing for low energy ad hoc sensor networks. In Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC'02), Orlando, Florida, USA, March 2002.

[44] : W.B. Heinzelman, J. Kulik, and H. Balakrishnan. Adaptive protocols for information dissemination in wireless sensor networks. In Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'99), Seattle, WA, USA, August 1999.

[45] : C. Intanagonwiwat, R. Govindan, and D. Estrin. Directed diffusion : A scalable and robust communication paradigm for sensor networks. In MOBICOM Boston MA USA, 2000.

[46] : F. Silva, J. Heidemann, R. Govindan, and D. Estrin. Directed diffusion. In USC/ISI Technical Report ISI-TR-2004-586, 2004.

[47] : D. Braginsky and D. Estrin. Rumor routing algorithm for sensor networks. In Proceedings of the First Workshop on Sensor Networks and Applications (WSNA), Atlanta, GA, USA, October 2002.

[48] : Y. Yao and J. Gehrke. The cougar approach to in-network query processing in sensor networks. In SIGMOD Record, September 2002.

[49] : N. Sadagopan, B. Krishnamachari, and A. Helmy. The acquire mechanism for efficient querying in sensor networks. In Proceedings of the First International Workshop on Sensor Network Protocol and Applications, Anchorage, AK, USA, May 2003.

[50] : K. Kalpakis, K. Dasgupta, and P. Namjoshi. Maximum lifetime data gathering and aggregation in wireless sensor networks. In Proceedings of IEEE International Conference on Networking (NETWORKS'02), Atlanta, GA, USA, August 2002.

[51] : K. Dasgupta, K. Kalpakis, and P. Namjoshi. An e_cient clustering-based heuristic for data gathering and aggregation in sensor networks. In Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC'03), New Orleans, LA, USA, March 2003.

[52] : F. Ye, A. Chen, S. Lu, and L. Zhang. A scalable solution to minimum cost forwarding in large scale sensor networks. In Proceedings of International Conference on Computer Communications and Networks (ICCCN), Dallas, TX, USA, October 2001.

[53] : K. Sohrabi, J. Gao, V. Ailawadhi, and G.J. Pottie. Protocols for selforganization of a wireless sensor network. In IEEE Personal Communications, Vol. 7, Issue 5, pp. 16-27, 2000.

[54] : T. He, J.A. Stankovic, C. Lu, and T. Abdelzaher. Speed : a stateless protocol for real-time communication in sensor networks. In Proceedings of International Conference on Distributed Computing Systems, Providence, RI, USA, May 2003.

[55] : V. Rodoplu and T.H. Ming. Minimum energy mobile wireless networks. In IEEE Journal of Selected Areas in Communications, Vol 17 (8) (1999) pp : 13331344, 1999.

[56] : L. Li and J.Y Halpern. Minimum energy mobile wireless networks revisited. In Proceedings of IEEE International Conference on Communications (ICC'01), Helsinki, Finland, June 2001.

[57] : Y. Xu, J. Heidemann, and D. Estrin. Geography-informed energy conservation for ad hoc routing. In Proceedings of the 7th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'01), Rome, Italy, July 2001.

[58] : Y. Yu, D. Estrin, and R. Govindan. Geographical and energyaware routing : a recursive data dissemination protocol for wireless sensor networks. In UCLA Computer Science Department Technical Report, UCLA-CSD TR-01-0023, May 2001.

[59] : B. Karp and H.T. Kung. Gpsr : greedy perimeter stateless routing for wireless sensor networks. In Proceedings of the 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'00), Boston, MA, USA, August 2000.

[60] : R. Kacimi, R. Dhaou, A.-L. Beylot, A. Delye de Mazieux, V. Gauthier, M. Marot, J. Vaudour, and M Becker. Etat de l'art sur les réseaux de capteurs sans fil. In Livrable Projet CAPTEURS, SP1, V1.2, IRIT-ENSEEIHT et INT, Rapport de recherche INT n_05001 RST, 2006.

[61] : J. Deng, R. Han, and S. Mishra. INSENS: intrusion-tolerant routing in wireless sensor networks. In *Technical Report CU-CS-939-02, Department of Computer Science, University of Colorado*, 2002.

[62] : B. Karp and H. T. Kung. GPSR: greedy perimeter stateless routing for wireless networks. In *Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 243–254. ACM Press, 2000.

[63] : P. Papadimitratos and Z. J. Haas. Secure routing for mobile ad hoc networks. In *Proceedings of the SCS Communication Networks and Distributed System Modeling and Simulation Conference (CNDS 2002)*, 2002.

[64] : S. Tanachaiwiwat, P. Dave, R. Bhindwale, and A. Helmy. Poster abstract secure locations: routing on trust and isolating compromised sensors in locationaware sensor networks. In *Proceedings of the 1st international conference on Embedded networked sensor systems*, pages 324–325. ACM Press, 2003.

[65] : D. Estrin, R. Govindan, J. S. Heidemann, and S. Kumar. Next century challenges: Scalable coordination in sensor networks. In *Mobile Computing and Networking*, pages 263–270, 1999.

[66] : L. Hu and D. Evans. Secure aggregation for wireless networks. In *SAINTW '03: Proceedings of the 2003 Symposium on Applications and the Internet Workshops (SAINT'03 Workshops)*, page 384. IEEE Computer Society, 2003.

[67] : S. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong. Tag: a tiny aggregation service for ad-hoc sensor networks. *SIGOPS Oper. Syst. Rev.*, 36(SI):131–146, 2002.

[68] : B. Przydatek, D. Song, and A. Perrig. Sia: Secure information aggregation in sensor networks, 2003.

[69] : N. Shrivastava, C. Buragohain, D. Agrawal, and S. Suri. Medians and beyond: new aggregation techniques for sensor networks. In *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 239–249. ACM Press, 2004.

[70] : F. Ye, H. Luo, S. Lu, and L. Zhang. Statistical en-route detection and filtering of injected fase data in sensor networks. In *IEEE INFOCOM 2004*, 2004.

[71] : A. R. Beresford and F. Stajano. Location Privacy in Pervasive Computing. *IEEE Pervasive Computing*, 2(1):46–55, 2003.

[72] : T. Kaya, G. Lin, G. Noubir, and A. Yilmaz. Secure multicast groups on ad hoc networks. In *Proceedings of the 1st ACM workshop on Security of Ad hoc and Sensor Networks (SASN '03)*, pages 94–102. ACM Press, 2003.

[73] : S. Rafaeli and D. Hutchison. A survey of key management for secure group communication. *ACM Comput. Surv.*, 35(3):309–329, 2003.

[74] : X. Fu, B. Graham, R. Bettati, and W. Zhao. On Countermeasures to Traffic Analysis Attack. In *Fourth IEEE SMC Information Assurance Workshop,* 2003.

[75]: Y. Guan, X. Fu, D. Xuan, P. U. Shenoy, R. Bettati, and W. Zhao. Netcamo: Camouflaging network traffic for QOS-guaranteed critical applications. *IEEE Trans. on Systems, Man, and Cybernetics Part a: Systems and Humans, Special Issue on Information Assurance,* pages 253–265, July 2001.

[76] : J. Raymond. Traffic Analysis: Protocols, attacks, design issues and open problems. In H. Federrath, editor, *Designing Privacy Enhancing Technologies: Proceedings of International Workshop on Design Issues in Anonymity and Unobservability,* volume 2009 of *LNCS,* pages 10–29,2001.

[77] : M. Franklin, Z. Galil, andM. Yung, "Eavesdropping games: a graph-theoretic
approach to privacy in distributed systems," *J. ACM*, vol. 47, no. 2, pp. 225– 243, 2000.

[78]: M. Abadi and J. Jürjens, "Formal eavesdropping and its computational interpretation," in *TACS '01: Proceedings of the 4th International Symposium on Theoretical Aspects of Computer Software*. London, UK: Springer- Verlag, 2001, pp. 82–94.

[79]: K. D. Murray, *Security Scrapbook Espionage* and *Privacy News of the Week*. [Online]. Available: http : //www.spybusters.com/SS0210.html.

[80] : C. Karlof and D. Wagner. Secure Routing in Sensor Networks: Attacks and Countermeasures. In *Proc. of First IEEE International Workshop on Sensor Network Protocols and Applications,* May 2003 & " *Elsevier's AdHoc Networks Journal, Special Issue on Sensor Network Applications and Protocols*, vol. 1, no. 2–3, pp. 293–315, September 2003.

[81]: V. Gupta, S. Krishnamurthy, and M. Faloutsos, "Denial of service attacks at the mac layer in wireless ad hoc networks." [Online]. Available: http : //www.cs.ucr.edu/ krish/milcomvik.pdf

[82]: I. A. Jean-Pierre, "Denial of service resilience in ad hoc networks." [Online]. Available: http://lcawww.epfl.ch/Publications/aad/aadHK04.pdf

[83]: A. D. Wood and J. A. Stankovic, "Denial of service in sensor networks," *Computer*, vol. 35, no. 10, pp. 54–62, 2002.

[84]: P. Michiardi and R. Molva, "Prevention of denial of service attacks and selfishness in mobile ad hoc networks," in *Institut Eurecom Research Report RR-02-063*, 2002.

[85]: A. A. Cardenas, S. Radosavac, and J. S. Baras, "Detection and prevention of mac layer misbehavior in ad hoc networks," in *Proceedings of the 2nd ACM workshop on security of ad hoc and sensor networks*, 2004.

[86]: E. D. Cardenas, "Mac spoofing–an introduction," 2003. [Online]. Available: http : //www.giac.org/practical/GSEC/EdgarC ardenasGSEC.pdf.

[87]: J. Newsome, E. Shi, D. Song, and A. Perrig, "The sybil attack in sensor networks: analysis & defenses," in *IPSN'04: Proceedings of the third international symposium on Information processing in sensor networks*. ACM Press, 2004, pp. 259–268.

[88] : John R. Douceur. The sybil attack. In *IPTPS '01 : Revised Papers from the First International Workshop on Peer-to-Peer Systems*, pages 251–260. Springer-Verlag, 2002.
[89]: C. R. Murthy and B.S.Manoj, "Transport layer and security protocols for ad hoc wireless networks," in *Ad Hoc Wireless Networks - Architectures and Protocols*, 2004.
[90]: Yacine Challal , "sécurité dans les réseaux de capteurs", support de cours, UTC.
[91] : A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar. SPINS: Security Protocols for Sensor Networks. *Wireless Networks Journal (WINET),* 8(5):521–534, September 2002.

[92] : E. SHI and A. PERRIG, "Designing secure sensor networks," *IEEE Wireless Communications*, vol. 11, no. 6, pp. 38–43, 2004.
[93]: J. Elson and D. Estrin, "Time synchronization for wireless sensor networks," in *IPDPS '01: Proceedings of the 15th International Parallel & Distributed Processing Symposium*. Washington, DC, USA: IEEE Computer Society, 2001, p. 186.
[94]: G. Khanna, A. Masood, and C. N. Rotaru, "Synchronization attacks against 802.11," in *Networks and Distributed Systems Symposium (NDSS) Workshop*, 2005.
[95] : A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar. SPINS: Security Protocols for Sensor Networks. *Wireless Networks Journal (WINET),* 8(5):521–534, September 2002.
[96]: C. Karlof, N. Sastry, and D. Wanger. "TinySec : A Link Layer Security Architecture for Wireless Sensor Networks". Proceeding 2nd ACM Conference on Embdded Networked Sensor Systems (SenSys 2004), Baltimore, Maryland, Etats-Unis, pp. 162-175, November 2004.
[97] : D. W. Carman, P. S. Krus, and B. J. Matt. Constraints and approaches for distributed sensor network security. Technical Report 00-010, NAI Labs, Network Associates, Inc., Glenwood, MD, 2000.
[98]: Q. Yi and N. Moayeri, "Design of secure and application-oriented VANETs," in *Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE*, 2008, pp. 2794-2799.
[99]: K. Plossl, T. Nowey, and C. Mletzko, "Towards a security architecture for vehicular ad hoc networks," in *Availability, Reliability and Security, 2006. ARES 2006. The First International Conference on*, 2006, p. 8 pp.
[100]: M. Raya, P. Papadimitratos, and J. P. Hubaux, "Securing vehicular communications," *Wireless Communications, IEEE,* vol. 13, pp. 8-15, 2006.
[101]: J. Hwang and Y. Kim. Revisiting random key pre-distribution schemes for wireless sensor networks. In *Proceedings of the 2nd ACM workshop on Security of Ad hoc and Sensor Networks (SASN '04)*, pages 43–52, New York, NY, USA, 2004. ACM Press.
[102]: H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in *SP '03: Proceedings of the 2003 IEEE Symposium on Security and Privacy*. Washington, DC, USA: IEEE Computer Society, 2003, p. 197.
[103]: L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in *CCS '02: Proceedings of the 9th ACM conference on Computer and communications security*. New York, NY, USA: ACMPress, 2002, pp. 41–47.
[104]: D. Liu and P. Ning, "Establishing pairwise keys in distributed sensor networks," in *CCS '03: Proceedings of the 10th ACM conference on Computer and communications security*. New York, NY, USA: ACM Press, 2003, pp. 52–61.
[105]: F. Armknecht, A. Festag, D. Westhoff, and K. Zeng, "Cross-layer privacy enhancement and non-repudiation in vehicular communication," *Communication in Distributed Systems (KiVS), 2007 ITG-GI Conference,* pp. 1-12, 2007.
[106]: B. Parno and A. Perrig, "Challenges in securing vehicular networks," in *Workshop on Hot Topics in Networks (HotNets-IV)*, 2005.

[107]: Menezes, A., P. Van Oorschost et S. Vanstone. Handbook of applied cryptography. CRC Press, 1996.
[108]: Naiara Escudero Sanchez naiara.escudero@googlemail.com , University of Paderborn, "The Rabin Cryptosystem.
[109]: Kao, M. Sécurité des réseaux: un guide pratique pour créer une infrastructure de réseau sécurisée. 2000.
[110]: NIST National Institute of Standards and Technology. «Secure hash standard
(SHS).» FIPS publication (1995): 180-181.
[111]: Michel Abdalla Thomas Claveirole, Marcelo Dias De Amorim and Yannis Viniotis. Résistance contre les attaques par capture dans les réseaux de capteurs. In *JDIR*, 2007
[112]: Jing Deng, Richard Han, and Shivakant Mishra. Countermeasures against traffic analysis attacks in wireless sensor networks. In *SECURECOMM '05: Proceedings of the First International Conference on Security and Privacy for Emerging Areas in Communications Networks*, pages 113–126, Washington, DC, USA, 2005. IEEE Computer Society.
[113]: Étude technique réalisée par CGI, " Étude technique Cryptographie à clé publique et signature numérique Principes de fonctionnement ". Septembre 2002.
[114]: Z. Yan, P. Zhang, and T. Virtanen. Trust evaluation based security solution in ad hoc networks. In *NordSec 2003, Proceedings of the Seventh Nordic Workshop on Secure IT Systems*, 2003.
[115]: H. Zhu, F. Bao, R. H. Deng, and K. Kim. Computing of trust in wireless networks. In *Proceedings of 60th IEEE Vehicular Technology Conference, Los Angles, California*, September 2004.
[116]: Pissinou Niki and Crosby Garth V. Cluster-based reputation and trust for wireless sensor networks. In *Consumer Communications and Networking Conference, 2007. CCNC 2007. 4th IEEE*, pages 604–608, January 2007.
[117]: Kui Ren, Tieyan Li, Zhiguo Wan, Feng Bao, Robert H. Deng, and Kwangjo Kim. Highly reliable trust establishment scheme in ad hoc networks. *Computer Networks*, 45(6):687–699, 2004.
[118]: Saurabh Ganeriwal and Mani B. Srivastava. Reputation-based framework for high integrity sensor networks. In Sanjeev Setia and Vipin Swarup, editors, *SASN*, pages 66–77. ACM, 2004.

[119]: Vladimir Oleshchuk and Vladimir Zadorozhny. Trust-aware query processing in data intensive sensor networks. In *SENSORCOMM '07: Proceedings of the 2007 International Conference on Sensor Technologies and Applications*, pages 176–180, Washington, DC, USA, 2007. IEEE Computer Society.
[120]: Zhengqiang Liang and Weisong Shi. Pet: A personalized trust model with reputation and risk evaluation for p2p resource sharing. In *HICSS*. IEEE Computer Society, 2005.
[121]: David MARTINS , Hervé GUYENNET. Etat de l'art Sécurité dans les réseaux de capteurs sans fil. *SAR-SSI 2008*.
[122]: Jochen Mundinger and Jean-Yves Le Boudec. Analysis of a Reputation System for Mobile Ad-Hoc Networks with Liars. In *The 3rd International Symposium on Modeling and Optimization in*, 2005.
[123]: Samuel Galice, thèse, « MODÈLE DYNAMIQUE DE SÉCURITÉ POUR RÉSEAUX SPONTANÉS », Institut National des Sciences Appliquées de Lyon, 29 Octobre 2007.
[124]: S. Buchegger, J. Mundinger, and J.-Y. Le Boudec, "Reputation systems for self organized networks," *Technology and Society Magazine, IEEE,* vol. 27, pp. 41-47, 2008.
[125]: R. Molva and P. Michiardi, "Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks," *Institute EurecomResearch Report RR-02-062,* 2001.
[126]: A. Srinivasan, J. Teitelbaum, and J. Wu, "DRBTS: distributed reputation-based beacon trust system," in *Dependable, Autonomic and Secure Computing, 2nd IEEE International Symposium on*, 2006, pp. 277-283.
[127]: S. Buchegger, J. Mundinger, and J.-Y. Le Boudec, "Reputation systems for self organized networks," *Technology and Society Magazine, IEEE,* vol. 27, pp. 41-47, 2008.

[128] : Y.-C. Hu, A. Perrig, and D. B. Johnson, “Ariadne: A secure ondemand routing protocol for ad hoc networks,” in *Proceedings of the Eighth Annual International Conference on Mobile Computing and Networking* (MobiCom 2002), Sept. 2002, to appear. [Online]. Available: citeseer.ist.psu.edu/article/hu02ariadne.html.
[129]: A. Srinivasan, J. Teitelbaum, J. Wu, M. Cardei, and H. Liang, "Reputation-and-Trust-Based Systems for Ad Hoc Networks," *Algorithms and protocols for wireless and mobile ad hoc networks,* p. 375, 2009.
[130] : X. Fu, B. Graham, R. Bettati, and W. Zhao. On Countermeasures to Traffic Analysis Attack. In *Fourth IEEE SMC Information Assurance Workshop,* 2003.
[131]: A. Abdul-Rahman and S. Hailes, "Supporting trust in virtual communities," in *System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference on*, 2000, p. 9 pp. vol. 1.
[132]: Philip R. Zimmermann. *The official PGP user’s guide*. MIT Press, 1995

[133]: Alfarez Abdul-Rahman and Stephen Hailes. A distributed trust model. In *NSPW ’97 : Proceedings of the 1997 workshop on New security paradigms*, pages 48–60. ACM, 1997.
[134]: Amir Herzberg, Yosi Mass, Joris Michaeli, Yiftach Ravid, and Dalit Naor. Access control meets public key infrastructure, or : Assigning roles to strangers. In *SP ’00 : Proceedings of the 2000 IEEE Symposium on Security and Privacy*, page 2. IEEE Computer Society, 2000.
[135]: D. Fraga, Z.Bankovi´c, J. M. Moya *Department Of Electrical Engineering ETSI Telecomunicaci´on Universidad Polit´ecnica de Madrid Madrid, Spain, “*A Taxonomy of Trust and Reputation System Attacks”, 2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications.
[136]: Lik Mui, Mojdeh Mohtashemi, Cheewee Ang, Peter Szolovits, and Ari Halberstadt. Ratings in distributed systems : A bayesian approach, 2001.
[137]: L. Mui, M. Mohtashemi, and A. Halberstadt. A computational model of trust and reputation. pages 2431 – 2439. IEEE Computer Society, jan. 2002.
[138]: Andrew Whitby, Audun Jøsang, and Jadwiga Indulska. Filtering out unfair ratings in bayesian reputation systems. In *Workshop on Trust in Agent Societies*, number 2, pages 106–117. Citeseer, 2004.
[139]: Buchegger, Sonja and Le Boudec, Jean-Yves. Nodes bearing grudges: towards routing security, fairness, and robustness in mobile ad hoc networks. *10th Euromicro Workshop on Parallel, Distributed and Network-Based Processing: Proceedings.* Canary Islands : Inst Elect & Electronic Engineers, 2002, pp. 403-410.
[140]: V. S. Yadav, S. Misra, and M. Afaque, "Security of self-organizing networks," ed, 2010.
[141]: J. T. Issac, S.Zeadally, and J. S. Camara, "Security attacks and solutions for vehicular ad hoc networks," *IET Communication,* vol. 4, pp. 894-903, 2010-04-30 2010.
[142]: S. Marti and H. Garcia-Molina, "Taxonomy of trust: Categorizing P2P reputation systems," *Computer Networks,* vol. 50, pp. 472-484, 2006.
[143]: A. Ravoaja, "Mécanismes et architectures P2P robustes et incitatifs pour la réputation," PhD, Rennes-1, 2008.
[144]: P. L. Mazenc, "Système de réputation préservant la vie privée," in *3ième édition Atelier Protection de la vie privée*, 2012.
[145]: G. Urdaneta, G. Pierre, and M. V. Steen, "A survey of DHT security techniques," *ACM Computing Surveys (CSUR),* vol. 43, p. 8, 2011
[146]: T. Reidemeister, K. Böhm, E. Buchmann, and P. A. Ward, "Man-in-the-middle attacks in distributed hash-tables," *IEEE Journal on Selected Areas in Communication,* 2006.

[147]: I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for internet applications," *ACM SIGCOMM Computer Communication Review,* vol. 31, pp. 149-160, 2001.

[148]: A. Fiat, J. Saia, and M. Young, "Making chord robust to byzantine attacks," *Algorithms–ESA 2005,* pp. 803-814, 2005.

[149]: D. Fraga, Z.Bankovi´c, J. M. Moya *Department Of Electrical Engineering ETSI Telecomunicaci´on Universidad Polit´ecnica de Madrid Madrid, Spain, "*A Taxonomy of Trust and Reputation System Attacks", 2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications.

[150]: Sergio Marti, T. J. Giuli, Kevin Lai, and Mary Baker. Mitigating routing misbehavior in mobile ad hoc networks. In *MobiCom '00 : Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 255– 265. ACM, 2000.

[151]: Johnson David B, Maltz David A, and Broch Josh. Dsr : The dynamic source routing protocol for multi-hop wireless ad hoc networks. In *In Ad Hoc Networking, edited by Charles E. Perkins, Chapter 5*, pages 139–172. Addison- Wesley, 2001.

[152]: Sonja Buchegger and Jean-Yves Le Boudec. Performance analysis of the confidant protocol. In *MobiHoc '02 : Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing*, pages 226–236. ACM, 2002.

[153]: Pietro Michiardi and Refik Molva. Core : a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks. In *Proceedings of the IFIP TC6/TC11 Sixth Joint Working Conference on Communications and Multimedia Security*, pages 107–121. Kluwer, B.V., 2002.

[154]: Amir R. Khakpour, Maryline Laurent-Maknavicius, and Hakima Chaouchi, WATCHMAN: An Overlay Distributed AAA Architecture for Mobile Ad hoc Networks. ARES: pages 144-152. IEEE Computer Society 2008.

[155]: M. Wang, L. Lamont, P Mason, and M. Gorlatova. An effective intrusion detection approach for olsr manet protocol. In *Proceedings of the First Workshop on Secure Network Protocols (NPSec), Boston, Massachusetts, États-Unis.*, jul 2005.

[156]: F. Cuppens, N. Cuppens-Boulahia, S. Nuon, and T. Ramard. Property based intrusion detection to secure olsr. In *Proceedings of the 3rd IEEE International Conference on Wireless and Mobile Communications (ICWMC'07)*, pages 52–60. IEEE Computer Society, 2007.

[157]: Z. Yao, D. Kim, and Y. Doh, "PLUS: Parameterized and localized trust management scheme for sensor networks security," in *Proc. of the 3rd IEEE Int. Conf. on Mobile Adhoc and Sensor Systems*, Vancouver, Canada, Oct. 2006, pp. 437–446.

[158]: S. Ganeriwal and M. B. Srivastava, "Reputation-based framework for high integrity sensor networks," in *Proc. of ACM Security for Ad-hoc and Sensor Networks*, Oct. 2004.

[159]: S. Ganeriwal, L. K. Balzano, and M. B. Srivastava, "Reputation-based framework for high integrity sensor networks," *ACM Trans. Sen. Netw.*, vol. 4, no. 3, pp. 1–37, 2008.

[160]:Riaz Ahmed Shaikh, Hassan Jameel, Brian J. d'Auriol, Member, IEEE Computer Society, Heejo Lee, Member, IEEE, Sungyoung Lee, Member, IEEE, and Young-Jae Song, "Group-Based Trust Management Scheme for Clustered Wireless Sensor Networks" IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 20, NO. 11, NOVEMBER 2009.

[161]: Riaz Ahmed Shaikh, Young-Koo Lee, Sungyoung Lee Dept. of Comp. Eng., Kyung Hee University, Global Campus, Korea, "An Extended Energy Consumption Analysis of Reputation-based Trust Management Schemes of Wireless Sensor Networks », JOURNAL OF NETWORKS, VOL. 5, NO. 3, MARCH 2010.

[162]: Riaz Ahmed Shaikh , Kyung Hee, Young-Koo, Kyun, Sungyoung Lee,Kyung Hee University, Global Campus, Korea, "Energy Consumption Analysis of Reputation-based Trust Management Schemes of Wireless Sensor Networks", C.2 [Computer Communication Networks]: Network Protocols, january 2009.

[163]: R. A. Shaikh, S. Lee, M. A. U. Khan, and Y. J. Song. LSec: Lightweight security protocol for distributed wireless sensor network. In *11th IFIP Int. Conf. On Personal Wireless Comm., LNCS 4217*, pages 367–377, Albacete, Spain, Sept. 2006.

[164]: X. Anita,1 J. Martin Leo Manickam,2 and M. A. Bhagyaveni1, 1Department of ECE, Anna University, Chennai 600025, India 2 Department of ECE, St. Joseph's College of Engineering, Chennai 600119, India, "Two-Way Acknowledgment-Based Trust Framework for Wireless Sensor Networks", International Journal of Distributed Sensor Networks Volume 2013 (2013), Article ID 952905, 14 pages, 4 April 2013.

[165]: Junqi Duan, Dong Yang, Haoqing Zhu, Sidong Zhang, and Jing Zhao, National Engineering Laboratory for Next Generation Internet Interconnection Devices, School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing 100044, China, "TSRF: A Trust-Aware Secure Routing Framework in Wireless Sensor Networks", International Journal of Distributed Sensor Networks Volume 2014 (2014), Article ID 209436, 14 pages.

[166]: Farruh Ishmanov, Sung Won Kim * and Seung Yeob Nam, Department of Information and Communication Engineering, Yeungnam University, 214-1 Dae-dong, Gyeongsan-si, Kyongsan 712-749, Gyeongsangbuk-do, Korea, "A Secure Trust Establishment Scheme for Wireless Sensor Networks », *Sensors* 2014, *14*, 1877-1897; doi:10.3390/s140101877.

[167]: Li, X.; Zhou, F.; Du, J. LDTS: A Lightweight and Dependable Trust System for Clustered Wireless Sensor Networks. *IEEE Trans. Inf. Forensics Security* 2013, *8*, 924–935.

[168]: Dodonov, Y.S.; Dodonova, Y.A. Robust measures of central tendency: Weighting as a possible alternative to trimming in responsetime data analysis. *Psikhologicheskie Issledovaniya* **2011**, *19*, 1–11.

[169]: Huber, P.J. *Robust Statistics*; John Wiley & Sons Ltd.: New York, NY, USA, 1981.

[170]: Li, X.; Zhou, F.; Du, J. LDTS: A Lightweight and Dependable Trust System for Clustered Wireless Sensor Networks. *IEEE*.

[171] : H. O. Tan and I. Korpeoglu, "Power efficient data gathering and aggregation in wireless sensor networks," *ACM SIGMOD Record*, vol. 32, no. 4, pp. 66–71, Dec. 2003.