République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université de $Batna - 2$
Faculté des Mathématiques et de l'informatique
Département d'informatique

Thèse

En vue de l'obtention du diplôme de

Doctorat en Sciences en Informatique

Un modèle de confiance pour la sécurité des communications dans les réseaux mobiles Ad hoc

Présentée Par :

Abdesselem Beghriche

Devant le jury composé de :

Président :	*SEGHIR Rachid*	*MCA. Université de Batna 2.*
Rapporteur :	*BILAMI Azeddine*	*Prof. Université de Batna 2.*
Examinateur :	*KHOLLADI Mohamed-Khireddine*	*Prof. Université d'El Oued.*
Examinateur :	*BABAHENINI Mohamed-Chaouki*	*Prof. Université de Biskra.*
Examinateur :	*CHERIF Fodil*	*Prof. Université de Biskra.*
Examinateur :	*AOUAG Sofiane*	*MCA. Université de Batna 2.*

Remerciements

Comme c'est difficile de trouver les mots exacts qui expriment mes profonds sentiments de gratitude envers les personnes qui étaient à mes côtés pour m'aider à surmonter tous les obstacles et à mener à bien ce travail. Ce n'est que justice de leur écrire cette page.

Ainsi, je tiens à exprimer ma très vive reconnaissance à l'égard de mon encadreur Azeddine Bilami, Professeur au département d'informatique de l'université de Batna 2, directeur du laboratoire LaSTIC, pour ses conseils, ses encouragements, sa disponibilité et son soutien tout au long de ces années. Les mots me manquent pour le remercier de m'avoir motivé et de m'avoir laissé la liberté nécessaire à l'accomplissement de mes travaux tout en gardant un œil critique et avisé.

Je remercie également Dr. Rachid Seghir, Maitre de conférences à l'université de Batna 2 qui m'a fait l'honneur de présider le jury de thèse. Mes remerciements les plus respectueux vont aussi à :

- Pr. Med Khireddine Kholladi, Professeur à l'université d'El oued ;
- Pr. Med Chaouki Babahenini, Professeur à l'université de Biskra ;
- Pr. Fodil Cherif, Professeur à l'université de Biskra ;
- Dr. Sofiane Aouag, Maître de Conférences à l'université Batna 2.

qui m'ont fait l'honneur d'accepter de participer à ce jury de thèse. Je leurs suis très reconnaissant pour l'intérêt qu'ils ont porté à mes travaux.

Je tiens à remercier :

- Pr. Abderrahmane Boumezbeur, Professeur à l'université de Tébessa ;
- Dr. Abdelhakim Ben machiche, Maître de Conférences à l'université de Biskra ;
- Dr. Djallel-Eddine Boubiche, Maître de Conférences à l'université de Batna 2 ;

pour leurs conseils et surtout leurs encouragements.

Je dédie ce travail à ma femme de m'avoir supporté (dans tous les sens du terme) pendant les deux dernières années.

Enfin, Merci à tous ceux qui m'ont aidé en m'accordant leurs temps, leurs encouragements et leurs savoirs.

"À *la mémoire de mon cher père*"

رحمت الله عليه

Résumé

Le sujet de cette thèse se focalise sur le problème de l'évaluation et de la gestion de la confiance dans les réseaux mobiles Ad hoc (MANET: Mobile Ad hoc NETwork), où les nœuds accumulent le rôle de routeur, de serveur et de client, les obligeant à coopérer pour un bon fonctionnement du réseau. L'absence d'une gestion centrale des fonctionnalités du réseau rend ces réseaux beaucoup plus vulnérables aux attaques que les réseaux sans fil (WLAN) et filaires (LAN). Plusieurs nouveaux protocoles de sécurité ont été proposés, parce que les solutions conventionnelles ne sont pas adaptées pour de tels réseaux (environnement dynamique). Ils ne prennent pas la contrainte des ressources en considération car non seulement l'environnement est dynamique, mais les ressources sont aussi limitées (la mémoire, la capacité de calcul et surtout l'énergie), ce qui complique davantage la problématique, car on sait bien que les solutions de sécurité sont gourmandes en termes de ressources.

La plupart des protocoles et des applications pour réseaux Ad hoc considèrent l'existence d'une parfaite coopération entre tous les nœuds du réseau. Il est supposé que tous les nœuds se comportent selon les spécifications des applications et des protocoles précédemment déterminés pour le réseau. Néanmoins, cette condition peut être fausse, à cause de contraintes de ressources ou de comportements malveillants. Par la suite, les nœuds peuvent ne pas se comporter comme prévu entrainant un mauvais fonctionnement du réseau. Prétendre que ces nœuds se comportent correctement peut entrainer des problèmes, tels qu'une faible efficacité du réseau, une consommation élevée de ressources et une vulnérabilité importante aux attaques. Par conséquent, un mécanisme permettant à un nœud d'avoir confiance en d'autres nœuds est nécessaire.

L'objectif principal de la thèse consiste à définir et proposer un nouveau modèle de gestion de la confiance, où les nœuds d'un réseau Ad-hoc établissent un rapport de confiance basé sur des expériences et des recommandations préalables. Le but est de rendre les nœuds du réseau capables de recueillir des informations pour raisonner apprendre et prendre leur propre décision. La solution envisagée est de faire reposer la prise de décision d'un échange sur la base de la confiance, sachant que chaque nœud ne pourra se protéger d'éventuels voisins malicieux qu'en faisant appel aux informations locales dont il dispose.

Notre modèle de gestion de la confiance a donc pour objectif d'intégrer des mécanismes contrant les attaques qui pourraient exister, en forçant la coopération entre les nœuds, et détectant les comportements défaillants. Le modèle proposé repose sur la combinaison de deux mécanismes principaux, dont le premier est basé sur la théorie des ensembles flous, et le deuxième applique la méthode d'analyse relationnelle grise "G. R. A" (Grey Relational Analysis) afin de calculer un niveau de confiance pour chaque nœud dans le réseau. Ces niveaux de confiance sont ensuite utilisés dans le processus de prise de décision de routage.

La performance de ces mécanismes est évaluée avec la prise en compte de plusieurs aspects primordiaux pour les réseaux mobiles Ad hoc, telles que la qualité d'interaction du réseau, l'efficacité du modèle de confiance, l'atténuation de nœuds malveillants et les améliorations de la sécurité du système.

Mots clés : Réseaux mobiles Ad hoc, MANETs, sécurité, protocoles de routage, confiance, modèle de gestion de la confiance, réputation, recommandation, logique floue, méthode d'analyse relationnelle grise, méthode de classification grise.

ملخص

يرتكز موضوع هذه الأطروحة حول مشكلة تقييم وإدارة الثقة في شبكات المحمول اللاسلكية المخصصة (أد-هوك)، حيث تلعب العقد دورا مهما في التشغيل المناسب للشبكة. نقوم بدراسة أمن هذه الشبكات نظرا لعدم وجود إدارة مركزية، الشيء الذي يجعل من هذه الأخيرة أكثر عرضة للهجوم مقارنة بالشبكات الأخرى (السلكية واللاسلكية). للأسف، بروتوكولات الأمن والحماية التقليدية ليست مصممة لمثل هذا النوع من الشبكات (محيط ديناميكي متحرك)، أضف إلى ذلك محدودية الطاقة والذاكرة وضعف القدرة على الحساب وذلك مما يزيد من تعقيد مشكلة الأمن في هذه الشبكات.

معظم بروتوكولات وتطبيقات الشبكات المخصصة تفترض وجود تعاون تام بين جميع عقد الشبكة. لذلك من المفترض أن كافة العقد تتصرف وفقا للمواصفات والتطبيقات التي سبق تحديدها لبروتوكولات الشبكة. غير أن هذا الشرط يمكن أن يكون غير صحيح، بسبب القيود المفروضة على موارد الشبكة أو السلوكيات الخبيثة. العقد قد لا تتصرف كما هو متوقع، مما قد يسبب خلل في الشبكة، والإدعاء بأن هذه العقد تتصرف بشكل صحيح يمكن أن يؤدي أيضا إلى مشاكل عديدة مثل أن تكون كفاءة الشبكة منخفضة، إستهلاك كميات كبيرة من الموارد أو ضعف كبير عند صد الهجمات.

الهدف الرئيسي من هذه الرسالة هو إقتراح وتصميم نموذج جديد لإدارة الثقة في شبكات المحمول اللاسلكية المخصصة، حيث عقد الشبكة تقوم بإنشاء علاقات من الثقة مبنية على أساس من التجارب والتوصيات الأولية. والهدف هو جعل هذه العقد قادرة على جمع المعلومات اللازمة لإتخاذ القرارات المناسبة في عمليات التوجيه. الحل المقترح إذن هو جعل إتخاذ قرار عمليات التوجيه مبني على أساس الثقة، علما أن كل عقدة لا يمكن أن تحمي نفسها من الهجمات المحتملة الخبيثة إلا بإستخدام المعلومات المحلية المتاحة لها.

لذلك، نموذج إدارة الثقة عندنا يهدف إلى دمج آليات مواجهة وصد الهجمات في حال وجودها، وذلك من خلال تشجيع التعاون بين العقد والكشف عن السلوك الخاطئ. ويستند النموذج المقترح على الجمع بين آليتين رئيسيتين، حيث تستند الألية الأولى على طريقة ومنهج التحليل العلائقي الرمادي، في حين تقوم الآلية الثانية على تطبيق نظرية المنطق الضبابي وذلك من أجل حساب مستوى الثقة لكل عقدة في الشبكة لإستخدامها لاحقا في إتخاذ قرار عمليات التوجيه.

يتم تقييم أداء هذه الآليات من خلال مراعاة عدة جوانب هامة لشبكات المحمول المخصصة، مثل جودة التفاعل في الشبكة، فعالية نموذج الثقة، التخفيف من العقد الخبيثة والتحسينات التي تشمل بشكل عام أمن النظام.

كلمات البحث الرئيسية:

شبكات المحمول المخصصة، أد-هوك، أمن الشبكات، بروتوكولات التوجيه، الثقة، نموذج إدارة الثقة، السمعة، التوصية، المنطق الضبابي، طريقة التحليل العلائقي الرمادي، طريقة التجميع الرمادية.

Abstract

The subject of this thesis focuses on the problem of trust evaluation and management in mobile ad hoc networks (MANET: Mobile Ad hoc NETwork), in which nodes accumulate the role of a router, server, and client compelling them to cooperate for the correct operation of the network. The lack of any central management of the network functions make MANETs more vulnerable to attacks than wireless (WLANs) and wired networks (LANs). Several new security protocols have been proposed and developed, because traditional solutions are not designed to adapt MANETs characteristics. They do not take into account the resource limits, while the environment is dynamic and the resources are limited (memory, storage, computation power and energy), and this complicates the problem, because, as we know, security solutions require a high amount of resources.

Most protocols and applications for Ad hoc networks considers the perfect cooperation among all nodes. It is assumed that all nodes behave according to the application and protocol specifications previously defined for the network. Nevertheless, this assumption may be false, due to resource restrictions or malicious behaviour. Consequently, the nodes may not behave as expected, causing the network to not work properly. The assumption that nodes behave correctly can lead to unforeseen pitfalls, such as low network efficiency, high resource consumption, and high vulnerability to attacks. Therefore, a mechanism that allows a node to infer the trustworthiness of the other nodes is necessary.

The main aim of this thesis is to define and propose a new trust management model. This model builds a trust relationship among the nodes of an Ad hoc network based on previous experiences and neighbour's recommendations. The goal is to make nodes capable of gathering information to reason and make their own decisions. The proposed solution is to make the decision-making process of an exchange based on trust, knowing that each node will not be able to protect itself against potential malicious neighbours, except using local information available.

Therefore, our trust management model aims to integrate mechanisms against attacks that might exist, by forcing cooperation between nodes and detecting faulty behaviour. The proposed model is based on a combination of two main mechanisms, the first of which is based on the method of Grey Relational Analysis (G.R.A) and the second applies the fuzzy set theory, in order to calculate a trust level for each node in the network. These trust levels are then used in the routing decision-making process.

In order to prove the applicability of the proposed solution, extensive experiments were conducted to evaluate the efficiency of our model, aiming at improving the network interaction quality, malicious node mitigation and enhancements of the system's security.

Keywords: Mobile Ad-hoc networks, MANET, security, routing protocol, misbehaviour, trust model, trust management, fuzzy set, grey relational analysis, grey clustering method.

Table des matières

Chapitre 3 : ***La sécurité dans les réseaux mobiles Ad hoc***

Partie 2 : Contributions

Chapitre 4 : *Contexte de notre approche*

Chapitre 5 : ***Un nouveau modèle de gestion de la confiance***

Chapitre 6 : ***Le protocole de confiance proposé***

Chapitre 7 : ***Évaluation des performances à travers la simulation***

Liste des figures

Liste des tableaux

Glossaire des acronymes

- **AdvSig:** Advanced Signature.
- **AEC:** Average Energy Consume.
- **ALOHA:** Areal Locations of Hazardous Atmospheres.
- **AODV:** Ad hoc On demand Distance Vector.
- **ARAN:** A secure Routing protocol for Ad hoc Network.
- **BIND:** Berkeley Internet Name Domain.
- **BGP:** Border Gateway Protocol.
- **BRP:** Bordercast Resolution Protocol.
- **CBR:** Constant Bit Rate.
- **Confidant:** Cooperation of nodes-fairness in dynamic Ad hoc networks.
- **CORE:** a COllaborative REputation mechanism to enforce node cooperation in mobile Ad hoc networks.
- **CSMA/CA:** Carrier Sense Multiple Access with Collision Avoidance.
- **DAC:** Discretionary Access Control.
- **DARPA:** Defense Advanced Research Agency.
- **DBLAR:** Distance-Based Location-Aided Routing.
- **DIFS:** Distributed Inter Frame Space.
- **DiffServ:** Differentiated Services.
- **DSDV:** Destination-Sequenced Distance Vector.
- **DSR:** Dynamic Source Routing.
- **EFSM:** Extented Finite State Machine.
- **ESP:** Encapsulating Security Payload.
- **FTP:** File Transfer Protocol.
- **GPRS:** General Packet Radio Service.
- **GRA:** Grey Relational Analysis theory.
- **HMAC:** keyed-Hash Message Authentication Code.
- **HTTP:** HyperText Transfer Protocol.
- **IARP:** Intrazone Routing Protocol.
- **IDM:** Intrusion Detection mechanism.
- **IEEE:** Institute of Electrical and Electronics Engineers.
- **IERP:** Interzone Routing Protocol.
- **IETF:** Internet Engineering Task Force.
- **IIS:** Internet Information Services.
- **IntServ:** Integrated Services.

- **IP:** Internet Protocol.
- **IPsec:** Internet Protocol Security.
- **IPv4:** Internet Protocol version 4.
- **MAC:** Message Authentication Code.
- **MANET:** Mobile Ad-hoc Network.
- **NHDP:** Neighborhood Discovery Protocol.
- **NS-2:** Network Simulator version 2.
- **OLSR:** Optimized Link State Routing Protocol.
- **OPNET:** Optimum Network Performance.
- **OSI:** Open Systems Interconnection.
- **PAN:** Wireless Personal Area Network.
- **PCF:** Point Coordination Function) et le mode DCF (Distributed Coordination Function.
- **PDR:** Packet Delivery Ratio.
- **PGP:** Pretty Good Privacy.
- **PRNet:** Packet Radio NETwork.
- **QoS:** Quality of Service.
- **RAL:** Route Acquisition Latency.
- **RDP:** Route Discover Packet.
- **REP:** REply Packet.
- **RFC:** Requests For Comments.
- **RREQ:** Route REQuest.
- **RREP:** Route REPly.
- **RSVP:** Resource ReSerVation Protocol.
- **RTS/CTS:** Request-To-Send/Clear-To-Send.
- **S-AODV:** Secure Ad hoc On demand Distance Vector.
- **SAR:** Security-aware Ad hoc Routing protocol.
- **SEAD:** Secure Efficient distance vector Routing for mobile Ad hoc networks.
- **SHA-1:** Secure Hash Algorithm.
- **SI :** Systèmes d'informations.
- **SLSP:** Secure Link State Protocol.
- **SMTP**: Simple Mail Transfer Protocol.
- **SRP:** Secure Routing Protocol.
- **SUCV:** Statistically Unique Cryptographically Verifiable identifiers and addresses.
- **T-AODV:** A Trusted AODV Routing Protocol for Mobile Ad Hoc Networks.
- **TBM:** Trust-Based Mechanism.
- **TBRPF:** Topology Dissemination Based on Reverse Path Forwarding.
- **TCL**: Tool Command Language.

- **TCP:** Transmission Control Protocol.
- **TESLA:** Timed Efficient Stream Loss-tolerant Authentication.
- **TFP :** Taux de Faux-Positifs.
- **THF :** Table d'Historique Favorite
- **TTL:** Time To Live.
- **UDP:** User Datagram Protocol.
- **VPN:** Virtual Private Network.
- **Wi-Fi:** Wireless Fidelity.
- **WiMAX:** Worldwide Interoperability for Microwave Access.
- **ZRP:** Zone Routing Protocol.

Introduction Générale

Introduction Générale

1. INTRODUCTION

Aujourd'hui, les communications de données constituent indéniablement le pilier fondamental de toute entreprise sans lequel elle ne pourrait atteindre les niveaux actuels d'efficacité et de réactivité. L'avènement des communications a permis d'améliorer considérablement la productivité et a contraint les entreprises à adapter leurs méthodes de travail pour pouvoir survivre face à la compétition du marché. Cependant, on ne peut ignorer l'existence des vulnérabilités intrinsèques et extrinsèques de ces systèmes de communications, ce qui confère à la sécurité des réseaux un rôle fondamental.

La sécurité des réseaux est en plein effervescence depuis plusieurs années. Même si le contexte a évolué, de la machine non connectée, aux réseaux filaires puis sans fil, l'objectif de la sécurité a toujours été globalement le même, à savoir préserver l'intégrité, la confidentialité et la disponibilité des ressources informatiques et réseaux. Pour atteindre cet objectif, plusieurs mécanismes de sécurité ont été développés tels que les mécanismes d'authentification, de chiffrement, de contrôle d'accès, etc., selon l'environnement réseau, certains mécanismes de sécurité sont plus mûrs que d'autres du fait de la jeunesse de certaines technologies réseau. Cependant, même avec l'ancienneté, et même s'ils sont déjà largement implémentés dans les produits du marché de la sécurité informatique, comme les firewalls[1] et les VPNs[2], certains mécanismes méritent encore des améliorations. Il est aussi important de considérer la limitation des ressources des terminaux mobiles et des radiocommunications afin d'adapter les mécanismes de sécurité des réseaux filaires au contexte sans fil.

Nous assistons à une évolution majeure des réseaux qui pourrait être résumée par : "*communiquer en tout lieu, en local ou à distance, tout en se déplaçant*". Ceci est rendu possible grâce à l'évolution des réseaux sans fil et mobiles et l'évolution des terminaux utilisateurs. Même si les solutions de sécurité ne sont pas encore au point, le déploiement de ces réseaux est déjà effectif et va tendre à se développer du fait des besoins croissants des utilisateurs en termes de mobilité, flexibilité et services.

Les réseaux sans fil *Ad hoc*, aussi appelés *MANET* (Mobile Ad hoc NETwork) sont des réseaux spéciaux qui apparaissent dans ce contexte. Se souciant de pouvoir communiquer et de partager l'information dans n'importe quelle situation, ces réseaux sont des systèmes autonomes composés par

[1] Firewall : est un logiciel et/ou un matériel, permettant de faire respecter la politique de sécurité du réseau, celle-ci définissant quels sont les types de communications autorisés sur ce réseau informatique. Il mesure la prévention des applications et des paquets.
[2] VPN (Virtual Private Network) : une connexion inter-réseau permettant de relier deux réseaux locaux différents par un protocole de tunnel.

un ensemble d'entités mobiles libres de se déplacer sans contraintes. Ces entités utilisent le médium radio pour communiquer et forment un réseau n'utilisant aucune infrastructure existante. De ces faits, ces réseaux qualifiés de spontanés présentent une architecture originale qui évolue à tout instant.

Parmi les solutions offertes par les réseaux mobiles Ad hoc, il existe par exemple l'extension de l'accès aux bornes d'une infrastructure fixe (téléphonie sans fil, Wi-Fi[3], etc.), en dépassant la portée radio de ces bornes grâce aux relais Ad hoc que les utilisateurs peuvent se fournir les uns aux autres pour accéder indirectement à l'infrastructure. Cependant, l'aspect le plus novateur des réseaux Ad hoc est leur capacité à fournir une couverture réseau mobile de manière automatique et autonome, et ce même sans accès à une infrastructure préexistante.

Il existe plusieurs domaines d'application aux réseaux Ad hoc. Le domaine militaire et celui des secours en cas de catastrophes restent des exemples fréquemment cités. Toutefois, plusieurs autres applications des réseaux Ad hoc ont vu le jour. Nous citons les réseaux véhiculaires résultant de l'interconnexion de véhicules en mouvement ou les réseaux de capteurs capable de récolter et de transmettre les données environnementales. D'autres situations de la vie courante sont adaptées à l'utilisation des réseaux Ad hoc. C'est le cas par exemple du réseau créé entre un professeur et ses étudiants pour le besoin d'une séance de cours ou le réseau créé entre les participants à une réunion ou même entre les voyageurs dans un train.

L'élargissement du domaine d'application des réseaux mobiles Ad hoc nécessite plus de sécurité pour assurer l'intégrité et la confidentialité des données qui circulent dans le réseau. En effet, les réseaux mobiles Ad hoc sont confrontés à de nombreux problèmes liés à leurs caractéristiques qui rendent les solutions de sécurité développées pour les réseaux filaires ou sans fil avec infrastructure inapplicables dans le contexte des réseaux mobiles Ad hoc. Parmi les vulnérabilités qui touchent les réseaux mobiles Ad hoc nous pouvons citer :

- L'absence d'une infrastructure centralisée : Ceci ne nous permet pas d'opter pour une architecture centralisée. En effet, l'absence d'une unité centralisée accentue le défi pour proposer une solution de sécurité comme c'est le cas dans les réseaux filaires ou sans fil avec infrastructure fixe. Cependant, une architecture centralisée est déconseillée dans les réseaux mobiles Ad hoc, car elle peut créer un point de vulnérabilité dans le réseau.

- La topologie réseau dynamique : Parmi les caractéristiques des réseaux mobiles Ad hoc, on trouve l'environnement dynamique, qui est dû à la mobilité des nœuds. Cette caractéristique nécessite le développement de protocoles de routage sophistiqués et de solutions de sécurité adaptées à un tel environnement, ce qui constitue un vrai défi.

[3] Wi-Fi (Wireless Fidelity) : est un ensemble de protocoles de communication sans fil régis par les normes du groupe IEEE 802.11

- La vulnérabilité des nœuds : Les nœuds ne sont pas physiquement protégés, ils peuvent être capturés par des attaquants, ce qui pose problème au niveau des relations de confiance entre les nœuds. Ainsi, n'importe quel modèle de sécurité dédié au réseau mobile Ad hoc doit prendre en compte la compromission des nœuds, ainsi que la résistance à cette attaque.

- La vulnérabilité du canal : Le support de transmission est l'air. Ce dernier est très vulnérable aux écoutes clandestines. N'importe quelle machine qui dispose d'une carte sans fil adaptée à la technologie utilisée, est capable de capturer le trafic, de l'analyser et même d'injecter du nouveau trafic, soit dans le but de surcharger le réseau, soit dans celui de faire circuler des fausses informations pour changer la topologie du réseau. De plus, le canal sans fil est fortement vulnérable au risque de brouillage (Jamming[4]), ce qui a des conséquences néfastes sur le réseau.

- Les ressources limitées : Les nœuds mobiles dans les réseaux mobiles Ad hoc ont des ressources très limitées, comme la capacité de calcul, de stockage et surtout d'énergie. La batterie ne tient pas longtemps si le nœud travaille sans arrêt, ce qui complique davantage le problème de la sécurité. En effet, nous savons que la plupart des solutions de sécurité sont basées sur la cryptographie, mais malheureusement cette dernière est gourmande en termes de ressources : capacité de calcul, consommation d'énergie et mémoire de stockage. Par conséquent, de nombreux thèmes de recherches ont surgi au cours des dernières années pour remédier à ces vulnérabilités et assurer les services de sécurité dans les réseaux mobiles Ad hoc.

Le routage dans les MANETs est une fonction primordiale, où chaque entité mobile joue le rôle d'un routeur et participe activement dans la transmission des paquets de données. Ainsi, une entité peut communiquer directement avec une autre si elle est dans sa portée radio. Sinon, elle compte sur la coopération des voisins pour relayer ses messages. Cette équivalence entre les entités fait que les schémas classiques de routage utilisés dans les réseaux filaires ne s'appliquent plus pour les réseaux Ad hoc, qui nécessitent donc la mise en place de protocoles de routage spécifiques. Ces protocoles de routage Ad hoc spécifient la manière avec laquelle les entités communiquent pour échanger des informations sur la topologie leur permettant de construire leur propre vision du réseau.

2. PROBLÉMATIQUE

Les protocoles de routage Ad hoc tel que conçus [1, 2, 3] manquent de contrôles de sécurité. La plupart des propositions présupposent une phase de distribution de clefs [4, 5] pour protéger le routage, et assurer l'authentification des participants. Cependant, ces travaux reposent implicitement sur une infrastructure de sécurité, ce qui est contradictoire avec la nature d'un réseau Ad hoc. De nombreux travaux focalisent sur les comportements malveillants débouchant sur des attaques actives en

[4] Le brouillage radio "Jamming", est une technique de transmission d'un signal radio, visant à interrompre des communications, en diminuant le rapport signal sur bruit.

négligeant les comportements égoïstes qui peuvent avoir des conséquences dramatiques dans le cas d'un réseau Ad hoc, ou bien concentrent sur le second type de comportement en négligeant le premier.

Un réseau Ad hoc devient pleinement fonctionnel lorsque les nœuds participants démontrent un réel comportement de coopération. Or, la réalité est souvent toute autre, car il existe toujours quelques nœuds qui essayent de saboter ou veulent simplement profiter de l'environnement social qui leur est très favorable. Cela a conduit au développement des modèles de gestion de la confiance, dans lequel les nœuds mobiles capturent des preuves de la fiabilité des autres nœuds et de représenter leur comportement, puis d'établir des relations de confiance avec eux. Un protocole d'établissement de la confiance entre les nœuds joue donc un rôle important dans l'installation de ces réseaux. Il est fondamental que son exécution soit sûre en toutes circonstances.

Sur la base de ce cadre, plusieurs modèles de gestion de la confiance ont été proposés pour les MANETs [6, 7, 8, 9], où la confiance peut être définie comme un degré de croyance sur le comportement des autres entités, c'est-à-dire, la valeur qui reflète l'histoire de comportement d'un nœud avec un voisin spécifique [10]. Cependant, ces modèles de confiance peuvent être inappropriés pour réagir à des attaques malignes ou de collision en raison des facteurs de décision qui sont souvent incomplets dans la dérivation de la confiance et qui ne sont pas pleinement intégrés aux caractéristiques intrinsèques de MANET. Dans ce contexte, nous nous concentrons sur la résolution de ces problèmes d'une manière globale, fiable et efficace, contrairement aux solutions actuelles qui se concentrent seulement sur certains sous-problèmes.

Ces problèmes essentiels de sécurité ne se situent pas uniquement au niveau du support physique mais également dans le fait que tous les nœuds sont équivalents et potentiellement nécessaires au bon fonctionnement du réseau. En outre, en absence d'infrastructure permettant l'authentification des nœuds, un nœud malicieux ou compromis pourrait s'insérer dans le réseau et effectuer différentes actions malveillantes. Par exemple, l'attaque man-in-the-middle[5] met en lumière la problématique de l'authenticité des interlocuteurs, qui est essentielle pour une réelle sécurité dans les réseaux Ad hoc. Or, par définition, un réseau Ad hoc ne possède d'infrastructure fixe ce qui complique très sérieusement la mise au point de protocoles d'authentification simples et efficaces. Cette absence soulève également de nombreuses autres interrogations sur la sécurité de tels réseaux :

a. Peut-on échanger sans risque et sans paralyser la circulation d'informations ?

b. Comment faire naître la confiance lors d'un échange au travers d'un canal hostile ?

c. Comment les entités peuvent-elles se reconnaitre ?

d. Comment préserver les données de la vie privée ?

[5] Man-in-the-middle attack (MITM) : est une attaque qui a pour but d'intercepter les communications entre deux parties, sans que ni l'une ni l'autre ne puisse se douter que le canal de communication entre elles a été compromis.

- La question (*a*) est centrale. Sa réponse doit constituer le pivot essentiel à la viabilité des services de sécurité. Elle en appelle une autre : comment un nœud est-il sûr de communiquer avec le bon interlocuteur et non avec un nœud malveillant ou un attaquant ? Les nœuds ne se connaissant pas et le peu de contrôles pouvant être effectués impliquent qu'il faut tenir compte des risques encourus. Il sera à la charge des nouveaux protocoles de participer à la prise de décision en définissant les politiques de confiance locales à mener et les niveaux de sécurité à garantir.

- La question (*b*) est fondamentale, car la confiance constitue un levier incontournable à l'émergence des réseaux du futur. La confiance est par essence une caractéristique des systèmes purement sociaux et s'observe dans les interactions humaines. Les modèles de confiance actuels, éprouvés dans les réseaux filaires ou sans fil en mode infrastructure, sont inopérants pour les réseaux Ad hoc parce qu'ils sont construits en général à partir d'un tiers de confiance défini à l'avance ou un système d'adressage stable. De tels protocoles de confiance sont en effet inadaptés à la forte mobilité qui implique sans-cesse l'apparition de nouveaux nœuds et leur disparition. Il naît alors un paradoxe entre la nécessité de fournir une confiance a priori envers des nœuds inconnus et préserver un niveau de sécurité optimal.

- La question (*c*) soulève la problématique de l'identification et de l'authentification. L'utilisateur est confronté à un manque de visibilité face à la multiplicité des entités mobiles qui gravitent autour de lui. Or, il est difficile de mettre en place un tiers de confiance au sens traditionnel qui assurerait l'établissement et la gestion de la confiance. Il n'existe donc pas de moyen simple pour garantir l'authenticité d'un nœud étranger car les identités ne sont jamais préétablies et l'emploi de pseudonymes est généralement la règle. Les protocoles d'identification et d'authentification classiques sont donc inopérants.

- La question (*d*) est essentielle pour l'utilisateur qui ne doit pas s'exposer aux risques d'une attaque telle l'usurpation de son identité ou le profilage de ses activités. Les contraintes sont nombreuses mais par soucis d'efficacité, de robustesse et de fiabilité, il faut qu'ils soient capables de garantir la confidentialité, la protection de la vie privée et l'intégrité des informations échangées pour que l'utilisateur soit protégé dans l'utilisation de tels réseaux.

Notre travail se concentre donc sur le problème de l'établissement de la confiance dans les réseaux mobiles Ad hoc, sur la signification de la confiance dans de tels réseaux et sur la manière de mettre en place des protocoles efficaces et fonctionnels de gestion de la confiance dans un environnement entièrement opérationnel.

3. OBJECTIF ET CONTRIBUTIONS

Les conditions de sécurité définies dans cette thèse mènent à la conclusion que les comportements malveillants et égoïstes doivent être pris en compte pour fournir un ensemble complet de service de

sécurité de réseau. En conséquence, les objectifs de recherches développés dans cette thèse visent la conception des mécanismes de sécurité faisant face au comportement égoïste et malveillant des nœuds.

Notre approche consiste alors à définir et proposer un nouveau modèle d'établissement et de gestion de la confiance, où les nœuds établissent un rapport de confiance basé sur des expériences et des recommandations préalables, le but est de rendre les nœuds du réseau capables de recueillir des informations pour raisonner, apprendre et prendre leur propre décision. La solution envisagée est de faire reposer la prise de décision d'un échange sur la base de la confiance, sachant que chaque nœud ne pourra se protéger d'éventuels voisins malicieux qu'en faisant appel aux informations locales dont il dispose. Notre modèle de gestion de la confiance a donc pour objectif d'intégrer des mécanismes contrant les attaques qui pourraient exister, en forçant la coopération entre les nœuds, et détectant les comportements défaillants.

Le premier mécanisme proposé dans notre modèle de gestion de la confiance repose sur la théorie des ensembles flous. La confiance est par nature un concept flou, ce qui pose une contrainte floue sur la prise de décision de l'itinéraire de confiance. Dans les relations humaines, la confiance est souvent exprimée linguistiquement plutôt que numériquement [11]. Il est bien établi que la logique floue est adaptée pour quantifier la confiance entre les entités qui composent un réseau ou un groupe. Un des avantages de l'utilisation de la logique floue pour quantifier la confiance entre les nœuds dans les réseaux Ad hoc, est sa capacité à quantifier des données imprécises ou de l'incertitude dans les mesures de l'indice de sécurité des nœuds Ad hoc.

Le deuxième mécanisme présenté pour ce modèle, est basé sur la théorie d'analyse relationnelle grise "G. R. A" (Grey Relational Analysis theory). Elle est l'une des théories les plus importantes des systèmes gris, elle utilise un concept spécifique de l'information, qui est défini comme étant, les situations avec aucune information en "noir", et ceux qui ont une information parfaite en "blanc". Cependant, aucune de ces situations idéalisées ne se produit jamais dans les problèmes du monde réel. En fait, les situations entre ces deux extrêmes sont décrites comme étant grises ou floues.

Avec cette définition, la quantité et la qualité des informations inférées forment un continuum allant d'un manque total d'information à une information complète, (du noir au gris jusqu'au blanc). Tant que l'incertitude existe toujours, on est toujours quelque part au milieu, quelque part entre les deux extrêmes, quelque part dans la zone grise.

L'analyse grise est alors un ensemble clair d'énoncés sur les solutions de système. À un extrême, aucune solution ne peut être définie pour un système sans aucune information. À l'autre extrême, un système avec une information parfaite a une solution unique. Au milieu, les systèmes gris donneront une variété de solutions disponibles. L'analyse grise ne cherche pas à trouver la meilleure solution, mais peut fournir des techniques pour déterminer une bonne solution.

L'idée de base de cette technique consiste à sélectionner certaines variables d'entrée qui montrent un impact plus fort à la sortie du système. Cette technique utilise les informations fournies par le système gris pour comparer dynamiquement et quantitativement chaque facteur, et d'établir une relation en fonction du niveau des facteurs de similarité ainsi que du niveau de variabilité. Ensuite, le rapport de décision peut être prise en fonction de la relation.

Les principales contributions techniques de notre travail peuvent être résumés comme suit :

a. Etablir un modèle d'évaluation de la confiance en utilisant la méthode d'analyse relationnelle grise combinée avec le procédé de la logique floue. La nouveauté du modèle de confiance proposé est sa structure entièrement hiérarchique et hybride. Il comprend la confiance directe, la confiance indirecte (par recommandation) et le degré actif pour déduire une valeur de confiance. Ce modèle combine à la fois les éléments classiques de la sécurité et de nouveaux éléments que nous suggérons, et qui sont nourris par les interactions de l'entité avec son environnement.

b. Accomplir et réaliser la confiance dans notre modèle, les valeurs de confiance obtenues par les nœuds peuvent être facilement utilisées dans le schéma de gestion de la confiance, y compris le mécanisme d'anti-attaque et des applications de prise de décision.

c. Proposer un algorithme de routage fiable basé sur notre modèle de confiance. Au même temps, un protocole de routage sécurisé est présenté comme une application de l'algorithme de routage proposé.

d. Effectuer des simulations pour évaluer l'efficacité de notre protocole en ce qui concerne l'identification, l'authentification, l'isolement des nœuds malveillants et l'efficacité contre les différentes attaques.

4. ORGANISATION DE LA THÈSE

Notre travail est organisé en deux parties, la première consiste à présenter le domaine de recherche, tandis que la deuxième est consacrée à nos contributions en termes de modèles et solutions de confiance et de sécurité dans les réseaux Ad hoc.

La première partie est structurée en trois chapitres. Le premier chapitre servira de préliminaires aux notions nécessaires à la compréhension globale de la thèse, particulièrement les concepts liés à la sécurité et à la confiance. Dans le deuxième chapitre, nous présentons les réseaux mobiles Ad hoc, leurs caractéristiques, leurs domaines d'application, et les protocoles que doivent suivre de telles

structures, ainsi que la couche MAC 802.11[6] utilisée pour ces réseaux. Ensuite, le problème de la sécurité va être abordé dans le troisième chapitre. Nous citons les principaux axes de sécurité dans les réseaux mobiles Ad hoc. Puis, nous étudions les différentes approches de sécurité dédiées à l'environnement de ces réseaux et nous expliquerons quelles sont les carences.

La deuxième partie de notre travail est aussi divisée en quatre chapitres, dont le premier chapitre consiste à présenter et à expliquer les mécanismes nécessaires au développement de nos contributions. Le deuxième chapitre dévoile notre nouveau modèle de gestion de la confiance et décrit les propriétés intuitives qui devraient être dans tous les modèles de ce contexte. Le troisième chapitre décrit la partie applicative de notre modèle de gestion de la confiance, il présente également notre protocole de routage. Le dernier chapitre est consacré à notre étude expérimentale correspondante dans le but d'évaluer les performances de nos contributions. Enfin, dans la conclusion de cette thèse, nous proposons une synthèse de nos contributions, puis nous évoquons les perspectives possibles de nos travaux.

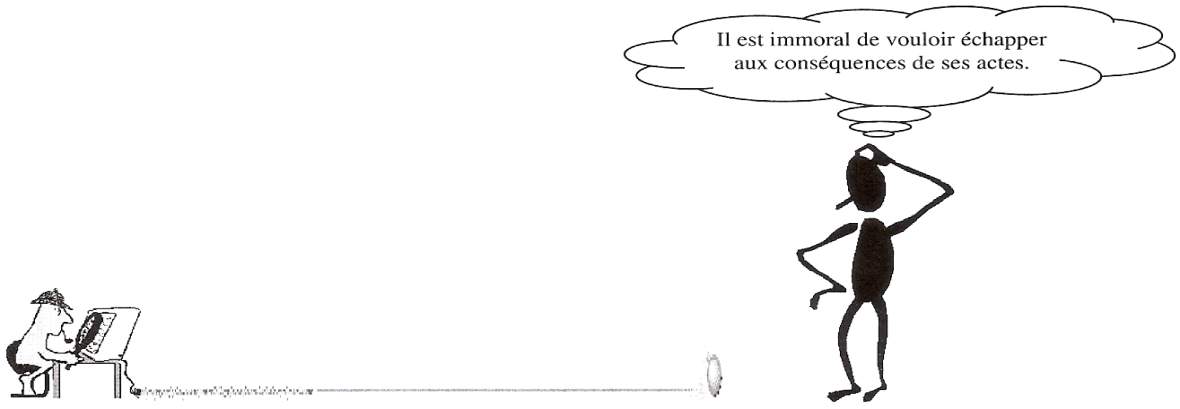
[6] IEEE 802.11 : est un ensemble de normes concernant les réseaux sans fil locaux (le Wi-Fi) qui ont été mises au point par le groupe de travail 11 du comité de normalisation.

Partie 1

Introduction sur le domaine de recherche

Chapitre 1

Sécurité, Risques
&
Attaques



1. INTRODUCTION

Pour une meilleure compréhension des solutions de sécurité décrites dans la suite de cette thèse, il est utile de présenter dans un premier temps les notions fondamentales de la sécurité dans les réseaux. Ce chapitre s'attache en particulier à introduire les définitions de sécurité et de confiance dans l'ère numérique. La première section de ce chapitre introduit notamment la terminologie habituellement utilisée dans ce contexte. Les besoins en mécanismes de sécurité sont illustrés par les liaisons et relations entre un bien, son propriétaire et l'environnement. La deuxième section présente des modèles de menaces pour les divers systèmes de communications actuels. Elle discute plusieurs aspects importants pour la sécurité comme l'hétérogénéité et l'homogénéité des systèmes. Cette partie introduit une classification des vulnérabilités rencontrées, les séparant en risques infrastructurels et personnels. Enfin, une dernière section illustre comment ces vulnérabilités peuvent être exploitées, à travers la description de plusieurs attaques spécifiques.

2. SÉCURITÉ DANS L'ÈRE NUMERIQUE

2.1 La propriété privée

"Chacun a le droit à la protection des intérêts moraux et matériels découlant de toute production scientifique, littéraire ou artistique dont il est l'auteur"[1].

Le principe de la propriété privée est un des piliers sur lesquels se base la société moderne. La protection de cette propriété, sous forme de biens, de patrimoine, d'investissements, ainsi que le respect des domaines implicitement annexes comme la sphère privée, les droits de l'homme, etc., sont des obligations morales et légales pour l'État, les entreprises et les citoyens.

Les développements culturels et industriels de la fin du 19ème et du 20ème siècle et la globalisation technologique ont changé la perception de la concrétisation des biens et des valeurs, en la poussant progressivement de sa forme purement matérielle (immobiliers, produits de base, etc.) vers des formes plus abstraites, comme le soulignent par exemple l'introduction des lois internationales sur la propriété intellectuelle (droit d'auteur) [12], l'évolution du secteur tertiaire, etc., ce processus a atteint son apogée avec le début de l'ère numérique qui, en introduisant la notion du patrimoine numérique, efface les dernières frontières entre les biens réels et les biens virtuels. Apparaissent enfin les notions de

[1] Art 27.2 - Déclaration universelle des droits de l'homme (1948).

produit logiciel (software mais aussi multimédia, jeux, etc.) et, par la suite l'apparition d'Internet, de ventes de numérique par le numérique au numérique (*iTunes*[2], etc.).

Pourtant, c'est cette information numérisée qui est particulièrement sensible et vulnérable à la volatilité, aux changements et à une duplication incontrôlable. En effet, le produit numérique ne connaissant point la notion d'original, toute instance doit être traitée comme un clone.

De plus la globalisation et le développement des technologies de télécommunications et des services informatiques résultent dans une normalisation et une ouverture de système d'informations (SI). La partie propriétaire dans les SIs diminue continuellement, et l'on observe ainsi la banalisation d'accès, l'interconnexion amplifiée des systèmes et une forte tendance vers une convergence des secteurs auparavant séparés, comme on le voit avec le secteur du multimédia, les communications classiques et informatiques. La facilité d'accès amplifie les échanges du bien avec son environnement, étant donné la vulnérabilité innée des biens numériques, le risque d'abus augmente avec l'exposition à l'utilisation diversifiée alors que la protection devient plus compliquée.

Dans l'ère numérique, les biens virtuels (produit en logiciel, savoir-faire, algorithmes, connaissances, renseignements, multimédia, données, etc.) deviennent partie intégrante des infrastructures de SI conçus pour rendre différents services. Confrontés à une telle répartition de la propriété numérique dans les infrastructures interconnectées de systèmes d'information, les opérateurs de ces systèmes, ses utilisateurs (les entreprises et les individus) et l'État doivent se poser des questions quant à la protection des informations contenues et échangées. Cette protection doit couvrir :

- L'intégralité des biens, c'est-à-dire aussi bien les contenus transportés que les parties utilisées ou stockées dans les infrastructures en question.
- L'intégralité des types des acteurs.
- L'intégralité du temps, respectant notamment les aspects de l'usage réel mais aussi les aspects légaux imposés (expiration versus audit).

Fournissant aujourd'hui des services critiques (contrôle aérien, défense, services d'urgence, transactions commerciales, etc.), les systèmes des télécommunications deviennent indispensables pour tous les acteurs de la société de l'information. Les indépendances entre ces infrastructures et les infrastructures critiques classiques (énergie, transport, l'eau) créent de nouveau systèmes dont la complexité et la vulnérabilité sont supérieures à celles des systèmes qui les composent. De nouvelles dispositions deviennent nécessaires pour prendre en compte les indépendances des infrastructures critiques modernes (robustesse, résiliences, innocuité, etc.).

[2] iTunes, est un logiciel de lecture et de gestion de bibliothèque multimédia numérique distribué gratuitement par Apple.

À l'autre bout du spectre, la démocratisation de l'informatique, poussée par le progrès technologique bouleversant, crée des véritables infrastructures personnelles dans l'espace privé des individus. Dans la plupart des cas, il ne s'agit plus de systèmes isolés, mais au contraire des systèmes de plus en plus ouverts, se chevauchant dans plusieurs dimensions, difficiles à délimiter en pratique. Aujourd'hui, les divers acteurs, les particuliers comme les États, doivent maitriser aussi bien les contenus et les opérations (transport, traitement et stockage) que sécuriser les info-sphères, c'est-à-dire les infrastructures virtuelles crées en partie dans l'espace privé respectif par les interconnexions des SIs et le partage des biens numériques.

2.2 *Qu'est-ce que la sécurité*

Il devrait alors nécessaire de trouver une définition de la sécurité commune à un bien, un service, une infrastructure et une info-sphère pour tout propriétaire concerné.

On trouve dans la littérature plusieurs définitions de la sécurité. Le dictionnaire de l'académie française [13] définit la sécurité comme suit "Sécurité. *n.f.* Confiance, tranquillité d'esprit qui résulte de l'opinion, bien ou mal fondée, qu'on n'a pas à craindre de danger".

Plusieurs aspects discutés ci-dessous sont visibles dans cette définition où la sécurité est vue comme une situation caractérisée par l'absence de tout risque pour les personnes concernées "Je me sens en sécurité". Bien que cette définition soit d'un niveau suffisamment haut et notamment applicable sur les SIs [14], elle définit la sécurité d'une manière faible "bien ou mal fondée" et se positionne comme constatation, c'est-à-dire une vue a posteriori, omettant toute notion de réalisation de la situation souhaitée.

Selon une autre vision, la sécurité est souvent vue comme l'art de partager les secrets. Cette définition de sécurité est celle donnée par les cryptologues, de très bas niveau, nécessaire mais insuffisante dans beaucoup de contextes actuels. Elle se révèle difficile à appliquer sur des systèmes d'informations modernes dans une approche "*top-down*[3]".

On peut définir la sécurité de l'ère numérique comme une *quête* pour la protection des biens numériques et la protection des systèmes de traitement des biens numériques contre tout acte non voulu ou perçu comme un abus par les propriétaires. Les actes non voulus sont typiquement possibles à cause des vulnérabilités présentes dans les SIs. L'exploitation des vulnérabilités crée des menaces et représente ainsi un risque du point de vue de propriétaire. Ainsi, les risques mènent à l'implémentation d'un ensemble de contre-mesures.

Cette définition se trouve alors au croisement de la définition habituelle, visant à installer la tranquillité d'esprit et de la définition militaire [14], visant à parler des mesures. Cette définition prend

[3] Une approche ascendante (dite *bottom-up*) ou descendante (dite *top-down*) caractérise le principe général de fonctionnement d'une démarche procédurale.

comme point de départ l'existence d'un bien méritant d'être protégé dans un certain environnement de traitement. Le terme 'quête' utilisé dans cette définition, souligne la continuité du processus et l'incertitude liée, typiques pour la sécurité, les contre-mesures doivent évoluer dans le temps, et généralement on ne sait pas si elles sont suffisantes, les contre-mesures elles-mêmes pourraient avoir des vulnérabilités (leur présence se traduit par des nouveaux risques contre lesquels le propriétaire doit se protéger). De plus, par la notion de "non voulu" la définition implique la présence d'au moins deux acteurs distincts, nommés respectivement *propriétaire* et *attaquant*, dont le dernier est présumé malveillant. C'est l'attaquant qui crée des menaces en exploitant les vulnérabilités dans ou autour du bien. Le propriétaire veut minimiser ses risques et impose des contre-mesures qu'il considère comme nécessaires pour protéger le bien. (Figure 1.1)

La complexité de cette problématique est due à plusieurs facteurs. Étant donné la complexité architecturale et technologique et le dynamisme des biens dans le contexte du SI, il est ardu de cerner toutes les vulnérabilités possibles.

Il est souvent difficile (c'est-à-dire couteux, trop contraignant) de réaliser l'ensemble de contre-mesures jugé nécessaire : le plus souvent, le propriétaire doit en pratique évaluer le compromis entre son estimation de la gravité d'un risque et le cout de la réalisation des contre-mesures.

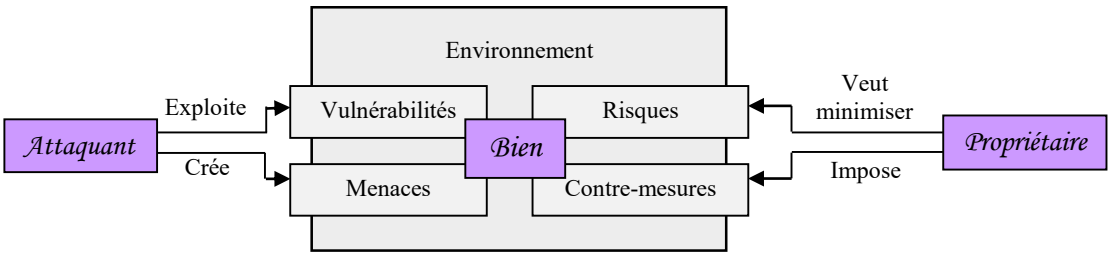


Figure 1.1 : Les relations entre le bien, l'attaquant et le propriétaire.

Il s'agit de l'analyse des risques. L'installation de l'ensemble jugé nécessaire augmente la complexité du SI initial. En effet, c'est ce nouveau système, résultant de l'ajout des contre-mesures au système initial, qui doit être évalué à nouveau. Les compromis acceptés par le propriétaire introduisent des risques résiduels, ce qui engendre souvent, à terme, de nouvelles vulnérabilités.

Ainsi, l'ensemble des contre-mesures est normalement insuffisant, d'une part à cause de l'ignorance de certaines vulnérabilités associées à la complexité des interactions entre le bien et son environnement, et d'autre part à cause de l'évaluation de risque pratiquée, typiquement liée à des modèles probabilistes (statistique de l'utilisation, confort de l'utilisation versus perception du risque, pertinence des services concernés, etc.). Évidemment, il n'existe pas de modèle suffisant, car un attaquant en s'affranchissant de toute hypothèse utilise son intelligence pour trouver les vulnérabilités.

Donc la sécurité des SIs est avant tout un processus continu [15] et pas un produit final. Dans le cas idéal, la perception de l'environnement et des risques, l'estimation de la valeur de l'objet de sécurité, la recherche des vulnérabilités dans l'ensemble {système initial, contre-mesure} doivent être refaits systématiquement et périodiquement. Il n'existe pas aujourd'hui de système standard pour répondre aux exigences de chacune de ces phases pour différentes cibles, ni des spécifications pour les périodes exactes.

Classiquement *le processus de sécurité* est décomposé en trois aspects se référant à l'objet de sécurisation en spécifiant notamment ce qui doit être protégé. Cette vue de la sécurité est connue sous la trinité *CIA* (confidentialité, intégrité et disponibilité[4]). Cette décomposition est aujourd'hui normalement insuffisante, car elle ne couvre pas bien certaines nouvelles menaces comme les virus informatiques, les messages non-sollicités, ou l'utilisation abusive.

Une autre approche se réfère à la question comment protéger le bien et décompose le processus de sécurité en phases de *Prévention, Détection* et *Réaction*, typiquement dénommées *PDR*. Il est évident que la réalisation de mécanismes pour ces phases sera également liée aux aspects de la *qualité de service.*

Commune à toutes les deux approches est l'estimation des vulnérabilités des menaces et des risques qui doit se faire auparavant.

2.3 Confiance et subjectivité

On note deux aspects importants inhérents à toute définition de la sécurité :

Le premier aspect est la notion de la *confiance*, il est évident que la confiance absolue dans tout acteur de l'environnement étudié enlève le besoin pour la sécurité de la même façon que la méfiance totale interdit toute exposition d'un actif à son environnement et met en question la notion de la propriété privée. En effet, si tout acte possible sur l'actif est perçu comme un risque, le système converge inévitablement vers la clôture totale. Ceci souligne l'interdépendance entre la confiance et la sécurité [14] : le partage d'un secret présume aussi bien une confiance initiale que la notion du confort vis-à-vis des risques. De même, l'existence des mesures de sécurité présume une confiance dans certains acteurs ou dans certaines parties du système. Pourtant dans le cas général, on ne connaît pas de transformation directe entre la confiance et la sécurité. Malgré leur influence mutuelle, il faut faire une distinction nette entre la sécurité et la confiance.

Le deuxième aspect important est la *subjectivité*. En effet, pour le même bien dans le même environnement, l'évaluation des risques peut être radicalement différente. Elle ne dépend pas seulement de la confiance présumée (se basant par exemple sur des connaissances et des expériences

[4] Availability, en anglais.

du propriétaire) mais aussi de son investissement et principalement de son positionnement par rapport à l'objet (c'est-à-dire ses cibles, ses intérêts, l'utilisation prévue).

La subjectivité et la confiance doivent être évaluées dans le contexte de l'environnement visé (militaire/hostile, civil/courtois). Par définition, la subjectivité de la sécurité ne pose pas de problèmes fondamentaux quant à l'évaluation de la sécurité, si le bien est isolé du monde extérieur, c'est-à-dire, si l'environnement du bien d'un propriétaire ne chevauche pas avec les environnements des autres propriétaires (systèmes fermés, systèmes propriétaires, etc.). Mais dans l'ère numérique, plus souvent le contraire est le cas, les biens numériques de différents propriétaires sont traités par différents systèmes numériques appartenant à d'autres propriétaires, il est souvent normal que le bien traverse pendant son traitement plusieurs dizaines de systèmes rendant des services différents. La complexité des interactions, les natures très différentes des biens mêmes et les évaluations de dangers très différentes posent un énorme problème quant à l'évaluation de risques pour la globalité du système.

Dans le cas général, il est en effet impossible de comparer deux ensembles de contre-mesures. De plus, on observe l'interdépendance naturelle entre la subjectivité et la confiance. À cause de cette interdépendance, les ensembles de contre-mesures demandés par deux propriétaires dans le cas d'un échange peuvent contenir des exigences contradictoires ou sémantiquement non recouvrables, une mesure de protection exigée pour un bien "X" peut se révéler irréalisable en vue d'une composition d'une série de traitements séquentiels, etc.

2.4 La relation service-sécurité

La définition de sécurité utilisée ci-dessus saisit explicitement le service comme une cible à protéger. En effet, tout service, étant économiquement parlant un équivalent immatériel d'un bien, possède dans le cas général une valeur pour son offreur. Cette valeur est justifiée par l'investissement initial dans l'infrastructure du service, par le cout du maintien quotidien et des évolutions possibles, et par les objectifs commerciaux ou autre de cette offre. De plus implicitement, chaque service présume une interaction de son offreur (dans le cadre du service, c'est-à-dire contractuelle) avec au moins un deuxième acteur, l'utilisateur. Chaque service sous-entend donc une ouverture vers l'extérieur représente par l'interface d'accès par les utilisateurs. De plus, généralement, l'ensemble des utilisateurs prévus d'un service est un vrai sous-ensemble de l'ensemble total des acteurs. Par conséquence, chaque service est naturellement exposé aux menaces : en absence de contre-mesures (contrôle d'accès), l'utilisation de chaque service (indépendamment de sa sémantique) est étroitement liée à la notion d'abus, c'est-à-dire, à une utilisation non contractuelle, hors contractuelle, etc. La sémantique ajoute d'autres risques et pour l'offreur et pour l'utilisateur : en effet, les données échangées dans le cadre du service doivent normalement être réservées à leur destinataire, le fait de participation à un service est également une information confidentielle (par exemple protection de la sphère privée). En conséquence, chaque service nécessite une analyse du système propre au service et prenant en compte

tout acteur impliqué dans son exécution. Le contrat de service est utile, parmi d'autres, pour homogénéiser la politique de sécurité des acteurs et créer une base de confiance entre les partenaires.

Au contraire, dans le cas général, la sécurité ne peut pas être vue comme un service. Le problème est dans la définition de la sécurité, souligné par la liaison intrinsèque et intime des mesures de la sécurité à leur cible. Tout d'abord, la notion de "*service de sécurité*" suggère une sécurisation d'un bien (par exemple d'un service) peu ou pas assez sécurisé. Or, il serait préférable de réfléchir à des besoins et des problèmes de sécurité avant l'exposition du bien à son environnement (par exemple avant le déploiement de service, notamment dans la phase de conception). Ensuite la subjectivité de l'appréciation de la sécurité d'un même bien est généralement impossible à résoudre, même en présumant une grande flexibilité du service (par exemple par personnalisation). La protection d'un bien par un service de protection ajoute au moins un partenaire "l'offreur du service de protection" ce qui peut contredire les exigences de certains propriétaires. L'exemple typique est le fournisseur de l'infrastructure du service, cette dernière exige des mesures de protection, mais elles sont à intégrer dans l'infrastructure même et ne peuvent donc pas appartenir à une tierce personne.

Généralement, la sécurité reste une propriété non fonctionnelle, souvent invisible mais intrinsèque à chaque service "on ne peut pas l'activer et la désactiver", on ne peut pas s'abonner à l'utilisation de sécurité[5]. Autrement dit, chaque service, il faut prévoir un sous-système de sécurité, même si celui-ci reste invisible à l'utilisateur de service. Néanmoins, en pratique la sécurité peut être proposée comme un service dans certains cas de figures. Cela semble notamment applicable pour des services statiques, bien analysés, bien déployés et acceptés, c'est-à-dire pour les cas où les modèles de menaces sont approuvés, et les mesures de protection sont jugées largement suffisantes (par exemple par la pratique quotidienne : observation du risque réel sous la protection appliquée).

Ce sont surtout les *nouveaux services* qui entrainent le déroulement socio-technologique d'une spirale de développement mutuel des vulnérabilités et des mesures de protection. Les nouveaux services définissent de nouveaux usages et donc de nouveaux scénarios, ils subissent alors de nouveaux abus. De plus, la pression commerciale pour le déploiement des nouveaux services (effet de concurrence féroce sur le marché, etc.) étant importante, les nouveaux services sont souvent déployés avec une analyse insuffisante : les vulnérabilités sont naturellement méconnues et les modèles de menaces ne correspondent pas à la réalité. Dans cette situation, le service est souvent déployé avec un accent sur sa forme fonctionnelle. Toutefois, en fonction de progrès du déploiement, de nombreuses vulnérabilités sont découvertes. Celles-ci, exploitées par les attaquants, créent des menaces réelles, car ils débouchent sur un ensemble d'attaques. Perçues comme des risques par les offreurs et les

[5] Même si en pratique on peut s'abonner à une version "plus sécurisée" d'un même service, il faut comprendre que l'opérateur est obligé d'implémenter des mesures de sécurité pour les deux versions du service proposées. Paradoxalement, les mesures de sécurité pour la version moins sécurisée seront typiquement plus difficiles à mettre en œuvre (floues, plus subtiles).

utilisateurs, elles freinent le déploiement du service rendant l'investissement nécessaire dans les contre-mesures, supérieur aux pertes liées à la gêne du déploiement.

2.5 *Besoins en sécurité*

Les besoins de base en sécurité (services de sécurité) font référence à des concepts de sécurité contrairement aux mécanismes de sécurité qui incluent l'ensemble des outils utiles à la mise en œuvre de ces besoins en sécurité. Ainsi le standard X.800 [16] définit *les besoins*[6] en sécurité de la façon suivante :

- ***Disponibilité*** **:** "propriété d'être accessible et utilisable sur demande par une entité autorisée".
- ***Contrôle d'accès*** **:** "précaution prise contre l'utilisation non autorisée d'une ressource, cela comprend les précautions prises contre l'utilisation d'une ressource de façon non autorisée".
- ***Intégrité des données*** **:** "propriété assurant que des données n'ont pas été modifiées ou détruites de façon non autorisée".
- ***Authentification de l'origine des données*** **:** "confirmation que la source des données reçues est telle que déclarée".
- ***Authentification de l'entité homologue*** **:** "confirmation qu'une entité homologue d'une association est bien l'entité déclarée". Notons la distinction qui doit être faite entre "identification" et "authentification". Une identification se réfère à la déclaration de l'identité par une entité (utilisateur, équipement) par fourniture de son identifiant (nom, pseudonyme, adresse de messagerie électronique, adresse IP, nom de domaine), ou bien à la procédure de retrouver l'identité d'un utilisateur parmi *N* utilisateurs répertoriés en fonction de certains paramètres le caractérisant. Une authentification consiste à prouver l'identité déclarée par la fourniture d'un ou plusieurs éléments d'authentification.
- ***Confidentialité des données*** **:** "*propriété* d'une information qui n'est ni disponible, ni divulguée aux personnes, entités ou processus non autorisés".
- ***Détection de rejeux***[7]**:** une détection de rejeux consiste pour une entité à détecter que des données reçues sont dupliquées d'un précédent échange. Des données peuvent avoir été envoyées de façon sécurisée par une entité légitime, copiées par un imposteur, elles sont réémises vers la même destination. Elles sont toujours authentiques, mais elles ont déjà été traitées, il est donc nécessaire de détecter ce rejeu pour éviter qu'elles ne soient traitées plusieurs fois.

[6] Excepté le besoin de détection de rejeux.
[7] Non défini dans X.800

3. RISQUES ET MENACES POUR LES SYSTÈMES DE TÉLÉCOMMUNICATIONS

3.1 Le rôle des systèmes des télécommunications

En tant que dénominateur commun de toute interconnexion de systèmes d'informations modernes, les systèmes des télécommunications sont au cœur de l'ère numérique. Ils forment ainsi l'interface cruciale et se retrouvent au point critique du point de vue de sécurité.

Historiquement formés par une infrastructure fermée sous le contrôle de l'état, ce sont les systèmes des télécommunications qui ont subi le changement le plus radical durant ces dernières décades. Aujourd'hui, les systèmes des télécommunications sont considérés comme des infrastructures critiques. Leur protection devient même une préoccupation politique [14] plus que commerciale et personnelle. Tout acteur impliqué (État, entreprise, particuliers) doit assumer les responsabilités autour des systèmes des télécommunications qui se trouvent au centre de leurs usages quotidiens. Ces responsabilités peuvent être dues aux risques perçus (et dépendent dans ce cas du positionnement de l'acteur par rapport au système en question) mais aussi d'une nature légale.

Les réseaux (optique, filaires et sans fil) sont des composants principaux des systèmes des télécommunications. Ces réseaux et leurs utilisateurs sont exposés à plusieurs risques. Nous classifions ici ces risques selon un modèle qui distingue les rôles des propriétaires des données et des propriétaires des infrastructures les traitant.

3.2 Modèles de menaces des systèmes des télécommunications

Les modèles des menaces décrivent le système, les acteurs de ce système, et leur position dans le système (par exemple lien, nœud). Enfin, le modèle introduit un attaquant dans le système et démontre son positionnement topologique et ses capacités potentielles.

Le modèle classique des menaces pour un canal de communications se base sur le modèle de communications minimaliste, définissant deux participants, dénommés par exemple Alice et Bob, et un canal de communications. (Figure 1.2)

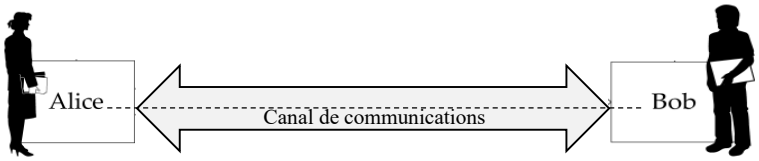


Figure 1.2 : Le modèle des communications.

Dans ce modèle, on présume normalement une confiance initiale mutuelle entre Alice et Bob. Ce modèle est souvent utilisé dans la cryptographie, il ne peut modéliser que des attaques contre le

canal de communications entre Alice et Bob. Or, dans le contexte des systèmes des télécommunications, ce modèle n'est pas exhaustif, car d'autres éléments et des vulnérabilités sont présents.

Dans la figure 1.3, on introduit alors un modèle plus approprié, distinguant les deux communicants (Alice et Bob) et au moins une infrastructure traversée et son autorité. En effet, dans le cas général, cette autorité est une personne tierce à Alice et Bob.

C'est l'apparition de cette tierce partie qui augmente la complexité du système, introduit des nouvelles interfaces et des vulnérabilités, peut nécessiter d'une chaîne de confiance plus compliquée et élargit ainsi le spectre des menaces possibles pour un attaquant "Eve".

Le modèle de confiance de la figure 1.3 peut avoir des formes très différentes, mais en pratique on présume un des cas suivants :

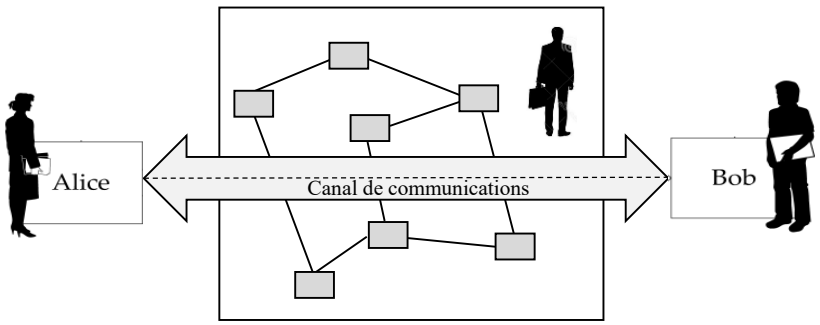


Figure 1.3 : Le modèle des communications avec un système des télécommunications.

- Alice et Bob se font mutuellement confiance, et Alice et Bob font confiance au système des télécommunications utilisé (réseau privé).
- Alice et Bob se font mutuellement confiance, mais n'ont pas confiance en infrastructure traversée (réseau public).
- Alice et Bob font confiance à l'infrastructure mais pas l'un à l'autre, ils s'en servent comme d'un tiers de confiance pour établir une confiance mutuelle.

Dans la figure 1.4, on présente de gauche à droite les menaces directes contre les parties du modèle :

Eve peut attaquer un des communicants (Alice, dans l'exemple) en utilisant les vulnérabilités présentes dans les logiciels et les protections d'Alice. Strictement vue, cette menace n'est pas liée au système des télécommunications. Toutefois, un terminal disposant d'une interface de connexion à un système des télécommunications est un système plus ouvert et donc plus vulnérable. Souvent, des attaques

sont possibles à cause des vulnérabilités présentes dans le terminal et par la visibilité du terminal qui participe dans un service des télécommunications. Un exemple typique est l'exécution du code malicieux sur la plate-forme utilisée par Alice par le biais d'un virus, ou par le débordement de tampons de réception.

Alternativement, Eve peut attaquer le canal de communications reliant Alice au système des télécommunications. Cette attaque peut être non intrusive (la lecture des données échangées) ou intrusive (modification des données échangées, injection des données, rejeu des données anciennes). La possibilité d'une telle attaque dépend des propriétés du canal. Par exemple, un canal sans fil est potentiellement plus vulnérable contre l'écoute passive par une tierce personne qu'un câble, qui normalement exige au minimum l'accès physique au medium.

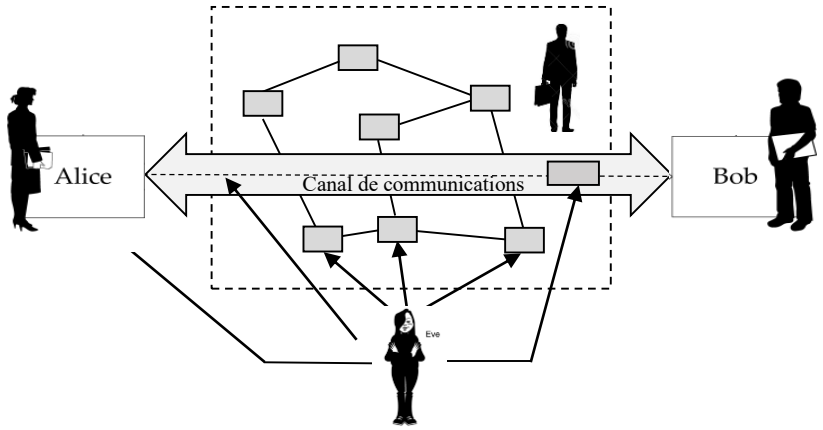


Figure 1.4 : Le modèle des menaces d'un système des télécommunications.

Une autre possibilité d'attaque contre le canal existe au sein du système des télécommunications. Pour cela, une forme d'accès au système des télécommunications est normalement nécessaire. Si Eve n'est pas le propriétaire du système, Eve peut essayer de se faire passer pour une partie légitime de l'infrastructure pour attirer Alice (ou Bob) d'utiliser ses services. Dans certains cas, Eve peut obtenir l'accès physique aux canaux de communications faisant partie du système ou utiliser des vulnérabilités pour obtenir l'accès aux composants du système. Ces formes d'accès peuvent permettre à Eve de collectionner des informations sur les communications entre Bob et Alice et de manipuler le flux des données échanges entre les deux.

Les intrusions dans l'infrastructure utilisée permettent de monter notamment des attaques du type "man-in-the middle". Dans ce scénario, Eve s'installe à la coupure du canal entre Alice (ou Bob) et l'infrastructure. Sans authentification mutuelle et fiable entre Alice (ou Bob) et l'infrastructure, Alice et l'infrastructure ne peuvent pas généralement s'apercevoir de ce genre d'intrusions. Mais les scénarios du type "man-in-the middle" sont également possibles, si Eve se fait respectivement passer pour le

communicant adversaire. Pour contrer ces attaques, une authentification mutuelle est fiable entre Alice et Bob devient alors nécessaire.

Dans chaque scénario décrit, l'attaque peut avoir un caractère d'une intrusion ou un caractère purement destructif visant l'indisponibilité (au moins temporaire) de l'élément attaqué. Un attaquant se sert typiquement d'une combinaison d'attaques destructives et ciblées pour arriver à son but.

Enfin, le canal de communications d'Alice et Bob passe typiquement par plus qu'un seul système des télécommunications, opéré par des autorités différentes, et souvent par une superposition de systèmes d'informations et d'autorités différentes. Dans chaque système traversé, toutes les menaces décrites auparavant sont imaginables. De plus, les interconnexions des systèmes introduisent des nouvelles interfaces et compliquent à nouveau la chaîne de confiance.

3.3 L'homogénéité versus l'hétérogénéité

L'hétérogénéité des systèmes d'informations est typiquement perçue comme une grande gêne au déploiement de la politique de sécurité. En effet, l'implémentation des mécanismes de sécurité dans un environnement hétérogène est naturellement plus difficile, car il faut faire extrêmement attention que les objectifs définis globalement soient atteints au niveau sous-système malgré la diversité des instanciations des mécanismes, c'est-à-dire, à travers des différents liens et connexions, des différents équipements avec des propriétés, des capacités, des vulnérabilités et des usages variés. Cela résulte normalement dans une explosion des spécifications et des conditions supplémentaires. Un tel déploiement nécessite alors une maîtrise élevée de l'ingénierie du SI cible.

De plus, en présumant, par exemple, que la probabilité de présence des vulnérabilités dans une instanciation d'une fonctionnalité est constante, l'hétérogénéité augmente les chances d'un attaquant de pouvoir trouver une vulnérabilité en multipliant nombre d'instanciations différentes.

En outre, la gestion d'une infrastructure hétérogène est également plus compliquée et engendre en pratique un ajout considérable en complexité du SI, ce qui introduit des nouvelles vulnérabilités.

Par conséquent, l'hétérogénéité est perçue comme une vulnérabilité importante d'un système d'information, plus difficile à protéger, plus facile à attaquer. Classiquement, les ingénieurs du système sont tentés d'introduire des moyens pour pallier ce problème : standardisation, centralisation, superposition *(overlay),* traduction (passerelle applicative).

Alors que c'est un argument valide pour un SI déterminé, sous le contrôle d'une autorité subissant la complexité élevée du SI, à l'échelle globale le contraire est vrai. L'homogénéité à l'échelle globale est une vulnérabilité majeure car elle expose toute vulnérabilité globalement. L'exploitation des vulnérabilités présentes devient quasi certaine, mais surtout la recherche des vulnérabilités devient une tâche attirante. L'attaquant se base alors sur un compendium des outils d'attaques potentiellement utilisable

partout. En exploitant les différences dans les cultures, les stratégies de gestion (par exemple les périodes d'application de correctif) et dans les politiques de sécurité déployées par des autorités différentes, il est pratiquement sûr de pouvoir trouver des sous-systèmes vulnérables et de pouvoir en prendre contrôle.

Le meilleur exemple pour un tel espéranto mondial est Internet avec sa suite protocolaire unique TCP/IP, conçue pour permettre l'accès partout dans le monde. Aujourd'hui Internet est devenu la plate-forme par excellence pour les attaques contre les Sis, tout le monde est accessible, tout le monde est exposé, tout le monde est standard et conforme. Avec un nombre de services communs, typiquement soutenus par les mêmes implémentations (*Bind*[8], *Apache*[9], *Sendmail/Exim*[10], *IIS*[11], etc.) tout le monde est également vulnérable. Enfin, ajoutons à cela l'homogénéité des plates-formes des utilisateurs, soulignée par le monopole de facto dans les systèmes d'opération (Windows, Internet Explorer/Firefox, etc.).

3.4 Internet

Dans un système comme Internet, interconnecté, standard, ouvert et géré par des autorités différentes (typiquement par des grands opérateurs) sous les législations différentes, les attaques sont une normalité. Elles sont de natures différentes (malicieuses, pannes, oublis, mauvaises configurations, etc.) et représentent les différentes implications, rôles et jugements des acteurs quant à cette ressource. Internet est vulnérable en tant qu'infrastructure sur les différents niveaux (routage et sa convergence dans *BGP*[12], résolution du nom par DNS, transports par IP, UDP et TCP) mais aussi en tant que somme des services rendus aux utilisateurs finaux (principalement *mail/SMTP*[13] et web/HTTP). La sécurité d'Internet aujourd'hui est un problème des niveaux multiples, il concerne la politique, la technique et l'espace personnel.

Dans la société d'information du 21ème siècle, Internet forme le dénominateur commun, il est vu de plus en plus comme une infrastructure critique par beaucoup d'États (e-government, e-voting, e-learning, source d'information, etc.). Ainsi, cette infrastructure nécessite une protection rigoureuse. Or, l'application de politique de sécurité doit être coordonnée et simultanée dans la globalité du réseau, parmi des acteurs de différents niveaux : aujourd'hui c'est une mesure, qui, surtout politiquement, est quasi impossible à accomplir. C'est un problème de cultures informatiques différentes chez des

[8] BIND (*Berkeley Internet Name Domain*) : est le serveur DNS le plus utilisé sur Internet (79 % des serveurs en 2008), spécialement sur les systèmes de type UNIX et est devenu un standard.
[9] Apache est un serveur HTTP créé et maintenu au sein de la fondation Apache. C'est le serveur HTTP le plus populaire du World Wide Web. Il est distribué selon les termes de la licence Apache.
[10] Sendmail/Exim, est un serveur de messagerie électronique dont le code source est ouvert. Il se charge de la livraison et de l'envoi de courriers électroniques.
[11] Internet Information Services, communément appelé IIS *(prononcé "2is")*, est un serveur Web (FTP, SMTP, HTTP etc.) des différents systèmes d'exploitation Windows NT.
[12] Border Gateway Protocol (BGP) est un protocole d'échange de route utilisé notamment sur le réseau Internet. Son objectif est d'échanger des informations de routage et d'accessibilité de réseaux (appelés *préfixes*) entre *Autonomous Systems* (AS).
[13] Simple Mail Transfer Protocol (SMTP) est un protocole de communication utilisé pour transférer le courrier électronique (courriel) vers les serveurs de messagerie électronique.

acteurs principaux, mais aussi un problème d'externalité : typiquement, un opérateur d'une infrastructure d'Internet n'est pas directement concerné par les attaques sortant de son réseau (à moins qu'il en soit directement responsable). Dans certains cas, il n'est même pas concerné par les attaques entrantes, car contrairement à la cible, il est souvent transparent à l'attaque (transport pur) et ses capacités sont immenses. Autrement dit, l'ajout des systèmes de contrôle et de vérification pour l'opérateur est coûteux, mais il ne représente aucun gain tant que les autres opérateurs ne réagissent pas de la même manière.

Pour les utilisateurs, Internet est probablement le plus grand danger de sécurité parmi tous les SIs disponibles. En connectant les utilisateurs (souvent ignorants) à un système d'information mondial et peu sécurisé, il les rend pour la première fois universellement accessibles et universellement vulnérables. C'est Internet qui ouvre pour un particulier la porte de la société d'information, mais il ne faut pas se tromper en pensant que cette porte est un passage à sens unique. Les applications qui utilisent les services proposés par les fournisseurs d'accès et dans Internet proviennent souvent des sources redoutables. Elles peuvent être de qualité moindre et sont en général toujours vulnérables. De plus, même en présumant une qualité suffisante des applications, il faut se douter par rapport au contenu, aux adresses affichées et aux sources auxquels on accède par le biais de ces dernières. Ainsi, les utilisateurs doivent être sensibilisés et formés. Ils doivent se poser les questions quant à la protection de leur sphère privée, de leur réputation et quant à la confidentialité de leurs données.

En revanche, pour les attaquants, Internet représente une véritable plate-forme d'appui en proposant des forums pour les échanges d'informations sur les nouvelles vulnérabilités découvertes, du code source informatique les exploitant, des listes des destinations vulnérables, des accréditations volées, etc. Il est même possible et courant de collaborer dans le développement d'un nouvel outil d'attaque.

3.5 Risques pour les infrastructures

Le propriétaire des infrastructures traitant et/ou transportant les données numérisées est principalement préoccupé par la protection de son propre investissement dans son infrastructure et la maintenance de cette dernière (visibilité intérieure). Il est donc dans un premier temps moins concerné par la protection des données que par la protection de son infrastructure et notamment par les risques suivants :

3.5.1 Accès illicites

Il s'agit de tout type d'accès non autorisé (qualitatif), ou d'accès hors contrat (quantitatif) à l'infrastructure, à ses éléments principaux ou un accès non autorisé par l'infrastructure. Nous pouvons citer comme exemples l'accès d'une personne non autorisée au réseau interne de l'entreprise, l'utilisation d'un service non contractuel, l'accès à certains contenus par le réseau d'un opérateur, etc.

L'usurpation de l'identité d'un tiers est fatale dans les systèmes qui pratiquent le contrôle d'accès basé sur les identités.

3.5.2 Espionnage de l'infrastructure

Les données sur l'infrastructure même représentent une valeur pour le propriétaire et l'attaquant, car les connaissances de telles données dévoileraient des informations potentiellement critiques (comme des faiblesses, vulnérabilités) pour son bon fonctionnement et son utilisation. La publication des statistiques diverses, ainsi que l'accès aux mesures, la divulgation des informations sur la topologie, l'échantillonnage des équipements et de leurs types sont des risques importants, puisque les données peuvent servir pour préparer des attaques. L'exemple typique dans un réseau de télécommunications est l'espionnage des services utilisables sur tous les éléments par la méthode appelée *port scan*[14].

3.5.3 Intrusions infrastructurelles

Le changement du comportement "normal" d'un élément infrastructurel est évidemment un risque pour un opérateur puisque quelqu'un s'approprie un bien qui appartient à l'opérateur. Toutefois, même un ajout illicite d'un équipement dans l'infrastructure est une intrusion et représente un grand risque pour les opérateurs des infrastructures, il permet notamment par extension de monter des attaques contre les utilisateurs ignorants ou contre des parties de l'infrastructure et collectionner des statistiques importantes. Comme exemple on peut citer le problème de faux point d'accès dans les réseaux sans fil *(rogue access point*[15]*),* ou une interconnexion d'un réseau sécurisé avec un réseau public par un terminal connecté physiquement aux deux infrastructures et créant un pont. Enfin rappelons que, une fois connecté, le terminal fait partie d'un réseau des télécommunications. Son intégrité est alors directement liée à l'intégrité de l'infrastructure. Autrement dit, du point de vue de l'opérateur, l'exécution du code malicieux (comme des virus, des chevaux de Troie, spyware, *malware*[16], etc.) représente un risque direct pour l'opérateur. En effet, une telle intrusion est une attaque réussie contre l'utilisateur qui possède le terminal, ensuite, elle peut être utilisée pour attaquer les utilisateurs dans le voisinage virtuel du réseau et, dans certains cas, attaquer l'infrastructure même (inondation, découverte, etc.).

3.5.4 Traçabilité insuffisante

Pour pouvoir détecter et comprendre les problèmes, pour retrouver le responsable en cas d'enquête, une bonne traçabilité est nécessaire. Le propriétaire de l'infrastructure doit être muni de moyens pour se protéger contre les accusations de tierces parties, par exemple suite aux attaques provenant de

[14] Port scan (le balayage de port) est une technique servant à rechercher les ports ouverts sur un serveur de réseau.
[15] Rogue access point (Un point d'accès non autorisé) est un point d'accès sans fil qui a été installé sur un réseau sécurisé sans l'autorisation d'un administrateur réseau local, qu'il s'agisse ajoutée par un employé bien intentionné ou par un utilisateur malveillant.
[16] Malware, Un logiciel malveillant parfois logiciel nuisible : est un programme développé dans le but de nuire à un système informatique, sans le consentement de l'utilisateur dont l'ordinateur est infecté.

son infrastructure. Dans certains états, la conservation de telles données peut être imposée par la loi [14]. Ainsi, les mesures et les statistiques "la métrologie" dans toute infrastructure susceptible d'être utilisée par plusieurs acteurs sont nécessaires non seulement pour des raisons de dimensionnement mais aussi pour des raisons de sécurité. Un point très important et peu pratiqué est le fait que ces observations doivent également servir les preuves pour juger l'efficacité des mesures de sécurité déployées, elles constituent ainsi une base pour l'évaluation de l'assurance de sécurité [17].

De plus, pour améliorer sa réputation (visibilité extérieure) le propriétaire veut augmenter la fiabilité des contrats de l'utilisation de son infrastructure par les tierces personnes. Pour maintenir les contrats, il doit se protéger contre tout risque rendu son infrastructure moins bien accessible par ses partenaires contractuels. Il doit également se préoccuper de la bonne maîtrise de son infrastructure.

3.5.5 Indisponibilité de l'infrastructure

La disponibilité de l'infrastructure doit être une des préoccupations principales du propriétaire, car il s'engage dans les contrats quant à son utilisation. Particulièrement dans Internet, les techniques de *déni de service*[17] ont fait d'énormes progrès. Débutant avec des attaques très naïves (inondation directe) lors de la commercialisation d'Internet, le déni de service devient intelligent et ciblé sur les vulnérabilités des systèmes (voir *ping of death*[18]), puis indirect (attaquer une machine et monter une attaque à partir de cette machine), peut utiliser sous formes distribuées (prendre la main sur plusieurs machines et monter une attaque à partir d'un cluster), souvent appelées des *botnets*[19], les *botnets* sont contrôlés à partir de plusieurs centres de contrôle qui se chargent de leur évolution, maintien et commandement. Des nouvelles capacités sont développée et intégrées régulièrement dans les réseaux déployés. Ainsi, la capacité d'attaque et les nouvelles fonctions se louent et se revendent dans un milieu très fermé cybercriminels.

3.6 Risques personnels

Cette catégorie regroupe les risques que court un utilisateur potentiel des infrastructures de SI modernes en utilisant les services accessibles par le biais de ces infrastructures, en faisant traiter ses données par ces dernières, etc.

Les principales préoccupations d'un utilisateur sont la protection de ses données (programmes, images, textes, etc.) et la protection de sa sphère privée. Ces considérations sont souvent mises à l'avant dans les débats autour de la sécurité, attribuant aux systèmes de télécommunications utilisés un rôle secondaire, et une sécurité qui résulte du besoin de sécurisation des données de l'utilisateur.

[17] Une attaque par déni de service (Denial of Service attack, d'où l'abréviation DoS) est une attaque informatique ayant pour but de rendre indisponible un service, d'empêcher les utilisateurs légitimes d'un service de l'utiliser.
[18] Ping of death (Le ping de la mort ou PoD) est une attaque historique de type déni de service réalisé par l'envoi de paquet ping malformé.
[19] Abréviation anglaise, de *robot network,* réseaux des robots.

Toutefois, nous rappelons ici la subjectivité de la sécurité et insistons sur le fait que l'importance d'une considération dépend seulement des deux critères : de la valeur que le propriétaire attache à son bien et de l'estimation des risques autour de ce bien. Dans cette optique, la protection de l'infrastructure ne peut pas être vue comme une extension logique de la protection des données.

Tout utilisateur est concerné par les risques suivants :

3.6.1 Accès aux données privées

Ce risque comprend toute lecture illicite des contenus consommés, produits, transmis, etc. Un exemple est *l'écoute des communications*[20], c'est-à-dire une lecture passive des données lors de leur transmission sur un réseau des télécommunications. Motivé par la protection de la sphère privée, également compris dans cette catégorie sont les accès aux données administratives liées au profil d'accès (telles que l'identité personnelle, la localisation, les statistiques de l'utilisation et de facturation).

3.6.2 Modification des données privées

Une modification inaperçue des données privées citées dans le paragraphe précédent est un risque, car elle permet de prendre possession des données, changer les rapports de l'utilisation, etc. Notons que, selon la technologie utilisée, la modification des données n'implique pas l'accès à ces mêmes données. Un exemple est la modification des trames chiffrées dans les réseaux sans fil de type 802.11.

3.6.3 Services imposteurs

Dans le monde numérique virtuel, l'utilisateur court le risque de se connecter à un service imposteur. Cette situation peut avoir lieu à cause de failles techniques : les exemples sont l'accès à un faux point d'accès dans un réseau sans fil, redirection sur un faux serveur web, et toute usurpation de l'identité du réseau. Aujourd'hui les services imposteurs profitent de plus en plus des combinaisons des failles techniques et des failles socio-technologiques, en utilisant par exemple des techniques telles que le *phishing*[21], en attirant les utilisateurs avec des interfaces mimant les interfaces connues et des fausses promesses, les attaquants visent à attirer les utilisateurs sur les services imposteurs. C'est seulement dans une phase suivante qu'ils essaieront d'exploiter les failles techniques éventuelles.

3.6.4 Propriétés non contractuelles de l'accès

L'utilisateur court le risque de ne pas obtenir les paramètres d'accès "spécifiés dans le contrat de service" comme la fiabilité de l'accès, le débit négocié, les heures et la durée de connexion, etc.

[20] Snooping ou Sniffing.

[21] L'hameçonnage, phishing ou filoutage, est une technique utilisée par des fraudeurs pour obtenir des renseignements personnels dans le but de perpétrer une usurpation d'identité.

3.6.5 Fragilité de la plate-forme d'exécution

C'est un des plus grands risques aujourd'hui, orthogonal aux risques discutés auparavant. Parallèlement aux informations sur la source, la destination et la protection des données discutées ci-dessus, l'utilisateur doit s'occuper de l'intégrité de la plate-forme et des programmes qu'il utilise. Si la plate-forme numérique utilisée n'est pas fiable, tout accès malveillant (par le biais d'un virus, d'un cheval de Troie et de tout type de malware) est possible aux données privées de l'utilisateur, et même sous son identité. Cette catégorie comprend également tout le spyware, utilisé (souvent illégalement) pour espionner les usages courants d'un utilisateur, son profil commercial, etc.

3.6.6 Usurpation de l'identité d'accès

C'est un risque pour les utilisateurs autorisés, puisque tout acte commis sous cette identité usurpée peut être faussement attribué à un utilisateur autorisé. Ceci permet notamment l'accès aux données privées, leur changement, etc.

4. DES VULNÉRABILITÉS FILAIRES AUX VULNÉRABILITÉS DANS LE SANS FIL

4.1 Le medium sans fil

Le médium sans fil est très vulnérable par sa nature, beaucoup plus vulnérable que le medium filaire. Le médium sans fil permet un accès libre de tout acteur : la lecture, l'injection, la suppression et la modification des données sont possibles dans la plupart des configurations. De plus toute communication est de nature purement virtuelle, en général, on ne peut ni limiter le périmètre du réseau (à cause de propriété physiques, l'affaiblissement est fort, mais la propagation multi-chemin, les réflexions/réfraction, etc., produisent souvent des résultats étonnants), ni distingues ses vis-à-vis. Autrement dit, le médium ne permet pas de limiter le cercle des acteurs impliqués dans le traitement des données envoyées. Il ne permet pas non plus de détecter si un accès au médium ou aux données a eu lieu pendant la transmission.

Pour un attaquant le medium sans fil est souvent plus attractif, car il ne nécessite pas de la présence physique de l'attaquant. Bien équipé, il est capable de monter des attaques contre les vulnérabilités naturelles du medium en restant en dehors du domaine attaqué (*parking lot attack*). De plus, les attaques peuvent être autorisées ou au moins semi-automatisées facilement. Les équipements peuvent enregistrer les trames reçues pour espionner l'infrastructure rencontrée (*wardriving*) ou même un traitement autonome a posteriori (attaque par dictionnaire, attaque par force brute), même sans

exploiter les failles éventuelles dans les contre-mesures de sécurité normalement implémentée dans ce genre de réseaux (principalement contrôle d'accès, confidentialité et intégrité).

Pour pallier les problèmes de transmission, les systèmes de gestion, les machines à état et les piles protocolaires employés dans ces réseaux exposent souvent une complexité élevée au niveau d'implémentation de la carte réseau, des pilotes, des applications dédiées, etc. Ils représentent ainsi des nouvelles vulnérabilités et donc cibles d'attaques.

4.2 Les terminaux sans fil

Les terminaux utilisés dans les réseaux sans fil sont caractérisés par leur probabilité. Ils sont alors petits, possèdent souvent une interface homme-machine (IHM) restreinte, limités au niveau des capacités de calcul et de stockage et alimentés par une batterie.

Ces caractéristiques ont un impact important sur la sécurité du terminal, et, par extension du SI les employant. L'interface IHM limitée pose souvent des problèmes dans les phases d'appairage et de contrôle d'accès (comment faire entrer un mot de passe dans un pair d'écouteurs, comment établir une identité unique d'une clé USB, etc.). Les capacités de calcul (CPU) et stockage (mémoire, disque) limitées introduisent des contraintes quant aux calculs possibles. Il est par exemple assez ardu de vouloir faire du calcul aux clés privées dans un équipement embarqué comme un capteur ou même un téléphone portable sans un module dédié.

L'alimentation par la batterie provoque aussi bien un changement de comportement (On/Off inattendu, techniquement proche de la mobilité) et nécessite de système de gestion supplémentaire (gestion de veille, des mécanismes de paging, etc.). De plus, le développement des technologies de batterie est constant mais linéaire, il ne peut pas suivre la vitesse du développement exponentiel de la micro-électronique (loi de Moore). On parle alors souvent d'un phénomène nommé *Wireless Security gap*. Pour pallier ce problème, une prudence et une qualité de travail élevées sont exigées lors de la conception des circuits et des systèmes compliqués adaptatifs sont employés, compliquant le terminal et le rendant ainsi potentiellement plus vulnérable.

4.3 Nouveaux services

Au-delà de ces aspects, les réseaux basés sur la transmission radio ajoutent un degré de liberté à toute une transmission, le contexte spatio-temporel. Il est alors raisonnable de parler de la mobilité, du nomadisme et de la localisation des utilisateurs connectés par ce medium. Ces nouvelles libertés justifient des services de la mobilité ou encore de la localisation (*location-based service*, etc.).

La mobilité représente en effet un problème connu pour la sécurité, car elle introduit non seulement des nouveaux mécanismes et sous-systèmes et donc une nouvelle complexité mais surtout la présence potentielle de plusieurs domaines d'autorité. Cet aspect complique considérablement les chaînes de

confiance. Il est souvent nécessaire d'offrir le service approprié à des utilisateurs de provenance d'un autre domaine, soumis à une autre politique de sécurité. Or, en général il est difficile de comparer deux politiques de sécurité. Mais même l'accueil des utilisateurs mobiles appartenant au domaine est compliqué : on doit en effet vérifier que, après s'être absenté de leur domaine d'origine, leur configuration est toujours conforme à la politique de sécurité du domaine. Dans la pratique, cela nécessite souvent des mesures draconiennes au niveau d'attribution des droits locaux sur le terminal d'utilisateur ou encore une mise en *quarantaine* du poste. De ce fait, les systèmes mobiles sont normalement plus vulnérables, et du point de vue de l'utilisateur et du point de vue du réseau. Il est ardu de répondre à toute exigence de sécurité dans le sens *CIA*, mais encore plus difficile de s'assurer de la non-répudiation, d'une traçabilité suffisante (par exemple pour la facturation) mais sans abus (anonymat, respect de la sphère privée), de l'absence des virus, de malware sur le poste.

Donc la sécurité de la mobilité doit être traitée avec une prudence élevée. Le problème c'est que les mécanismes de sécurité interviennent souvent en même temps que les mécanismes typiques de mobilité. Ces mécanismes s'intercalent et prolongent les délais, ils deviennent alors critiques pour la performance du SI résultant. Des compromis sont souvent requis pour arriver à des résultats acceptables pour le service fourni.

5. CONCLUSION

La sécurité constitue un problème essentiel dans les systèmes informatiques modernes. Il se posera certainement de manière encore plus importante dans les technologies du futur (réseaux autonomes, 4G, etc.). La démocratisation des technologies de l'information et des communications, matérialisée par l'interconnexion de divers systèmes (sans fil, autonomes ou autres), rend la protection des données et des infrastructures considérablement plus complexes.

Malgré les problèmes de sécurité intrinsèque, les réseaux sans fil continueront à se développer dans plusieurs marchés verticaux comme les télécommunications, les applications industrielles et le *M2M* (machine-to-machine) et la domotique. Il est important de bien connaître les difficultés liées à la mise en place de ce type de réseaux, mais aussi les nouvelles opportunités qu'ils proposent et les particularités de provisionnement du service dans ce nouveau monde interconnecté et communicant.

Incapable de trouver un bon compromis entre la sécurité et son coût pour les services requis par leurs propres moyens, les entreprises et les particuliers deviennent plus exigeants sur les garanties de sécurité que leur apportent les fournisseurs des services. La sécurité apparaît donc comme l'un des enjeux majeurs pour la commercialisation des services et des produits dans le domaine des SIs et dépasse les dimensions purement techniques. Aujourd'hui, la sécurité concerne tous les acteurs impliqués (opérateurs des réseaux, fournisseurs des services, intégrateurs des systèmes, utilisateurs,

État, etc.). La législation, les industriels, les académiques et les utilisateurs sont appelés à collaborer pour développer des meilleures méthodologies pour les processus de protection.

Une sensibilité accrue aux problématiques de sécurité se reflète aujourd'hui dans les débats politiques, le marketing des produits et les demandes des clients. En même temps, une panoplie de travaux est menée par les académiques, les industriels et dans le cadre de consortiums et d'organismes de normalisation afin d'apporter des améliorations et de définir des solutions plus robustes.

En revanche, il faut comprendre qu'il ne peut pas y voir de sécurité standardisée, suffisante pour tout le monde. Cela est dû à l'appréciation très différente des risques autour d'un même bien dans un environnement donné mais surtout à cause de la complexité croissante des SIs. De plus le développement constant des nouveaux services et de nouveaux produits amène des nouvelles vulnérabilités, dont la gravité ne peut être mesurée à l'avance, car elle dépend parmi d'autre de l'échelle du déploiement. La sécurité reste un processus qui doit accompagner le développement d'un système d'information. Le monde devenant plus connecté et plus communicant, les problèmes de sécurité vont probablement s'aggraver dans le futur.

Chapitre 2

Les réseaux mobiles Ad hoc

1. INTRODUCTION

L'évolution récente de la technologie dans le domaine de la communication sans fil et l'apparition des unités de calculs portables, poussent les chercheurs à faire des efforts pour mieux assurer la fonction des réseaux, à savoir l'accès rapide à l'information indépendamment du lieu et du temps.

Dans un passé proche, les réseaux sans fil se basaient exclusivement sur des infrastructures planifiées et dimensionnées ainsi que sur un contrôle hiérarchique des opérations (Figure 2.1). Avec la croissance importante qu'ont connue les applications sans fil, en particulier dans les réseaux personnels et locaux, des besoins en termes d'auto-organisation, d'indépendance, d'adaptabilité et de réduction de coûts se sont fait ressentir.

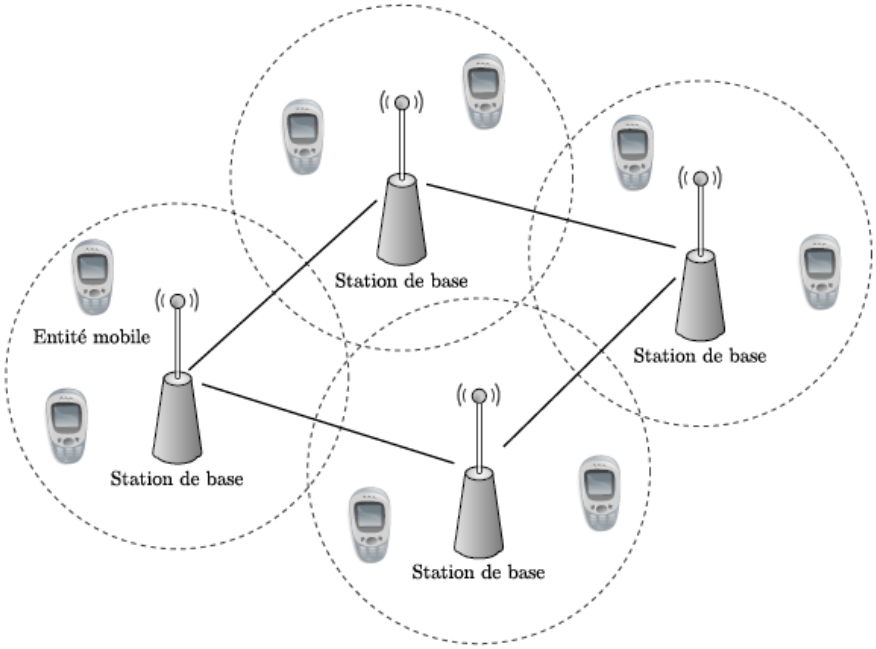


Figure 2.1 : Réseau avec infrastructure fixe.

Une solution pour prendre en charge ce type d'exigences est de considérer un réseau sans fil dont la mobilité n'est plus seulement liée aux utilisateurs mais directement à l'infrastructure elle-même. Un tel réseau est composé non plus de points d'accès fixes mais au contraire, d'entités entièrement mobiles. Ces entités (aussi appelées nœuds), communiquent entre elles par le biais d'ondes radio et lorsque deux d'entre elles sont trop éloignées pour communiquer directement, elles utilisent d'autres nœuds qui sont chargés de relayer les paquets depuis l'émetteur jusqu'au destinataire (Figure 2.2).

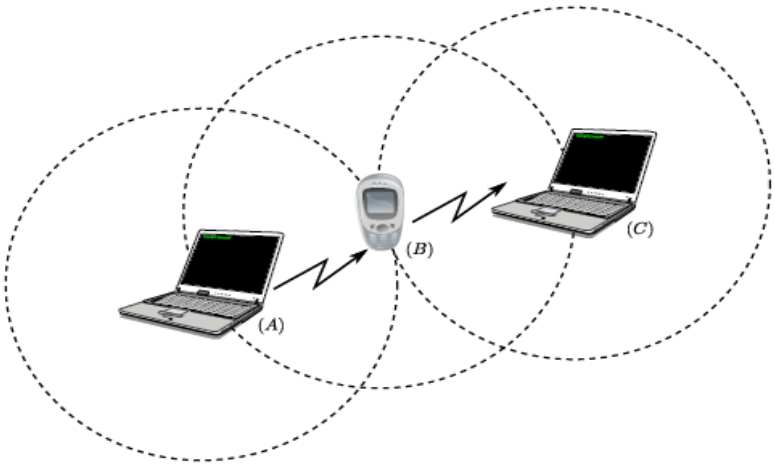


Figure 2.2 : Réseau Ad hoc avec routage multi saut.

Ce concept peut être vu comme un réseau à architecture plate, dans le sens où toutes les entités communicantes sont équivalentes et peuvent en fonction des besoins, être utilisées tantôt en tant que clients (pour émettre ou recevoir des paquets) tantôt comme des routeurs (pour relayer des paquets entre deux autres nœuds). De tels réseaux sont appelés des réseaux *Ad hoc,* c'est-à-dire des réseaux dédiés spécifiquement à un environnement sans infrastructure fixe. Pour s'adapter à l'absence de routeur fixe, des protocoles de routage spécifiques doivent être employés, c'est la raison pour laquelle les réseaux *Ad hoc* (aussi appelés *MANET* pour *Mobile Ad hoc NETworks)* utilisent un routage multi saut pour acheminer les paquets. De par la forte mobilité des entités, le protocole de routage doit également s'adapter aux fluctuations importantes de la connectivité, des liens peuvent apparaître ou disparaître à tout moment, devenir successivement unidirectionnels, bidirectionnels et offrir des capacités disparates. De par ces différentes caractéristiques (hétérogénéité des liens, non-pérennité des entités, protocoles de routage entièrement distribués).

Les avantages des réseaux *Ad hoc* sont multiples, ils permettent la mise en place de réseaux dont les nœuds sont capables en quelques instants, avec très peu d'intervention humaine et à moindre cout, d'initier une communication et échanger des informations.

2. RESÉAUX MOBILES AD HOC

2.1 Définition

Ad hoc est un terme issu du latin, qui signifie littéralement : "*qui va vers ce vers quoi il doit aller*" [18]. Il est souvent utilisé pour décrire des solutions qui sont développées et/ou configurées à la volée pour répondre à un but spécifique.

Dans le contexte des réseaux informatiques, les réseaux Ad hoc auxquels nous nous sommes intéressés sont ceux décrits et étudiés par le groupe de travail *MANET* de l'*IETF*[1]. Une définition de ces réseaux est donnée formellement dans la *RFC 2501* [19]. Il s'agit de réseaux sans fil auto-adaptatif. Ils sont capables de s'organiser spontanément et de manière autonome dans l'environnement dans lequel ils sont déployés, et ce, sans intervention humaine ou une quelconque infrastructure dédiée. Ainsi, afin de s'affranchir de la nécessité d'un câblage long et fastidieux, les entités qui les composent communiquent les unes les autres par voie aérienne au moyen d'une interface radio. De plus, contrairement aux réseaux filaires dans lesquels toutes les opérations fondamentales de découverte et de maintenance de l'infrastructure de communication sont assurées par des équipements dédiés tels que des routeurs ou des stations de base sans fil (points d'accès "AP"), dans un réseau Ad hoc, ces tâches sont réparties sur l'ensemble des entités. Cette décentralisation impose l'élaboration d'algorithmes totalement distribués. Ainsi, chaque entité gère ses communications à partir des informations dont elle dispose sur l'état du réseau et sur son état interne, ceci dans le but de faire émerger, à l'échelle du réseau, une structure de communication cohérente, c'est-à-dire la formation de chemins valides.

Les réseaux Ad hoc sont divers, nous pouvons en citer quelques-uns :

- ✓ Les réseaux poste-à-poste (*Peer-to-Peer*) sont des réseaux dont le fonctionnement est décentralisé entre les différents utilisateurs du réseau, dont les machines sont simultanément clients et serveurs et aussi routeur, en passant les messages et les données vers leur(s) destinataire(s).
- ✓ Les réseaux personnels : PAN (*Personal Area Network*) désigne un réseau restreint d'équipements informatiques habituellement utilisés dans le cadre d'une utilisation personnelle. Parmi les technologies sans fil utilisées par les réseaux PAN, nous pouvons citer le Bluetooth[2], l'infrarouge[3], ou le Zig-Bee[4].
- ✓ Les réseaux de capteurs [20] sont des réseaux composés de nœuds intégrant :
  - Une unité de mesure chargée de capter des grandeurs physiques (chaleur, humidité, vibrations) et de les transformer en grandeurs numériques.
  - Une unité de traitement informatique et de stockage de données.
  - Un module de transmission sans fil (Wireless).
- ✓ Les réseaux de voitures : les voitures de nos jours embarquent de plus en plus de technologie et ont de plus en plus besoin de communiquer avec l'extérieur. Les voitures équipées par des capteurs

[1] Internet Engineering Task Force.
[2] Bluetooth est un standard de communication permettant l'échange bidirectionnel de données à très courte distance et utilisant des ondes radio UHF. Son objectif est de simplifier les connexions entre les appareils électroniques en supprimant des liaisons filaires.
[3] Egalement connu sous le sigle IrDA, permet de transférer des fichiers avec le rayonnement infrarouge.
[4] Zig-Bee est un protocole de haut niveau permettant la communication de petites radios, à consommation réduite, basée sur la norme IEEE 802.15.4 pour les réseaux à dimension personnelle.

dans les toits et/ou les pare-chocs sont capables de créer des plateformes de réseaux mobile Ad hoc et de relier en réseau les automobiles passant à proximité les unes des autres.

Les réseaux Ad hoc peuvent également être connectés au monde filaire (Figure 2.3) par l'intermédiaire d'une ou plusieurs passerelles, que nous appellerons, 'en référence au monde cellulaire IP', des points d'accès (AP). De tels réseaux sont communément appelés réseaux hybrides [21]. Chaque terminal du réseau Ad hoc, s'il possède une double interface filaire et sans fil peut donc agir en tant que passerelle pour les autres clients de la bulle Ad hoc.

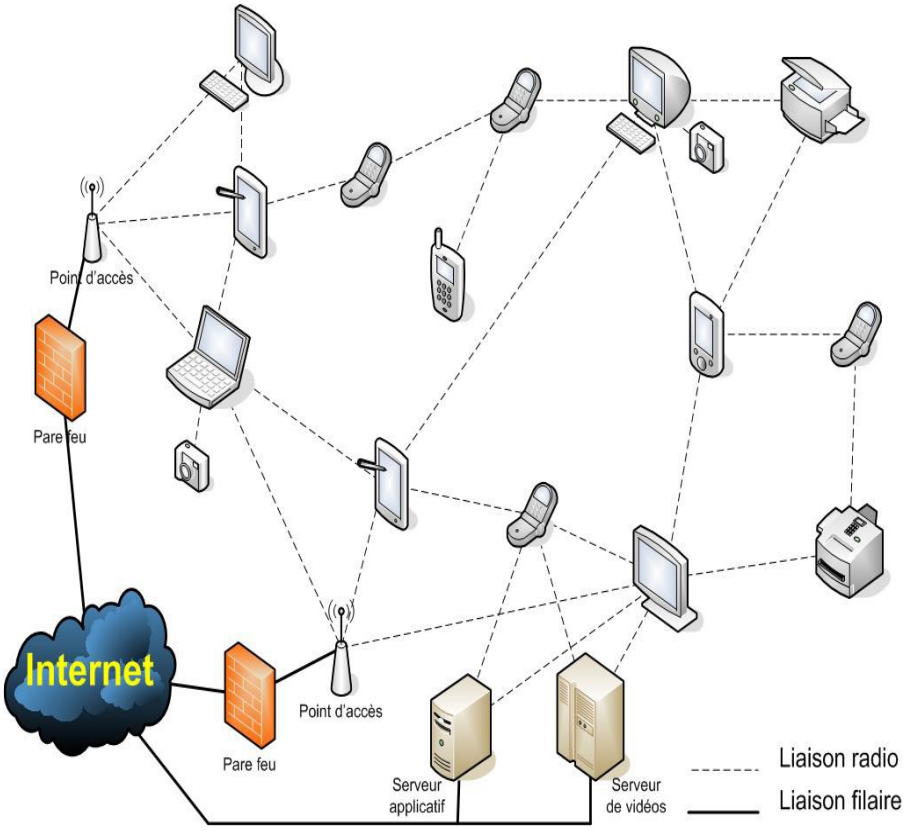


Figure 2.3 : Un réseau Ad hoc.

2.2 *Caractéristiques des réseaux mobiles Ad hoc*

Les réseaux mobiles Ad hoc héritent des mêmes propriétés et problèmes liés aux réseaux sans fil. Particulièrement, le fait que le canal radio soit limité en termes de capacité, plus exposé aux pertes (comparé au médium filaire) et sujet à des variations dans le temps. Le canal est confronté aux problèmes de "station cachée" et "station exposée". En outre, les liens sans fil sont asymétriques et pas sécurisés.

D'autres caractéristiques spécifiques aux réseaux Ad hoc conduisent à ajouter une complexité et des contraintes supplémentaires qui doivent être prises en compte lors de la conception des algorithmes et des protocoles réseaux, à savoir :

2.2.1 L'absence d'une infrastructure centralisée

En l'absence de toutes entités fixe, il devient difficile de mettre en place une infrastructure à clé publique classique avec la définition d'une autorité de certification centralisée. D'autres part, dans le cas des systèmes de détection d'intrusion, cela pose le problème de la supervision du réseau, le trafic est entièrement distribué. Enfin, cela pose également le problème crucial de la synchronisation des nœuds, en l'absence de système de guidage par satellite, il devient très délicat de synchroniser les nœuds entre eux sur une même horloge. Cette fonctionnalité est pourtant cruciale pour vérifier la fraîcheur des messages dans certains protocoles.

2.2.2 Topologie dynamique

Les entités du réseau sont potentiellement libres de se déplacer indépendamment les unes des autres. La conséquence est que la topologie du réseau tend à changer rapidement, à n'importe quel moment, et de manière imprévisible. De plus, le fait qu'une entité quitte un groupe de communication est considéré comme un état normal qui ne doit pas perturber les autres participants.

2.2.3 Canal de communication sans fil

Généralement, les liaisons sans fil offrent significativement moins de capacité que les liaisons câblées. De plus, le débit obtenu sur une liaison de communication sans fil, (après prise en considération des effets liés à la régulation des accès au média et les phénomènes d'atténuation du signal, de bruit, ou d'interférence), est sensiblement inférieur à son débit théorique maximal que permet une liaison radio. De ce fait, la bande passante réservée à un nœud est faible, et la congestion des liaisons est courante dans ces réseaux. Aussi, les attaques par déni de service ont elles plus d'impact dans les réseaux Ad hoc, puisque la bande passante disponible est aisément saturée.

2.2.4 Ressources limitées

Les entités envisagées dans ces réseaux sont des terminaux légers et de taille réduite qui fonctionnent sur batterie. Par ailleurs, les capacités de calcul, de mémoire et de stockage sont plus restreintes que dans les réseaux filaires. Par conséquent, une préoccupation majeure pour ces entités est l'économie d'énergie, car elle permet d'augmenter leur durée de vie dans le réseau. D'autre part, les coûts en termes de calcul des opérations influent sur la réactivité du système et doivent de ce fait également être pris en considération lors de la conception de solutions.

2.2.5 *La taille des réseaux Ad hoc*

Elle est souvent de petite ou moyenne taille (une centaine de nœuds), le réseau est utilisé pour étendre temporairement un réseau filaire, comme pour une conférence ou des situations où le déploiement du réseau fixe n'est pas approprié (ex : catastrophes naturelles). Cependant, quelques applications des réseaux Ad hoc nécessitent une utilisation allant jusqu'à des dizaines de milliers de nœuds, comme dans les réseaux de senseurs [20]. Des problèmes liés au passage à l'échelle tels que : l'adressage, le routage, la gestion de la localisation des senseurs et la configuration du réseau, la sécurité, etc., doivent être résolus pour une meilleure gestion du réseau.

2.2.6 *La faible sécurité*

Il est facile d'espionner un canal radio de manière passive. Les protections ne pouvant pas se faire de manière physique (il est en général difficile d'empêcher quelqu'un de placer discrètement une antenne réceptrice très sensible dans le voisinage), elles devront être mises en place de manière logique, avec de la cryptographie ou éventuellement des antennes très directionnelles. Mais le canal radio restera quoiqu'il en soit vulnérable à un brouillage massif (attaque de type Déni de service).

Les problématiques de la sécurité dans les réseaux Ad hoc sont donc très complexes, puisque l'on cherche à autoriser de nouveaux mobiles à participer au réseau, tout en évitant des nœuds "malins" qui détourneraient ou perturberaient le fonctionnement même du routage.

2.2.7 *La qualité de service (QoS)*

De nombreuses applications ont besoin de certaines garanties relatives par exemple au débit, au délai ou encore à la gigue. Dans ces réseaux Ad hoc, ces garanties sont très difficiles à obtenir. Ceci est dû à la nature du canal radio d'une part (interférences et taux d'erreur élevés) et au fait que des "liens" entre des mobiles peuvent avoir à se partager les ressources (alors qu'en filaire, deux liens sont par définition indépendants). De ce fait, les protocoles de qualité de service habituels (par exemple IntServ[5] / RSVP[6] ou Diff-Serv[7]) ne sont pas utilisables directement dans le monde Ad hoc et des solutions spécifiques doivent être proposées [22, 23].

[5] Le modèle IntServ (Integrated Services) définit une architecture capable de prendre en charge la Qualité de service en définissant des mécanismes de contrôle complémentaires sans toucher au fonctionnement IP.
[6] Resource ReSerVation Protocol (RSVP) est un protocole de la couche transport du modèle OSI, permettant de réserver des ressources dans un réseau informatique.
[7] DiffServ (Differentiated Services) est une architecture de réseau qui spécifie un mécanisme pour classer et contrôler le trafic tout en fournissant de la qualité de service, en différenciant les services des données.

2.3 *Applications*

Les motivations qui conduisent à développer des applications qui vont reposer sur les réseaux Ad hoc peuvent être physiques (impossibilité de déployer une infrastructure filaire) ou encore économiques.

L'origine des réseaux Ad hoc remonte au début des années 1970 avec le projet ALOHA[8] [24]. Commandité par l'université d'Hawaï, ce projet a été conduit dans le but de permettre aux ordinateurs des îles d'Hawaï d'être reliés entre eux par le biais d'ondes radio, et ce, dans un système de communication à un saut. Directement inspiré d'ALOHA, l'agence militaire états-unienne DARPA[9] commandita en 1973 le projet PRNet[10] [24] afin d'étudier les communications radio en mode paquet dans les réseaux autonomes. Le but est la construction d'un réseau capable de s'adapter dynamiquement aux changements topologiques afin de réagir à la mobilité des équipements d'une part, et de faire face à d'éventuelles pannes ou destructions, d'autre part. Dans PRNet, le protocole proposé permet le déploiement d'une infrastructure de communication multi sauts entre différentes unités sur un champ de bataille, en passant par l'intermédiaire de véhicules communiquant par liaison radio.

Si les premiers travaux dans le contexte des réseaux Ad hoc ont été menés dans une optique militaire, c'est en partie dû à leur mode de fonctionnement particulièrement bien adapté aux environnements hostiles et mobiles. Puisque le réseau est auto-organisé et qu'il ne requiert la présence d'aucune infrastructure, il peut être déployé rapidement et avec très peu d'intervention humaine dans n'importe qu'elle situation. Ainsi, les communications entre fantassins, véhicules et engins aérodynamiques deviennent autonomes et spontanées.

Le recours aux réseaux Ad hoc se justifiant dès lors que l'installation d'une infrastructure s'avère inappropriée, que ce soit pour des raisons de temps, des raisons économiques ou pour des raisons physiques (par exemple pour des zones géographiques dévastées ou à accès difficiles), de nouveaux champs d'applications ont récemment été envisagés.

Ils peuvent être utilisés pour la mise en communication d'unités de secours, lorsqu'une catastrophe naturelle (telle qu'un tremblement de terre, une inondation) a détruit les infrastructures de télécommunications et que l'établissement d'une liaison satellite pour chaque entité en communication représente un coût trop élevé.

Un autre domaine d'application particulièrement bien adapté pour les réseaux Ad hoc concerne les réseaux de capteurs [25]. De tels réseaux sont constitués d'équipements de taille réduite qui embarquent un système de communication sans fil et qui sont caractérisés par des ressources très limitées en termes de mémoire, de capacité calcul, de bande passante et de portée radio. Ces équipements, disséminés par

[8] Areal Locations of Hazardous Atmospheres.
[9] Defense Advanced Research Agency.
[10] Packet Radio NETwork.

centaines voire par milliers dans un environnement potentiellement hostile ou inaccessible, sont chargés de mesurer certaines propriétés physiques (telles que la température, la pression, la lumière, les sons, etc.) et de les transmettre de manière autonome. En s'affranchissant de toute limitation physique imposée par une infrastructure filaire, les réseaux Ad hoc permettent la mise en communication de tous ces équipements et facilitent ainsi la transmission des informations mesurées vers un point de collecte.

Outre les applications scientifiques et militaires, des applications civiles ont commencé à tirer profit des caractéristiques des réseaux Ad hoc. Ils peuvent être utilisés pour la mise en place instantanée d'un réseau reliant plusieurs ordinateurs entre eux. Ils s'avèrent particulièrement utiles lors de l'organisation d'évènements tels que des colloques, des salons afin de proposer un réseau de partage de l'information.

Un autre domaine de recherche concerne l'utilisation des réseaux Ad hoc pour l'établissement de communications inter-véhicules [26]. Les objectifs envisagés par un tel usage sont, entre autres, de permettre aux véhicules d'obtenir des informations locales sur les conditions de circulation (alertes d'accidents, bulletins sur les risques d'encombrement), des informations touristiques, une téléphonie entre véhicules, et un accès à l'Internet. Le recours à un réseau Ad hoc se justifie ici par ses faibles coûts de déploiement, car il permet d'éviter les investissements financiers importants que peut représenter la mise en place d'une infrastructure dédiée sur l'ensemble du réseau autoroutier.

L'informatique ambiante ou ubiquitaire prévoit, selon la définition donnée par *Weiser* [27], une omniprésence de l'informatique dans notre quotidien à tel point qu'elle deviendrait invisible. Les réseaux Ad hoc peuvent servir pour la mise en relation instantanée et transparente des équipements informatiques présents dans un environnement et rendent alors quasi réalité cette idée. Nous pouvons donner en exemple l'informatique vestimentaire [28] où chaque utilisateur transporte sur lui des ordinateurs qui ont la capacité de communiquer entre eux ou avec l'environnement.

Enfin, de plus en plus d'applications motivées par des aspects purement économiques ont recours aux réseaux Ad hoc. Nous pouvons citer en exemple le cas des réseaux citoyens où le but est de permettre aux habitants d'une même ville de communiquer entre eux grâce à un réseau libre d'accès, et ce, sans obligation de contrôle ou de gestion par un quelconque opérateur. Ce type de réseau a suscité un fort engouement ces dernières années, ce qui s'est traduit par l'apparition d'associations dans de grandes agglomérations telles que, Bruxelles [29], Lille [30], etc.

Comme nous pouvons le constater, les réseaux Ad hoc sont promis à un large spectre d'applications, qu'elles soient civiles ou militaires. Cependant, de nombreux défis se posent avant que les réseaux Ad hoc puissent être réellement utilisés. Parmi ces défis, nous pouvons citer la conception de protocoles de routage et la mise en place d'architectures de sécurité adaptées à l'environnement Ad hoc.

3. DESCRIPTION DE LA COUCHE MAC IEEE 802.11

Le protocole *MAC IEEE 802.11* est la technologie d'accès au canal que nous avons utilisée dans notre thèse. Il existe deux modes de fonctionnement de ce protocole, le mode *PCF* (*Point Coordination Function*) et le mode *DCF* (*Distributed Coordination Function*). Le mode *PCF* est utilisé pour supporter les trafics synchrones tels que les trafics en temps réel. En général, ce mode est pris en considération dans le cas des réseaux avec infrastructure, car un point d'accès est nécessaire. Le mode *DCF* est utilisé par les réseaux mobiles Ad hoc. Nous ne nous focalisons que sur le mode *DCF* qui se base sur l'utilisation de *CSMA/CA*[11] pour assurer la transmission asynchrone des données.

Le principe de fonctionnement du *DCF* consiste à écouter le canal de communication pour détecter si le canal est libre (*IDLE*) ou bien si un autre nœud est en train d'émettre. Avant chaque transmission, le nœud doit vérifier que le canal est libre pour une certaine durée appelée *DIFS*[12]. Dans le cas où le canal est occupé, la transmission est différée d'un certain temps appelé *Back-off* qui est choisi de manière aléatoire dans une fenêtre de contention (*Contention Window*). La valeur du *Back-off* n'est décrémentée que si le canal est libre. Mais, une collision peut apparaitre, si au moins deux stations transmettent en même temps, cependant la station émettrice n'a pas la possibilité de détecter cette collision. Ainsi, un mécanisme d'acquittement (*ACK*) est nécessaire pour informer la station émettrice de la bonne réception du paquet.

De plus, pour éviter le problème des stations cachées et réduire le nombre de collisions, le mécanisme *RTS/CTS*[13] est utilisé. Avec ce mécanisme, la station émettrice envoie un paquet de contrôle *RTS* dans le but d'informer les stations voisines de son souhait de transmettre ainsi que de la durée de transmission appelée *NAV(RTS).* Une fois que la station réceptrice reçoit le paquet *RTS* correctement, elle va répondre avec un autre paquet de contrôle *CTS* dans le but d'informer ses voisins de son état de réception pendant une durée *NAV(CTS).* Toutes les stations qui reçoivent soit le paquet de contrôle *RTS*, soit le *CTS* doivent bloquer leur transmission pendant *NAV(RTS)* ou *NAV(CTS)* respectivement.

Lorsque la station émettrice reçoit le paquet *CTS*, elle en déduit que son paquet *RTS* a bien été reçu par la station réceptrice et donc qu'elle a bien réservé le canal pour transmettre. Elle va donc lancer la transmission du paquet *DATA*. Si la station réceptrice reçoit le paquet *DATA* avec succès, elle va répondre avec un acquittement pour informer la station émettrice de la bonne réception du paquet *DATA*. Ce mécanisme d'accès canal dans le mode *DCF* est décrit dans la figure 2.4.

[11] Carrier Sense Multiple Access with Collision Avoidance.
[12] Distributed Inter Frame Space.
[13] Request-To-Send/Clear-To-Send.

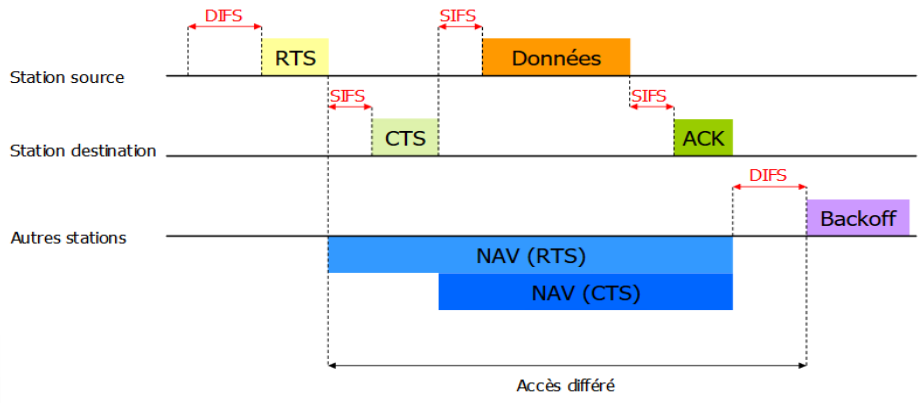


Figure 2.4 : Méthode d'accès au canal avec le mode DCF.

4. LE ROUTAGE DANS LES RESÉAUX MOBILES AD HOC

Afin de permettre les communications multi-sauts entre des nœuds hors de portée de transmission, une des fonctions fondamentales dans les réseaux Ad hoc est le *routage*. C'est un mécanisme qui sert à trouver et maintenir des chemins, ceci dans le but de permettre, à n'importe quel moment, l'établissement d'une communication entre une paire de nœuds distants. Il fonctionne selon deux phases distinctes : une phase de signalisation assurée par des échanges de messages de contrôle afin de permettre la construction et le maintien de chemins, et une phase d'acheminement des paquets de données de bout en bout. Au regard de la phase d'acheminement, les paquets de données sont relayés par chaque nœud intermédiaire appartenant au chemin établi vers la destination. En l'absence d'équipement dédié, toutes ces opérations sont supportées par l'ensemble des nœuds qui forme le réseau Ad hoc.

En raison des caractéristiques particulières des réseaux Ad hoc, les protocoles de routage classiques ne peuvent être utilisés tels quels dans ce contexte. En effet, les protocoles doivent ici prendre en compte la forte mobilité des nœuds et l'absence de routeur préconfiguré. Les approches utilisées sont classiques : inondations, routage par vecteur de distance, par état de lien.

Selon la manière dont les nœuds établissent les chemins, les protocoles de routage Ad hoc sont classés en trois catégories : les protocoles réactifs, proactifs, et hybrides. Pour cette dernière classe, il s'agit essentiellement d'une combinaison des protocoles proactifs et réactifs afin de tirer parti des avantages de chacun d'eux. Dans les trois sous-sections qui suivent, nous présentons chacune de ces catégories et fournissons des exemples représentatifs de protocoles.

4.1 Protocoles de routage Ad hoc proactifs

Les protocoles de routage proactifs se basent sur l'établissement de routes à l'avance. Les nœuds mettent à jour périodiquement les données de routage de façon à obtenir en permanence le plus court chemin (calculé en termes du nombre de nœuds intermédiaires, aussi appelé nombre de sauts) vers tous les nœuds du réseau. Ainsi, si un nœud désire transmettre un paquet vers une destination, il consulte sa table de routage qui lui indique immédiatement le chemin à suivre. Il existe deux approches pour ce type de protocoles :

- L'approche vecteur de distance où chaque nœud diffuse les distances qui le séparent de tous les autres nœuds du réseau.
- L'approche à état des liens où il s'agit de diffuser des descriptions des liens avec les nœuds voisins.

Dans ce qui suit, nous détaillons ces approches par l'intermédiaire de trois exemples de protocoles proactifs de routage Ad hoc : *DSDV* [31], *OLSR* [32] et *TBRPF* [33].

4.1.1 Le protocole DSDV

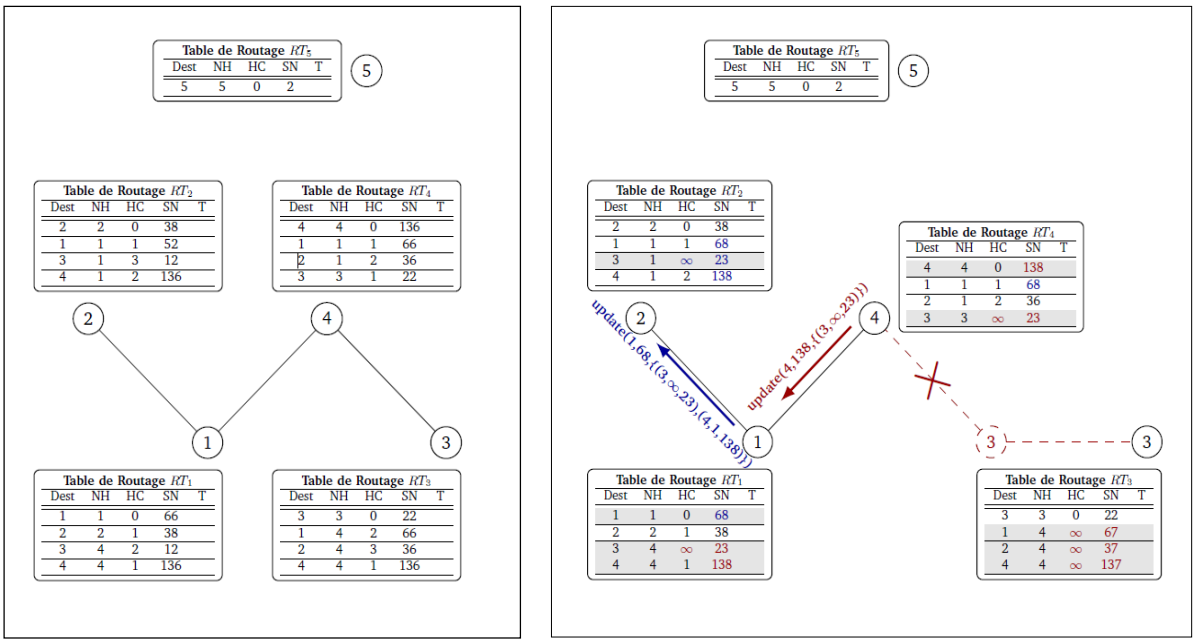
DSDV (*Destination-Sequenced Distance Vector*) [31], est l'un des premiers protocoles de routage Ad hoc proactifs à vecteur de distance. Il se base sur l'algorithme distribué Bellman-Ford ou DBF[14] [34] qui a été modifié pour s'adapter aux réseaux Ad hoc.

Comme il s'agit d'un protocole proactif, chaque nœud a une vision complète du réseau, à chaque instant. Pour ce faire, chaque nœud récupère les distances le séparant de chaque autre nœud du réseau et ne garde que le plus court chemin. Ceci est fait grâce à des échanges périodiques d'informations sur leurs tables de routage respectives. Ces échanges sont classés en deux types :

✓ Les mises à jour incrémentales (incremental updates) pour lesquelles seules les données qui ont subi des modifications depuis la dernière mise à jour sont envoyées. Un exemple est présenté dans la figure 2.5{a} [35].

Suite au déplacement du nœud 3 qui n'est plus à portée radio, le nœud 4 initie une procédure de mise à jour (update) qui ne concerne que l'entrée correspondant au nœud 3 dans sa table de routage (Figure 2.5{b}) [35]. Chaque nœud recevant ce message et le transfère en incluant les entrées qui viennent d'être modifiées. C'est le cas du nœud 1 qui initialise une mise à jour suite à la réception de celle du nœud 4.

[14] Distributed Bellman-Ford.



(a) Après échanges de messages de mises à jour, les tables de routage se sont stabilisées.

(b) Propagation du paquet et répercussion sur les tables de routage.

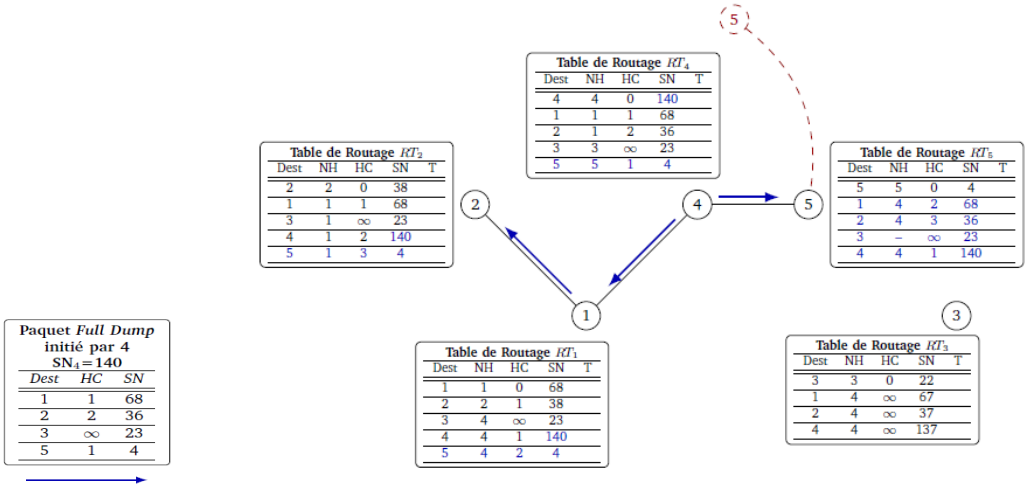
Figure 2.5 : Mise à jour incrémentale.

✓ Les mises à jour complètes (*full dump*) pour lesquelles la totalité de la table de routage est envoyée. La figure 2.6 [35] montre un exemple de cette procédure où le nœud 4 envoie la totalité de sa table de routage à tous les nœuds du réseau ce qui induit des changements au niveau de leurs tables de routage.

Outre son adresse et son propre numéro de séquence, chaque paquet de mise à jour doit contenir une liste des routes ajoutées/modifiées pour laquelle chaque entrée est un triplet formé par : l'adresse de la destination "$Dest$", le nombre de sauts "$Hop -_{Count}$" pour l'atteindre et le dernier numéro de séquence connu associé à cette destination "$Sequence\ Number$" qui permet notamment de distinguer les nouvelles routes des anciennes et évite ainsi la formation de boucles de routage. La figure 2.6{a} montre un exemple de ce paquet de mise à jour.

Pour gérer la mobilité des nœuds, *DSDV* associe à chaque nœud un minuteur (*timer*) qui est mis à jour à la valeur maximale à chaque fois qu'un message est reçu du voisin, c'est un indicateur de validité du lien. Ainsi, lorsque ce minuteur expire, le nœud considère que le voisin en question n'est plus à porter radio et que le lien est rompu. La détection d'un lien rompu se traduit au niveau de l'entrée correspondante dans la table de routage par l'assignement de la valeur "∞" au nombre de sauts (en pratique, il s'agit de n'importe quelle valeur supérieure au maximum autorisé) et l'incrémentation du

numéro de séquence au prochain numéro impair[15]. Toutes les routes utilisant ce nœud qui n'est plus joignable sont aussi mises à jour comme étant des routes invalides. Ces changements sont envoyés en priorité à tous les voisins en utilisant un paquet de mise à jour. Il est à noter que c'est le seul cas où un nœud autre que la destination pourra changer le numéro de séquence de la destination qui n'est plus joignable (Figure 2.6{b}, cas des nœuds 3 et 4).



(a) Paquet de mise à jour émis par le nœud 4.

(b) Propagation du paquet et répercussion sur les tables de routage.

Figure 2.6 : Mise à jour complète (full dump).

À la réception d'un paquet de mise à jour, les routes avec les plus grands numéros de séquences sont privilégiées pour le choix des routes, puisque cela signifie une route plus fraîche. Dans le cas de numéros de séquences égaux, le plus court chemin est retenu en se basant sur le nombre de saut.

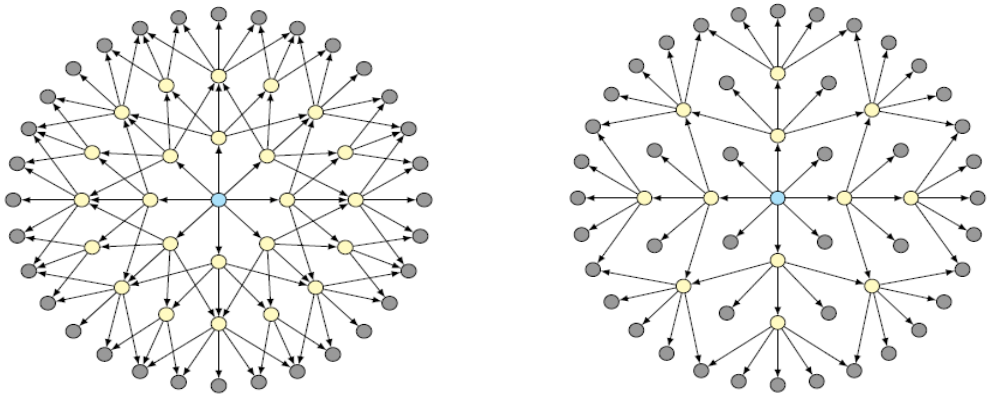
Le nœud intermédiaire procède ensuite à la rediffusion des informations qu'il vient de modifier dans sa table de routage tout en incrémentant son numéro de séquence.

Malgré les améliorations qu'il propose par rapport à *DBF* en éliminant le problème des boucles de routage[16] et le problème du comptage à l'infini[17], grâce notamment à l'utilisation des numéros de séquence, *DSDV* reste long et coûteux. Il nécessite des mises à jour régulières de ses tables de routage même lorsque le réseau est inactif. À chaque mise à jour, un nouveau numéro de séquence est nécessaire ce qui augmente le temps avant que le réseau converge. Ceci rend *DSDV* peu adapté aux réseaux très dynamiques.

[15] Ce mécanisme est utilisé par DSDV pour différencier les routes valides ayant des numéros de séquence pairs, des routes invalides ayant des numéros impairs. Par exemple, un nœud détectant qu'un lien n'est plus disponible change le numéro qui est pair dans sa table de routage vers le numéro impair qui est immédiatement au-dessus.
[16] Routing loops.
[17] Counting to infinity.

4.1.2 *Le protocole OLSR*

OLSR [32] (*Optimized Link State Routing*) est un protocole proactif à état de lien (Link state), inspiré du protocole de routage filaire classique *OSPF* (*Open Short Path First*) [36]. Il utilise ainsi des envois de paquets de contrôle périodiques afin d'informer chaque nœud des changements survenus dans la topologie. *OLSR* se distingue des protocoles à état de lien classiques en introduisant une optimisation de la stratégie de diffusion de base, effectuée par le biais de nœuds particuliers : les "relais multipoints[18]" (*MPR*). Ces *MPR* sont des nœuds auxquels est confiée la responsabilité exclusive d'émettre certaines informations de routage. Chaque nœud choisit son ou ses *MPR* parmi ses voisins symétriques à un saut, de telle manière qu'à travers son ensemble de *MPR*, il puisse joindre n'importe quel voisin à deux sauts. Chaque *MPR* lui-même maintient une liste de tous les nœuds qui l'ont choisi comme tel, ces nœuds sont appelés des sélecteurs *MPR.* Par la suite, le rôle des *MPR* est de relayer tout message en provenance de leurs sélecteurs *MPR* et d'ignorer les messages en provenance des autres nœuds (Figure 2.7).



(a) Routage par inondation (24 transmissions pour atteindre tous les nœuds à 3 sauts).

(b) Routage avec les nœuds MPR (12 transmissions pour atteindre tous les nœuds à 3 sauts).

Figure 2.7 : Avantage de l'utilisation des MPR.

La découverte de voisins et la détermination du type de lien est effectuée par l'échange de messages de type *HELLO*. Ces messages sont émis périodiquement (deux secondes par défaut) par chacun des nœuds à l'attention de tous leurs voisins à deux sauts. Ils contiennent pour chaque nœud, la liste de tous ses voisins connus, ainsi que le type de lien qui les relie. Celui-ci peut être asymétrique (si un échange a été effectué dans une seule direction) ou symétrique (lorsqu'un échange a été effectué dans les deux directions). Il peut être également relais multipoint lorsqu'il s'agit de spécifier à un nœud qu'il est choisi comme *MPR* et enfin perdu lorsqu'un lien est détecté comme rompu après un certain délai. Ainsi, sur la réception d'un message *HELLO*, un nœud examine la liste des adresses et les informations associées pour mettre à jour sa table de routage.

[18] En anglais, Multi Point Relays "MPR".

En plus des connaissances sur le voisinage à un saut, chaque nœud maintient également des informations sur son voisinage à deux sauts. Les adresses de ces voisins sont stockées dans une liste et sont utilisées ultérieurement pour déterminer l'ensemble *MPR* optimal de couverture de ces nœuds.

Afin de mettre à jour leurs tables de routage, les nœuds doivent être informés régulièrement des changements de topologie qui interviennent en dehors de leur voisinage. C'est là le rôle des messages de type *topologie control* (*TC*). Ces paquets de contrôle sont émis périodiquement par chaque *MPR* à l'attention de tous les nœuds du réseau, afin de déclarer leurs ensembles de sélecteurs *MPR*. La conséquence est que chaque nœud reçoit un graphe de topologie partielle, constitué par tous les nœuds du réseau et l'ensemble des liens entre un nœud et ses éventuels sélecteurs *MPR*. À partir de ces informations, chaque nœud peut déterminer très rapidement les routes optimales (en termes de sauts) vers n'importe quelle destination.

4.1.3 Le protocole TBRPF

À l'instar d'*OLSR*, le protocole *TBRPF* (*Topology Dissemination Based on Reverse Path Forwarding*) [33] est, lui aussi, un protocole à état de liens dans lequel chaque nœud maintient un arbre de routage des plus courts chemins. Les protocoles OLSR et TBRPF se distinguent principalement au niveau des techniques utilisées pour disséminer les informations de routage. Contrairement à OLSR pour lequel seuls les nœuds *MPR* diffusent l'état des liens, chaque nœud diffuse à ses voisins directs l'état des liens du réseau. Plus précisément, chaque nœud diffuse périodiquement à tous ses voisins directs, d'une part la liste de ses voisins directs et d'autre part l'arbre de routage qu'il a construit. À partir de ces informations, chaque nœud construit itérativement la topologie à une distance de deux sauts de lui ainsi qu'un arbre de plus courts chemins. En revanche, pour réduire l'utilisation de la bande passante, TBRPF possède un mécanisme d'optimisation. Ce mécanisme consiste à ne pas diffuser la totalité des informations topologiques dans chaque message de contrôle, mais seulement les différences par rapport au dernier message émis, réduisant ainsi le volume de trafic de contrôle.

4.2 Protocoles de routage Ad hoc réactifs

Dans des réseaux de grandes tailles, l'approche proactive peut s'avérer peu performante, car trop gourmande en bande passante. C'est pourquoi certains protocoles se basent sur une autre approche, plus spécifique au domaine. C'est ainsi le cas des protocoles réactifs qui ne maintiennent pas d'information sur la topologie du réseau. Au contraire, ils n'établissent une route que lorsqu'un nœud désire envoyer un message. Ce sont des protocoles de routage à la demande.

L'avantage de ces protocoles est que le réseau n'est inondé par les paquets de contrôle que lorsque cela est vraiment nécessaire, c'est-à-dire uniquement à la demande d'un nœud et non pas régulièrement comme c'est le cas des protocoles proactifs. Ils sont donc globalement moins coûteux en termes de signalisation et d'énergie. En revanche, le délai pour établir une route peut s'avérer bien

supérieur aux protocoles proactifs, surtout si la distance entre la source et la destination est grande. En outre, tous les nœuds reçoivent les requêtes, y compris ceux qui ne sont pas concernés. L'approche réactive engendre donc, elle aussi, une certaine surcharge de trafic.

4.2.1 Le protocole AODV

AODV (*Ad hoc On demand Distance Vector*) [37, 38] est un protocole de routage réactif à vecteur de distance qui s'inspire de *DSDV*. Contrairement à celui-ci, il ne construit pas a priori la table de routage mais réagit à la demande et essaie de trouver un chemin avant de router les informations.

Tant que la route reste active entre la source et la destination, le protocole de routage n'intervient pas, ce qui diminue le nombre de paquets de routage échangés entre les nœuds constituant le réseau. Lorsqu'un nœud "S" essaie de communiquer avec un nœud "D", l'échange de messages se fait en plusieurs étapes décrites ci-dessous à l'aide de l'exemple de la figure 2.8.

4.2.1.1 Découverte de route

Lorsqu'un nœud source a besoin d'une route vers une certaine destination (Par exemple, le nœud 1 dans la figure 2.8 désire envoyer des données au nœud 5) et qu'aucune route n'est disponible (la route peut être non existante, avoir expiré, ou être défaillante), la source 1 diffuse en broadcast (Figure 2.8{a}) un message de demande de route *RREQ* (*Route REQuest*). Ce message contient un identifiant ($id\#$) associé à l'adresse de la source ($Source_{\_Addr}$) qui servira à identifier de façon unique une demande de route. Le nœud 1 enregistre cet identifiant de paquet *RREQ* ([$id\#$, $Source_{\_Addr}$]) dans son historique (*buffer*) et l'associe à un *timer* qui décomptera sa durée de vie au-delà de laquelle cette entrée sera effacée.

Quand un nœud intermédiaire (cas des nœuds 2 et 4 dans la figure 2.8{b}) qui n'a pas de chemin valide vers la destination, reçoit le message *RREQ*, il ajoute ou met à jour le voisin duquel le paquet a été reçu. Il vérifie ensuite qu'il ne l'a pas déjà traité en consultant son historique des messages traités. Si le nœud s'aperçoit que la *RREQ* est déjà traitée, il l'abandonne et ne la rediffuse pas. Sinon, il met à jour sa table de routage à l'aide des informations contenues dans la requête afin de pouvoir reconstruire ultérieurement le chemin inverse vers la source. Il incrémente ensuite le nombre de sauts "$Hop_{\_count}$" dans la demande de route et la rediffuse.

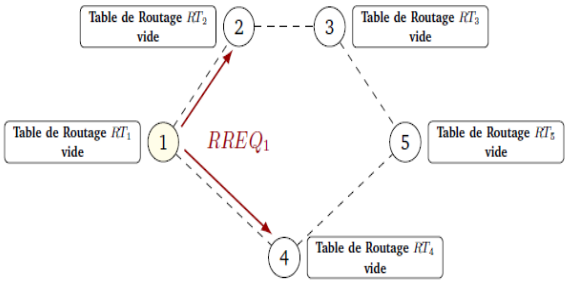
Il est à noter qu'*AODV* utilise le principe des numéros de séquence pour pouvoir maintenir la cohérence des informations de routage. Ce numéro, noté $Seq\#$ (*Sequence Number*), est un champ qui a été introduit pour indiquer la fraicheur de l'information de routage et garantir l'absence de boucles de routages.

À la réception d'un paquet *RREQ* (Figure 2.8{c}), la destination 5 ajoute ou met à jour dans sa table de routage un chemin vers le nœud voisin duquel il a reçu le paquet (nœud 4) ainsi qu'un chemin vers

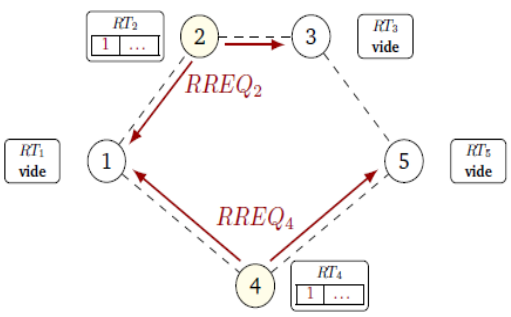
la source 1. La destination 5 génère ensuite une réponse de route *RREP* (*Route Reply*) qu'elle envoie en unicast vers le prochain saut en direction de la source (Figure 2.8{c}). Notons qu'un nœud intermédiaire peut aussi générer un *RREP* si la requête l'autorise à le faire et qu'il dispose déjà dans sa table de routage d'un chemin valide vers la destination 5.

Les nœuds intermédiaires qui reçoivent la *RREP* (cas du nœud 4 dans la figure 2.8{d}) vont mettre à jour le chemin qui mène à la destination dans leur table de routage et retransmettre en unicast le message (après avoir incrémenté le nombre de sauts) vers le nœud suivant en direction de la source sachant que cette information a été obtenue lors du passage de la *RREQ*. Lorsque la réponse de route atteint la source (nœud 1 dans l'exemple), un chemin bidirectionnel est établi entre la source et la destination (Figure 2.8{e}) et la transmission de paquets de données peut débuter.

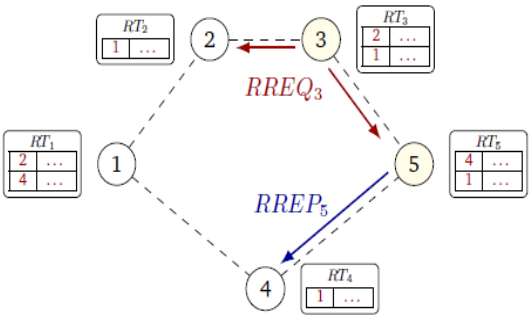
4.2.1.2 Maintenance des routes



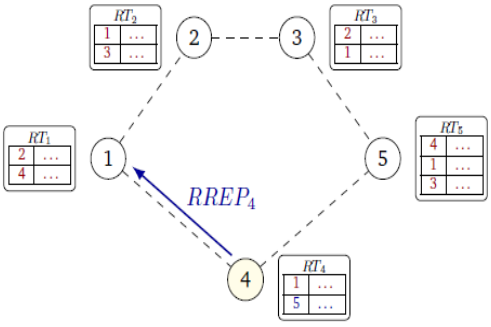
(a) Le nœud 1 initialise une demande de route pour obtenir un chemin vers 5.



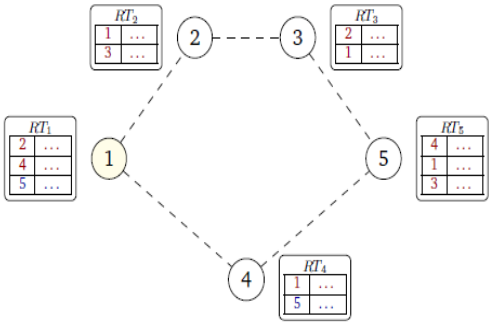
(b) 2 et 4 Retransmettent la demande de route.



(c) 5 Initialise une réponse de route.



(d) 4 Retransmet la réponse de route.



(e) Arrivée de la réponse de route à la source 1.

Figure 2.8 : Exemple d'établissement de route dans le protocole AODV.

Afin de maintenir les routes, une transmission de messages *HELLO* est effectuée. Ces messages sont en fait des réponses de route (*RREP*) diffusés aux voisins avec un nombre de sauts égal à un. Si au bout d'un certain temps, aucun message n'est reçu d'un nœud voisin, le lien en question est considéré défaillant. Alors, un message d'erreur *RERR* (*Route ERRor*) se propage vers la source et tous les nœuds intermédiaires vont marquer la route comme invalide et au bout d'un certain temps, l'entrée correspondante est effacée de leur table de routage. Le message d'erreur *RERR* peut être diffusé ou envoyé en unicast en fonction du nombre de nœuds à avertir de la rupture de liaison détectée. Ainsi, s'il y en a un seul, le message est envoyé en unicast sinon, il est diffusé.

AODV a l'avantage de réduire le nombre de paquets de routage échangés étant donné que les routes sont créées à la demande et utilise le principe du numéro de séquence pour éviter les boucles de routage et garder la route la plus fraîche. Cependant, l'exécution du processus de création de route provoque des délais importants avant la transmission de données.

4.2.2 Le protocole DSR

À l'instar d'AODV, le protocole DSR (*Dynamic Source Routing*) [39] emploie le routage à la demande mais il se distingue par une approche de routage par la source. Ainsi, il recourt lui aussi au mécanisme de découverte de routes mais à la différence du protocole AODV, le nœud source indique dans l'en-tête de chaque paquet la liste de tous les nœuds qui composent la route jusqu'à la destination. Cette liste est renvoyée à la source dans un paquet de réponse de route (Figure 2.9). Nous donnons plus de détails sur le processus de découverte de route dans ce qui suit.

4.2.2.1 Découverte de route

Lorsqu'un nœud source désire envoyer des données à une destination et qu'il ne trouve pas de route disponible pour cette destination dans son cache (route cache), il initialise une demande de route *RREQ* (*Route REQuest*). C'est le cas du nœud 1 dans la figure 2.9{a}. La *RREQ* contient un identifiant unique (*route request identifier*), la destination à atteindre et une liste d'adresses de nœuds qui contient initialement uniquement l'adresse de la source (cette liste constituera le chemin entre la source et la destination à la fin du processus de découverte).

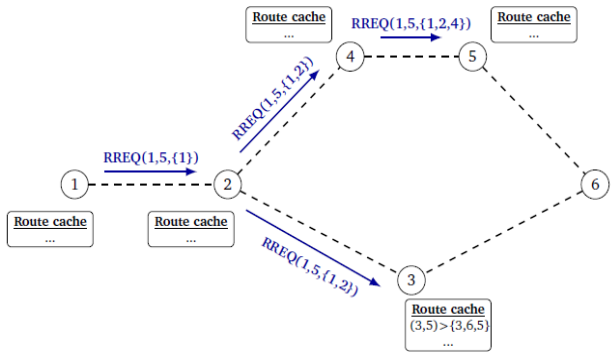
Lorsqu'un nœud intermédiaire reçoit la demande de route *RREQ*, il commence par vérifier s'il ne s'agit pas d'une requête déjà traitée en cherchant dans l'historique l'existence du couple (*identifiant de la requête, adresse de la source*) identifiant cette *RREQ*. Si c'est le cas, le paquet est ignoré, sinon le nœud rajoute son adresse dans la liste du paquet et rediffuse ce paquet à son tour après l'avoir ajouté dans son historique. C'est le traitement que les nœuds 2 et 4 ont suivi dans l'exemple de la figure 2.9{a}.

Lorsque le paquet *RREQ* arrive à la destination, la liste contenue dans le paquet constitue le chemin complet pour l'atteindre (cas du nœud 5 dans la figure 2.9{a}). La destination crée alors une réponse

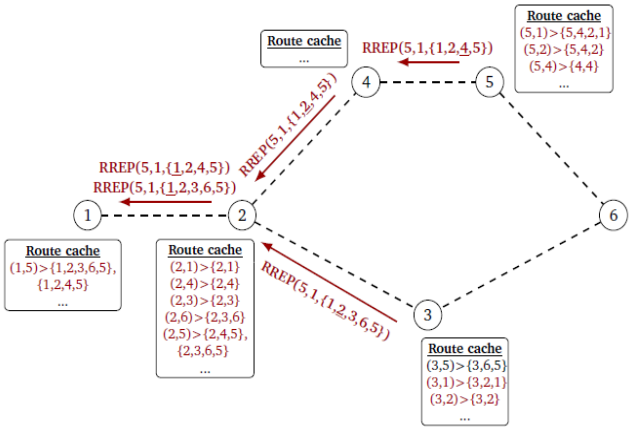
de route *RREP* en y copiant la liste contenue dans la *RREQ* reçue et en insérant son adresse à la fin de cette liste. Une fois envoyée, cette réponse de route suivra le chemin contenu dans la liste jusqu'à atteindre la source. Ainsi, le chemin est établi entre la source et la destination et la transmission de données peut débuter.

Dans certains cas, un nœud intermédiaire peut avoir une route qui mène à la destination dans son route-cache (nœud 3 dans la figure 2.9). Dans cette situation, le nœud intermédiaire peut générer une réponse de route en concaténant le chemin qu'il a reçu dans le paquet *RREQ* avec celui qui se trouve dans son route-cache en s'assurant qu'il n'y a pas de nœud qui figure dans les deux parties auquel cas il devra renoncer à la création de la *RREP* pour éviter la création des boucles de routage.

À la fin du processus de découverte de route, un nœud peut avoir dans son cache plus d'une route pour certaines destinations auquel cas il devra choisir une route en se basant sur le plus court chemin ou en utilisant une autre métrique (rapidité, confiance, etc.).



(a) Propagation de la demande de route.



(b) Propagation de la réponse de route.

Figure 2.9 : Exemple d'établissement de route dans le protocole DSR.

4.2.2.2 Maintenance des routes

Lors de la transmission d'un paquet, chaque nœud est responsable de l'acheminement des données sur le lien en direction du prochain saut. Il devra s'assurer que les données sont bien parvenues au prochain saut. Un accusé de réception peut garantir la confirmation de la validité du lien. Si un nœud ne reçoit pas un accusé de réception suite à un envoi de paquet, il considère que le lien est rompu et supprime cette route du cache. Il crée alors un paquet erreur de route *RERR* qu'il envoie à tous les nœuds ayant envoyé un paquet sur ce lien depuis le dernier accusé de réception.

Parmi les avantages de *DSR* sur *AODV*, on peut noter l'utilisation indifférente des liens symétriques ou asymétriques. En effet, un nœud destinataire peut indiquer dans l'en-tête de ses paquets une route différente de celle indiquée par le nœud. Un autre avantage important de *DSR* est l'absence de boucles de routage. En revanche, *DSR* induit un certain *overhead*[19] au niveau de la signalisation puisque la route présente dans l'en-tête des paquets augmente leur taille. On peut estimer que cet *overhead* est compensé par l'absence de messages *HELLO*.

4.3 Protocoles de routage Ad hoc hybrides

Partant de ce constat, certains protocoles proposent de combiner les deux approches précédentes afin d'éliminer leurs inconvénients respectifs, pour n'en garder que les avantages. C'est le cas des protocoles hybrides. Ce type de protocoles adopte une méthode proactive pour établir les chemins à l'avance dans un voisinage ne dépassant pas quelques sauts (2 ou 3 sauts) et utilise une méthode réactive au-delà de cette limite. La combinaison de ces deux techniques partage le réseau en zones où un nœud peut décider directement à la réception d'un message si la destination fait partie de la même zone ou non, auquel cas il devra rediriger le message vers une autre zone. Le protocole le plus représentatif de cette catégorie est le protocole *ZRP* (*Zone Routing Protocol*) [40]. Celui-ci divise le réseau en zones géographiques. Par la suite :

- Un protocole de routage proactif *IARP (Intrazone Routing Protocol)* qui fournit une vue détaillée du voisinage à k-sauts (par exemple $k = 2$) appelé zone de routage (*Routing zone*). Pour pouvoir construire cette zone, chaque nœud a besoin du voisinage à un saut qui est obtenue grâce au protocole de la couche liaison ou en utilisant un protocole prévu à cet effet comme le NHDP[20]. La figure 2.10, présente les zones de routage pour les nœuds 1 et 4. Ces deux zones se chevauchent du fait que chaque nœud maintient sa propre zone de routage.

Les protocoles de routage proactifs à état de lien peuvent être modifiés et utilisés comme un *IARP* en limitant les mises à jour de l'état de lien au rayon de la zone de routage. De cette manière, un nœud peut décider immédiatement lors de la réception d'un paquet s'il a un chemin vers la destination ou non

[19] L'overhead est un anglicisme qui désigne le rapport entre la charge utile d'un paquet et la charge de contrôle.
[20] Neighborhood Discovery Protocol.

et peut ainsi répondre au nom de tous les nœuds de la zone à laquelle il appartient. Ce qui évite l'effort d'explicitement interroger le reste des nœuds de la même zone.

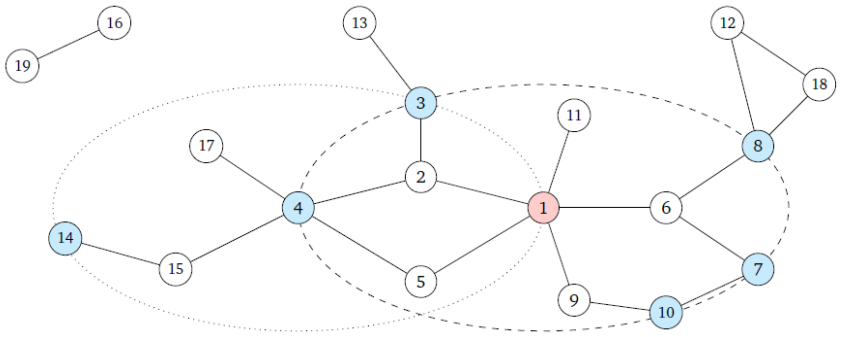


Figure 2.10 : Zone de routage de rayon = 2 (les nœuds 1 et 4).

- Si un nœud n'a pas de chemin vers la destination, l'*IERP* (*Interzone Routing Protocol*) prend le relais pour la recherche de routes en dehors de la zone dans laquelle le nœud se trouve. Ainsi, une demande de route est créée et envoyée aux nœuds périphériques de la zone de routage (opération appelée *Bordercast*). Par exemple, les nœuds 3, 4, 7, 8 et 10 sont des nœuds périphériques de la zone de routage du nœud 1 dans la figure 2.10.

Les demandes de route créées sont acheminées vers la périphérie de la zone en utilisant le protocole BRP[21]. Ce dernier se base sur la topologie obtenue grâce à l'IARP pour la construction d'un arbre multicast (*bordercast tree*) donnant les différents chemins pour atteindre les nœuds périphériques d'une zone. Ces nœuds périphériques vérifient à leur tour l'existence de destination dans leurs zones et si c'est le cas, un paquet de réponse de route est retourné à la source. Dans le cas contraire, ils diffusent la demande de route à leurs propres nœuds périphériques qui, à leur tour, effectuent le même traitement. Chaque nœud retransmettant la demande de route fait attention à ne pas retransmettre la demande de route aux nœuds l'ayant déjà traité afin d'optimiser le temps de découverte de route et éviter les boucles de routage.

Pour résumer, lorsqu'un nœud implémentant ZRP veut joindre une destination, il commence sa recherche dans la zone à laquelle il appartient. S'il la trouve, il peut l'atteindre immédiatement étant donné que cette entrée est maintenue dans son cache grâce au protocole proactif (*IARP*). Sinon, il envoie une requête de demande de route aux nœuds périphériques grâce au protocole réactif (*IERP*) en utilisant BRP pour la livraison de ces demandes. Ainsi, les nœuds périphériques recevant la demande de route recherchent dans leurs zones respectives et ainsi de suite jusqu'à atteindre la destination. Ce protocole présente l'avantage de diminuer le nombre de messages de contrôle qui transitent sur le réseau

[21] Bordercast Resolution Protocol.

compare aux protocoles proactifs ou réactifs. De plus, il permet de diminuer le temps de latence pour trouver de nouvelles routes.

4.4 Performances

D'une manière générale, s'agissant d'évaluer les performances, il est difficile de comparer les approches "proactive & réactive". En effet, toutes les simulations effectuées montrent que les performances varient considérablement suivant les caractéristiques du réseau (mobilité des nœuds, densité, diamètre du réseau, etc.) et également du modèle de mobilité choisi. Toutefois, il semble établi qu'un protocole comme *OLSR* est plus adapté à des réseaux denses avec une mobilité élevée tandis qu'un protocole comme *DSR* est plus efficace sur des réseaux peu dynamiques, à faible densité. Ceci s'explique par le fait que les routes n'ont alors pas besoin d'être redécouvertes régulièrement, le mécanisme de découverte de routes étant la phase la plus coûteuse des protocoles réactifs.

5. ANALYSE DE VULNERABILITÉS DU ROUTAGE AD HOC

Les réseaux Ad hoc sont exposés à un grand nombre de vulnérabilités, surtout au niveau routage. Ces vulnérabilités engendrent des attaques spécifiques contre lesquelles les mesures de sécurité traditionnelles sont inefficaces. Étudier les vulnérabilités dans les réseaux MANETs peut nous permettre de reconnaître toutes les attaques à éviter, afin d'établir un environnement sécurisé qui satisfait les besoins de sécurité de chacune des applications de MANETs.

5.1 Description des attaques

Le terme "attaque" désigne une action visant à compromettre la confidentialité ou l'intégrité des informations transitant sur le réseau, ou, d'une manière générale, à altérer son bon fonctionnement.

Dans les réseaux Ad hoc, selon le niveau d'intrusion des actions menées par un attaquant, on distingue généralement deux catégories : les attaques *passives* et les attaques *actives*. Une attaque est passive lorsqu'un nœud non autorisé obtient un accès à des informations échangées sur le réseau, et ce, sans altérer les opérations du réseau. A contrario, une attaque est active lorsqu'un nœud non autorisé altère des informations en transit par des actions de modification, suppression, ou fabrication, ce qui conduit à des perturbations dans le fonctionnement du réseau.

Selon le domaine d'appartenance d'un nœud, les attaques actives peuvent elles-mêmes être classées en deux catégories, à savoir les attaques *externes* et *internes*. Tandis que les attaques externes sont réalisées par des nœuds qui n'appartiennent pas au domaine du réseau, les attaques internes sont menées par des nœuds compromis qui sont autorisés à participer au fonctionnement du réseau. Étant donné que les attaquants font d'ores et déjà partie du réseau de nœuds autorisés, les attaques internes sont généralement plus pernicieuses et difficiles à détecter que les attaques externes.

Enfin, les attaques peuvent être de type individuelles ou par collusion. Les attaques individuelles sont menées par un seul nœud attaquant. Puisque les capacités de communication et de calcul de l'attaquant sont en général similaires à celles des autres nœuds du réseau, ces attaques demeurent relativement simples, et sont d'autant plus limitées que des mécanismes de sécurité sont mis en œuvre. En revanche, rien n'empêche à des nœuds attaquants de mutualiser leurs informations et leurs ressources, en exploitant les connexions qu'ils ont entre eux. Ces attaques par collusion, issues de plusieurs nœuds répartis à différents endroits dans le réseau, sont généralement plus évoluées et plus pernicieuses. Par ailleurs, en raison de l'intervention de plusieurs nœuds intermédiaires, leur détection et l'identification précise de leur origine sont rendues plus complexes.

5.1.1 Attaques passives

En raison de la nature même du médium d'accès, il est très facile pour un nœud quelconque, d'écouter des communications à l'insu des participants. En effet, dans les réseaux Ad hoc, les nœuds communiquent par l'interface air avec un accès partagé au support, de type "évitement de collision" [41]. Une attaque triviale consiste alors pour un attaquant à se mettre en mode écoute (*Promiscuous listening*) pour capter tout ce qui passe par l'interface air et ainsi, analyser le trafic. Partant de ce constat, suivant la signalisation utilisée par le protocole, un nœud peut extraire toutes sortes d'informations stratégiques comme bien sûr le contenu des données, mais aussi la connectivité du réseau, la localisation de certains nœuds, leurs adresses (*IP*, *MAC*[22], etc.).

En ce qui concerne la protection des données, un protocole de chiffrement de type IPsec[23] suffirait à la confidentialité des informations. Cependant, dans le cadre des réseaux Ad hoc, sa mise en place ne constitue pas une protection suffisante puisque les nœuds ne peuvent pas à l'origine, avoir confiance les uns dans les autres et donc, s'échanger facilement des clés. En outre, il n'est pas adapté au modèle Ad hoc car il a ce essentiellement été conçu pour assurer la confidentialité des données et non pour sécuriser les informations de signalisation. Une autre solution envisageable consiste à effectuer un codage au niveau physique, en fonction du temps ou de la longueur de l'onde radio [42].

5.1.2 Attaques actives

Ces attaques peuvent être menées aux différents niveaux du modèle OSI[24] (Tableau 2.1). Ainsi, à l'instar des autres types de réseau sans fil, les réseaux Ad hoc sont tout d'abord vulnérables à des attaques au niveau de la couche physique. La raison est que les ondes radio sont très sensibles aux interférences. Ainsi, sans nécessairement prendre part au réseau Ad hoc formé, un attaquant est susceptible de générer des signaux à une fréquence proche de celle utilisée dans le réseau afin de brouiller les transmissions [43]. En conséquence, toute communication devient impossible.

[22] Medium Access Control (MAC) est la moitié basse de la couche de liaison de données du modèle OSI.
[23] IPsec (Internet Protocol Security), défini par l'IETF comme un cadre de standards ouverts pour assurer des communications privées et protégées sur des réseaux IP, par l'utilisation des services de sécurité cryptographiques.
[24] Open Systems Interconnection.

Table 2.1 : Attaques contre les réseaux Ad hoc par couche de la pile réseau.

Couche réseau	*Nom de l'attaque*	*Influence sur la propriété de sécurité*			*Technique*			Motivations/Résultats
		Confidentialité	*Disponibilité*	*Intégrité*	*Rejeu*	*Modification*	*Fabrication*	
Application	Écoute et analyse du trafic	✓			Pas d'actions			- Intercepter des informations.
	Location disclosure [44]	✓				✓		- Découvrir l'emplacement des nœuds.
Transport	SYN flood [45]		✓	✓				- Vol de session, inondation de requêtes.
Réseau	Detour attack [46]		✓			✓		- Ne pas participer dans le routage.
	Resource consumption [47]		✓				✓	- Epuisement de la batterie.
	Sleep deprivation torture [48]		✓				✓	- Epuisement de la batterie.
	Routing table overflow [49]		✓				✓	- Débordement de la table de routage.
	Selfish behavior [50]		✓		Pas d'actions			- Conserver son énergie.
	Black Hole attack [51]			✓		✓	✓	- Absorber le trafic.
	Routing table poisoning [52]			✓			✓	- Routage non optimal, congestion du réseau.
	Rushing attack [53]			✓	Exécution plus rapide			- Division du réseau, attirer le trafic.
	Wormhole attack [54] [55]			✓	✓			- Création d'un tunnel/perturber le routage.
Liaison (MAC)	Man-in-the-middle attack [56]	✓	✓	✓		✓	✓	- Utiliser l'usurpation d'identité.
	Sybil attack [57]	✓	✓	✓	✓	✓	✓	- Identités multiples.
Physique	Jamming [58]							- Terminal, ondes radio,

Au niveau de la couche liaisons de données, des protocoles ont été définis afin de maintenir la connectivité à un saut entre des nœuds voisins. Ils visent à offrir aux nœuds un accès équitable au media de communication. Pour ce faire, leur fonctionnement repose sur des échanges de trames de contrôle et suppose une coopération inconditionnelle entre les nœuds. De manière évidente, un nœud n'est pas contraint à suivre les spécifications de ces protocoles. Dès lors qu'un attaquant sature le media en émettant des trames de contrôle ou de données, les autres nœuds à proximité se trouvent dans l'incapacité de communiquer. En raison de l'indisponibilité des ressources de communication, on parle alors de déni de service. Des attaques spécifiques contre la couche de contrôle d'accès au support (MAC) définie par la norme IEEE 802.11, et exploitant certaines caractéristiques du protocole, ont déjà été mises en évidence [59].

Les protocoles définis au niveau de la couche réseau servent à étendre la connectivité à un saut à tous les nœuds dans le réseau. C'est donc à ce niveau que fonctionnent les protocoles de routage et le mécanisme de retransmission des paquets de données. Un simple détournement du fonctionnement normal de ces protocoles entraine une perturbation des communications, et l'ensemble du réseau peut être paralysé. La sécurité de la couche réseau est donc primordiale dans la mesure où le but du réseau est avant tout de mettre en relation des nœuds et d'acheminer leurs données.

Au niveau de la couche application, les attaques sont communes à tous les types de réseau et leur mise en œuvre dépend de l'application visée. Les mécanismes de contre-mesures envisagés ont pour la plupart recours à la cryptographie afin de protéger les échanges de bout en bout.

Nous pouvons constater qu'à la fois les attaques au niveau de la couche physique, liaison de données et réseau portent atteinte à la disponibilité des services de communication offerts par le réseau. Par conséquent, la seule protection des protocoles de la couche réseau n'est en soi pas suffisante pour en assurer le bon fonctionnement. Dans le meilleur des cas, une protection à ce niveau s'avère efficace contre les attaques présentes sur cette couche. Néanmoins, la protection des couches inférieures fait appel à des techniques d'accès au média pour la couche liaison de données et des techniques de transmission radio pour la couche physique. Étant donné que ces techniques étaient en dehors de notre domaine de recherche, nous avons orienté nos travaux vers des solutions de protection des protocoles de la couche réseau.

5.1.2.1 Attaques sur la phase de signalisation

La génération et le traitement des messages de contrôle sont entièrement sous la responsabilité des nœuds qui forment le réseau. Or ces derniers représentent un des éléments de base dans le processus de routage, car ils servent à établir et maintenir les relations de connectivité entre les nœuds. À partir du moment où un attaquant génère des messages de contrôle dont le contenu n'est pas conforme au regard de la logique du protocole, ou bien s'il ne participe pas correctement à la retransmission de ces messages, alors l'intégrité de la topologie peut être compromise. En raison de leur importance, ils constituent la principale cible des attaquants pour perturber le fonctionnement des réseaux Ad hoc. Les

attaques contre les messages de contrôle prennent de nombreuses formes telles que : la fabrication, la modification, le rejeu, le refus de retransmission. Elles incluent par ailleurs tous les types d'attaquant décrits précédemment.

Dans une attaque par modification, l'attaquant met à jour certains champs des messages de contrôle qu'il reçoit et les retransmet, sans suivre les recommandations du protocole. La modification de nombreux champs est susceptible d'influer l'intégrité des données de routage. Figurent parmi ces champs : le numéro de séquence, le nombre de sauts, les identités des nœuds (adresses IP), le descriptif du chemin en construction. Par exemple, en modifiant la métrique de routage ou le descriptif du chemin, un attaquant parvient à manipuler la construction des chemins en les faisant apparaître soit plus longs, soit plus courts que ce qu'ils sont réellement.

Les messages de contrôle échangés entre les nœuds décrivent un état de la topologie du réseau à un instant donné. Une attaque qui exploite ces informations est le rejeu. Elle consiste en la réémission en un point du réseau de messages de contrôle caducs (car anciens). La conséquence est que des nœuds sont amenés à réaliser des mises à jour sur une base d'informations qui n'a plus lieu d'exister, entraînant alors des incohérences dans les chemins construits. Une particularité de cette attaque est qu'elle est effective même si les messages de contrôle sont protégés par une empreinte ou une signature numérique.

De manière générale, il en résulte un détournement du trafic de son cheminement normal. Par détournement de trafic, nous entendons le fait qu'un attaquant est capable d'attirer le trafic vers lui-même, de créer des boucles de routage, de créer des chemins sous optimaux, ou de créer des chemins non existants. En outre, des attaquants en collusion peuvent empêcher un nœud source de trouver un chemin vers une destination, conduisant alors à un partitionnement du réseau.

La non-retransmission est une attaque selon laquelle un nœud supprime les messages de contrôle qu'il reçoit au lieu de les retransmettre. Bien que triviale à mettre en œuvre, la réalisation de cette attaque, ne serait-ce que par un nombre limité d'attaquants, nuit sévèrement au fonctionnement du réseau. Elle entraine non seulement une réduction de la connectivité globale et du nombre de chemins de communication disponibles, mais aussi un raccourcissement de la durée de vie du réseau puisque le trafic sera moins bien réparti entre les nœuds.

Il existe également des attaques qui visent en particulier la phase de maintenance de chemin des protocoles réactifs. Si la destination ou un nœud intermédiaire le long d'un chemin actif se déplace, le nœud en amont de la rupture du lien diffuse un message d'erreur vers le nœud source. Tous les nœuds amenés à traiter ce message (et a fortiori la source) invalident le chemin vers cette destination dans leur table de routage. Un attaquant peut tirer avantage de ce mécanisme de deux façons :

- En l'absence d'événement de rupture, l'attaquant forge un faux message d'erreur. Il s'ensuit un déclenchement d'opérations coûteuses de maintenance et de reconstruction d'un chemin valide entre la source et la destination victime de l'attaque.
- L'attaquant ne retransmet pas un message d'erreur pourtant valide, ce qui conduit à une augmentation du temps pour détecter et corriger l'erreur.

5.1.2.2 Attaques sur la phase d'acheminement des paquets de données

En sus des attaques contre les messages de contrôle des protocoles de routage, un attaquant peut perturber l'opération d'acheminement des paquets de données. Dans ce type d'attaque, le but n'est pas de compromettre l'intégrité de la topologie construite. Ici, un attaquant participe aux phases de découverte et de maintenance des chemins. En revanche, il ne relaie pas correctement les paquets de données, même s'il appartient au chemin établi entre une source et une destination. Il agit par des actions de suppression, de modification, de rejeu sur les paquets de données qu'il reçoit. Toujours dans un but de perturber l'opération d'acheminement, une autre attaque plus subtile consiste à ralentir la retransmission de paquets de données sensibles aux retards. Lorsqu'un attaquant ne fait que supprimer les paquets de données, cette attaque est assimilée à une attaque de non-coopération.

5.1.2.3 Autres attaques spécifiques

Une attaque inhérente aux réseaux Ad hoc et qui nuit au fonctionnement de tous les protocoles de routage est l'attaque du "*Wormhole*" [55], cette attaque, se manifeste par un détournement de trafic : un paquet capturé en un point du réseau est rejoué en un autre point du réseau plus ou moins distant. Pour mener cette attaque, l'attaquant établit un tunnel privé entre deux points éloignés du réseau au moyen d'une liaison câblée ou d'une liaison sans fil longue portée. Ensuite, au cours de cette attaque, l'attaquant capture les transmissions sur une des extrémités, les envoie (éventuellement de manière sélective) à travers le tunnel vers l'autre extrémité, d'où elles sont rejouées dans le réseau. La conséquence directe de cette attaque est l'établissement d'une vision erronée des nœuds au regard de leur connectivité dans le réseau. En outre, de sorte à paraître virtuellement invisible, l'attaquant n'effectue aucune manipulation sur les transmissions relayées. La sévérité de l'attaque vient du fait qu'elle est difficilement détectable et que d'autre part, elle est effective même dans le cadre d'un réseau où l'authentification, l'intégrité et la confidentialité sont préservées.

En détournant le trafic vers lui-même, un attaquant se donne la possibilité d'analyser un maximum d'informations. À partir du moment où l'intégralité des paquets de données est retransmise vers la destination, l'attaque est quasiment transparente. Seuls des délais supplémentaires dans l'acheminement des données pourront apparaître. En revanche, si l'ensemble des paquets reçus par l'attaquant est supprimé, alors des dégradations sévères dans les communications de bout en bout seront occasionnées. Cette attaque, décrite par [52], est plus connue sous le nom de trou noir "*black hole*". Dans le but de rendre l'attaque moins intrusive (et donc plus difficilement détectable), la suppression des paquets peut

être réalisée de manière sélective, c'est-à-dire par la suppression des paquets pour une source ou une destination spécifique, par la suppression d'un paquet toutes les "t" secondes, par la suppression aléatoire des paquets, etc.

Enfin, les réseaux Ad hoc se caractérisent par des ressources limitées en termes d'énergie et de bande passante. En effet, les nœuds fonctionnent sur batterie et même si ces dernières sont de plus en plus performantes en terme d'autonomie, elle n'en demeure pas moins une ressource capitale. Les réseaux Ad hoc sont alors particulièrement vulnérables aux attaques brutales de type déni de service. Une des plus simples est l'attaque par harcèlement selon laquelle un attaquant inonde le réseau de paquets superflus, ceci dans le but d'augmenter la charge sur le réseau ainsi que d'épuiser les ressources des nœuds.

6. CONCLUSION

Nous avons vu dans ce chapitre que tous les protocoles de routage classiques dans les réseaux Ad hoc (DSDV, OLSR, AODV) sont particulièrement vulnérables à un grand nombre d'attaques qui peuvent aller de la capture d'informations sensibles à la paralysie complète du réseau. Or, à l'époque actuelle, où l'utilisation des réseaux sans fil connaît un essor sans précédent (notamment grâce au Wi-Fi, au WiMax[1] et aux téléphones mobiles) et où parallèlement, le nombre d'attaques contre les systèmes informatiques n'a jamais été aussi élevé, l'enjeu de la sécurité des réseaux est devenu considérable. Ainsi, même si les réseaux Ad hoc constituent une solution tout à fait prometteuse aux problèmes actuels liés à la mobilité des utilisateurs et des réseaux eux-mêmes, leur développement est freiné aujourd'hui par l'absence de mécanismes de sécurité suffisamment efficaces pour subvenir aux besoins actuels en protection des données tels que ceux des applications commerciales.

Partant de ce constat, les recherches qui autrefois étaient concentrées sur l'amélioration des performances, se réorientent aujourd'hui sur la sécurisation des protocoles de routage. Cependant les procédés employés sont souvent très différents d'un algorithme à l'autre et les caractéristiques inhérentes au modèle Ad hoc telles que la mobilité, l'absence d'infrastructure et la limitation des ressources imposent de repenser complètement les dispositifs de protection classiques utilisés dans le domaine filaire et obligent les concepteurs à faire des compromis entre la sécurité des protocoles et les contraintes de performances. En effet, dans un contexte totalement distribué, ces mécanismes doivent être adaptés en conséquence, au risque d'engendrer une surcharge conséquente du réseau.

[1] Worldwide Interoperability for Microwave Access.

Chapitre 3

La sécurité dans les réseaux mobiles Ad hoc

1. INTRODUCTION

Pendant de nombreuses années, la problématique de la sécurité a été totalement ignorée dans le domaine des réseaux Ad hoc, la plupart des recherches s'appliquent à améliorer les performances (rendement des protocoles, limitation de l'overhead, etc.). Par la suite, plusieurs mécanismes ont été envisagés pour accroitre la robustesse des protocoles de routage sans pour autant trop affecter les performances. Certains d'entre eux consistent simplement en des optimisations basiques des protocoles, en vue de prolonger leur utilisation en milieu hostile. D'autres, en tel revanche s'inspirent de techniques plus avancées mais également plus coûteuses telles que la cryptographie pour garantir des fonctionnalités essentielles comme la confidentialité et l'authentification. Il est clair que le problème de la sécurité dans les réseaux mobiles Ad hoc est large et qu'il n'existe pas de solution générale. Il est aussi clair que les différentes applications des réseaux mobiles Ad hoc n'ont pas les mêmes besoins en sécurité.

Après avoir abordé le domaine des réseaux mobiles Ad hoc et de la sécurité de l'ère numérique dans les chapitres précédents, ce chapitre propose de traiter des solutions et architectures de sécurité propres à ce type de réseaux. Dans un premier temps, les domaines et les principaux axes de sécurité dans les réseaux mobiles Ad hoc vont être détaillés. Puis, nous étudions les différentes approches de sécurité dédiées à l'environnement des réseaux Ad hoc. Enfin, une discussion sur les stratégies de prévention proposées pour les réseaux Ad hoc sera présentée.

2. BESOINS EN SÉCURITÉ POUR LES RÉSEAUX MOBILES AD HOC

Les besoins de base en sécurité pour les réseaux mobiles Ad hoc sont plus ou moins les mêmes que pour les réseaux filaires ou sans fil avec infrastructure. Les services de sécurité sont basés sur quatre concepts fondamentaux : l'authentification des utilisateurs, la confidentialité, l'intégrité des données et du trafic du réseau, et enfin la non-répudiation des utilisateurs.

2.1 Authentification

L'authentification permet de vérifier l'identité d'une entité ou d'un nœud dans le réseau. C'est une étape incontournable pour le contrôle de l'accès aux ressources réseau. Sans l'authentification, un nœud malicieux peut facilement usurper l'identité d'un autre nœud dans le but de bénéficier des privilèges attribués à ce nœud ou d'effectuer des attaques sous l'identité de ce nœud et de nuire à la réputation du nœud victime. De manière générale, l'authentification est un processus basé sur trois principes qui peuvent être définis en une seule phrase : "*c'est quelque chose qu'on est, quelque chose qu'on connaît*

et quelque chose qu'on a" [60]. En d'autres termes, "*c'est quelque chose qu'on est*" : c'est la biométrie, comme la rétine des yeux, l'empreinte digitale, etc., "*c'est quelque chose qu'on connaît*" : c'est un mot de passe ou une clé, etc., "*c'est quelque chose qu'on a*" : c'est une carte d'accès ou un certificat, etc.

Dans le cadre des réseaux filaires ou des réseaux sans fil avec infrastructure, le processus d'authentification est basé sur un tiers de confiance en qui toutes les entités du réseau ont confiance. Le tiers de confiance n'est que l'autorité de certification (CA) qui distribue les certificats aux nœuds qui ont le droit d'accéder à un certain service du réseau. Ce schéma d'authentification est centralisé, et est connu sous le nom d'infrastructure à clé publique (*PKI*) [61].

Appliquer le modèle PKI directement au réseau mobile Ad Hoc n'est pas possible pour des raisons de changement dynamique et fréquent de topologie réseau, car la disponibilité du service d'authentification est limitée en raison de la limite des capacités des nœuds (énergie, calcul, etc.).

2.2 *Confidentialité*

La confidentialité est un service essentiel pour assurer une communication privée entre les nœuds. C'est une protection contre les menaces qui peuvent causer la divulgation non autorisée d'informations, alors qu'il faut veiller au caractère privé de l'information. Elle est principalement basée sur la cryptographie, (*en particulier les algorithmes de chiffrement*). Le chiffrement peut être appliqué à différents niveaux des couches de protocoles. Au niveau de la couche réseau, nous pouvons citer le protocole ESP[1][62], qui assure la confidentialité aux datagrammes IP en utilisant le chiffrement. Les algorithmes de chiffrement, qu'ils soient symétriques ou asymétriques, nécessitent une clé de chiffrement pour chiffrer le message avant de l'envoyer à la destination. Cependant, la destination doit avoir une clé de déchiffrement pour pouvoir déchiffrer le message. Par conséquent, un mécanisme de gestion de clés adapté au contexte du réseau mobile Ad hoc est primordial, mais réaliser un tel mécanisme constitue un vrai défi.

2.3 *Intégrité*

Ce service assure que le trafic de la source à la destination n'a pas été altéré ou modifié sans autorisation préalable pendant sa transmission. C'est la protection contre les menaces qui peuvent causer la modification non autorisée de la configuration du système ou des données. Les services d'intégrité visent à assurer le bon fonctionnement des ressources et la transmission de données. Ces services assurent une protection contre la modification délibérée ou accidentelle et non autorisée des fonctions du système (*intégrité du système*) et de l'information (*intégrité des données*). Dans le réseau sans fil, le message peut être modifié pour des raisons non malicieuses, telles que la corruption du paquet au niveau de la propagation radio. Cependant, le risque qu'un nœud malicieux modifie le paquet

[1] Encapsulating Security Payload (ESP) : est un protocole appartenant à la suite IPsec, permettant de combiner plusieurs services de sécurité.

est toujours présent. En fait, ce service peut être appliqué de manière indirecte avec des protocoles de sécurité qui assurent la confidentialité ou l'authentification.

2.4 Non-répudiation

La non-répudiation est la possibilité de vérifier que l'émetteur et le destinataire sont bien les parties qui disent avoir respectivement envoyé ou reçu le message. Autrement dit, la non-répudiation de l'origine prouve que les données ont été envoyées, et la non-répudiation de l'arrivée prouve qu'elles ont été reçues. En d'autres termes, la non-répudiation permet de garantir qu'une transaction (émission/réception/action) ne puisse pas être niée. Cela est très pratique pour détecter et isoler les nœuds compromis. N'importe quel nœud qui reçoit un message (*paquet*) erroné peut accuser l'émetteur avec une preuve et cela permet de convaincre d'autres nœuds de la compromission du nœud émetteur. Généralement, la non-répudiation peut être atteinte seulement en utilisant la technique du certificat numérique. En effet, cette technique permet de prouver l'identité d'une personne qui possède sa propre clé privée.

2.5 Autres services de sécurité

Nous définissons d'autres paramètres de sécurité utilisés dans l'analyse des aspects de sécurité réseaux mobiles Ad hoc qui sont les suivants :

2.5.1 Disponibilité

La disponibilité consiste à assurer la continuité du service fourni par un nœud même en présence d'une ou plusieurs attaques. En d'autres termes, les nœuds doivent assurer la continuité des services réseau quelle que soit l'attaque. Pour cela, la protection contre les menaces qui peuvent causer la perturbation des fonctions du réseau est nécessaire pour assurer à tous les nœuds l'accès aux ressources réseau comme le routage, l'accès aux données, etc.

2.5.2 Autorisation d'accès

Un utilisateur ou un nœud autorisé à utiliser un service, doit posséder un certificat ou une référence de l'autorité de certification (*un tiers de confiance*). Cette référence spécifie les privilèges et les autorisations associées à l'utilisateur ou le nœud.

2.5.3 Contrôle d'accès

Le contrôle d'accès détermine les méthodes et les politiques qui permettent à un utilisateur ou à un nœud d'accéder aux données ou services. Seuls les nœuds autorisés peuvent former, détruire, rejoindre ou quitter le réseau. Parmi les approches de contrôle d'accès, nous pouvons citer :

- Contrôle d'accès discrétionnaire[2] (*DAC*) : Cette approche permet aux utilisateurs (nœuds) de définir eux-mêmes la politique de contrôle d'accès.
- Contrôle d'accès mandataire[3] (*MAC*) : Cette approche introduit un mécanisme de contrôle d'accès centralisé avec une politique d'autorisation formelle bien définie.
- Contrôle d'accès basé sur des rôles : Cette approche introduit le concept de rôles pour autoriser l'accès.

2.5.4 Anonymat

L'anonymat permet d'assurer l'absence de lien entre *l'identité d'un nœud dans le réseau* qui peut être : adresse (MAC ou IP) et *l'identité de l'utilisateur*. Cela consiste à assurer la sécurité des nœuds sensibles, dont le rôle est crucial pour le bon fonctionnement de réseau, où l'identité de l'utilisateur qui peut être la cible d'une attaque.

2.5.5 Auto-stabilisation

Un protocole de routage dans un réseau mobile Ad hoc doit être capable de détecter des anomalies et de reprendre le fonctionnement normal du protocole sans intervention humaine. N'importe quel protocole destiné aux réseaux mobiles Ad hoc doit être capable de s'auto-organiser et d'assurer la continuité du service quelle que soit la situation, sans lourde intervention.

2.5.6 Tolérance aux pannes

Le principe de tolérance aux pannes permet à un système de fonctionner avec la présence d'attaques ou de possibles pannes. Si un attaquant veut affecter le bon fonctionnement du réseau dans le but de créer un déni de service, l'attaque sera détectée par le protocole tolérant aux pannes et le service de reprise après pannes est enclenché pour réduire l'impact de l'attaque et assurer la continuité du service. Ce principe est très important pour la phase de réaction contre les attaques, dans le but de les isoler et de réduire leur impact.

2.5.7 Relations de confiance

Dans le cas où le niveau de sécurité physique est faible et où la relation de confiance entre les nœuds est dynamique, la probabilité d'avoir un problème de sécurité, augmente rapidement. Si un certain nombre de nœuds de confiance est compromis, cela risque de compromettre tous les modèles

[2] Le Contrôle d'accès discrétionnaire (DAC pour *Discretionary access control*) est un genre de contrôle d'accès, défini par le Trusted Computer System Evaluation Criteria (TCSEC) comme "*des moyens de limiter l'accès aux objets basés sur l'identité des sujets ou des groupes auxquels ils appartiennent*".
[3] Le Mandatory access control (MAC) ou contrôle d'accès obligatoire : est une méthode de gestion des droits des utilisateurs pour l'usage de systèmes d'information.

de confiance. Construire des liens de confiance au début n'est pas une tâche difficile, mais maintenir ces liens de confiance et supporter les changements dynamiques des liens est un vrai défi.

3. OUTILS DE SÉCURITÉ EXISTANTS

Comme on l'a vu dans la 5ème section du chapitre 2, les attaques se caractérisent pour la plupart, entre autres, par une corruption des paquets de contrôle et par des usurpations d'identité. Typiquement, un nœud malicieux va altérer le contenu de ces paquets en vue de créer des boucles de routage ou encore, supprimer des routes illégitimement. Ainsi, un protocole visant à sécuriser le routage doit permettre, pour prévenir ces attaques, de garantir l'intégrité et l'authenticité des paquets. C'est la raison pour laquelle les protocoles de routage sécurisés les plus aboutis recourent le plus souvent aux outils classiques que sont les fonctions de hachage et les mécanismes de chiffrement symétriques-asymétriques [63]. Ainsi, pour garantir l'authenticité d'un message, la solution la plus efficace consiste à munir chacun des nœuds de clés secrètes (privées), utilisées pour chiffrer puis déchiffrer les messages reçus (Figure 3.1) [64]. Dans la mesure où il existe une seule et unique clé pour chaque paire de nœuds possible, un tel procédé garantit la provenance de chaque message.

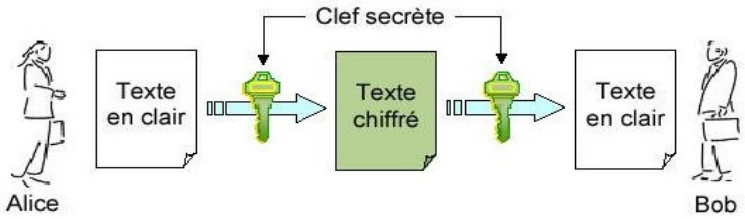


Figure 3.1 : Chiffrement symétrique "à clé secrète".

En raison du caractère fortement distribué des réseaux Ad hoc, certains protocoles ont préféré s'orienter vers les mécanismes de chiffrement asymétriques. Dans ce cas, on n'utilise plus une seule clé secrète pour chiffrer et déchiffrer un message mais un couple clé publique/clé privée. Les mécanismes de chiffrement asymétriques permettent eux aussi d'assurer l'authenticité des messages ou leur confidentialité. Dans le premier cas, l'émetteur d'un message le chiffre préalablement avec sa clé privée tenue secrète. Le destinataire peut ensuite le déchiffrer avec la clé publique de l'émetteur, qui peut-elle, être connue de tous. Puisque à chaque clé privée est associée une et une seule clé publique et sous réserve que le protocole utilisé soit suffisamment fiable, cette opération garantit l'authenticité du message. Si en revanche, le but recherché est la confidentialité, l'émetteur préfèrera chiffrer le message avec la clé publique du destinataire. Par la suite, seul ce dernier pourra déchiffrer le message, puisqu'il est le seul à posséder la clé privée correspondante. (Figure 3.2) [64]

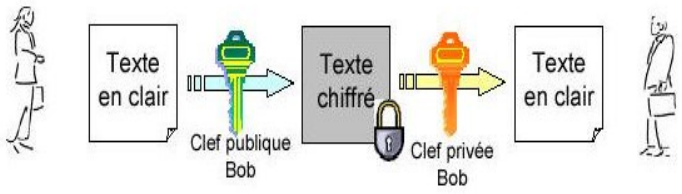


Figure 3.2 : Chiffrement asymétrique "à clé publique".

De tels mécanismes permettent indirectement de garantir également l'intégrité des messages. En effet, si ceux-ci sont corrompus en cours d'acheminement, ils ne peuvent plus ensuite être déchiffrés. Par conséquent, un message déchiffré est un message qui n'a pas été altéré. Cependant, s'agissant seulement de l'intégrité des paquets, ces mécanismes sont relativement coûteux dans la mesure où ils nécessitent de chiffrer puis déchiffrer l'intégralité du message. Une solution plus adaptée et plus économique consiste à recourir aux *fonctions de hachage à sens unique*. (Figure 3.3) [64]

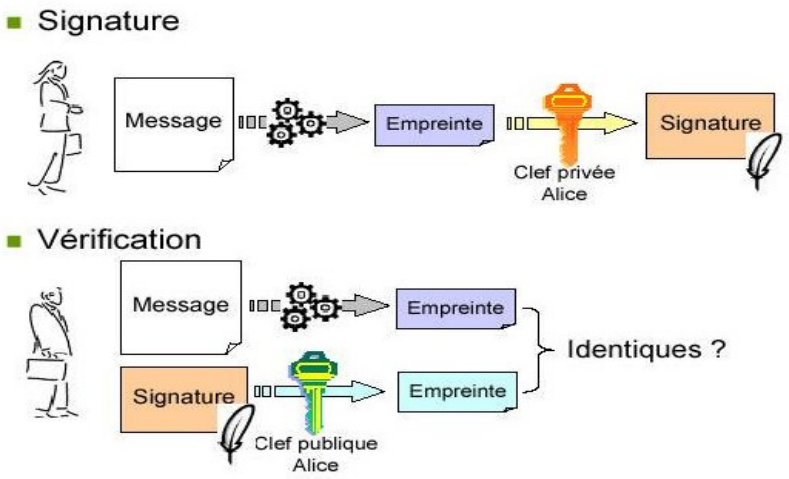


Figure 3.3 : Signature numérique avec fonction de hachage.

Les fonctions de hachage sont des objets mathématiques qui, à un ensemble de données fourni en entrée : $\{0,1\}^*$, associent un ensemble beaucoup plus petit "de l'ordre de quelques centaines de bits" $\{0,1\}^\rho$, ρ étant la longueur en bits de l'image de la fonction. On appelle cette image une *empreinte*, ou un *condensé*. En pratique, lorsque l'émetteur d'un message souhaite permettre au destinataire de vérifier son intégrité, il applique une fonction de hachage sur le message et y joint l'empreinte calculée. Par la suite, ce destinataire recalcule l'empreinte de lui-même et compare le résultat obtenu avec l'empreinte reçue. Si celles-ci sont différentes, cela signifie que le message a été altéré. Pour que l'intégrité soit formellement vérifiée, la fonction de hachage doit satisfaire aux propriétés suivantes :

- Il est très difficile de trouver le contenu du message à partir de l'empreinte (attaque sur la première pré-image).
- À partir d'un message donné et de son empreinte, il est très difficile de générer un autre message qui donne la même signature (attaque sur la seconde pré-image).
- Il est très difficile (c'est-à-dire que cela dépasse les capacités de calcul actuelles) de trouver deux messages aléatoires qui donnent la même signature (*résistance aux collisions*).

Si l'on désire garantir en plus de l'intégrité, l'authenticité d'un message, la fonction de hachage cryptographique peut être combinée à un mécanisme de chiffrement. Dans ce cas, on parle alors de code d'authentification de message HMAC[4] [65], le message est d'abord soumis à la fonction de hachage qui génère l'empreinte correspondante. Ensuite, celle-ci est chiffrée de manière à prouver à l'émetteur l'authenticité du message.

Une autre caractéristique intéressante des fonctions de hachage, est leur capacité à être utilisées de manière récurrente, pour produire des chaines de hachage à sens unique. La construction d'une chaine de hachage consiste ainsi à appliquer successivement une fonction de hachage sur la sortie précédemment calculée. En pratique, un nœud choisit un élément de départ $x \in (\{0,1\}^{\rho})$ et calcule une liste de valeurs $h_0, h_1, \dots h_n$ où $h_0 = x$ et $h_1 = H(h_{i-1})$ pour tout $i < n$. Par la suite, à partir d'un élément authentifié de la chaine de hachage, on peut aisément vérifier un élément antérieur en appliquant autant de fois que nécessaire la fonction de hachage et en comparant les valeurs des éléments. Par exemple, à partir d'une valeur authentifiée h_i, un nœud peut authentifier la valeur h_{i-3} en calculant $H(H(H(h_{i-3})))$ et en vérifiant que la valeur obtenue est identique à h_i.

Un grand avantage des chaines de hachage est qu'elles ne requièrent pas de grosses capacités de stockage et de calcul. À titre d'exemple, [66, 67] ont mis au point un mécanisme de stockage de chaines de hachage tel qu'une chaine composée de n éléments ne requiert que $O(\log(n))$ opérations de stockage et $O(log(n))$ opérations de calcul pour accéder à un des éléments. Ces caractéristiques en font un outil tout à fait adapté à une utilisation dans un réseau Ad hoc où les ressources sont par définition, limitées.

Ces procédés sont très efficaces mais certains d'entre eux ne sont pas tout à fait appropriés à l'environnement des réseaux Ad hoc. Ainsi, la distribution de clés secrètes au sein d'un réseau dont on ne connait pas tous les participants constitue un problème délicat. En effet, dans un contexte où les nœuds sont mobiles et où tout un chacun peut espionner les informations transitant à proximité, il est difficile de mettre en place un canal sûr pour échanger les clés. Celles-ci doivent donc être mises en place préalablement au déploiement du réseau, pendant la phase d'initialisation. Certains préconisent une distribution manuelle en dotant chaque nœud d'une carte à puce sur laquelle la clé sera stockée, en dur. Si cette solution s'avère envisageable pour des réseaux de petite taille et parfaitement contrôlés,

[4] keyed-hash message authentication code.

tels les réseaux militaires ou les terminaux d'un opérateur téléphonique, il n'en va pas de même pour des réseaux ouverts comme les réseaux citoyens ou encore les réseaux de forte densité de nœuds tels les réseaux de capteurs. Puisque chaque paire de nœuds susceptibles de communiquer ensemble doit posséder une clé, le nombre total de clés d'un réseau de n nœuds est de $n * (n-1)/2$. Cela peut représenter un nombre considérable de clés à gérer dans certaines circonstances.

La cryptographie asymétrique semble alors constituer une solution plus appropriée en raison de sa souplesse. Cependant, elle souffre également d'un défaut pénalisant dans le contexte Ad hoc. Ainsi, lorsque l'on désire envoyer un message chiffré, on doit y joindre un certificat délivré par une autorité de certification. Le rôle de ce certificat est de prouver que l'on possède bien la clé publique revendiquée pendant une certaine période. L'inconvénient de cette approche est que dans le contexte des réseaux Ad hoc qui, par nature, sont dépourvus de toute infrastructure, il n'est pas envisageable de recourir à une autorité de certification centralisée et fixe, comme on peut le faire dans les réseaux classiques.

Dans la suite de ce chapitre, nous allons voir que pour pouvoir être utilisé efficacement, ces mécanismes vont devoir être adaptés afin de satisfaire les contraintes du modèle Ad hoc.

4. SOLUTIONS ET MÉCANISMES DE SÉCURITÉ

La littérature est riche de propositions et solutions de contre-mesures aux attaques discutées en 5ème section du chapitre 2. Chacune d'elles se base sur un raisonnement différent suivant le type d'application visée. Parmi celles-ci, nous pouvons distinguer deux catégories principales : (i) celles qui sont établies à l'avance pour assurer la sécurité et empêcher à l'avance les attaques de nœuds compromis sur le réseau grâce notamment aux solutions à base de cryptographie et (ii) celles qui réagissent (adaptation/prise de décision immédiate) selon le comportement du voisinage, car ils visent à détecter en temps réel les attaques du réseau ainsi qu'à favoriser la coopération entre les nœuds afin de restreindre l'impact des nœuds malicieux. (Figure 3.4)

Dans chacune des sous sections suivantes, nous détaillons les solutions présentées dans la littérature pour chaque axe de recherche.

4.1 Protections basiques

Si les caractéristiques spécifiques des réseaux Ad hoc constituent souvent un obstacle à la sécurisation du routage, elles peuvent également être exploitées a contrario comme un atout, pour renforcer l'acheminement des données. C'est le cas par exemple de la redondance de routes. Chaque nœud dans un réseau Ad hoc est susceptible à tout moment de servir de routeurs. Dès lors, pour peu que le nombre de nœuds soit suffisant, il est souvent possible de trouver plusieurs chemins différents entre deux nœuds. Or, la plupart des protocoles classiques (*AODV*, *OLSR*, *ZRP*, etc.) ont justement la

faculté d'établir plusieurs routes entre deux nœuds s'échangeant des informations. Une solution simple consiste alors à profiter de cette multiplicité de routes pour sécuriser le transfert [68]. D'une part, lorsqu'un nœud malveillant est identifié, le protocole peut presque toujours trouver une route qui permette de le contourner. D'autre part, il devient possible de transmettre de l'information redondante à travers des routes additionnelles afin de permettre au destinataire de vérifier l'intégrité de l'information envoyée. On peut ainsi adjoindre des codes détecteurs d'erreur, correcteurs d'erreur, ou des hachages des données transmises. Par exemple, s'il existe n routes disjointes entre deux nœuds, on peut employer $(n - r)$ canaux pour transmettre les données et utiliser les "r" canaux restants pour transmettre l'information redondante.

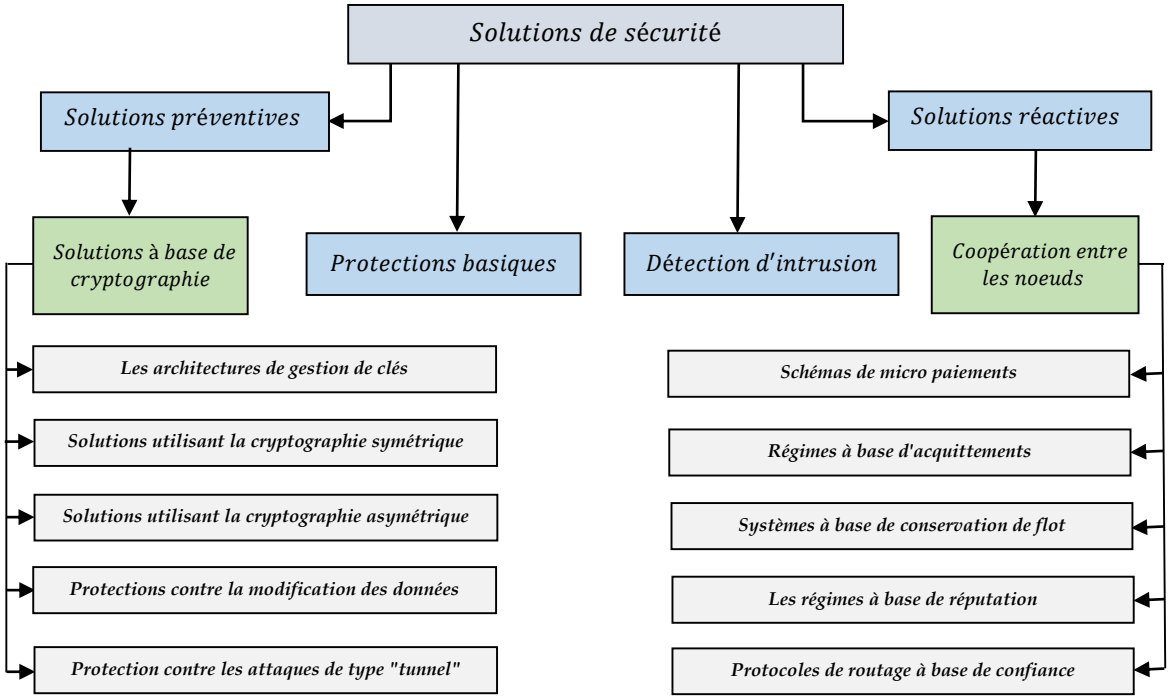


Figure 3.4 : Principales pistes de solutions de sécurité Ad hoc.

Bien qu'offrant un bon niveau de sécurité, cette technique a pour inconvénient de réduire sensiblement la bande passante disponible en augmentant le trafic de contrôle. D'autre part, elle ne répond pas à un certain nombre de problèmes évoqués en première partie comme l'usurpation d'identité, l'injection de faux paquets de signalisation ou la redirection de route.

Une autre approche consiste à utiliser la nature du médium, à savoir l'onde radio pour s'assurer que l'information est réellement transmise. En effet une autre des caractéristiques des réseaux Ad hoc étant un médium totalement ouvert avec un accès partagé, tous les nœuds peuvent écouter les informations transmises par leurs voisins à un saut. Ainsi, une solution [69], consiste à modifier le protocole de base *DSR* de manière à ce que chaque réponse à un "*RREQ*" fasse l'objet d'une confirmation par un voisin de l'émetteur. Lorsqu'un nœud reçoit un paquet de type *RREQ* auquel correspond une route valide

dans son cache de route, il répond bien sûr par un paquet de type *RREP*, mais envoie également un paquet de demande de confirmation (*CREQ*) auprès du premier voisin en aval. Celui-ci examine son cache de roule à la recherche d'une route vers la destination. S'il en trouve une, il répond à la source par un paquet (*CREP*) contenant cette information, dans le cas contraire, il ne répond rien. De son côté, le nœud source compare les informations envoyées par le premier nœud intermédiaire avec la confirmation reçue par le voisin. Si elles s'avèrent différentes ou plus simplement, si le voisin en aval n'a rien envoyé, le nœud source ne prend pas en compte la réponse du nœud intermédiaire et recherche une autre route.

On constate que ce procédé a pour objectif de sécuriser la découverte des routes en s'assurant que le chemin annoncé existe réellement. Cependant il est beaucoup trop simple et limitatif pour sa sécurité soit jugée suffisante. Tout d'abord, il ne prend en compte que la sécurisation des *RREP* des nœuds intermédiaires, ce qui ne représente, somme toute, qu'une amélioration de la technique de *source caching* du protocole *DSR*. Ensuite, il nécessite obligatoirement l'emploi d'outils supplémentaires, capables de fournir l'authentification des paquets car sinon, rien n'empêcherait le nœud intermédiaire de falsifier une confirmation et de l'envoyer au nœud source en usurpant l'adresse de son voisin. D'autre part, supposons qu'un autre nœud intermédiaire décide de supprimer la confirmation des paquets *RREP* qu'il relaie, la route ne sera jamais établie et le soupçon pèsera immédiatement sur le nœud à l'origine du *RREP*. La confirmation de route doit donc obligatoirement être transmise à travers un autre chemin. Enfin, si deux nœuds malveillants s'associent pour mener une attaque, le procédé de protection peut très bien être contourné. Le premier nœud envoie un *RREP* qui est confirmé par le second nœud à l'aide d'un *route confirmation reply* et une fausse route sera établie.

Ces mécanismes permettent d'offrir un niveau de sécurité supérieur aux protocoles classiques en permettant de renforcer le processus d'acheminement des paquets. En revanche, il ne s'avère pas suffisants dès lors qu'il s'agit de satisfaire les exigences de sécurité de bases que sont la confidentialité, l'authentification et l'intégrité.

4.2 Solutions basées sur la cryptographie

Les solutions basées sur la cryptographie sont souvent utilisées contre les attaquants externes. Elles font la distinction entre les nœuds qui sont autorisés à prendre part au réseau (*qui sont supposés se comporter correctement*) et les nœuds qui n'y sont pas autorisés et qui sont considérés a priori comme étant des attaquants. La plupart de ces solutions se basent sur des mécanismes de chiffrement et de signature numérique pour assurer l'authentification des nœuds et la confidentialité et l'intégrité des messages. Nous présentons dans ce qui suit les propositions utilisant les mécanismes cryptographiques pour sécuriser le routage dans les réseaux Ad hoc.

4.2.1 *Les architectures de gestion de clés*

Comme on vient de le voir, les outils mathématiques classiques peuvent être utilisés pour concevoir des protocoles de routage sécurisés. Détaillons maintenant les solutions et leurs limites.

4.2.1.1 Le Resurrecting duckling

Dans une volonté de permettre une distribution facilitée des clés dans un réseau Ad hoc, Franck Stajano et Ross Anderson ont proposé dans [49] un mécanisme pour échanger une clé secrète entre deux nœuds. Ce modèle, appelé *The Resurrecting duckling*, repose sur la relation de *maître/esclave* et sur le concept d'imprégnation. Ainsi, pendant une phase d'initialisation (avant son introduction au sein du réseau), un nœud esclave doit être imprégné par son nœud maître (éventuellement, le propriétaire) par le biais d'un contact physique (par exemple électrique). Lors de ce contact, une clé secrète est échangée en toute confidentialité. Par la suite, cette clé peut être utilisée pour chiffrer et authentifier des informations, comme une liste d'autres clés partagées par exemple.

Bien qu'innovante, cette approche laisse plusieurs questions en suspens. La première concerne la phase d'imprégnation. Si un contact physique est possible dans le cadre d'un petit réseau (*un piconet* [70] par exemple) avec un leader désigné, il devient moins envisageable dans le cadre d'un grand réseau ouvert. Le deuxième problème porte sur la gestion de clés. En effet, l'approche ne propose pas comment faire pour échanger une clé secrète entre chaque paire de nœuds du réseau. Par ailleurs, si l'un des nœuds est corrompu, toutes les autres clés liées à ce nœud peuvent se trouver menacées et rien n'est mentionné quant à la répudiation d'une clé. Une réinitialisation systématique paraît difficile à mettre en place.

4.2.1.2 SUCV

Dans [71], les auteurs ont mis au point une autre approche appelée *SUCV* (*Statistically Unique Cryptographically Verifiable identifiers and addresses*) dans laquelle chaque nœud construit une adresse basée sur sa clé publique. Chaque nœud génère une paire *clé publique/clé privée* et choisit ensuite son adresse calculée à partir de la clé publique, à l'aide d'une fonction de hachage cryptographique. Les auteurs proposent deux mécanismes. Dans le premier, l'adresse *IPv6* d'un nœud correspond au résultat complet de la fonction de hachage sur la clé publique. Dans l'autre approche, seuls les 64 bits les moins significatifs correspondent au résultat de la fonction de hachage. Ainsi, si un attaquant désire compromettre une adresse *SUCV* donnée, il devra effectuer 2^{63} (approximativement $4{,}8 \times 10^{18}$) essais pour trouver une clé publique dont l'empreinte est identique à celle de cette adresse *SUCV*. Si cet attaquant a la possibilité de calculer un milliard d'empreintes par seconde, il lui faudra approximativement 142 années pour trouver cette collision.

L'inconvénient de cette approche est qu'elle ne résout pas entièrement le problème de mise en place des clés. Ainsi, si dans un réseau normal, le problème consiste à obtenir une liste de couples (nœuds, clés publiques) de confiance, ici on doit malgré tout déterminer une liste de nœuds de confiance.

Une approche alternative consiste à définir une ou plusieurs autorités de certification. En effet, la seule présence d'une clé publique ne suffit pas, encore faut-il que les nœuds puissent vérifier la légitimité de la clé publique utilisée par chaque nœud, c'est là le rôle de l'autorité. Chaque nœud du réseau possède un certificat qui contient son adresse IP, sa clé publique et bien sûr, une signature de l'autorité de certification. Lorsqu'un nœud désire envoyer un message, il le signe et y joint son certificat. Par la suite, le nœud récepteur vérifie dans un premier temps le certificat puis utilise la clé publique contenue dans ce certificat pour vérifier la signature du message. Plusieurs problèmes se posent cependant. Le premier concerne la disponibilité de l'autorité, En effet, dans un réseau exempt de toute infrastructure fixe, la question de l'accès à l'autorité se pose pour vérifier le certificat. Certains liens se rompent, les nœuds sont amenés à bouger et ainsi, il n'est pas sûr que chaque nœud ait à tout instant un accès à l'autorité et donc au service de certification. Le deuxième problème concerne la dépendance mutuelle entre sécurité et routage.

En effet, pour valider un certificat auprès d'une autorité de certification, il faut au préalable établir une route, mais pour que cette route puisse être établie de manière sûre, il faut d'abord vérifier les clés publiques de chacun des nœuds qui la composent.

4.2.1.3 L'architecture de certification distribuée

Pour remédier aux contraintes induites par l'absence d'infrastructure centralisée, Zhou et Haas ont imaginé profiter des caractéristiques intrinsèques des réseaux Ad hoc pour concevoir une nouvelle approche de gestion des certificats. Ils ont ainsi imaginé un système de certification de clés [72] dont l'autorité est non plus confiée à une seule entité fixe mais qui est au contraire distribuée entre plusieurs nœuds du réseau. Ainsi, le service de certification obtenu revient à définir une autorité de certification distribuée disposant d'une paire de clés publique/privée. La clé publique est connue de chaque nœud du réseau, ce qui leur permet de vérifier en confiance tout certificat signé avec cette clé privée. La clé privée n'est connue d'aucun nœud particulier, mais se trouve en fait partiellement distribuée sur des nœuds appelés *contributeurs*. Ainsi, un nœud client qui souhaite obtenir les clés publiques des autres clients ou lancer des mises à jour pour changer sa propre clé publique, émet une requête vers le service de certification. Pour garantir un niveau suffisant de sécurité même dans un contexte distribué, le service de certification repose sur la cryptographie à seuil. Un schéma de cryptographie à seuil $(n, t+1)$ est conçu de telle manière que parmi les n nœuds qui se partagent la gestion des clés, "$t+1$" auront la possibilité de procéder aux opérations de chiffrement, tandis que t nœuds seuls en seront incapables, même en coalition, Ainsi, lorsque le service doit signer un certificat, chaque nœud serveur génère une signature partielle en utilisant sa clé privée, et transmet le résultat à un autre serveur appelé *assembleur* qui sera chargé d'assembler les portions de signature des "t" nœuds. Lorsque ce serveur a reçu "$t+1$"

signatures partielles correctes, il est capable de calculer la signature finale du certificat. On notera que ce rôle d'assembleur peut être rempli par n'importe lequel des "n" nœuds. Pour renforcer la robustesse du dispositif et déjouer la compromission éventuelle de ce serveur, les auteurs préconisent d'affecter éventuellement ce rôle à "$t+1$" nœuds simultanément (bien sûr, la phase de vérification des signatures en est alors considérablement alourdie). L'avantage de ce modèle [42] réside dans le fait que "t" nœuds malveillants complices ne peuvent créer de certificat valide puisque "$t+1$" signatures partielles valides sont nécessaires, Bien entendu, nous ne sommes pas à l'abri d'un attaquant qui génère systématiquement de fausses signatures, en vue de conduire à la création d'un certificat invalide. Toutefois, Le nœud assembleur a toujours la possibilité de vérifier la validité d'une signature en utilisant la clé publique du service. Dans le cas où la vérification échoue, l'assembleur se doit de désigner un autre ensemble de "$t+1$" signatures partielles. Cette procédure continue jusqu'à ce qu'il parvienne à générer une signature correcte.

Le point négatif de l'architecture proposée par Zhou et Haas est sa complexité de mise en œuvre [42]. En effet, la sécurité repose pour une grande partie sur le choix des nœuds assembleurs. Si le nombre de nœuds malicieux ou corrompus dépasse un certain seuil, le service devient inopérable. En outre, il est probable que le fait de requérir des certificats de plusieurs nœuds pour chaque message chiffré engendre un *overhead* conséquent au niveau de la charge réseau, dans la mesure où l'on doit envoyer et recevoir de l'information de tous les nœuds assembleurs.

4.2.1.4 L'approche de type PGP

Une autre solution [2] propose de s'affranchir du modèle de certification en ligne classique en s'inspirant du concept de graphes de certificats (les sommets du graphe représentent les clés publiques des utilisateurs tandis que les arêtes représentent les certificats) du protocole *PGP* (*pretty good privacy*). Dans ce modèle, chaque nœud signe des certificats pour les participants en qui il a confiance, en fonction de ses propres critères. Les certificats reposent sur une confiance transitive, c'est-à-dire que si A fait confiance à B et que B fait confiance à C, alors A fait confiance à C. Mais à la différence de *PGP*, les certificats sont stockés puis distribués par les nœuds eux-mêmes et non pas par un serveur en ligne. Ainsi, chaque nœud possède un "*dépôt de certificats local*". Par la suite, lorsque deux nœuds désirent mutuellement vérifier leurs identités, ils fusionnent leurs dépôts respectifs dans le but de trouver une chaîne de certificats qui les lie dans une relation de confiance.

Le succès de cette approche dépend en grande partie des caractéristiques des graphes de certificats mais également de la construction des dépôts de certificat locaux. D'autre part, avant d'être à même de générer des certificats, chaque nœud doit d'abord construire son dépôt de certificat, ce qui constitue une opération complexe, En outre, si le nombre de certificats révoqués devient trop important, les dépôts de certificats deviennent obsolètes dans la mesure où les chaines de certificats ne sont plus valides.

4.2.1.5 TESLA

Le protocole *TESLA* (*Timed Efficient Stream Loss-tolerant Authentication*) [73] a été conçu pour permettre une authentification de la source d'un flux multicast, tolérant les pertes. Le principe de base de *TESLA* est le suivant : l'émetteur d'un message lui associe un code d'authentification de message (MAC) obtenu à l'aide d'une clé secrète qu'il ne révèlera qu'après un certain délai. Plus en détail, voici comment cet émetteur procède.

Le mécanisme débute par l'élaboration des clés MAC. Pour cela, l'émetteur génère une série de clés $K_1, K_2, \dots K_t$ à l'aide d'une fonction de hachage à sens unique. Il détermine une première clé de manière aléatoire et calcule les clés suivantes en appliquant successivement la fonction de hachage $K_i = h(K_{i+1})$. Ensuite, il génère les clés MAC à l'aide d'une autre fonction de hachage à sens unique $K'_i = h'(K_{i+1})$ (Figure 3.5).

L'utilisation de deux fonctions de hachage distinctes est une précaution prise par les auteurs pour renforcer encore davantage la sécurité (l'utilisation d'une même fonction pour tous les calculs cryptographiques peut constituer une vulnérabilité potentielle exploitable par un attaquant).

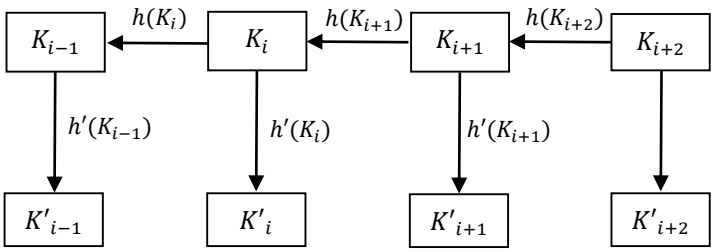


Figure 3.5 : L'utilisation des fonctions de hachage dans TESLA.

Par la suite, l'émetteur attache à chaque paquet un MAC calculé à partir de son contenu et généré grâce à la fonction de hachage à sens unique. Il divise ensuite le temps en plusieurs intervalles de durée fixe. Pendant un même intervalle, l'émetteur peut envoyer zéro ou plusieurs paquets. Une fois l'envoi effectué et à l'expiration d'un délai prédéfini, il peut divulguer la clé correspondante qui servira à authentifier le paquet (par exemple, la clé utilisée pendant l'intervalle i est divulguée pendant l'intervalle "$i + 3$").

De son côté, lorsque le récepteur reçoit un paquet comportant un indice d'intervalle "i", il doit estimer l'intervalle dans lequel se trouve l'émetteur à l'aide de son horloge locale (en prenant en compte le temps de transmission d'un paquet et une estimation de l'horloge de l'émetteur). Cette estimation sert à vérifier que l'émetteur n'est pas encore parvenu dans l'intervalle de temps où il divulgue la clé K_i. Si cette condition n'est pas respectée, l'intégrité n'est plus formellement garantie et le paquet est rejeté. Dans le cas contraire, le récepteur ne peut pas pour l'instant vérifier l'authenticité du message envoyé

pendant l'intervalle i sans la clé correspondante K_i qui sera délivrée plus tard. Il enregistre donc un triplet (contenant le message, l'indice de l'intervalle ainsi que le MAC) dans un tampon jusqu'à ce qu'il reçoive la clé K_i. Une fois celle-ci reçue, le récepteur s'assure de sa légitimité en la hachant successivement plusieurs fois et en comparant l'empreinte obtenue à la valeur d'une clé antérieure. Pour un certain nombre de passes successives, les valeurs doivent être identiques. Ainsi, en notant "d" le délai de divulgation des clés, et K_v $(v < i - d)$ une clé antérieure, on doit avoir $K_v = h^{(i-d-v)}K_{i-d}$.

On peut noter ici un des principaux avantages de *TESLA* lié aux propriétés des chaînes de hachage, à partir d'une clé révélée, on peut calculer toutes les clés précédentes, de sorte que même si plusieurs paquets d'un même intervalle sont perdus [42], un nœud est toujours capable de les vérifier à partir d'une clé obtenue dans un intervalle ultérieur. Ainsi, si la valeur de $(i - v)$ est supérieure à 1, le récepteur peut vérifier l'authenticité de tous les paquets enregistrés pendant les intervalles compris entre $(v + 1)$ et $(i - 1)$. Ceci caractérise la capacité de tolérance aux pertes de *TESLA*.

Une autre propriété importante est le caractère unidirectionnel des flux, c'est-à-dire d'une source vers une ou plusieurs destinations. La source divulgue ainsi les clés correspondant aux intervalles des paquets envoyés indépendamment du nombre de récepteurs. C'est cette capacité de passer à l'échelle qui permet à *TESLA* d'être utilisé dans le cadre de flux multicast.

4.2.2 Solutions utilisant la cryptographie symétrique

4.2.2.1 SRP

Les auteurs de [74] ont proposé un protocole de routage sécurisé nommé *SRP* (*Secure Routing Protocol*), spécialement adapté aux caractéristiques du protocole *DSR* et du protocole de routage interzone *ZRP*. Ainsi, ils ont conçu *SRP* comme une extension de l'en-tête des paquets *RREQ* et *RREP*. SRP utilise des numéros de séquence à l'intérieur des requêtes, de manière à garantir leur fraîcheur, cependant, ce numéro de séquence ne peut être vérifié qu'au niveau de la destination. Il établit en outre des associations de sécurité, entre les nœuds communicants uniquement. Cette association est ensuite utilisée pour authentifier les paquets *RREQ* et *RREP* par le biais de MAC. Au niveau de la destination, *SRP* permet de détecter des modifications de paquets de type *RREQ* tandis qu'au niveau de la source, c'est l'intégrité des *RREP* qui sera analysée.

Puisque *SRP* ne nécessite des associations de sécurité qu'entre les nœuds communiquant entre eux, il est relativement léger. En contrepartie, certains défauts ont assez pénalisants et limitent son intérêt, Tout d'abord, *SRP* ne sécurise pas le mécanisme de maintenance des routes et délègue cette tâche à un autre protocole. De plus, *SRP* ne permet pas de détecter les modifications portant sur les informations de routage habituellement soumises à modification lors du routage. Par exemple, un nœud peut aisément corrompre voire supprimer le contenu de la liste de nœuds comprise à l'intérieur d'un paquet de type *RREQ*. Enfin, l'intégrité des messages n'étant vérifiée qu'au niveau des nœuds source

et destination, un attaquant eut toujours corrompu des paquets de manière à gaspiller les ressources du réseau en retransmissions inutiles.

4.2.2.2 SAR

Le protocole *SAR* (*Security-aware Ad hoc Routing protocol)* [75] se base lui aussi sur des procédés de chiffrement symétrique. Il a été élaboré à l'origine pour prévenir les attaques de type "*trou noir*" qui consiste à supprimer l'intégralité des paquets au niveau d'un nœud malicieux. À l'instar des protocoles précédents, *SAR* est conçu pour être employé conjointement avec des protocoles réactifs tels qu'*AODV* ou *DSR*. Il utilise la notion de "*niveaux de confiance*" pour établir la sécurité d'un chemin. Ainsi, lorsqu'un nœud désire établir une route avec un certain niveau de sécurité, il génère un nouveau paquet *RREQ* indiquant le niveau requis. Par la suite, le mécanisme de découverte de routes diffère légèrement du schéma classique des protocoles réactifs en ce sens que seuls les nœuds satisfaisant le niveau de sécurité requis peuvent rediffuser la requête à ses voisins. Dans le cas contraire, la requête est rejetée par le nœud. Une fois la route établie jusqu'à la destination, celle-ci génère en retour un paquet *RREP* avec le même niveau de sécurité. Dans l'éventualité où aucune route en retour ne garantit le niveau de sécurité requis, celui-ci peut être ajusté par le nœud source.

Bien sûr, cette approche implique de lier l'identité d'un nœud à un certain niveau de sécurité. Pour ce faire, il existe une clé secrète pour chaque niveau de sécurité défini et celle-ci doit être distribuée à tous les nœuds du réseau satisfaisant ce niveau de sécurité. Le contenu ainsi que l'en-tête des paquets sont ensuite chiffrés par la clé de sorte que les nœuds de niveau inférieur ne puissent pas le lire, Par conséquent, même l'information sur la topologie peut être cachée aux nœuds non sûrs.

Cette capacité à partitionner le réseau en fonction de différents niveaux de sécurité fait de *SAR* un protocole original. En contrepartie, il souffre de plusieurs défauts importants, le principal réside dans la distribution des clés [42]. Celle-ci doit être effectuée préalablement à la mise en place du réseau, par le biais d'un canal sûr. Ensuite, on peut imaginer que les nœuds de plus hauts niveaux de confiance sont utilisés pour distribuer les clés correspondant à des niveaux inférieurs. Mais ceci ouvre la voie à des attaques sévères de type usurpation d'identité si un nœud vient à être corrompu. En effet, dans ce cas, ce sont les clés de tous les niveaux de sécurité inférieurs qui deviennent obsolètes, menaçant de fait, la sécurité globale du réseau. D'autre part, le fait de chiffrer et déchiffrer tous les paquets (y compris les en-têtes) risque d'avoir un impact important sur les ressources du réseau. Et ceci peut être utilisé par un nœud malicieux pour lancer une attaque de type déni de service. Enfin, un effet de bord inhérent à cette approche est que Les routes ne sont plus optimales en termes de sauts. Encore, le taux d'établissement d'une route dépend directement du nombre de nœuds de confiance mais aussi et surtout, de leur disposition. Et il est probable que certaines topologies ne sont pas adaptées à cette approche.

4.2.2.3 Ariadne

Les auteurs dans [76], ont mis au point un protocole de routage réactif sécurisé : *Ariadne*, inspiré du protocole classique *DSR* et s'appuyant uniquement sur des mécanismes de chiffrement symétriques. L'enjeu était de proposer un protocole qui puisse être implémenté aussi bien sur des portables puissants que sur des assistants personnels, c'est pourquoi les auteurs ont choisi de l'associer trois méthodes d'authentification, afin de s'adapter aux capacités de calcul des nœuds :

- La première est basée sur le partage d'un secret entre chaque paire de nœuds et requiert par conséquent une configuration au préalable de $n * (n-1)/2$ clés, avec n le nombre de nœuds dans le réseau.
- La seconde est basée sur l'utilisation de la cryptographie asymétrique, ce qui rend particulièrement couteuses les opérations de génération et de vérification de signatures.
- La dernière s'appuie sur le partage d'un secret entre chaque paire de nœuds communicants combiné à une authentification par diffusion qui est supportée par le protocole *TESLA*. Cette méthode permet à une destination d'authentifier la source d'un message diffusé sur le réseau, à la condition que les nœuds aient une synchronisation approximative de leurs horloges.

Contrairement à *SRP*, *Ariadne* vise à limiter les attaques par modification des données variables inclues dans les messages de contrôle. Pour ce faire, *Ariadne* permet à un nœud destinataire d'authentifier l'initiateur de la demande de chemin grâce à code d'authentification de message (MAC). Ensuite, il est requis que chaque nœud intermédiaire impliqué dans un processus de découverte de chemin signe la nouvelle information contenue dans le message reçu avant de le propager. Ce mécanisme permet à un nœud destinataire d'authentifier chaque nœud inclus dans la demande avant d'envoyer un message de réponse. *Ariadne* a recours à l'utilisation de chaines de hachage à sens unique pour garantir l'intégrité de la liste des nœuds incluse dans la demande. Grâce à ce mécanisme, aucun nœud intermédiaire peut supprimer ou ajouter l'adresse d'un nœud présent dans la liste sans que cette modification passe inaperçue.

Dans la phase de maintenance, pour signaler une rupture de lien sur un chemin, un nœud envoie en direction de la source un message d'erreur signe dans lequel il annonce le lien à l'origine de l'erreur. Tous les nœuds appartenant au chemin (*et a fortiori la source*), après authentification de son émetteur, réagissent à ce message en invalidant le chemin.

La prévention contre les attaques par modification offerte par *Ariadne* génère un surcoût : au fur et à mesure de la propagation d'un message de découverte, un MAC est apposé par chaque nœud intermédiaire, il s'ensuit que la taille de ce message augmente linéairement avec la longueur du chemin. Concernant les différentes méthodes d'authentification d'*Ariadne*, il est à noter que l'utilisation du protocole *TESLA* présente l'avantage d'être plus souple dans le sens où elle s'affranchit de la distribution

des clés privées à toutes les paires de nœuds. En revanche, *TELSA* occasionne une augmentation du délai d'authentification, ce qui vient au détriment de la réactivité du protocole.

Dans *Ariadne*, les mécanismes de sécurité mis en œuvre sont efficaces contre les attaquants indépendants. Un attaquant seul ne peut pas supprimer les identités de nœuds dans le chemin annoncé dans un message de demande sans être détecté, puisque la vérification des MAC par la source ou par la destination échouera. En revanche, il affiche des faiblesses face aux attaques par collusion de nœuds internes. Dans les travaux de [77], les auteurs montrent comment un attaquant peut s'ajouter en tant que successeur d'un nœud intermédiaire légitime dont il n'est pas voisin dans un chemin en construction entre une source et une destination. Cette attaque aboutit éventuellement à l'utilisation d'un chemin inopérant par la source, puisqu'aucun message ne peut franchir le lien entre le nœud légitime et l'attaquant.

Dans d'autres travaux [78], les auteurs décrivent une autre attaque possible où deux attaquants en collusion parviennent à supprimer une séquence de nœuds intermédiaires dans le descriptif d'un chemin en construction. Bien que physiquement situés à différents endroits le long du chemin en construction, la particularité de ces deux attaquants est qu'ils partagent une même identité réseau.

4.2.2.4 endairA

Dans le but de corriger certains défauts de sécurité du protocole *Ariadne*, Buttyán et Vajda ont proposé une adaptation nommée facétieusement *endairA* [77]. Contrairement à *Ariadne* qui protège la construction du chemin dans la phase de protocole de découverte, ici ce ne sont que les réponses (en point à point) à une demande de chemin qui sont protégées par des signatures numériques. Au fur et à mesure que la réponse à la demande de chemin revient vers la source, chaque nœud intermédiaire vérifie son appartenance au chemin, vérifie que les nœuds suivant et précèdent dans le chemin sont ses voisins, et appose une signature numérique. Cette signature est calculée à partir de tous les champs, incluant le chemin et les signatures des nœuds intermédiaires précédents. Finalement, il diffuse le message vers le prochain nœud sur le chemin de retour vers le nœud source. À la réception d'une réponse, le nœud source s'assure que le message a bien été délivré par un voisin, puis vérifie toutes les signatures pour le chemin formé.

De manière similaire à *Ariadne*, l'inconvénient est que la taille du message de réponse augmente linéairement avec le nombre de nœuds appartenant au chemin. En outre, *endairA* est vulnérable à une attaque par collusion de nœuds internes [18].

4.2.2.5 Secure OLSR

Dans la littérature, plusieurs extensions de sécurité pour le protocole *OLSR* ont été proposées [79, 80]. Leur point commun réside dans l'utilisation de signature numérique pour assurer

l'authentification et intrinsèquement l'intégrité des messages de contrôle. Une telle authentification peut être réalisée soit de saut en saut, soit de bout en bout.

Dans *Secure OLSR* [79], les auteurs ont proposé une approche d'authentification de saut en saut dans laquelle chaque nœud signe les paquets *OLSR* au fur et à mesure de leur retransmission. Ainsi, à la réception d'un paquet *OLSR* (un tel paquet pouvant contenir plusieurs messages *OLSR* de type *HELLO* et *TC*), un nœud intermédiaire vérifie la signature du nœud précèdent, la retire, puis appose sa propre signature. Cette approche permet d'inclure dans le calcul de la signature numérique les champs devant être modifiés en transit, tels que le *TTL*[5] et le nombre de sauts. Cependant, la signature assure seulement que le nœud qui a transmis le trafic est bien celui qui a signé le paquet, mais n'apporte aucune garantie sur l'intégrité du paquet original. Ici, les auteurs suggèrent l'utilisation de clés symétriques et d'une fonction de hachage telle que *SHA-1*[6] pour la génération des signatures numériques.

De manière similaire à *Secure OLSR*, les auteurs de [80] ont proposé une extension de sécurité pour le protocole *OLSR* basée sur l'utilisation de signatures numériques. Une première différence se situe au niveau du type des données protégées. Dans leur approche, une signature numérique est associée à chaque message de contrôle *OLSR* (c'est-à-dire, *HELLO* ou *TC*) et non plus à chaque paquet *OLSR*. Ensuite, les auteurs proposent une approche d'authentification de bout en bout selon laquelle un nœud récepteur d'un message de contrôle authentifie le nœud d'origine plutôt qu'un nœud intermédiaire dans son cheminement. Ici, les champs *TTL* et le nombre de sauts ne sont pas protégés par la signature, car ces derniers doivent être modifiés en transit par chaque nœud intermédiaire. En remplacement au *TTL* et pour déterminer si un paquet est trop ancien et s'il doit être rejeté, les auteurs proposent d'horodater chaque message *OLSR*. En outre, cette information temporelle est un indicateur qui sert à détecter les rejeux de messages de contrôle.

L'authentification des messages de contrôle représente une première ligne de défense pour contrecarrer efficacement les attaques externes contre le protocole *OLSR*. Or à elle seule, l'authentification n'empêche pas l'inoculation de fausses informations de routage par des attaquants internes au réseau. Pour pallier ce problème, d'autres méthodes plus originales ont été proposées (par exemples : *AdvSig*[7] [81, 82], *AdvSig+* [83]).

4.2.3 Solutions utilisant la cryptographie asymétrique

Les protocoles détaillés dans cette section supposent généralement l'existence d'un système de gestion et de distribution de clés préétabli. Ils utilisent ensuite les mécanismes de sécurité décrits au section 3 pour assurer l'intégrité et l'authenticité des paquets de contrôle. Pour prévenir les risques d'usurpation dans un réseau, le protocole doit pouvoir être à même de garantir l'authentification des

[5] Time To Live.
[6] Secure Hash Algorithm.
[7] Advanced Signature.

nœuds. Pour cela, les mécanismes de chiffrement apparaissent comme les plus efficaces. La différence entre les différents protocoles se fait alors sur le choix du procédé cryptographique.

4.2.3.1 SAODV

M.G. Zapata et N. Asokan ont mis au point un protocole dédié à la sécurisation du protocole AODV, appelé *SAODV* (*Secure Ad hoc On demand Distance Vector*) [84]. L'idée principale de *SAODV* consiste à utiliser des signatures afin d'authentifier la plupart des champs des paquets *RREQ* et *RREP* et d'utiliser des chaines de hachage pour protéger l'intégrité du compteur de sauts. Ainsi, *SAODV* constitue-t-il une extension d'*AODV* avec des signatures, afin de contrer les attaques de type "usurpation d'identité". *SAODV* nécessite la présence d'une autorité de certification afin de vérifier les paquets signés, assurant ainsi leur authenticité. Dans *SAODV*, chaque paquet *RREQ* inclut une extension de signature simple. L'initiateur du paquet choisit un nombre de sauts maximal en se basant sur une estimation du diamètre du réseau et il génère ensuite une chaîne de hachage à sens unique d'une longueur égale au nombre de sauts, plus un.

A l'instar du protocole *SEAD* détaillé au section 4.2.4, les chaînes de hachage dans *SAODV* sont utilisées pour authentifier les métriques présentes dans l'en tête des paquets de signalisation. L'initiateur *S* du paquet *RREQ-SSE* inclut le type de message (*RREQ*), un identifiant (i), l'adresse du nœud source (respectivement, Destination) ainsi qu'un numéro de séquence Seq_S (respectivement, Seq_D). En outre, cet en-tête inclut également un élément de la chaîne de hachage (h_0) basé sur l'estimation du nombre de sauts (N) de l'en-tête *RREQ*. Cette valeur est appelée l'authentifiant du nombre de sauts. Ainsi, si par exemple les valeurs de la chaîne de hachage $h_0, h_1, \dots, h_N$ ont été générées de sorte que $h_i = H(h_{i+1})$, alors l'authentifiant du nombre de sauts h_i correspond à un nombre de sauts de valeur $(N - i)$. Par la suite, le nœud source signe le tout à l'aide de sa clé privée K_S, ajoute un compteur de sauts et l'empreinte correspondante. Avant de relayer une requête *RREQ-SSE*, un nœud commence par vérifier l'authenticité du message afin de s'assurer que chaque champ est valide. Il supprime ensuite les éventuelles duplications (paquet en provenance de plusieurs nœuds). Il incrémente ensuite le compteur de sauts, le hache, ajoute l'empreinte et rediffuse le tout. Lorsque la requête parvient jusqu'à la destination, celle-ci vérifie son authenticité. Si la requête est invalide, elle est simplement supprimée. Autrement, le processus est similaire à *AODV* : la destination répond par un paquet *RREP-SSE* très similaire à la requête *RREQ-SSE*. La différence se situe dans la présence du champ *lifetime* qui correspond au nombre de nœuds exact pour renvoyer la réponse. Le paquet est ensuite signé et complété par un compteur de sauts de manière identique.

À l'exception du nombre de sauts et de son authentifiant, les champs contenus dans les en-têtes des paquets *RREQ* et *RREQ-SSE* ne sont pas modifiables et peuvent donc être aisément authentifiés en vérifiant la signature dans l'extension *RREQ-SSE*. Lorsqu'il relaie une requête *RREQ*, un nœud peut dans *SAODV* authentifier d'abord le paquet *RREQ* pour s'assurer que chaque champ est valide. Ensuite, il effectue une suppression des paquets dupliqués afin de ne pas retransmettre plus d'un *RREQ* pour

chaque exploration de route. Le nœud incrémente ensuite le nombre de sauts de l'en-tête *RREQ*, calcule l'empreinte qui va authentifier le nombre de sauts et rediffuse la requête ainsi que l'extension *RREQ-SSE*. Lorsque la requête parvient à destination, celle-ci vérifie l'authentifiant dans l'extension. Si la requête est valide, la destination retourne un *RREP* comme dans *AODV*. Comme pour le *RREQ*, le seul champ modifiable des *RREP* est le nombre de sauts. Par conséquent, la sécurisation se fait exactement de la même manière.

SAODV utilise également les signatures pour protéger les messages *RRER* lors du mécanisme de maintenance de route (route maintenance). Ainsi, chaque nœud utilisant *SAODV* signe les messages *RRER* qu'il émet. En revanche, les nœuds ne changent pas l'information sur le numéro de séquence lorsqu'ils reçoivent un paquet *RRER* car le nœud destination n'authentifie pas le numéro de séquence.

Ce protocole assure une bonne authentification des messages de contrôle ainsi qu'une bonne intégrité. Cependant, l'utilisation de chaînes de hachage ne permet pas d'empêcher à 100% les attaques sur le nombre de sauts. Ainsi, bien que le hachage du nombre de sauts empêche un éventuel nœud malicieux d'annoncer des routes plus courtes qu'en réalité, rien n'empêche un attaquant d'augmenter arbitrairement la longueur des routes, En effet, un tel nœud peut appliquer la fonction de hachage plusieurs fois consécutives avant de relayer un paquet, la route apparait ensuite plus longue qu'elle n'est en réalité.

D'autre part, dans l'éventualité où il y aurait plusieurs attaquants complices, une attaque de type tunnel peut toujours être lancée et le nombre de sauts peut même être décrémenté à l'arrivée, de manière transparente pour les autres nœuds.

4.2.3.2 ARAN

Les concepteurs du protocole *ARAN* (*A secure Routing protocol for Ad hoc Network*) [56] ont également choisi d'utiliser la cryptographie à clés publiques pour sécuriser les routes. *ARAN* est un protocole à la demande, qui fournit un service d'authentification de saut en saut par le biais d'une infrastructure à clés publiques. Il suppose donc l'existence d'un serveur d'authentification "*T*", dont le rôle est de gérer les certificats et dont la clé publique est connue de tous les participants. Ainsi, avant d'entrer dans le réseau, chaque nœud doit s'identifier auprès du serveur et solliciter auprès de lui un certificat qui lui servira à signer les messages qu'il enverra. Ce certificat contient l'adresse IP du nœud, sa clé publique, une première estampille qui rend compte de la date de création du certificat, et une seconde qui indique sa date d'expiration. De manière classique, ce certificat est ensuite signé par "*T*" et doit être mis à jour régulièrement.

Le principe d'*ARAN* est de sécuriser le mécanisme de découverte de routes de nœuds en nœud. Ainsi lorsqu'un nœud désire envoyer un message, il génère, signe puis diffuse un paquet de type *RDP* (*Route Discover Packet*). Par la suite, chaque nœud intermédiaire recevant ce paquet vérifie le certificat du nœud précédent, appose son propre certificat et rediffuse le paquet. Une fois ce paquet arrivé au

nœud destination, celui-ci vérifie à son tour le certificat et répond en unicast, par un message de type *REP* (*REply Packet*) qui est à son tour vérifié de nœuds en nœuds.

Le protocole *ARAN* spécifie également comment protéger le mécanisme de maintenance des routes. Ainsi, lorsqu'un nœud intermédiaire détecte qu'une route est rompue, il envoie un paquet de type *route error* (*ERR*) au nœud suivant en amont (en direction de la source). Ce paquet inclut les adresses des nœuds source et destination, le certificat du nœud intermédiaire ainsi qu'un nonce[8] et une estampille de temps. Le paquet est ensuite relayé sans être resigné par les nœuds intermédiaires.

Parce que les paquets ne contiennent aucun compteur de sauts et surtout parce que l'authentification est effectuée de nœuds en nœuds, d'éventuels nœuds malicieux ne peuvent pas créer de boucles de routage, ni rediriger le trafic en insérant des adresses non légitimes dans les paquets de découverte de routes. En ce sens, *ARAN* fait preuve d'une grande robustesse contre ce type d'attaques. En contrepartie, l'utilisation de mécanisme de chiffrement à clé publique ouvre la voie à des attaques de type déni de service.

En effet, dans ce protocole, pour chaque paquet de découverte do route, il faut vérifier le certificat fourni, déchiffrer le paquet, le re-chiffrer avec sa propre clé et apposer son certificat. Lorsque le nombre de paquets devient important, cela peut se révéler extrêmement coûteux. Aussi une attaque par déni de service consistera à inonder le réseau de faux paquets de contrôle, dont la vérification va monopoliser exagérément les ressources des nœuds. D'autre part, si un nœud ne peut effectuer cette vérification en temps réel, il peut dire amené par un attaquant à supprimer certains paquets aléatoirement, y compris des paquets valides.

4.2.4 Protections contre la modification des données

Les chaines de hachage sont un outil très efficace et permettent d'offrir une protection très satisfaisante à bien moindre coût par rapport aux approches cryptographiques détaillées précédemment. Ainsi le protocole *SEAD* (*Secure Efficient distance vector Routing for mobile Ad hoc networks*) propose de renforcer la sécurité du protocole *DSDV* en utilisant les chaînes de hachage à sens unique. Celles-ci permettent de prévenir d'éventuels attaquants d'incrémenter artificiellement le nombre de sauts dans l'en-tête des paquets de signalisation. Un nœud génère une chaîne de hachage et la décompose en plusieurs segments de m éléments :

$$((h_0, h_1, \dots, h_{m-1}), \dots, (h_{km}, h_{km+1}, \dots, h_{km+m-1}), \dots, h_n)$$

$$\text{avec } k = {}^{m}/_{n-i}$$

[8] Nonce (Number once) désigne ici un nombre arbitrairement choisi et utilisable une seule fois, dont le but est d'éviter les attaques de type "rejeu".

m correspondant au diamètre maximal du réseau et i étant le numéro de séquence (Figure 3.6).

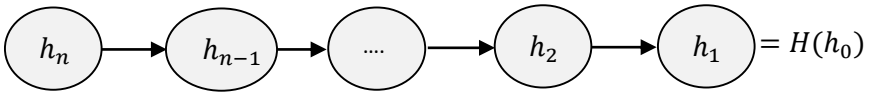


Figure 3.6 : Les chaines de hachage dans SEAD.

Puisque $h_i = H(h_{i-1})$, connaissant h_i, il est facile de vérifier l'authenticité de h_j, tant que j reste inférieur à i. De plus, comme des fonctions de hachage différentes sont utilisées pour des diamètres et des métriques différentes, un attaquant ne peut jamais forger une valeur de métrique inférieure ou un plus grand numéro de séquence. Enfin, le protocole *DSDV* spécifie que lorsqu'un nœud reçoit un message de signalisation, il met à jour sa table de routage si le numéro de séquence est plus grand ou identique avec une métrique inférieure. Donc, *SEAD* permet d'empêcher un attaquant potentiel de décrémenter artificiellement le nombre de sauts ou d'incrémenter le numéro de séquence des paquets.

En plus de leurs travaux sur *SRP*, [74] ont également mis au point un mécanisme destiné à sécuriser les protocoles de routage à état de lien, appelé *SLSP* (*Secure Link State Protocol*). À l'instar de *SEAD*, ce protocole utilise les signatures numériques ainsi que les chaînes de hachage à sens unique pour garantir l'intégrité des mises à jour de l'état des liens. *SLSP* peut être utilisé seul, de manière indépendante ou bien comme le protocole de routage interzone (*IARP*) qui est une composante du protocole *ZRP*. Le protocole *SRP* comporte quatre mécanismes principaux, à savoir : un protocole de surveillance des voisins (*NLP*), un protocole de distribution de clés (*PKD*), un protocole de mises à jour des états des liens (*LSU*) et enfin un mécanisme de prévention des attaques de type déni de service.

4.2.5 Protection contre les attaques de type "tunnel"

Les procédés cryptographiques employés dans les schémas précédents permettent de contrer efficacement un nombre important d'attaques. Pourtant, aucun d'entre eux, qu'il soit asymétrique ou à clés secrètes, ne permet de remédier au problème du tunnel (*Wormhole*). En effet, même si toutes les entrées d'un chemin semblent parfaitement identifiées, rien n'empêche un nœud chargé de transférer un paquet, de requérir parallèlement une route jusqu'à un nœud complice et de transférer le paquet encapsulé vers ce complice, lequel sera ensuite chargé d'acheminer le tout vers la destination.

Plusieurs solutions peuvent être envisagées pour résoudre ce problème. Tout d'abord, lors du processus de découverte de route, le paquet *RREQ* est inondé à travers le réseau et puisque le tunnel passe forcément par un nombre de nœuds plus important, si la destination établit le temps comme critère de choix d'un chemin, il y a fort à parier que la route passant par le tunnel ne sera pas choisie car elle sera moins rapide. D'autre part, on peut imaginer que les nœuds situés autour du premier nœud malveillant relaient le paquet jusqu'à la destination avant même que les nœuds complices aient eu le temps de l'encapsuler dans un tunnel.

Toutefois, ces solutions ne sont pas viables en toutes circonstances et notamment dans le cas où le nœud complice est un nœud indispensable sur le chemin. C'est pourquoi, les chercheurs ont mis au point une parade basée sur la localisation des nœuds d'une part et sur leur synchronisation temporelle d'autre part : *Packet Leashes* [85]. Dans la version de base, l'émetteur d'un paquet y inclut sa localisation et un horodateur correspondant à son horloge lors de l'émission. Lorsque la destination reçoit le paquet, elle compare ces valeurs avec sa propre localisation et son horloge au moment de la réception du paquet. Si les deux nœuds sont synchronisés à un coefficient près, le destinataire peut estimer, à partir des marqueurs temporels, une approximation de la distance qui les sépare et ainsi vérifier si cela correspond bien à la distance réelle. Néanmoins, il existe certaines circonstances pour lesquelles cette technique est inefficace. C'est le cas par exemple lorsque des obstacles s'immiscent entre deux nœuds voisins. Dans de telles circonstances, un schéma de protection basé sur la corrélation entre distances et temps de transfert ne pourrait empêcher une attaque de type tunnel.

C'est pourquoi les chercheurs ont développé un deuxième schéma dans lequel seule la métrique temporelle est prise en compte. Pour ce faire, les nœuds doivent être synchronisés entre eux à quelques microsecondes, voire nanosecondes près, cette différence doit être connue de tous les nœuds. Le procédé est alors identique lorsqu'un paquet est envoyé, on y inclut un horodateur (horloge d'émission). Ensuite, le nœud destination compare cette valeur avec son horloge au moment de la réception du paquet. Il est ainsi capable de déterminer si la distance parcourue est raisonnable en comparant le temps de transfert avec la vitesse de propagation de l'onde, Une variante consiste à inclure dans le paquet une date d'expiration au-delà de laquelle le paquet doit être purement et simplement ignoré.

La différence entre les deux approches réside dans le fait que lorsque la position géographique des nœuds est utilisée, la synchronisation n'a pas besoin d'être aussi précise. D'autre part, le fait de connaître la position des nœuds permet de déceler un nœud qui prétend être à plusieurs endroits à la fois.

4.3 Détection d'intrusion

Puisque les systèmes de sécurité basés sur la cryptographie sont parfois coûteux et qu'ils ne permettent pas d'empêcher toutes les catégories d'attaques, d'autres travaux, souvent considérés comme complémentaires à la prévention, ont porté sur la conception de mécanismes de *détection d'intrusions*. Le principe général des systèmes de détection d'intrusions consiste en la surveillance et l'examen du comportement du protocole qui doit être protégé, de sorte à identifier et à réagir contre d'éventuelles attaques. Lors de l'examen, sont pris en considération le comportement de l'intrus et les comportements normaux et espérés du protocole. Selon la nature des comportements examinés, deux stratégies de détection se distinguent : (i) la détection basée sur les anomalies et (ii) la détection basée sur les mauvais usages. Dans le premier cas, les actions réalisées sont comparées aux comportements normaux et attendus par le protocole. Si elles sont considérablement différentes, alors le protocole présente des

anomalies et fait l'objet d'une intrusion. Dans le second cas, les actions réalisées sont comparées à une base de signatures représentant les actions illégales habituellement effectuées par un intrus. Une intrusion est détectée dès lors qu'une signature est identifiée parmi les actions réalisées.

Compte tenu de la dynamique du réseau et du caractère parfois imprévisible des situations, la définition de la base de signature se révèle particulièrement difficile. À cela s'ajoute l'absence d'une entité centrale pour collecter et vérifier le trafic, ce qui a pour conséquence de compliquer la détection précise d'un intrus. En raison des limitations des systèmes conventionnels basés sur un répertoire de signatures, une grande partie de la littérature [42] utilise la détection d'anomalies.

Pour représenter le comportement normal attendu par un protocole, des techniques basées sur la spécification ont été proposées. En particulier, une détection basée sur une telle spécification nécessite dans une première étape la définition d'un ensemble de contraintes qui décrit les opérations correctes du protocole étudié. Ensuite, la conformité de l'exécution du protocole est contrôlée au regard des contraintes définies.

Les auteurs de [57] proposent une modélisation du protocole AODV sous la forme d'une machine à états finis. Dans leur approche, ils distinguent deux types de nœuds, les nœuds ordinaires et les nœuds moniteurs. Ce sont ces derniers qui ont pour responsabilité de détecter les anomalies dans les traces observées sur le réseau, en utilisant la machine à états finis d'AODV comme référence. Certaines hypothèses fortes telles que l'unicité des adresses MAC (servant à identifier un nœud à l'origine d'une anomalie) font que cette approche n'est pas viable dans le contexte des réseaux Ad hoc.

Dans le but de détecter les attaques sur la phase de découverte du voisinage, les auteurs de [86] ont proposé une modélisation formelle du protocole OLSR sous la forme d'une machine à états finis étendue (EFSM[9]). L'EFSM définit le comportement correct du protocole sous la forme d'évènements d'entrée/sortie avec ou sans paramètres, de prédicats à satisfaire et d'actions à effectuer. Le processus de vérification et de détection des anomalies consiste à comparer les traces d'exécution du protocole, c'est-à-dire les messages reçus et envoyés, avec sa modélisation. La comparaison est effectuée à l'aide d'un algorithme de recherche en arrière [87] défini par les auteurs. Néanmoins, cette approche est limitée dans le sens où seules les attaques locales, qui violent directement le modèle OLSR, sont détectées.

Les auteurs de [88], ont proposé une technique de détection d'intrusion basée sur la vérification sémantique du protocole OLSR. L'idée de base est de dériver les propriétés sémantiques implicites de la définition du protocole, afin de décrire le comportement valide dans les mises à jour des informations topologiques. Ces propriétés sont vérifiées par chaque nœud, à partir des messages de contrôle (HELLO et TC) qu'il émet et reçoit de son voisinage direct, et une intrusion est détectée lorsqu'une de ces propriétés n'est pas satisfaite. Toutefois, puisque les vérifications réalisées par un

[9] EFSM: Extented Finite State Machine.

nœud concernent uniquement sa connaissance locale du réseau, toutes les attaques contre le protocole ne peuvent pas être détectées (par exemple, les attaques distantes et distribuées). Un autre désavantage est lié à l'absence d'un modèle formel pour spécifier et vérifier les propriétés sémantiques [18]. Or à partir du moment où la sémantique ne permet pas de capturer finement tous les aspects d'un comportement, il est possible pour un attaquant de contourner le mécanisme de protection.

Dans d'autres travaux assez similaires, [89, 90, 91] ont précisément cherché à dériver formellement les propriétés sémantiques du protocole OLSR à partir de sa spécification. Pour ce faire, ils ont fait appel à la logique déontique et temporelle. Les travaux de [90, 91] se distinguent au niveau des réactions déclenchées : des contre-mesures de sécurité sont prises dès qu'une propriété n'est pas satisfaite. Ainsi, pour limiter les impacts de nœuds attaquants, les auteurs proposent de les exclure de la formation des chemins. À l'instar des travaux de [88], seules les traces de trafic local sont utilisées pour la vérification sémantique de la connaissance locale.

En général, ces techniques présentent l'avantage de détecter des intrusions sans recourir à une base de connaissances sur les signatures d'attaques, tout en produisant peu de fausses alarmes (i.e. détection de faux). En comparaison avec les solutions basées sur la cryptographie, elles affichent des coûts plus légers en termes de puissance de calcul. Néanmoins, un défi réside dans la définition des contraintes qui décrivent les opérations correctes du protocole : si le modèle n'est pas décrit assez finement (complexité), alors les attaques ne seront pas détectées efficacement. Entre autres, des attaques éventuellement plus complexes, et où les spécifications du protocole ne sont pas directement transgressées, peuvent passer inaperçues.

4.4 Coopération entre les nœuds

Les mécanismes décrits précédemment se révèlent efficaces pour fournir des services de sécurité classiques que sont l'authentification et l'intégrité, et ainsi assurer qu'aucune modification non conforme au protocole n'a été apportée sur les messages de contrôle. Alors qu'ils permettent de réduire le nombre d'attaques possibles (aussi bien externes qu'internes), ils s'avèrent inopérants lorsqu'il s'agit de traiter le problème de non-participation des nœuds dans l'opération de retransmission des paquets de contrôle et de données. Or dans le contexte des réseaux Ad hoc, cette opération est fondamentale, car elle rend possible l'établissement et le maintien des communications entre les nœuds distants, sans recourir à une infrastructure prédéfinie.

C'est pourquoi en sus des mécanismes de sécurité, certains protocoles visent plus spécifiquement l'incitation à détecter les nœuds non coopératifs dans les opérations de retransmission des paquets ou à atténuer les effets néfastes de leurs comportements. Parmi ceux-ci, on distingue généralement trois catégories, ceux qui se basent sur une réputation des nœuds élaborée au cours du temps en fonction des observations. Figurent dans cette catégorie : les régimes à base de réputation, les protocoles à base de confiance et les régimes à base d'acquittements. Une autre catégorie consiste non plus à

détecter, mais à fournir des encouragements aux nœuds qui participent aux opérations du réseau. Dans cette catégorie figurent les systèmes basés sur des échanges d'argent virtuel "micro-paiement". Une dernière approche est la tolérance à la non-coopération. Elle consiste à concevoir l'opération de retransmission de sorte que son fonctionnement soit le moins compromis possible par d'éventuels nœuds non coopératifs. Dans la suite de cette section, nous présentons et discutons certaines de ces solutions.

4.4.1 Mécanismes de micro-paiements

Dans cette approche, Le concept consiste à monnayer les services auxquels les nœuds souhaitent accéder en échange de crédits virtuels. Pour obtenir ces crédits, chaque nœud doit fournir des services aux autres nœuds. Les crédits sont ultérieurement dépensés pour pouvoir acheter des services. Si un nœud n'a plus assez de crédits pour acheter le moindre service, cela signifie alors qu'il n'a pas suffisamment participé au bon déroulement du processus de routage. Dans ce cadre il existe plusieurs modèles tel que : Nuglets [92], Sprite [93], Token-Based Cooperation Enforcement [94] et Ad-hoc-VCG [95]. Dans la suite de ce paragraphe, nous présentons une description du protocole *Nuglets*.

4.4.1.1 Nuglets

Le protocole *Nuglets*, s'inscrit dans cette optique. Son objectif est à la fois d'inciter les nœuds à participer et de limiter les inondations du réseau, dès lors qu'elles deviennent payantes. Afin, de sécuriser les crédits virtuels, le protocole suppose l'existence de matériels inviolables. L'hypothèse principale est donc qu'aucune attaque ne peut être lancée contre la monnaie virtuelle. Deux modèles sont spécifiés par le protocole.

Dans le premier, un nœud désirant envoyer un paquet doit au préalable y incorporer suffisamment de crédits. Par la suite, chaque nœud intermédiaire sur la route prélève une quantité de crédits. Si le nombre de crédits est insuffisant, le paquet est rejeté. L'intérêt de cette approche est qu'elle limite les attaques de type déni de service dans la mesure où aucun nœud ne peut se permettre de financer une inondation. En revanche, elle implique que chaque nœud connaisse par avance le nombre de nœuds sur la route. Si le nombre de crédits est trop grand, ils sont gaspillés. Dans le cas contraire, le paquet est perdu et davantage de crédits doivent être dépensés pour sa réémission.

Dans le second modèle, le routage fait l'objet de transactions puisque ce sont ici les nœuds destinataires qui doivent payer pour recevoir les paquets qui leur sont destinés. En effet, chaque nœud achète les paquets reçus de son voisin amont et le destinataire d'un paquet l'achète donc au dernier nœud intermédiaire. Cette approche souffre d'un inconvénient encore plus conséquent que la précédente puisqu'elle ne permet pas d'empêcher un attaquant d'inonder le réseau [42]. Au contraire, un nœud peut être tenté de relayer beaucoup de paquets vers de nombreux nœuds afin de maximiser ses profits lors des transactions.

D'une manière générale, ces protocoles ne collent pas suffisamment au modèle Ad hoc pour être efficaces. Tout d'abord, ils ne prennent pas assez en compte la mobilité des nœuds. En effet, si un nœud intermédiaire quitte la route, le paquet est perdu ainsi que l'investissement en termes de crédits, soit pour l'émetteur (cas du premier modèle) soit pour le dernier nœud intermédiaire (deuxième modèle). Enfin, cette approche pose de gros problèmes concernant le fonctionnement même du protocole de routage [42]. Ainsi dans le cas d'un protocole réactif les nœuds peuvent être tentés de ne pas envoyer de messages d'erreur *RRER* lors de la détection de la rupture d'un lien puisqu'ils auraient alors à payer pour cela. Dans le cas d'un protocole proactif, cela concernerait les messages de contrôle qui deviendraient alors trop coûteux. Enfin, le protocole devrait aussi s'assurer que les nœuds ne puissent pas voler des crédits simplement en espionnant les conversations de ses voisins.

4.4.2 Régimes à base d'acquittement

L'idée de cette approche n'est pas récente et des techniques d'acquittement des couches hautes du modèle OSI, telles que le TCP-ACK, permettent de détecter des fautes[10] dans des communications de bout en bout. Néanmoins, la technique du TCP-ACK n'est pas utilisable telle quelle pour résoudre les problèmes de sécurité du routage dans les réseaux Ad hoc. Ceci vient essentiellement du fait qu'elle ne donne aucune information précise de l'endroit où la faute s'est produite. Pour pallier ce problème, de nouvelles techniques ont été proposées.

4.4.2.1 Trace − route

Trace-route est un protocole utilisé pour déterminer le chemin emprunté par un paquet entre une source et une destination à travers un réseau. Un nœud source émet plusieurs paquets de contrôle vers la destination, avec une valeur de TTL de plus en plus grande (en commençant à 1) puis attend en retour les messages d'avertissement en provenance des nœuds intermédiaires qui ont reçu un paquet avec une valeur de TTL expirée. Les auteurs de [96] ont proposé "Secure Trace-route", une version sécurisée de Trace-route, pour localiser l'origine d'une faute sur un chemin de communication. Dans cette version, l'idée de base consiste à empêcher un attaquant de traiter de manière différente les paquets de contrôle des paquets de données. Pour ce faire, tous les paquets de contrôle sont authentifiés et masqués afin d'être confondus avec le trafic ordinaire. L'inconvénient ici est que la recherche d'un nœud fautif s'accompagne d'un coût en temps qui est linéaire avec la longueur du chemin.

Les auteurs de [97, 98] ont proposé un protocole de routage réactif résistant aux comportements byzantins, c'est-à-dire aux actions complètement arbitraires et imprévisibles d'un nœud interne (seul

[10] Ici, une faute désigne toute perturbation qui provoque des pertes ou des délais significatifs dans le réseau.

ou en collusion) qui résultent en une dégradation des services de routage. Les auteurs supposent l'existence d'un dispositif cryptographique pour assurer l'authenticité et l'intégrité des paquets en transit. Pour détecter les fautes byzantines, et plus particulièrement la suppression arbitraire de paquets de données sur des liens dans la phase d'acheminement, les auteurs proposent, un peu à la manière du TCP-ACK, une technique d'acquittement de bout en bout. Pour ce faire, les destinations doivent envoyer des acquittements explicites pour chaque paquet de données valide reçu. Les pertes sont caractérisées par la non-réception d'un accusé de réception et sont détectées par le nœud source. Une faute byzantine est décelée à partir du moment où le taux de pertes dépasse un seuil prédéfini, ce qui conduit le nœud source à déclencher une procédure de recherche du nœud défaillant dans le chemin. Selon cette procédure, Le nœud source procède par dichotomie pour identifier le nœud défaillant, et pour un chemin de longueur n, il y'a $log(n)$ fautes successives de suppression sont nécessaires pour trouver sa position. Comme dans l'approche Secure Traceroute, les paquets de contrôle sont masqués. Cependant, la procédure de recherche peut être mise en défaut et ne pas identifier immédiatement un attaquant lorsque ce dernier supprime aléatoirement les paquets qu'il reçoit. Une fois le nœud défaillant trouvé, son poids associé est augmenté et un autre chemin vers la destination est calculé.

4.4.2.2 $Two - ACK$

Afin d'accélérer la détection et l'identification des nœuds non coopératifs, [99] ont proposé Two-Ack, une technique de détection qui repose sur des émissions d'acquittements à deux sauts dans le sens inverse au chemin de retransmission. Two-Ack est conçue comme étant une technique additionnelle pour n'importe quel protocole de routage à la demande, tel que DSR. Le principe de fonctionnement est le suivant. Un nœud qui émet un paquet de données attend la réception d'un acquittement en provenance d'un autre nœud à deux sauts suivant le chemin de routage vers la destination. S'il ne reçoit aucun acquittement pendant le temps imparti, alors un compteur local comptabilisant le nombre de paquets non retransmis est décrémenté. Lorsqu'un nœud non coopératif est détecté, un message d'erreur de route est envoyé en direction de la source. Tous les nœuds intermédiaires sur le chemin peuvent entendre ce message et enregistrer le nœud incriminé comme étant défaillant. Tout chemin impliquant ce nœud sera évité.

Une limitation à cette technique est qu'elle ne permet pas de détecter des comportements de non-coopération lorsque deux nœuds attaquants forment une collusion. Dès lors que deux nœuds attaquants se succèdent sur un chemin de retransmission, l'un peut couvrir les actions de suppression de l'autre, soit en les ignorant soit en générant de faux acquittements. De plus, le système d'annonce de nœuds défaillants est lui-même une source d'attaques, et ce pour deux raisons, La première vient du fait que les messages d'erreur qui incriminent des nœuds peuvent être générés par tout nœud inclus dans le chemin de retransmission. Un attaquant est alors en mesure d'injecter de fausses informations, conduisant éventuellement à des déclenchements intempestifs de reconstructions de chemins ou un partitionnement du réseau. Alors que la deuxième vient du fait que des délais supplémentaires sont

introduits dans la phase de réaction lorsqu'un attaquant supprime en chemin les messages d'annonce de nœuds défaillants qu'il reçoit au lieu de les retransmettre.

4.4.3 Systèmes à base de conservation de flot

Le principe de conservation de flot dans un réseau stipule que dans un intervalle de temps, tous les octets de données envoyés à un nœud et qui ne lui sont pas destinés doivent quitter ce nœud. Une mesure additionnelle pour identifier des anomalies dans la phase de retransmission des paquets de données consiste alors à traiter les paquets en transit comme des flots (qui sont comptabilisés) puis à vérifier ce principe dans le graphe topologique.

De façon évidente, il n'est pas envisageable de laisser les nœuds vérifier par eux-mêmes le respect du principe de conservation de flot exclusivement à partir de compteurs locaux des paquets qu'ils ont reçus et émis. De même, ces vérifications ne peuvent être réalisées uniquement sur la base d'une comparaison d'égalité entre les valeurs de leurs propres compteurs et celles de leurs voisins directs (i.e. un test d'égalité des compteurs des deux extrémités d'un lien). En effet, étant donné que ces compteurs sont manipulés par les nœuds eux-mêmes, un attaquant est en mesure de modifier les valeurs de ses compteurs afin de masquer ses actions de suppression.

Basée sur le principe de conservation de flot, [100] ont proposé une solution distribuée de détection nommée WATCHERS[11], dans laquelle les nœuds contrôlent les flots de leurs voisins directs grâce aux compteurs des voisins de leurs voisins. Dans une première phase, il s'agit pour chaque nœud de maintenir des compteurs distincts de paquets émis et reçus pour chacun de ses voisins. Ensuite, ces valeurs de compteur sont périodiquement échangées entre les nœuds du réseau. Ceci leur permet, par comparaison de valeurs de compteur concernant un même flot, de vérifier pour chacun de leurs voisins si le principe a bien été respecté. Dans cette approche, il est requis que chaque nœud soit en possession d'une copie de la table de routage de ses voisins. Par ailleurs, la technique WATCHERS a été proposée pour être utilisée dans les réseaux filaires. Par conséquent, certaines hypothèses qui ont été prises telles que l'immobilité des nœuds et une synchronisation de ces derniers pour la phase d'échange de compteurs font qu'elle n'est pas adaptée au réseau Ad hoc.

Dans le contexte des réseaux Ad hoc, [82] ont proposé une technique similaire appliquée sur le routage OLSR. Cependant, les changements topologiques et la synchronisation des nœuds ne sont pas pris en considération.

4.4.4 Systèmes à base de réputation

La réputation est un indicateur qui est employé par les individus dans la vie de tous les jours pour d'une part valoriser une image, et d'autre part faciliter la prise de décision. Dans le cadre des réseaux

[11] Watching for Anomalies in Transit Conservation: a Heuristic for Ensuring Router Security.

Ad hoc, la réputation peut être définie comme étant le niveau de participation d'un nœud dans l'opération de retransmission des paquets, tel que vu par d'autres nœuds. La réputation d'un nœud est ensuite utilisée pour différencier les "bons nœuds" (à savoir ceux qui coopèrent), des mauvais qui adoptent un comportement égoïste. Watchdog-Pathrater [1], Confident [101], CORE [102], SORI [103] et OCEAN [104] sont des exemples de tels systèmes. Dans la suite de ce paragraphe, nous présentons une description des protocoles Confidant et CORE.

4.4.4.1 Confidant

Le protocole Confidant [101] (*Cooperation of nodes-fairness in dynamic Ad hoc networks*) s'inscrit dans ce cadre, il utilise une infrastructure à clés publiques auto-organisée inspirée du protocole PGP. L'objectif de Confidant est de traiter à la fois les nœuds malicieux et égoïstes à travers la supervision et l'analyse de deux processus du routage à savoir le transfert de données et la découverte de voisins. Il est ainsi conçu pour être utilisé conjointement avec un protocole réactif, typiquement DSR.

Confidant se compose de quatre éléments complémentaires : le moniteur, le moniteur de confiance, le système de réputation et le mécanisme de gestion de chemins. Le rôle du moniteur consiste à s'assurer que les voisins du nœud auquel il est rattaché relaient correctement le paquet. Lorsque le moniteur détecte une anomalie ou une incohérence, il avertit le système de réputation, qui de son côté maintient à jour des listes de notes pour chaque nœud observé. Les listes peuvent être éventuellement échangées entre les nœuds. Ainsi, si une liste est reçue d'un nœud de grande confiance, le récepteur peut directement enregistrer les informations à l'intérieur de sa propre liste. Dans le cas contraire, si la liste est envoyée par un nœud suspect, le récepteur peut l'ignorer totalement ou bien encore l'accepter tout en lui donner significativement moins d'importance qu'une liste reçue d'un nœud sûr. Finalement, le mécanisme de gestion de chemin détermine les routes les plus sûres à partir des listes de nœuds exclus et des nœuds de confiance. En outre, il peut décider de refuser de relayer les requêtes en provenance de nœuds mal notés.

Concernant la gestion de la confiance, l'approche s'inspire de celle utilisée dans PGP. Ainsi, les nœuds disposent de quatre niveaux de confiance : *ami*, *marginal*, *inconnu* ou *ennemi*. Chaque nœud enregistre ses amis dans une liste dédiée. Par la suite, si un nœud "A" parvient à détecter un comportement malicieux de la part d'un nœud "B", le nœud "A" va avertir tous les amis contenus dans la liste à l'aide d'un message d'alarme signé. De tels messages peuvent être diffusés à travers le réseau, Il appartient ensuite à chaque nœud de décider si le message doit être pris en considération, suivant que l'émetteur est de confiance ou non. Une version améliorée de Confidant utilise une approche bayésienne afin de différencier plus efficacement les vraies alarmes de mensonges destinés à faire baisser la réputation d'un nœud.

4.4.4.2 CORE

Une des motivations principales qui peuvent pousser un nœud à ne pas participer au routage est l'économie d'énergie. Celle-ci étant parfois une ressource critique, certains nœuds peuvent être tentés de l'épargner en adoptant un comportement égoïste. Pour lutter contre ce phénomène, [109] ont mis au point le protocole CORE (a COllaborative REputation mechanism to enforce node cooperation in mobile Ad hoc networks). L'objectif est ici non plus d'exclure définitivement les nœuds mais au contraire de les encourager à participer en rejetant leurs paquets jusqu'à ce qu'ils coopèrent au processus de routage. CORE prend entre autres comme hypothèses que : les identités des nœuds sont uniques et non modifiables, qu'un mécanisme de routage adapté est à même de sécuriser la phase de découverte de voisins et enfin, que le trafic à l'intérieur du réseau est suffisamment dense. Le fonctionnement est très similaire à celui de Confidant à savoir des moniteurs qui analysent le trafic et qui transmettent les résultats à un système de gestion des réputations. L'échange des réputations entre les nœuds est ici optionnel. Les auteurs ont en outre validé leur approche à la fois par la simulation et la théorie des jeux.

CORE souffre malheureusement de défauts importants [42]. Tout d'abord, il ne résout pas réellement le problème de non-participation. Certes, les nœuds égoïstes voient leurs paquets systématiquement rejetés et en ceci, le protocole est efficace. Mais en contrepartie de grandes quantités de données demeurent perdues, diminuant significativement le rendement du réseau. Enfin, le protocole repose sur des hypothèses très fortes (routage sécurisé, adresses uniques et non usurpables) qui restent encore à concrétiser. Il s'agit en fait d'un inconvénient commun à tous les protocoles basés sur la réputation. En effet, ceux-ci reposent sur l'information observée sur les nœuds et par conséquent requièrent un mécanisme d'authentification forte afin d'affecter les valeurs aux nœuds légitimes. D'autre part, il est difficile d'éviter le problème de "dénonciation calomnieuse" dans lequel un nœud malicieux génère de fausses alertes pour mettre sur liste noire des nœuds honnêtes. Ce type de mécanisme est en outre potentiellement très vulnérable face à des nœuds complices qui se mettent d'accord entre eux pour s'attribuer de bonnes valeurs et affecter en contrepartie, de mauvaises valeurs aux nœuds honnêtes.

4.4.5 Protocoles de routage à base de confiance

La notion de confiance est souvent confondue abusivement à la réputation même s'il existe des associations entre les deux. On peut définir la confiance comme étant la croyance et la conviction dans l'honnêteté, la sincérité, la franchise, la compétence, la fiabilité et la crédibilité d'une entité ou d'un service. Cette notion est omniprésente dans les protocoles de routage Ad hoc où les nœuds se font mutuellement confiance et se basent sur l'hypothèse d'un comportement honnête. Ainsi, les solutions à base de systèmes de gestion de confiance tirent avantage des propriétés intrinsèques des protocoles afin de détecter les comportements malhonnêtes. Dans un sens, comme pour les systèmes de gestion

de réputation, ces systèmes se comportent comme un système de détection d'intrusion essayant de créer une deuxième ligne de défense contre les attaquants internes.

Dans ce contexte, plusieurs protocoles de routage basées sur la confiance ont été proposées pour le routage Ad hoc.

Les auteurs de [105], ont introduit une approche pour mesurer la confiance entre les entités d'un réseau (DBLAR[12]), en se basant sur les statistiques de trafic du réseau, cette approche utilise la confiance directe et la recommandation de confiance pour éviter les nœuds malveillants de se joindre à l'expédition. Néanmoins, les propriétés présentées dans cette approche ne concernent que la vision locale et directe de chaque nœud et ne permettent pas de partager les observations. De plus, les auteurs n'ont pas spécifié les mesures à prendre pour stopper et isoler les nœuds malveillants.

Dans les travaux de [106], Xia et al., ont construit un modèle de confiance simple pour évaluer le comportement des paquets des voisins transitaires, en proposant un protocole de routage réactif multivoie fondé sur la confiance et l'extension du protocole AODV.

Sirotheau et al., [107] ont proposé un mécanisme d'évaluation qui visait à atténuer la mauvaise conduite de routage et d'autres défaillances du réseau. Quatre attributs des routes ont été envisagées : *le niveau d'activité*, *la confiance*, *la mobilité* et *le nombre de sauts*. Lors de la transmission d'un paquet vers une destination donnée, un nœud peut avoir deux voies : l'une est courte mais incrédible tandis que l'autre est longue mais crédible.

Les auteurs de [8], ont introduit un système de gestion de la confiance dans un protocole bien connu tel que le protocole S-AODV pour protéger l'acheminement contre les attaques de nœuds malveillants, y compris à la fois les attaques de Black-hole et Grey hole qui seront détectés sans perturber la communication. Par ailleurs, pour veiller à ce que la consommation d'énergie due à d'autres mécanismes de sécurité mis en œuvre, une analyse de l'énergie est également effectuée sur S-AODV avec et sans gestion de la confiance.

Dans le même contexte, [9] ont proposé un protocole nommé "Trust-based AODV" qui est basé sur un mécanisme de détection d'intrusion (IDM[13]) et d'un autre mécanisme fondé sur la confiance (TBM[14]), afin de pénaliser les nœuds égoïstes dans le protocole AODV.

Dans [108], les auteurs ont conçu une solution de routage qui permet au protocole DSR de trouver un itinéraire libre de nœuds de black-hole, en coopérant avec les voisins. Cette solution peut également

[12] Distance-Based Location-Aided Routing.
[13] Intrusion Detection mechanism.
[14] Trust-Based Mechanism.

protéger le réseau en présence d'attaquants de collusion sans la nécessité de surveiller la promiscuité des nœuds voisins.

4.5 Discussion

Dans le tableau 3.1, nous récapitulons les possibilités de défense qu'offrent les différentes solutions de sécurité décrites dans ce chapitre. On constate que les protocoles ont tendance à cibler certaines attaques en particulier, de sorte qu'aucun n'offre une protection efficace face à toutes les attaques décrites ici. La conclusion que l'on peut en tirer est que la solution la plus prometteuse est probablement dans l'utilisation d'un protocole combinant ces approches : un protocole basé sur la cryptographie pour assurer l'authentification des nœuds et l'intégrité des messages de contrôle et un protocole basé sur les modèles de confiance pour déceler puis ignorer les nœuds présentant un comportement malicieux.

Table 3.1 : Protocoles sécurisés, prévention des attaques.

	Ecoute indiscrète	Usurpation	Grey-Hole	Black-Hole	Tunnel	Non-coopération
ARAN	Oui	Non	Oui	Oui	Oui	Oui
Ariadne	Oui	Non	Oui	Oui	Oui	Oui
SRP	Non	Oui	Non	Non	Oui	Non
CORE						
SAODV	Non	Non	Oui	Oui	Oui	Oui
Confidant	Oui	Non	Non	Non	Oui	Non
Packet leashes	Oui	Oui	Oui	Oui	Non	Oui
T-AODV	Non	Non	Oui	Oui	Oui	Oui
Trust-based AODV	Oui	Non	Oui	Oui	Oui	Oui

4.6 Positionnement

La 4ème section de ce chapitre présente un aperçu presque détaillé des différentes solutions qui ont été mises en place pour sécuriser les protocoles de routage Ad hoc. Ces solutions sont classées en général en deux catégories principales :

- Les solutions utilisant les mécanismes cryptographiques, ces solutions assurent l'authenticité et l'intégrité des messages. Toutefois, ces mécanismes sont assez lourds à mettre en place notamment lors de la distribution des clés, d'une part, et d'autre part, ils n'empêchent pas des nœuds ayant le matériel

cryptographique nécessaire de se comporter malhonnêtement. Ces solutions sont donc plutôt destinées à fournir une protection contre les attaques externes.

- Les solutions basées sur la coopération entre les nœuds, notamment les régimes à base de réputation et les solutions basées sur la confiance, ces solutions tentent de profiter des propriétés intrinsèques des protocoles de routage pour chercher des incohérences entre les messages reçus. Ces solutions essaient de construire une opinion sur les autres nœuds sous forme de valeur numérique comparée à un seuil souvent obtenu par des méthodes heuristiques.

Nous nous intéressons plus particulièrement à la dernière catégorie (*les solutions basées sur la confiance)*, dont la proposition d'un système de gestion de la confiance est l'objectif principal de cette thèse. La notion de confiance est plus sollicitée dans le contexte des réseaux Ad hoc vu les caractéristiques particulières de ce type de réseaux (mobilité, distribué, auto-organisé, etc.) où les nœuds communiquent entre eux sans connaissance préalable de l'identité ou même de la topologie. Il nous semble judicieux de permettre à chaque nœud du réseau de mettre en place un mécanisme qui lui permet de raisonner sur les comportements des autres. Ce raisonnement utilise les observations qu'il collecte au fur et à mesure d'échange de messages et en les corrélant, lui permet de détecter des incohérences caractéristiques d'actions malhonnêtes.

5. CONCLUSION

Dans un réseau Ad hoc, tous les nœuds doivent participer aux opérations de routage. Ils gèrent entre autre l'établissement des chemins, la dissémination de notifications de ruptures de chemins et la retransmission des données. Étant donné cette caractéristique, il devient relativement facile pour un nœud malveillant de mener divers types d'attaques, rendant ainsi le réseau inopérant. Qu'il s'agisse de nœuds malveillants internes ou bien de nœuds normaux compromis par un attaquant au cours de l'exploitation du réseau, ces nœuds déviants sont particulièrement difficiles à contenir. La raison en est qu'ils ont accès à toutes les clés cryptographiques nécessaires pour y participer, ce qui leur permet de mener des attaques sur par exemple les points les plus critiques des opérations du réseau ou sur les mécanismes de sécurité mis en œuvre.

L'expérience dans le domaine de la cryptographie a déjà montré que la conception de protocoles sécurisés est souvent sujette à des failles difficilement détectables même en admettant que le chiffrement est parfait. Ainsi, même si les protocoles détaillés dans ce chapitre permettent d'améliorer sensiblement la sécurité du processus de routage, ils offrent en contrepartie une vulnérabilité accrue aux attaques de type déni de service.

Quoi qu'il en soit, aucun protocole ne peut en l'état contrer toutes les attaques, la plupart se contentant de cibler une menace (non-participation, usurpation d'identité, détournement de trafic) et

de fournir une solution relativement adaptée. C'est pourquoi la tendance la plus probable est à une utilisation *combinée de différentes approches* (cryptographie symétrique/asymétrique, modèles de confiance) au sein d'un même protocole, pour sécuriser le réseau.

En ce sens, nous proposons de mettre en place un mécanisme permettant à chaque nœud participant à un réseau Ad hoc de distinguer les nœuds dignes de confiance des nœuds malhonnêtes en se basant sur des expériences et des recommandations préalables, afin de rendre les nœuds du réseau capables de recueillir des informations pour raisonner, apprendre et prendre leur propre décision. La solution envisagée est de faire reposer la prise de décision d'un échange sur la base de la confiance, sachant que chaque nœud ne pourra se protéger d'éventuels voisins malicieux qu'en faisant appel aux informations locales dont il dispose.

Partie 2

Contributions

Chapitre 4

Contexte de notre approche

1. INTRODUCTION

Pour sécuriser les réseaux Ad hoc, nous envisageons un modèle de gestion de la confiance, ce modèle devra être capable de proposer un niveau de sécurité adapté à l'enjeu de la communication et dont le niveau pourra évoluer dans le temps en fonction du contexte. Bien entendu, une solution complètement opérationnelle respectant l'ensemble de toutes les spécificités exigées par un tel réseau n'existe pas encore. Mais, il nous a paru que la notion de confiance constitue le levier incontournable à l'émergence d'une solution globale au problème de la sécurité dans des réseaux d'objets autonomes. En effet, il est admis que la construction d'une relation de confiance entre deux entités autonomes, en l'absence de tiers, est un enjeu très complexe.

Dans ce chapitre, nous développons d'abord un cadre abstrait pour faciliter notre étude sur les protocoles de routage basés sur la confiance, en mettant l'accent sur les définitions de la relation de confiance, la logique floue et la méthode d'analyse relationnelle grise, ainsi que les exigences relatives à la conception de notre modèle. Ces définitions sont nécessaires pour justifier notre motivation afin de développer une nouvelle approche formelle adoptée pour la présente thèse.

2. LA CONFIANCE

Dans cette section, nous allons discuter des méthodes utilisées pour établir les relations de confiance entre les entités d'un réseau de communication et tout particulièrement dans le cas d'un réseau ad hoc. Après une courte introduction, nous donnerons quelques définitions de la notion de confiance telle qu'elle est habituellement utilisée dans la vie de tous les jours et aussi telle qu'elle est utilisée en informatique. Ensuite, nous étudierons la manière avec laquelle la confiance est évaluée, propagée et utilisée dans les réseaux distribués tels que les réseaux Ad hoc.

Les systèmes de gestion de la confiance et de réputation tiennent une place très importante dans la littérature relative à la sécurité des réseaux Ad hoc. Un des plus importants problèmes de sécurité des réseaux Ad hoc réside essentiellement dans leur nature distribuée. Ces réseaux Ad hoc reposent sur la collaboration des nœuds les composant afin de réaliser les fonctionnalités courantes du système. De ce fait, afin que la collaboration soit productive, il est nécessaire que la majorité des participants adoptent un comportement honnête. Pour cela, il est nécessaire de donner au réseau les moyens d'inciter les nœuds à collaborer et à adopter des comportements en adéquation avec les exigences des applications. Les systèmes de gestion de la confiance répondent à ce besoin.

2.1 *Définition de la confiance*

La confiance est un mécanisme de coordination des échanges en situation d'ignorance ou d'incertitude, c'est elle qui permet de prendre une décision malgré l'existence d'un risque [109]. Elle peut être accordée à une personne physique ou morale (institution, système). Dans un réseau Ad hoc, un nœud compte sur la coopération de nœuds qu'il ne connait pas éventuellement, pour pouvoir échanger des informations. Ceci suppose qu'il faut faire l'hypothèse d'accorder sa confiance aux autres nœuds et aux informations qu'ils lui présentent.

La notion de confiance a été traité à maintes reprises dans la littérature. Cependant, chacune l'abordant selon des points de vue différents :

- Du point de vue psychologique : la définition la plus répandue est celle de [110] qui considère que la confiance est basée sur une perception individuelle. Pour une même situation, chaque entité perçoit la situation différemment. L'auteur associe deux variables pour chaque cas à traiter : "$V_a +$ " pour un évènement bénéfique et "$V_a -$ " dans le cas contraire.

- Du point de vue sociologique : [111] affirme que le concept de confiance est un moyen de réduction de la complexité de la société. Elle joue un rôle important dans les interactions qui y prennent part. L'auteur de [112][1] lui aussi considère la confiance comme un bien social qu'il faut protéger à tout prix.

- Du point de vue mathématique : l'auteur de [113] considère la confiance comme un niveau particulier de la probabilité subjective avec laquelle un agent évalue qu'un autre agent ou un groupe d'agents effectue une action particulière. Elle devient ainsi quantifiable (généralement variant entre 0 et 1). [114] trouve plus judicieux de s'intéresser au comportement de la confiance plutôt qu'à la confiance. Il propose un modèle théorique et assez complexe qui reste générique et non applicable à des cas réels.

En général, on distingue deux types de confiance :

- La confiance assurée : On parle parfois de *confiance aveugle*, c'est-à-dire que la confiance est acquise a priori, sans réelle évaluation du risque, ceci peut être le cas, parce que l'on estime que la réalisation du risque est très improbable, que les inconvénients possibles sont minimes par rapport aux avantages attendus, ou encore que l'on n'a pas vraiment d'alternative.

- La confiance décidée : Résultat d'un réel processus d'appréciation du risque (évaluation des avantages attendus de la décision et des inconvénients qui peuvent en découler) et décision

[1] trust is a social good to be protected just as much as the air we breathe or the water we drink. When it is damaged, the community as a whole suffers; and when it is destroyed, societies falter and collapse. D'après [112], pages 26-27

parfaitement consciente. Celle-ci sous-entend que la décision prise peut conduire à une déception, et un regret de l'avoir prise. Elle ne peut donc être requise que dans le cas où les dommages possibles sont supérieurs aux avantages reçus.

Sauf à vivre dans un état d'incertitude et d'indécision permanente "ce qui conduirait à ne jamais décider" un certain niveau de confiance assurée est indispensable : dans l'histoire, cette confiance a pu être placée en : la famille, la tribu, Dieu, le roi, l'État, la Science, etc. La frontière entre confiance assurée et confiance décidée n'est d'ailleurs pas fixe : en fonction de l'expérience notamment, on peut être amené à revoir notre confiance assurée.

Dans notre travail, nous nous intéressons au comportement de la confiance dans les protocoles de routage Ad hoc puisque chaque nœud est supposé se conduire correctement, impliquant ainsi une confiance implicite. Cependant, les nœuds malhonnêtes profitent des faiblesses des protocoles pour servir leurs intérêts. Pour cela, et afin de donner une description formelle de la confiance et de la fiabilité dans ce cadre, nous définissons [11] la confiance comme la croyance d'un nœud nommé "*confiant*", qu'un autre nommé "*crédible*" (ou digne de confiance) a la compétence et la volonté de coopérer pour accomplir une tâche en faveur du *confiant*. Ce dernier est souvent invité à choisir le *crédible* parmi un groupe de voisins. Sa croyance est basée sur : les interactions précédentes avec chacun de ses voisins, les conseils de ses voisins fiables, les caractéristiques de la tâche demandée, ainsi que d'autres composantes subjectives (l'apparence du voisin, sa façon de s'exprimer, etc.). Cela a donné naissance à la confiance computationnelle, qui est censée représenter une adaptation de la confiance sociale au l'ère numérique.

2.2 *Les zones de la confiance*

Généralement, le scenario de confiance commence par un besoin exprimé par un nœud. L'objectif est d'accomplir une tâche qu'il ne peut pas faire lui-même pour des raisons diverses (portée de transmission, mobilité, etc.). Le nœud confiant doit trouver un nœud crédible pour exécuter la tâche. Pour ce faire, le confiant pense à plusieurs voisins possibles. Il estime la confiance qu'il peut attribuer à chacun d'entre eux selon deux facteurs : son estimation des intentions du voisin à son égard, et son estimation de la compétence du voisin à propos de la tâche à effectuer.

Selon les travaux de [115], les auteurs placent la confiance entre deux entités sur une barre en trois zones. Le nœud confiant classe les entités voisines selon sa confiance en eux sur les zones de la barre. La figure 4.1 montre la barre de confiance : la confiance varie entre deux valeurs " -1" (défiance complète) et " $+1$" (confiance complète). La valeur "0" représente la neutralité. Le seuil de coopération est la valeur minimale de confiance pour laquelle le confiant accepte de coopérer avec un voisin.

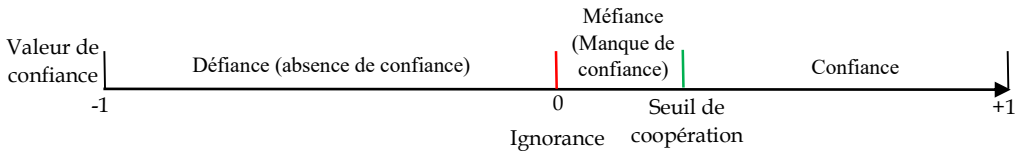


Figure 4.1 : Les zones de la confiance.

Dans ce schéma, on peut identifier trois zones :

- La défiance : La valeur de confiance dans cette zone est négative parce que le confiant pense que le voisin a des intentions négatives.
- La méfiance : L'estimation du confiant est positive, les intentions du voisin ne présentent pas de danger pour lui, mais la valeur reste en dessous du seuil de coopération. Cela parce que le confiant juge que le voisin est incapable d'exécuter la tâche, même si ses intentions sont positives. Par conséquent, le confiant ne va pas coopérer avec ce voisin tant qu'il émet des doutes sur sa compétence.
- La confiance : La valeur ici est au-dessus du seuil de coopération, alors le confiant prendra le risque de coopérer avec ce voisin.

Les zones de confiance numérisent l'opinion du confiant à l'égard de ses voisins. Le point le plus déterminant sur cette barre est le seuil de coopération : où se positionne-t-il exactement sur la barre ? Quand et comment change-t-il sa position ? Afin de mieux comprendre la notion du seuil de coopération, il faut prendre conscience du contexte de la confiance.

2.3 Relation de confiance "Trust relationship"

En MANET, la confiance est représentée et réalisée par les relations de confiance en interaction les uns avec les autres. En conséquence les relations de confiance sont déterminées par les règles d'évaluation afin d'évaluer et de juger les éléments de preuve d'une manière quantitative générés par les comportements antérieurs d'un nœud [116]. Pour augmenter la portée et pouvoir établir la confiance entre les nœuds du réseau, le contact entre le confiant et le crédible (le nœud qui sera évalué) se fait par les trois moyens suivants (Figure 4.1).

2.3.1 La Confiance directe

La confiance directe est un contact personnel direct ou une interaction directe entre un couple de nœuds dans le réseau (le nœud confiant et le nœud crédible), ce type de confiance est défini comme étant la fiabilité calculée pour un nœud par rapport à la preuve capturée pendant les interactions de

type un à un "one-to-one". La confiance directe est établie grâce à des observations préliminaires si les interactions précédentes entre les nœuds voisins ont réussi. (Figure 4.1{a})

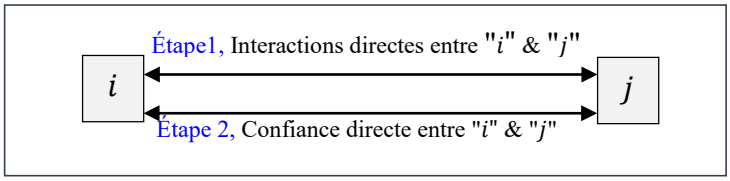


Figure 4.2 {a} : Confiance directe.

2.2.2 *La confiance à base de recommandation (Confiance indirecte)*

Dans ce type de confiance, le nœud confiant et le nœud crédible ne se connaissent pas directement. La confiance est établie par un ou plusieurs tiers intermédiaires, d'une manière que l'opinion sur un nœud pourrait le recevoir sur d'autres nœuds. (Figure 4.1{b})

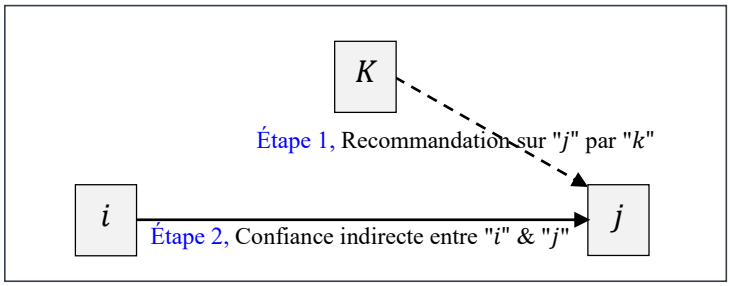


Figure 4.2 {b} : Confiance indirecte (Recommandation).

2.2.3 *Contact à travers l'examen de l'historique*

Quand un nœud confiant n'a pas assez d'expérience directe sur un nœud crédible, le nœud confiant peut enquêter à un troisième nœud pour la recommandation. Nous supposons que le troisième nœud a une certaine valeur de confiance sur le nœud crédible à base de sa propre évaluation. (Figure 4.1{c})

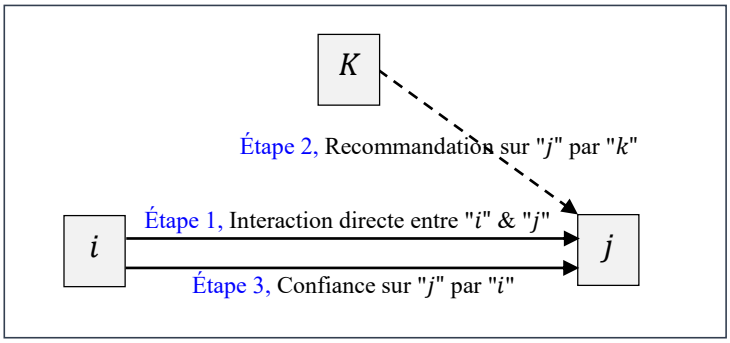


Figure 4.2 {c} : Contact à travers l'examen de l'historique.

3. LOGIQUE FLOUE POUR LA CONFIANCE "FUZZY TRUST"

Comme nous le savons, le cerveau humain interprète les informations sensorielles imprécises et incomplètes fournies par les organes perceptifs. La théorie des ensembles flous représente et manipule numériquement ces informations linguistiques de façon naturelle par l'intermédiaire des fonctions d'appartenance et des règles floues. Il peut être une méthodologie générale pour intégrer les connaissances, les heuristiques ou la théorie en contrôleurs et les décideurs. Par conséquent, la théorie des ensembles flous a été appliqué dans un système de décision de contrôle, soit pour améliorer les performances ou pour gérer les problèmes que la théorie classique de commande ne peut pas aborder avec succès, car celui-ci repose sur un modèle valide et précis qui n'existe pas toujours.

La confiance est par nature un concept flou, ce qui pose une contrainte floue sur la prise de décision de l'itinéraire de confiance. Dans les relations humaines, la confiance est souvent exprimée linguistiquement plutôt que numériquement [11]. Il est bien établi que la logique floue est adaptée pour quantifier la confiance entre les entités qui composent un réseau ou un groupe. Un des avantages de l'utilisation de la logique floue pour quantifier la confiance entre les nœuds dans les réseaux Ad hoc, est sa capacité à quantifier des données imprécises ou de l'incertitude dans la mesure de l'indice de sécurité des nœuds Ad hoc.

La logique floue est une extension de la logique booléenne créée par L. Zadeh [117] en se basant sur sa théorie mathématique des ensembles flous, qui est une généralisation de la théorie des ensembles classiques, la logique floue confère une flexibilité très appréciable aux raisonnements qui l'utilisent, ce qui permet la modélisation des imperfections des données et se rapproche dans une certaine mesure de la flexibilité du raisonnement humain.

Comme la science s'appuie sur la notion de mesure, les questions qui se posent ici sont :

- Comment représenter les valeurs non mesurables ? (Dans notre cas, les valeurs de confiance)
- Comment représenter ce qui est incertain ou subjectif ?
- Comment représenter les termes du langage humain ?

Les descriptions linguistiques d'un système sont souvent vagues. Mais le flou n'est pas imprécis. Si une donnée n'est pas connue précisément, elle peut être exprimée par un intervalle de confiance précis. Cet intervalle est un ensemble de valeurs possibles pour la donnée.

La logique floue est très utile lorsque le modèle mathématique du problème à traiter n'existe pas ou existe mais difficile à implémenter, ou il est trop complexe pour être évalué assez rapidement pour des opérations en temps réel [118], elle est aussi supposée de travailler dans les situations où il y a de large incertitude et des variations inconnues dans les paramètres et la structure du système.

Un système flou est caractérisé par le système d'inférence qui contient la base de règles du système, des fonctions d'appartenance qui *fuzzifer* les variables d'entrée et le processus de dé-fuzzification pour dé-fuzzifer les variables de sortie, comme le montre la figure 4.3.

3.1 Les sous-ensembles flous

La logique floue repose sur la théorie des ensembles flous, qui est une généralisation de la théorie des ensembles classiques. Par abus de langage, suivant les us de la littérature, nous utiliserons indifféremment les termes sous-ensembles flous et ensembles flous. Les ensembles classiques sont également appelés ensembles nets, par opposition à flou, et de même la logique classique est également appelée logique booléenne ou binaire.

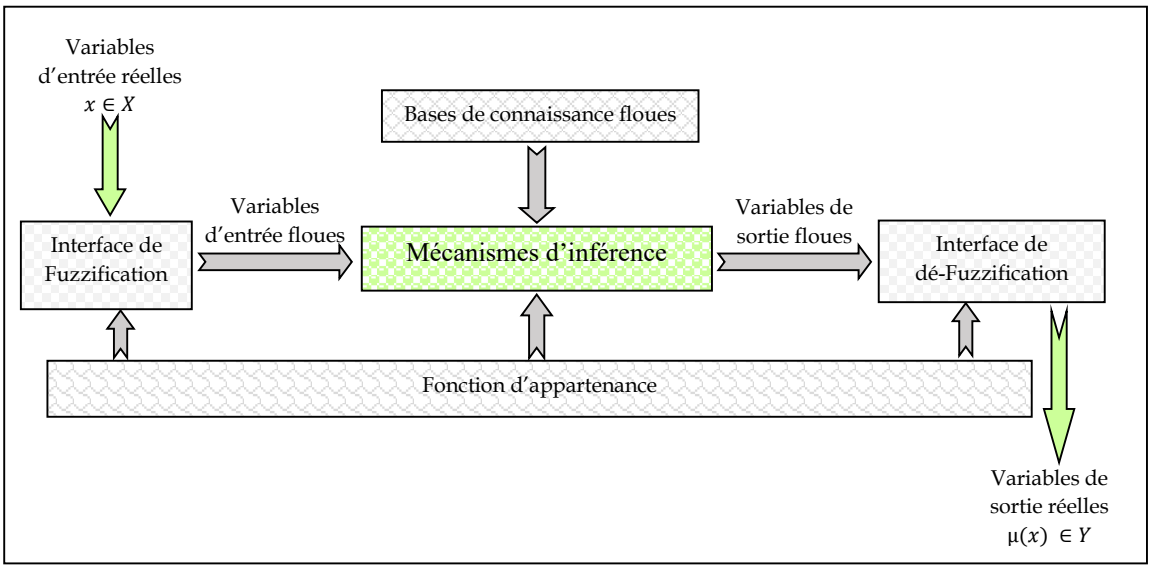


Figure 4.3 : Aperçu synoptique d'un système flou.

Définition 4.1 : *Soit X un ensemble. Un sous-ensemble flou A de X est caractérisé par une fonction d'appartenance*[2] μ_A. $x \rightarrow \mu_A(x)$, *tel que :* $\mu_A(x) \in [0,1]$.

Pour chaque élément x dans l'ensemble X, il existe une fonction $x \rightarrow \mu_A(x)$ dans laquelle : $\mu_A(x) \in [0,1]$. L'ensemble $\Delta = \{(x, \mu_A(x)\}$ est défini comme un ensemble flou de la confiance dans les MANETs et $\mu_A(x)$ est défini comme étant la fonction d'appartenance pour tous les x dans Δ. Une fonction d'appartenance définit le degré auquel une variable floue est un membre d'un ensemble.

La figure 4.4 montre un exemple de la fonction d'appartenance choisie pour caractériser le sous-ensemble "*bon*" de la variable d'entrée "confiance".

[2] Cette fonction d'appartenance est l'équivalent de la fonction caractéristique d'un ensemble classique.

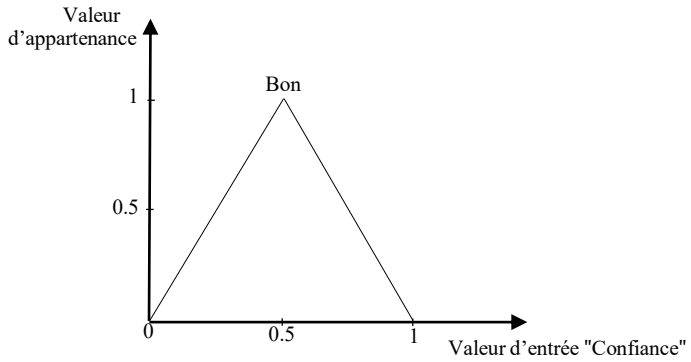


Figure 4.4 : Fonction d'appartenance caractérisant le sous-ensemble "bon" de la confiance.

La forme de la fonction d'appartenance est choisie arbitrairement en faisant des études statistiques : formes sigmoïdes, tangente hyperbolique, exponentielle, gaussienne ou de toute autre nature est utilisables. La figure 4.5 montre graphiquement la différence entre un ensemble classique et un ensemble flou.

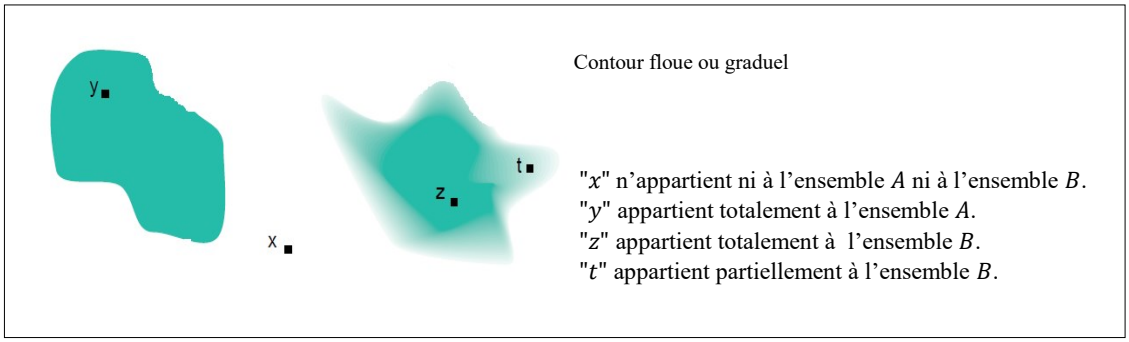


Figure 4.5 : Représentation graphique d'un ensemble classique et d'un ensemble flou.

Pour pouvoir définir les caractéristiques des ensembles flous, nous redéfinissons et étendons les caractéristiques usuelles des ensembles classiques.

Soit X un ensemble, A un sous-ensemble flou de X et μ_A la fonction d'appartenance le caractérisant.

Définition 4.2 : *La hauteur de A, notée* $h(A)$*, correspond à la borne supérieure de l'ensemble d'arrivée de sa fonction d'appartenance :* $h(x) = \sup \{\mu_A(A) \mid x \in X\}$.

Définition 4.3 : *A est dit normalisé si et seulement si* $h(A) = 1$. *En pratique, il est extrêmement rare de travailler sur des ensembles flous non normalisés.*

Définition 4.4 : *Le support de A est l'ensemble des éléments de X appartenant au moins un peu à A. Autrement dit, c'est l'ensemble* $supp(A)$*, tel que :* $\text{supp (A)} = \{x \in X \mid \mu_A(x) > 0\}$.

Définition 4.5 : *Le noyau de A est l'ensemble des éléments de X appartenant totalement à A. Autrement dit, c'est l'ensemble $noy(A)$, tel que : $noy(A) = \{x \in X \mid A(x) = 1\}$. Par construction, $noy(A) \subseteq supp(A)$.*

Définition 4.6 : *Une $\alpha - couple$ de A est le sous-ensemble classique des éléments ayant un degré d'appartenance supérieur ou égal à α : $\alpha - couple\ (A) = \{x \in X \mid \mu_A(x) \geq \alpha\}$.*

Nous remarquons que : Si A entait un ensemble classique, nous aurions simplement $supp(A) = noy\ (A)$ et $h(A) = 1$ (ou $h(A) = 0\ si\ A = \phi$). Nos definitions permettent donc bien de retrouver les propriétés usuelles des ensembles classiques.

3.2 Les variables linguistiques

Le concept de fonction d'appartenance vu précédemment nous permettra de définir des systèmes flous en langage naturel, la fonction d'appartenance faisant le lien entre logique floue et variable linguistique que nous allons définir à présent.

Définition 4.7 : *Soit V une variable (par exemple : confiance, qualité de service, réputation, etc.), X la plage de valeur de la variable V et T_v un ensemble fini ou infini de sous-ensemble flous. Une variable linguistique correspond au triplet (V, X, T_v).* (Figure 4.6)

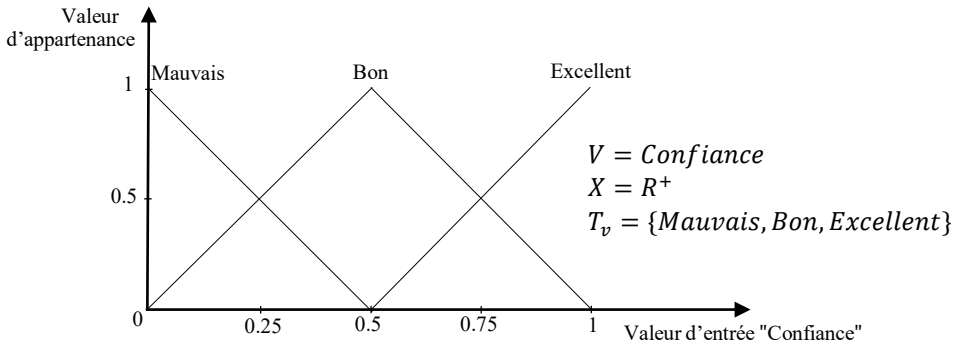


Figure 4.6 : Variable linguistique 'Confiance'.

3.3 Les opérateurs flous

Afin de pouvoir manipuler aisément les ensembles flous, nous redéfinissons les opérateurs de la théorie des ensembles classiques afin de les adapter aux fonctions d'appartenance propres à la logique floue permettant des valeurs strictement entre 0 et 1.

Contrairement aux définitions des propriétés des ensembles flous qui sont toujours les mêmes, la définition des opérateurs sur les ensembles flous est choisie, à l'instar des fonctions d'appartenance.

Voici les trois ensembles d'opérateurs les plus utilisés : pour le complément "Non", l'intersection "Et" et l'union "Ou".

Table 4.1 : Opérateurs sur les ensembles flous.

$Donémination$	$Intersection\ (ET)$ $\mu_{A\cap B}(x)$	$Réunion\ (OU)$ $\mu_{A\cup B}(x)$	$Complément\ (Non)$ $\mu_{\bar{A}}(x)$
$Opérateur\ de\ Zadeh\ (Min/Max)$	$min(\mu_A(x), \mu_B(x))$	$max(\mu_A(x), \mu_B(x))$	$1 - \mu_A(x)$
$Probabiliste\ (Prod/Probor)$	$(\mu_A(x) * \mu_B(x)$	$(\mu_A(x) + \mu_B(x)$ $- (\mu_A(x) * \mu_B(x)$	$1 - \mu_A(x)$

Avec les définitions usuelles des opérateurs flous, nous retrouvons toujours les propriétés de commutativité, distributivité et associativité des opérateurs classiques. Cependant, relevons deux exceptions notables, en logique floue :

- Le principe du tiers exclu est contredit : $A \cup \bar{A} \neq X$, autrement dit : $\mu_{A\cup\bar{A}}(x) \neq 1$.
- Un élément peut appartenir à A et non A en même temps : $A \cap \bar{A} \neq \phi$, autrement dit : $\mu_{A\cup\bar{A}}(x) \neq 0$. Notons que ces éléments correspondent à l'ensemble $supp(A) - noy(A)$.

3.4 Le raisonnement en logique floue

En logique classique, les raisonnements sont de la forme : { $Si\ P\ alors\ Q.$ ** $P\ vrai\ alors\ Q\ vrai.$ ** }.

En logique floue, le raisonnement flou, également appelé raisonnement approximatif, se base sur des règles floues qui sont exprimées en langage naturel en utilisant les variables linguistiques dont nous avons donné la définition précédemment. Une règle floue aura cette forme :

$$Si\ x \in A\ et\ y \in B\ alors\ z \in C, avec\ A, B, C\ des\ ensembles\ flous.$$

Les règles sont généralement exprimées sous la forme : "Si la variable est réglée Alors : l'action".

À l'instar des autres opérateurs flous, il n'existe pas de définition unique de l'application floue : le concepteur du système flou devra choisir parmi le large choix d'implications floues déjà définies, ou bien la définir à la main. Voici les deux définitions de l'implication floue les plus couramment utilisées :

$$Mamdani \rightarrow min(f_a(x), f_b(x))$$
$$Larsen \rightarrow f_a(x) * f_b(x)$$

Fait notable, ces deux implications ne généralisent pas l'implication classique. II existe d'autres définitions d'implication floue la généralisant, mais elles sont moins utilisées.

Le résultat de l'application d'une règle floue dépend donc de trois facteurs :

- La définition d'implication floue choisie.
- La définition de la fonction d'appartenance de l'ensemble flou de la proposition située en conclusion de la règle floue.
- Le degré de validité des propositions situées en prémisse.

Comme nous avons défini les opérateurs flous "Et", "Ou" et "Non", la prémisse d'une règle floue peut très bien être formée d'une conjonction de propositions floues. L'ensemble des règles d'un système flou est appelé la *matrice des décisions*.

4. MÉTHODE D'ANALYSE RELATIONNELLE GRISE "G. R. A"

4.1 Définition

La théorie du système gris proposée par J. Deng en 1982 [119] a été largement appliquée dans divers domaines. Il a été prouvé qu'il était utile pour traiter des informations insuffisantes, incomplètes et incertaines. L'analyse relationnelle grise "G. R. A" [120, 121], fait partie de la théorie des systèmes gris qui convient pour résoudre des problèmes avec des interrelations compliquées entre de multiples facteurs et variables, elle utilise un concept spécifique de l'information, qui est défini comme étant, les situations avec aucune information en "noir", et ceux qui ont une information parfaite en "blanc". Cependant, aucune de ces situations idéalisées ne se produit jamais dans les problèmes du monde réel. En fait, les situations entre ces deux extrêmes sont décrites comme étant grises ou floues. (Figure 4.7)

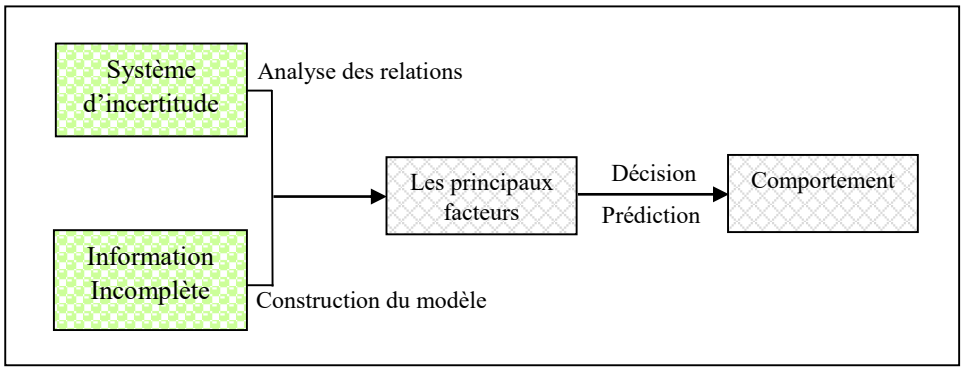


Figure 4.7 : Théorie des systèmes gris.

Avec cette définition, la quantité et la qualité des informations inférées forment un continuum allant d'un manque total d'information à une information complète, (du noir au gris jusqu'au blanc). Tant que l'incertitude existe toujours, on est toujours quelque part au milieu, quelque part entre les deux extrêmes, quelque part dans la zone grise.

L'analyse grise est alors un ensemble clair d'énoncés sur les solutions de système. À un extrême, aucune solution ne peut être définie pour un système sans aucune information. À l'autre extrême, un système avec une information parfaite a une solution unique. Au milieu, les systèmes gris donneront une variété de solutions disponibles. L'analyse grise ne cherche pas à trouver la meilleure solution, mais peut fournir des techniques pour déterminer une bonne solution.

L'idée de base de cette technique consiste à sélectionner certaines variables d'entrée qui montrent un impact plus fort à la sortie du système. Cette technique utilise les informations fournies par le système gris pour comparer dynamiquement et quantitativement chaque facteur et d'établir une relation en fonction du niveau des facteurs de similarité ainsi que du niveau de variabilité. Ensuite, le rapport de décision peut être prise en fonction de la relation.

4.2 Fonctionnement

L'analyse relationnelle grise est une méthode utile pour trouver l'importance des facteurs dans un système avec des données d'échantillonnage limitées. L'analyse relationnelle grise a été proposé à l'origine pour relier le facteur principal avec d'autres facteurs de référence dans un système donné. Pour cette raison, nous avons appliqué cette technique pour déterminer les variables d'entrée qui montrent un effet crucial à la sortie du système. En d'autres termes, la sortie du système peut saisir certaines informations utiles sur la variété de points de données de la séquence d'entrée. Les procédures d'analyse relationnelle grise sont présentées à la figure 4.8.

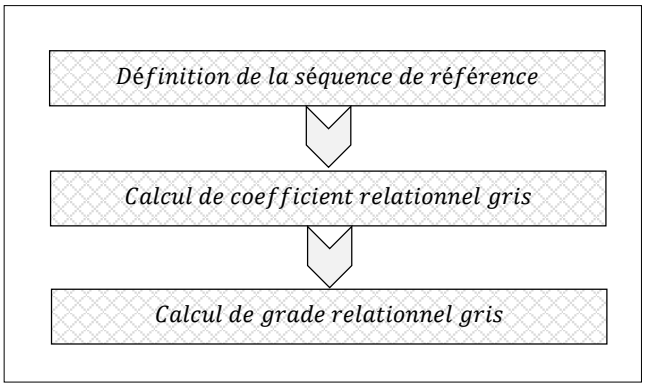


Figure 4.8 : Procédure d'analyse relationnelle grise.

Les détails de la procédure proposée sont présentés ci-dessous :

$Entrée$: La matrice comparative X, la suite optimale x^0.

$Sortie$: Le grade de corrélation γ_i.

$Étape\ 1\ (Prétraitement\ des\ données\ -\ Initialisation)$: La première étape est le prétraitement des données. Le prétraitement des données est normalement requis, lorsque les unités dans lesquelles la performance est mesurée sont différentes pour différents facteurs (attributs), l'influence de certains facteurs peut être négligée. Cela peut également se produire si certains facteurs de performances ont une portée très grande. En outre, si les buts et les directions de ces facteurs sont différents, il entraînera des résultats incorrects dans l'analyse [121]. Par conséquent, les données expérimentales originales doivent être prétraitées pour éviter de tels effets.

Le prétraitement des données est une méthode de transfert de la séquence de données originale "x_0" à une séquence comparable "x_i". À cet effet, les résultats expérimentaux sont normalisés dans une plage entre 0 et 1, cette étape de traitement est appelé *générateur relationnel gris*.

Selon les caractéristiques d'une séquence de données, il existe diverses méthodes de prétraitement des données disponibles pour l'analyse relationnelle grise :

- Si la valeur cible de la séquence d'origine est infinie, alors il a une caractéristique de "la plus grande est meilleure". La séquence originale peut être normalisée comme montré dans l'équation 4.1.

$$x_i^*(k) = \frac{x_i^0(k) - min\ x_i^0(k)}{\max x_i^0(k) - \min x_i^0(k)} \qquad (4.1)$$

- Quand la "plus petite est meilleure" est une caractéristique de la séquence originale, alors la séquence originale devrait être normalisée comme suit : (Équation. 4.2)

$$x_i^*(k) = \frac{max\ x_i^0(k) - x_i^0(k)}{max\ x_i^0(k) - min\ x_i^0(k)} \qquad (4.2)$$

- Cependant, s'il existe une valeur cible définie (valeur souhaitée) à atteindre, la séquence originale sera normalisée sous la forme suivante : (Équation. 4.3)

$$x_i^*(k) = 1 - \frac{|x_i^0(k) - x_i^0|}{\max x_i^0(k) - x_i^0} \qquad (4.3)$$

Où, $(1 \leq i \leq m)\ et\ (1 \leq k \leq n)$

m est le nombre d'éléments de données expérimentaux et n est le nombre de paramètres. $x_i^0(k)$ désigne la séquence originale, $x_i^*(k)$ est les séquences après le prétraitement des données. $min\ x_i^0(k)$ et $max\ x_i^0(k)$ sont la plus petite et la plus grande valeurs de $x_i^0(k)$ respectivement, et x^0 est la valeur désirée.

Définition 4.8 : *Soit X un ensemble relationnel gris étant utilisé comme un ensemble de facteurs d'évaluation, X est défini comme suit :* $X = \{x_1, x_2, \dots, x_i, x_{i+1}, \dots, x_n\}$, où, x_i est un facteur d'évaluation.

Définition 4.9 : *La matrice comparative X est constituée par les valeurs d'évaluation de chaque facteur, tel que l'ensemble de* m *séquences de comparaison est défini par :* $X = \{x_1, x_2, \dots, x_i, x_{i+1}, \dots, x_m\}$. (Équation. 4.4)

$$X = [x_1, x_2, \dots, x_m]^T = \begin{pmatrix} x_{11} & \cdots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mn} \end{pmatrix} \qquad (4.4)$$

Définition 4.10 : *La suite optimale* x^0 *est fondée sur la sélection de la valeur maximale de chaque facteur d'évaluation dans la matrice comparative X.*

$$x^0 = [x_{01}, x_{02}, \dots, x_{0n}] \qquad (4.5)$$

La sélection de la valeur optimale est principalement déterminée par le type de facteurs d'évaluation, elle est principalement divisée en trois types de cas, comme a été expliqué plus haut (Équations : 4.1, 4.2 et 4.3).

Étape 2 ($Calcul\ de\ coefficient\ relationnel\ gris$) : À la suite de données de prétraitement, un coefficient relationnel gris est calculé pour exprimer la relation entre les résultats expérimentaux actuels et les résultats expérimentaux normalisés. Le coefficient relationnel gris peut être exprimé comme suit :

$$\xi_i(k) = \frac{\Delta_{min} + \rho\, \Delta_{max}}{\Delta_{oi}(k) + \rho\, \Delta_{max}} \qquad (4.6)$$

Tel que, Δ_{oi} est la séquence d'écart entre la séquence de référence et la séquence de comparabilité, à savoir. (Équation 4.7)

$$\left\{\begin{array}{c}\Delta_{oi}= \|x_0^*(k) - x_i^*(k)\| \\ \Delta_{min}= min_{\forall j\in i} min_{\forall k}\left\|x_0^*(k) - x_j^*(k)\right\| \\ \Delta_{max}= max_{\forall j\in i} max_{\forall k}\left\|x_0^*(k) - x_j^*(k)\right\|\end{array}\right\} \qquad (4.7)$$

Où, $\xi_i(k)$ est le coefficient relationnel gris pour toutes les données, $x_0^*(k)$ désigne la séquence de référence et $x_i^*(k)$ désigne la séquence de comparabilité. ρ est le coefficient d'identification, $\rho \in [0,1]$ (la valeur de ρ peut être ajustée en fonction des exigences réelles du système). $\rho = 1/2$ est généralement utilisé.

Étape 3 (*Calcul de grade relationnel gris*) : Une fois que le coefficient relationnel gris est dérivé, il est habituel de prendre la valeur moyenne des coefficients relationnels gris comme grade relationnel gris. Le grade relationnel gris est défini comme suit : (Équation 4.8).

$$\{\gamma_i = \frac{1}{n}\sum_{k=1}^{n} w_k \xi_i(k), \quad \sum_{k=1}^{n} w_k = 1\} \qquad (4.8)$$

Où, n est le nombre de la fonction objective ou de la séquence de référence $x_0^*(k)$ et w_k représente le poids normalisé du facteur k.

Le grade relationnel gris représente le niveau de corrélation entre la séquence de référence et la séquence de comparabilité. Si les deux séquences sont identiques, alors la valeur du grade relationnel gris est égale à 1. Le grade relationnel gris indique également le degré d'influence que la séquence de comparabilité pourrait exercer sur la séquence de référence. Par conséquent, si une séquence de comparabilité particulière est plus importante que les autres séquences de comparabilité par rapport à la séquence de référence, alors le grade relationnel gris pour cette séquence de comparabilité sera plus élevé que les autres grades [121].

5. EXIGENCES RELATIVES À LA CONCEPTION DE NOTRE MODÈLE

Dans ce travail, le modèle de gestion de la confiance que nous allons proposer dans le chapitre suivant pourrait être utilisé dans tous les protocoles de routage MANETs pour renforcer la coopération entre tous les nœuds du réseau. Ce modèle nécessite les fonctionnalités suivantes pour accomplir ses fonctions correctement.

1. Notre modèle de confiance devrait être :

   a. *Sans infrastructure* : L'infrastructure de routage réseau est formée de manière Ad hoc.

b. *Subjectif* : La confiance dépend du nœud cible et de la proximité des recommandations au nœud source, de sorte que la confiance est de nature subjective.

c. *Dynamique* : La confiance est de nature dynamique, car la valeur de la confiance est mise à jour après chaque interaction.

d. *Évolutif* : les composantes du modèle de confiance permettent d'admettre plusieurs facteurs dans le processus d'évaluation de la confiance.

2. Tous les nœuds communiquent via un canal sans fil partagé.
3. Tous les canaux de communication sont bidirectionnels et les intervalles de transmission des nœuds sont plutôt égaux.
4. Les interfaces sans fil prennent en charge le mode de fonctionnement en promiscuité.
5. Tous les nœuds utilisent des émetteurs-récepteurs omnidirectionnels.

6. CONCLUSION

Dans le présent chapitre, nous avons présenté deux mécanismes qui nous semblent nécessaires pour fournir une solution de sécurité efficace et fiable pour les réseaux mobiles Ad hoc. Comme on peut le voir d'après l'analyse présentée ci-dessus, ce qui manque dans les réseaux Ad hoc est la "confiance", car chaque nœud ne doit pas faire confiance à un autre nœud immédiatement. Si la relation de confiance entre les nœuds de réseau est disponible pour chaque nœud, il sera beaucoup plus facile de sélectionner la mesure de sécurité appropriée pour établir la protection requise. Il sera aussi plus sage d'éviter les nœuds non fiables en tant que routeurs. En outre, il sera plus judicieux de rejeter ou d'ignorer les demandes hostiles de service.

Pour réduire les dangers des nœuds malveillants et renforcer la sécurité du réseau, il est important de développer de tels mécanismes de gestion de la confiance qui permettent à un nœud d'évaluer la fiabilité des autres nœuds sans que les autorités centrales construisent un environnement de confiance. De tels mécanismes aident non seulement à la détection de nœuds malveillants, mais améliorent également la performance du réseau puisque les nœuds honnêtes peuvent éviter de travailler avec des nœuds non fiables.

En effet, les mécanismes proposés doivent être au service de la politique de sécurité imposée par l'usage et non le contraire. Une politique trop restrictive n'offrira que très peu de possibilités d'interaction et donc rendra le système inopérant. Il en est de même pour une politique très permissive qui n'engendra aucune confiance entre les utilisateurs.

Chapitre 5

Un nouveau modèle de gestion de la confiance

1. INTRODUCTION

Précédemment, nous avons donné un état de l'art sur les différents protocoles de routage basés sur la sécurité et qui sont dédiés aux réseaux mobiles Ad hoc. La recherche dans ce domaine a été très fructueuse ces dernières années avec de nombreuses propositions et améliorations au cours du temps. Ces protocoles peuvent être optimaux dans certaines applications mais pas dans tous les cas. La recherche dans la sécurité des réseaux Ad hoc est ouverte pour de nouvelles idées afin d'optimiser encore les protocoles existants pour obtenir de meilleures performances.

En fait, la plupart des protocoles de communication dans la communauté Ad hoc tendent à trouver le chemin le plus court d'une source à la destination sans tenir compte de la présence de tout nœud malveillant dans le chemin de routage. Nous pouvons argumenter qu'une menace interne dans le réseau qui peut être un nœud compromis ou déloyal est une préoccupation importante, c'est-à-dire, un chemin libre du nœud malveillant est plus important que le chemin le plus court [122].

Ce chapitre est dédié à nos contributions de recherche dans le domaine de la sécurité des communications dans les réseaux mobiles Ad hoc, il s'agit d'un nouveau modèle de gestion de la confiance qui permet à un propriétaire de ressources d'inférer la fiabilité d'un demandeur d'une manière Ad hoc. Le point clé de notre modèle de confiance est que la fiabilité est calculée sur la base du modèle de la théorie floue, chaque nœud peut calculer la valeur de confiance pour ses voisins et la maintenir dans sa table de confiance.

Comme le processus de routage est également flou, le modèle proposé vise à rendre la décision de routage de confiance réalisable dans les MANETs.

Dans ce chapitre, nous allons tout d'abord présenter notre modèle de gestion de la confiance. Après, nous allons aborder la description et le fonctionnement de ce modèle. Le point suivant qui va être présenté est un algorithme de routage sécurisé basé sur le modèle de confiance proposé. Enfin, un protocole de routage est présenté comme une application de l'algorithme de routage fourni.

2. LE MODÉLE DE CONFIANCE PROPOSÉ

2.1 Modèle de réseau de confiance "Trust Network Model"

Dans un environnement distribué, comme dans les réseaux mobiles Ad hoc, la gestion de la confiance dans le réseau considère le nœud comme un agent pour obtenir les informations de confiance. Pour cette raison, notre modèle de confiance peut être représenté comme un graphe pondéré dirigé appelé le graphe de confiance (Figure 5.1).

Nous modélisons notre modèle de confiance par un graphe de confiance, qui est notée $G = < V, E, \omega >$, telle que :

- L'ensemble d'entités de confiance (Vertices de confiance) peut être défini comme $V = \{v_1, v_2, \dots, v_n\}$, où n est la taille du réseau.
- E est un ensemble de liaisons sans fil (Liens) et $|E|$ est le nombre de liens de réseau dirigés. Chaque e_{ij} dans E représente un lien dirigé du nœud V_i à son nœud voisin V_j.
- $\omega: \omega(e_{ij}) \rightarrow R \in [0,1]$, désigne la valeur de confiance (un nombre réel entre 0 et 1) de chaque lien e_{ij}.

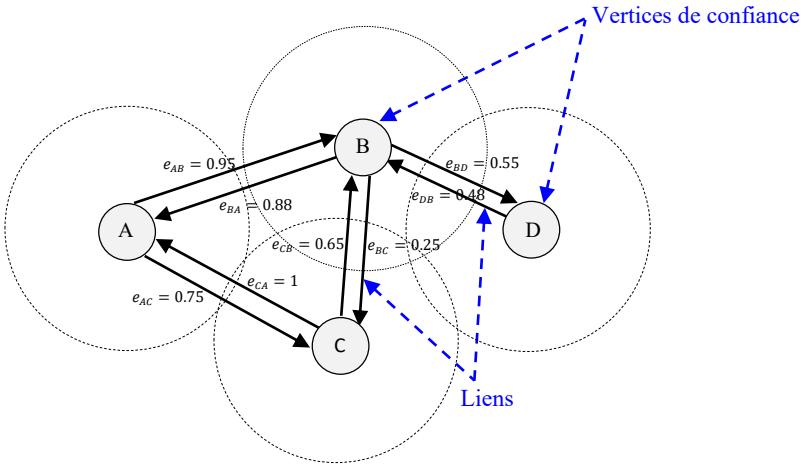


Figure 5.1 : Graphe de confiance.

2.2 *Le schéma proposé*

Les systèmes de gestion de confiance et de réputation tiennent une place très importante dans le domaine relatif à la sécurité des communications dans les réseaux Ad hoc. Il existe trois avantages majeurs liés à l'utilisation de ces systèmes. 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.

Dans ce contexte, nous nous intéressons à la mise en place d'une solution visant à rendre un protocole de routage résistant à l'action des nœuds malveillants, les nœuds sont autorisés de fonctionner

de manière distribuée, de surveiller le réseau et de fournir une mesure efficace et fiable de toute menace pouvant être initiée par chacun des autres nœuds.

Dans cette section, nous donnons une description détaillée du modèle de confiance proposé, le schéma adopté pour choisir le chemin le plus sûr dans le réseau est présenté ainsi que le mécanisme d'évaluation de la confiance. Notre modèle de gestion de la confiance est présenté à la figure 5.2 et ses composants sont décrits dans les paragraphes suivants. Comme le montre la figure 5.2, les étapes de notre modèle de confiance sont énumérées comme suit :

2.2.1 Module de surveillance et de collecte d'informations

C'est la phase où le réseau est nouvellement déployé, ou lorsqu'un nouveau nœud rejoint le réseau. Dans un réseau nouvellement déployé, les nœuds n'ont aucun historique de statistiques de trafic. Le scénario d'un nouveau nœud rejoignant le réseau indique que ce nœud n'a aucune information de confiance sur ses voisins et vice-versa. C'est la période où les nouveaux nœuds n'ont pas établi de clés avec leurs voisins à deux sauts ou plus.

La fonction principale de ce module est de surveiller les transmissions de paquets (telles que l'envoi de paquets, l'abandon, l'injection de fausse route, etc.) et de recueillir autant que possible les informations sur le comportement de l'entité. Ces informations peuvent provenir de plusieurs sources telles que, les expériences directes, les expériences de connaissances, les entités pré-approuvées, etc. Si ces informations ne proviennent pas d'expériences directes, l'intégrité de l'information en tant qu'autre point devrait être prise en compte. Autrement dit, la confiance accordée aux recommandations ou aux opinions des autres entités. Les expériences pourraient être représentées comme des valeurs de réputation, ou comme un ensemble de transactions antérieures positives et négatives.

Dans notre scénario, chaque nœud surveille indépendamment les activités d'acheminement de paquets de ses voisins dans le mode de "$promiscuous$". Cette surveillance est liée à la proportion de paquets correctement acheminés par rapport au nombre total de paquets transmis pendant une session de temps fixe (intervalle de temps).

2.2.1.1 Principe

Ce module consiste à fournir les mécanismes et les paramètres qui sont nécessaires pour associer une valeur de confiance (valeur numérique) à chaque nœud dans le système, via sa table de routage. Un mécanisme basé sur la notion de $réputation$ est mis en place. Toutefois, si un nœud réussi très régulièrement à acheminer un paquet de données avec un même nœud, sa réputation peut devenir importante et donc autoriser des accès à des services plus évolués dans le réseau (notamment le service d'authentification).

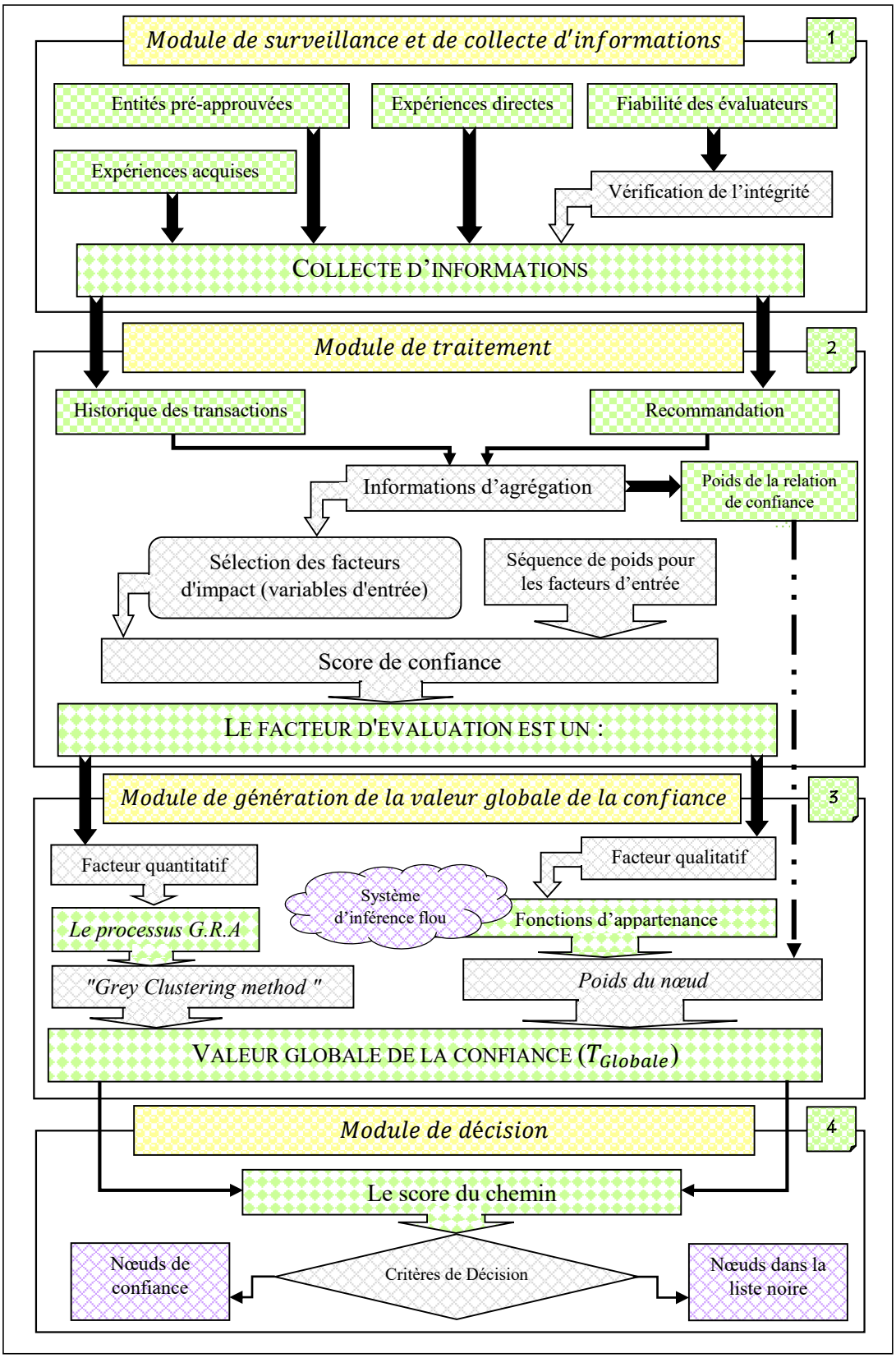


Figure 5.2 : Le modèle de confiance proposé.

Dans ce type de réseaux, la notion de réputation est limitée à des interactions de type un à un, et donc n'aura que très peu d'impact sur le réseau. Pour augmenter la portée dans notre modèle, nous proposons d'introduire un autre mécanisme basé sur le principe de $recommandation$. La confiance locale pour une entité (nœud ou participant) peut donc être transmise, l'acceptation d'une recommandation étant assujettie également au degré de confiance accordé à l'entité qui propose cette recommandation.

Cependant, la question qui se pose ici, est comment publier la confiance dans notre modèle tout en garantissant sa validité ? Pour cela, on va définir quelques paramètres qui peuvent nourrir le déroulement de notre modèle :

- Nous supposons qu'il existe une relation sociale entre les nœuds dans le but d'établir des relations de confiance.
- Une valeur de confiance (*Trust value*) T_v (T_v: valeur continue dans l'intervalle :]0, 1]).
- Une valeur de réputation (*Reputation value*) R_v (R_v : valeur continue dans l'intervalle : [0, 1]).
- Un nœud (i) possède une valeur de confiance plus élevé ($T_v(i) = 1$), s'il est connu par d'autres nœuds de confiance et a échangé ses clés via un canal sécurisé (rencontre physique par exemple) [123] avec un ou plusieurs nœuds de confiance.
- Une valeur de confiance très élevé, existe aussi si le nœud a prouvé sa totale coopération et son bon comportement (R_v=1) (principe de réputation).
- Si un nouveau nœud est ajouté à la liste des nœuds de confiance par un ou plusieurs nœuds de confiance, les autres nœuds doivent mettre à jour leurs listes des nœuds de confiance.
- Chaque nœud dispose dans sa table de routage de deux tables "$champs$" (un champ de confiance et un champs de réputation), qui seront actualisés à chaque changement de T_v et/ou de R_v.
- Chaque nœud inconnu commence avec la plus bas valeur de confiance (T_v= 0,1) et le plus bas niveau de réputation (R_v= 0). L'idée de ce principe consiste à obliger les nœuds inconnus à coopérer et bien se comporter [124].
- Pour estimer le chemin de confiance entre deux nœuds, on propose de prendre la valeur minimale entre leurs deux valeurs de confiance.

2.2.1.2 Fonctionnement

Dans cette étape de notre modèle de confiance, lorsque deux nœuds veulent communiquer sans connaissance préalable, ils s'échangent leur liste de certificats et vont essayer de créer une chaîne de confiance entre eux (Figure 5.3). Supposons qu'un élément "i" veuille communiquer avec un autre élément (nœud) "k", si "i" fait confiance en un troisième élément "j" (étape 1), et "k" fait aussi

confiance en "j" (étape 2), alors une chaîne de confiance entre "i" et "k" pourra être établie via "j" (le principe de recommandation).

Dans ce cas, "i" peut donner physiquement sa clé publique à "j" (main à main ou par téléphone par exemple, etc.) (étape 3), l'élément "j" connaît "i" et donc signe sa clé publique (étape 4), puis il redonne la clé signée (étape 5) et en garde une copie (étape 6). Quand "i" veut communiquer avec "k", il lui envoie une copie de la clé que "j" a signée (étape 7). Le nœud "k", qui a déjà la clé publique de "j" (il l'a eu à un autre moment) et qui fait confiance à "j" pour certifier les clefs d'autre nœuds, vérifie sa signature sur la clé de "i" et l'accepte (étape 8). De ce fait "j" a recommandé "i" à "k".

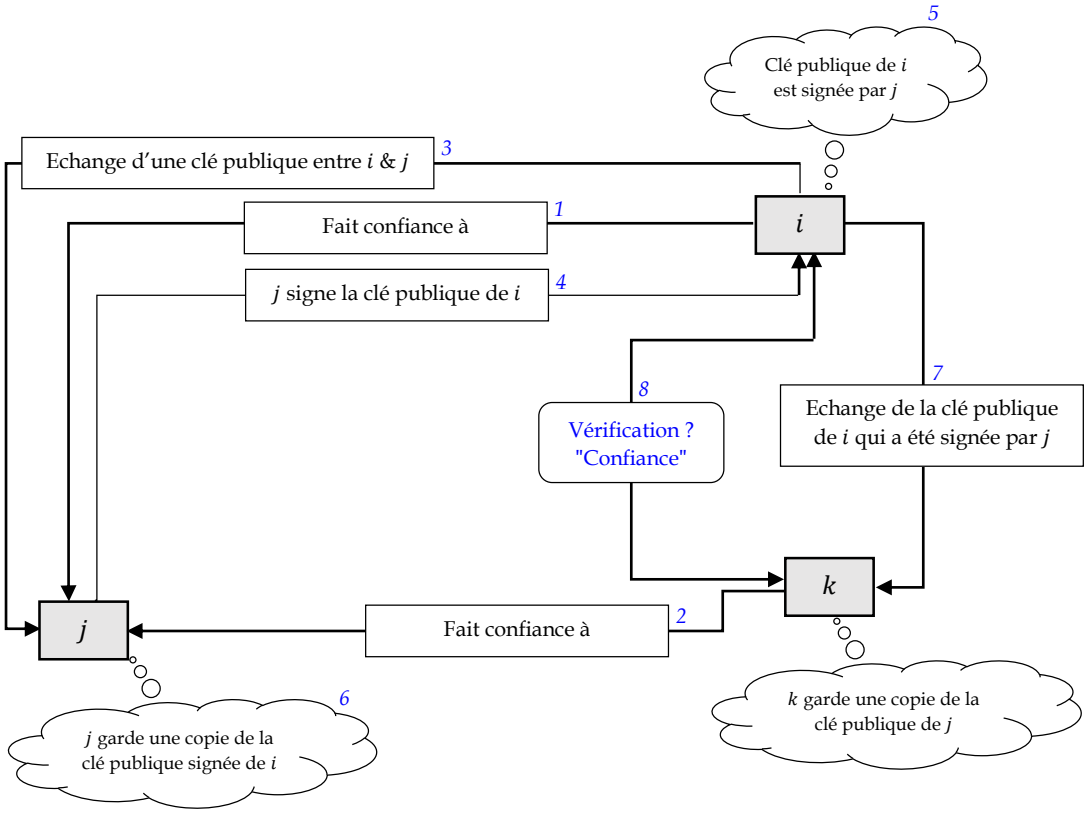


Figure 5.3 : Création d'une chaîne de confiance.

Dans ce module, chaque nœud ayant un degré de confiance élevé contrôle ses nœuds voisins, c'est-à-dire ceux qui ont un degré de confiance faible. Dans le cas que nous étudions, le processus de contrôle agit sur deux couches différentes du réseau :

a) La couche Mac : Les nœuds responsables du contrôle surveillent l'occupation du canal de communication par leurs voisins. Cette opération consiste à mesurer la durée de l'occupation du canal par les nœuds. Le but de cette fonction est de détecter les nœuds qui exercent un certain type de comportement égoïste [124], les nœuds égoïstes trichent en choisissant leur back-off, dans le but

d'obtenir une bande plus importante et de pénaliser les nœuds qui se comportent bien. Nous supposons que les nœuds chargés du contrôle à ce niveau génèrent un rapport noté (R_1) sur ses voisins qui ont un degré de confiance faible.

b) La couche réseau : Les nœuds chargés du contrôle surveillent les activités de transmission de paquets de leurs nœuds voisins, qui ont un degré de confiance faible. Cette idée est basée sur le paramètre de coopération des nœuds dans le réseau. La définition de ce paramètre consiste à calculer pour chaque nœud la proportion de paquets bien retransmis par rapport au nombre total de paquets devant être transmis sur une certaine période. Cette période consiste à collecter les informations données par les nœuds pour calculer le niveau de réputation. Soient deux nœuds "i" et "j" avec ($T_v(i) > T_v(j)$), dans ce cas, le nœud "i" peut contrôler le nœud "j". Le nœud "i" envoie un certain nombre de paquets de données au nœud "j" avec un autre nœud comme destination, et après une période de temps limitée, le nœud peut calculer le niveau de réputation :

$$R_2(i,j) = \frac{Nombre\ des\ paquets\ acheminés}{Nombre\ total\ des\ paquets} \qquad (5.1)$$

Comme nous avons déjà expliqué précédemment, chaque nœud inconnu commence avec une valeur de réputation la plus faible ($R_v = 0$) et ce degré augmente au fur et à mesure que le nœud prouve sa coopération et son bon comportement. Les niveaux de réputation générés par les nœuds sont liés aux degrés de confiance correspondant à chaque nœud. Le rapport final concernant le nœud "j" généré par chaque nœud chargé du contrôle "i", est :

$$R(i,j) = \frac{R_1(i,j)+R_2(i,j)}{2} \qquad (5.2)$$

2.2.2 *Module de traitement*

Ce module devrait agréger correctement toutes les informations reçues et calculer un score pour chaque nœud du réseau. Autrement dit, il recevrait soit une série de recommandations sur la fiabilité de l'entité, soit un ensemble de transactions pondérées ou les deux. Une fois qu'un historique de transaction d'entité a été collecté et correctement pondéré, une nouvelle valeur de confiance doit être calculée pour cette entité.

Lors de la pondération des informations comportementales collectées et du calcul du score de confiance, les résultats de la transaction la plus récente devraient avoir le plus de poids. En accordant une plus grande attention aux interactions plus récentes, il est plus facile d'estimer avec précision le comportement actuel d'une entité et d'empêcher les oscillations comportementales dans le temps.

La subjectivité devrait être autorisée dans l'évaluation d'une interaction, c'est-à-dire que chaque entité peut avoir ses propres critères lors de l'évaluation d'un service reçu. Une entité peut évaluer une interaction réalisée, d'une manière individuelle ou avec des opinions et même un consensus d'autres parties. Permettre l'évaluation subjective de l'interaction rend le score de confiance plus approprié avec les objectifs spécifiques et les nécessités de chaque entité dans le système.

2.2.3 *Module de génération de la valeur globale de la confiance*

Dans ce module, nous utilisons plusieurs paramètres pour mesurer les valeurs de confiance, en fonction des ensembles flous et du processus *G.R.A* que nous avons mentionnés dans la section précédente. Ces paramètres sont choisis en tant qu'éléments de base du processus de communication, c'est-à-dire, l'ensemble minimal des paramètres requis pour couvrir tous les types d'attaques contre des protocoles de niveau inférieur. Selon le processus *G.R.A*, nous sélectionnons certaines variables d'entrée qui montrent un impact plus fort sur la sortie du système.

Dans notre modèle, nous considérons les facteurs d'impact suivants :

le taux de perte de paquets, *la vitesse de transmission*, *la puissance du signal reçu* et *le taux de changement de signal*.

Par conséquent, le jugement sur le fait de savoir si un nœud doit être approuvé ou non, n'est pas seulement déterminé par la probabilité d'interactions réussies mais aussi par ces différents paramètres dans les couches physique et MAC.

2.2.4 *Module de décision*

Après avoir calculé l'approbation résultante, ce module est appelé pour catégoriser les nœuds en fonction de leur métrique de confiance. L'objectif de l'étape de décision est de sélectionner les solutions les plus appropriées afin d'obtenir le meilleur effet global de l'ensemble du processus de décision.

2.3 *Description et fonctionnement du modèle*

Nous considérons un MANET avec "n" nœuds mobiles. Les nœuds communiquent entre eux par l'intermédiaire d'un canal sans fil à bande passante limitée, sujet aux erreurs et non sécurisé. "n" peut changer dynamiquement lorsque les nœuds mobiles se joignent, quittent ou échouent avec le temps. En outre, "n" n'est pas limité, il peut y avoir un grand nombre de nœuds dans le réseau. Le réseau ne fournit aucun support d'infrastructure physique ou logique. Pour notre conception, la fiabilité du transfert de paquets à sauts multiples basée sur la couche de transport sous-jacente et le routage Ad hoc n'est pas supposée.

Pour assurer la praticité de notre modèle de confiance, nous faisons les hypothèses suivantes :

- La plupart des nœuds du réseau sont des nœuds normaux (c'est-à-dire qu'ils fonctionnent bien et se comportent en coopération). Les comportements inacceptables ou malveillants participant au réseau doivent être sanctionnés.
- Les nœuds normaux sont stimulés pour coopérer de manière adéquate sur le réseau.
- La communication entre nœuds voisins à un saut est considérée plus fiable que la communication multi-sauts, en raison de l'exposition limitée aux erreurs de canal, ainsi que la détection des erreurs intégrées et les mécanismes de retransmission à un saut de la norme IEEE 802.11.
- Chaque nœud devrait également être capable de détecter les comportements présentés par les nœuds malveillants, cette hypothèse est basée uniquement sur les mécanismes de détection locaux (*Observations directes et/ou indirectes*). Cette dernière repose sur le constat que la détection d'intrusion dans les réseaux Ad hoc est généralement beaucoup plus difficile que dans leur homologue filaire [125], la surveillance et la détection des mauvais comportements parmi les voisins à un saut sont facilement plus aisés et plus pratiques en raison de la nature de diffusion de la transmission sans fil [1].

Comme un premier pas vers la solution proposée, nous considérons le réseau comme un réseau basé sur l'opinion. À cet effet, le réseau est adapté à une architecture où il y a un groupe de nœuds voisin et leur nœud superviseur. Ce réseau hiérarchique peut également être considéré comme un système d'agrégation d'informations distribué. Chaque nœud du réseau conserve une table d'informations de transfert de données. La table inclut uniquement les informations de transaction de transmission de données en entendant les nœuds voisins, afin d'évaluer la confiance dans notre modèle.

Notre modèle de gestion de la confiance doit effectuer essentiellement l'évaluation de la confiance, le calcul et l'application.

2.3.1 Évaluation de la confiance (Trust evaluation)

L'évaluation de la confiance est considérée comme le noyau du système de gestion de la confiance, y compris la synthèse et la mise à jour de la confiance. Un nœud évaluateur (*le nœud confiant*) quantifie toutes les informations pertinentes concernant un nœud évalué (*le nœud crédible*), y compris les comportements, les observations, les enregistrements d'interaction et les vues provenant d'autres nœuds. Ensuite, il utilise un modèle approprié pour quantifier la crédibilité du nœud évalué.

Dans ce cadre, nous évaluons la confiance sur une échelle continue en tenant compte à la fois la confiance des nœuds et la confiance dans l'itinéraire. Une confiance d'un nœud peut être considérée comme une mesure subjective de la qualité d'acheminement des nœuds, tandis qu'une confiance d'acheminement peut être utilisée pour anticiper la qualité des transferts de paquets le long de l'itinéraire. Pour ce but, nous allons définir les trois paramètres suivants : "T_m, V_i et V_j", où T_m (métrique de confiance) indique la confiance d'un nœud, qui est définie dans une plage continue entre

$'0'$ et $'1'$ $(0 \leq T_m \leq 1)$, V_i et V_j représentent *l'entité évaluatrice*[1] et *l'entité évaluée*[2] respectivement. La valeur de confiance $'0'$ signifie une méfiance totale, tandis que la valeur $'1'$ implique une confiance absolue. La valeur de confiance exprime le degré auquel un nœud s'attend à ce qu'un autre nœud offre certains services. La confiance n'est pas nécessairement symétrique. Le fait que "i" fasse confiance à "j" ne signifie pas nécessairement que "j" fait confiance à "i", où "i" et "j" sont deux entités.

Basé sur le schéma présenté dans la dernière section, la gestion de la confiance peut être modélisée comme une procédure avec trois phases séquentielles.

2.3.1.1 Établissement de la confiance directe

Une confiance directe d'un nœud est l'information de première main sur les voisins, cette information est basée sur les preuves capturées lors des expériences individuelles avec l'autre nœud. Le module de traitement transforme ensuite la relation de confiance ou les relations de confiance inhérentes en métriques de confiance directe (c'est-à-dire, des descriptions quantitatives de confiance).

Généralement, les relations de confiance ont simplement des propriétés floues. Nous pouvons ensuite utiliser la théorie floue pour décrire le modèle de confiance directe. Selon la table d'acheminement de données, le nœud "i" peut faire l'évaluation de la confiance au nœud "j", noté $\{Source, Destination, Score, t\}$, où : $'la\ source'$ est les nœuds évaluateurs de confiance, $'la\ destination'$ est les nœuds évalués et $'le\ score'$ est la valeur de confiance à la destination à l'instant $'t'$. La fonction d'appartenance floue du modèle de confiance directe est définie comme suit :

$$DT_{ij}(t) = DT(v_i, v_j) = \frac{PT_{ij}(t)}{PT_{ij}(t) + \mu * NT_{ij}(t) + \theta} \qquad (5.3)$$

Où :

- $DT_{ij}(t)$ Représente la valeur de confiance directe du nœud "i" au nœud "j" à l'instant $'t'$. $\{0 \leq DT_{ij} \leq 1\}$
- PT_{ij} Indique le nombre de transactions positives et NT_{ij} indique le nombre de transactions négatives que le nœud "i" a été effectué avec le nœud "j".
- "μ" Représente les poids du comportement négatif passé qui peut être régulé pour punir l'action du nœud égoïste, c'est-à-dire, plus la valeur de "μ" est grande, plus le degré de punition est grand.

[1] Monitoring node.
[2] Monitored node.

- La constante "θ" est une confiance d'incertitude pour la valeur de poids, elle est utilisée pour ajuster le taux de défaillance. C'est-à-dire que, plus la valeur de "θ" est grande, plus le rythme de déclin de l'échec est lent.

Puisque le comportement d'un nœud n'est pas toujours constant mais change souvent avec le temps, les expériences récentes sont plus crédibles que les expériences antérieures. Ainsi que, les événements historiques doivent être juxtaposés aux événements récents afin de mettre à jour le PT_{ij}, comme indiqué dans l'équation (5.4).

$$\left\{ \begin{array}{l} PT_{ij} + NT_{ij} = AT_{ij} \\ PT_{ij} - NT_{ij} = SF_{ij} \end{array} \right\} \qquad (5.4)$$

AT_{ij} Dénote le nombre total de transactions que le nœud "i" a fait avec le nœud "j", et SF_{ij} (*facteur de satisfaction*) est la somme des évaluations de toutes les interactions du nœud "i" avec le nœud "j". Si ($SF_{ij} > 0$), cela indique que le nœud "i" a une évaluation positive au nœud "j", sinon il a une évaluation négative au nœud "j".

2.3.1.2 Inférence de la confiance indirecte (Recommandation de confiance)

La confiance indirecte est l'information de seconde main obtenue à partir d'autres nœuds. Ce modèle de confiance est construit sur la confiance directe ainsi que les recommandations de nœud. En d'autres termes, le modèle de confiance indirecte intègre le modèle de confiance directe et le modèle de recommandation. Notez que, lorsque les relations de confiance directes sont toutes non transitives, aucune confiance indirecte ne peut être dérivée dans cette phase. Nous notons également que la confiance indirecte est importante pour que tout routage basé sur la confiance soit pratique et réalisable dans les MANETs.

Lorsque le nœud "i" veut communiquer avec le nœud non-voisin "j", ils échangent leur liste de certificats et essaient de créer une chaîne de confiance entre eux. S'il n'y a pas de certification ou si la certification a expiré, le nœud "i" rassemblera les informations d'évaluation de confiance du nœud "j" à partir de ses voisins ou d'autres nœuds, puis les combinera.

Semblable à la fonction de confiance directe, la valeur de confiance recommandée "RT" du nœud évaluateur sur le nœud évalué (Figure 5.4) est calculée comme suit :

$$RT_{kj}(t) = RT(v_k, v_j) = \frac{PR_{ij}(t)}{PR_{ij}(t) + \mu * NR_{ij}(t) + \theta} \qquad (5.5)$$

Où :

- $RT_{kj}(t)$ représente la valeur de confiance recommandée du nœud "k" sur le nœud "j".

$\{0 \leq RT_{kj} \leq 1\}$

- PR_{ij} indique le nombre de fois de la recommandation positive et NR_{ij} indique le nombre de fois de la recommandation négative que le nœud "i" a été effectué avec le nœud "j".

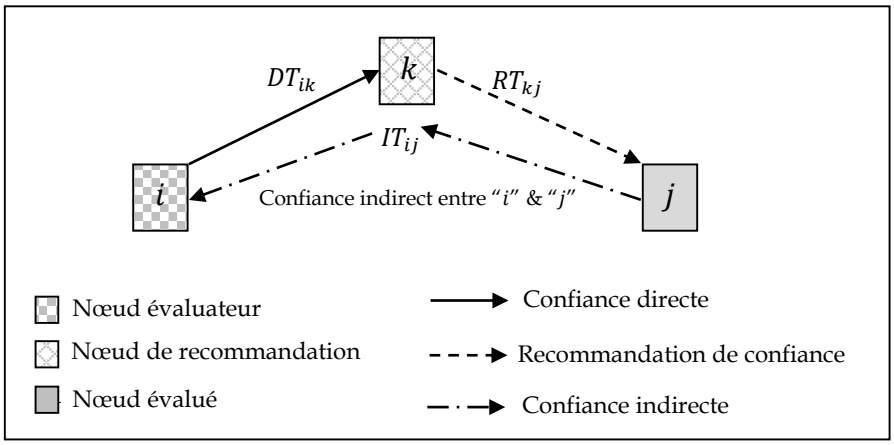


Figure 5.4 : Évaluation de la confiance indirecte dans notre modèle.

Supposons que le nœud "i" possède les expériences de transaction directes et l'évaluation de confiance directe DT_{ik} vers le nœud "k", au même temps le nœud "k" a une recommandation de confiance RT_{kj} du nœud "j". Alors le nœud "i" peut obtenir une évaluation de confiance indirecte IT_{ij} vers le nœud "j" à travers le nœud "k". La relation de confiance indirecte IT_{ij} est une fonction composite d'une relation de recommandation et d'une relation de confiance directe, une équation pour IT_{ij} peut être calculée comme suit :

$$IT_{ij}(t) = IT(v_i, v_j) = RT_{kj}(t) \text{ o } DT_{ik}(t) \qquad (5.6)$$

Où, $IT_{ij}(t)$ Indique la relation de confiance indirecte entre le nœud "i" et le nœud "j" à l'instant "t" $\{0 \leq IT_{ij} \leq 1\}$.

Dans cette section, il convient de mentionner que divers nœuds dans notre modèle pourraient fournir un certain nombre de recommandations sur les mêmes nœuds, c'est-à-dire que les différents nœuds auraient des évaluations de confiance différentes et même opposées vers le même nœud.

Dans ce cas, comme illustré dans la figure 5.5, le nœud "i" a des expériences d'interaction directe avec les nœuds "k_1", "k_2" et "k_3", donc il a trois relations de confiance directes. En même temps, le nœud "k_1" donne la recommandation d'évaluation de confiance $RT_{k_1 j}$ au nœud "j", le nœud "k_2" fait la recommandation de confiance $RT_{k_2 j}$ au nœud "j" et le nœud "k_3" fait aussi la recommandation d'évaluation de confiance $RT_{k_3 j}$ au nœud "j".

Le but de combiner ces recommandations avec les évaluations de confiance directes utilisées, est de fournir une évaluation relativement objective du nœud "j", qui sera calculée comme suit :

$$IT_{ij} = IT(v_i, v_j) = \frac{1}{n}\sum_{i=1}^{n} IT_{ij}(k_i) \qquad (5.7)$$

Où, n est le nombre de nœuds de recommandation.

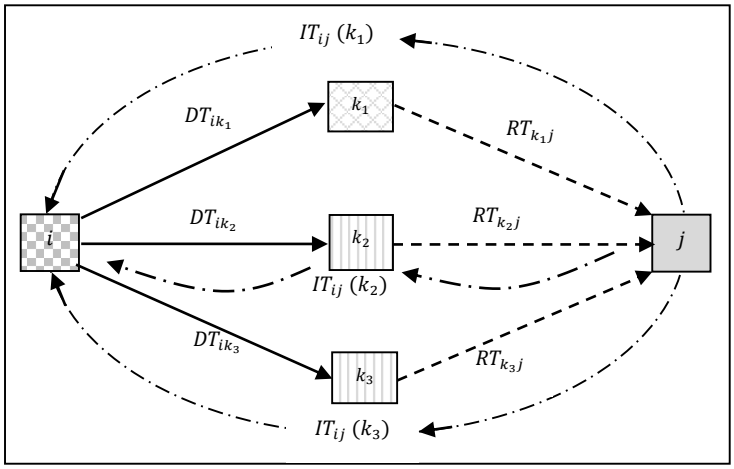


Figure 5.5 : Évaluation de la confiance indirecte avec plusieurs nœuds de recommandation.

2.3.1.3 Actions basées sur la confiance

Les concepts de la confiance directe et la confiance indirecte qui ont été établis au cours des deux phases précédentes seront utilisés dans cette étape pour soutenir toutes les activités. Pour le routage basé sur la confiance, des métriques de routage liées à la confiance seront formées (par exemple, combinées avec d'autres métriques) et utilisées comme critères pour sélectionner les chemins les plus fiables entre n'importe quelle paire "$source - destination$".

2.3.2 Le processus des méthodes d'évaluation

Selon les méthodes d'évaluation mentionnées ci-dessus (chapitre 4 : section 3 et 4), y compris les facteurs qualitatifs et les facteurs quantitatifs, la quantité d'informations peut être calculée en fonction de divers facteurs. Après avoir calculé l'information de divers facteurs, la quantité totale d'informations peut être déterminée en utilisant la méthode de la théorie floue.

Dans notre modèle, le processus d'évaluation des facteurs est basé sur la logique floue et la méthode *G.R.A*. C'est-à-dire, juger si les facteurs d'évaluation sont quantitatifs ou qualitatifs. Si les facteurs sont qualitatifs, alors déterminez les valeurs linguistiques et leurs fonctions d'appartenance correspondantes, sinon, si les facteurs sont quantitatifs, alors calculez le montant de l'information en utilisant le processus *G.R.A*.

2.3.3 *Utilisation du processus G.R.A*

G.R.A est une méthode utile pour estimer l'importance des facteurs dans un système avec des données d'échantillonnage limitées. *G.R.A* a été proposé à l'origine pour relier le facteur principal avec d'autres facteurs de référence dans un système donné. Pour cette raison, nous avons appliqué cette technique pour déterminer les variables d'entrée qui montrent un impact décisif sur la sortie. En d'autres termes, la sortie du système peut saisir des informations utiles sur la variété des points de données de la séquence d'entrée.

2.3.4 *Évaluation de la confiance en utilisant les ensembles flous*

La confiance d'un nœud peut être interprétée comme l'évaluation progressive et dynamique par un nœud sur d'autres nœuds dans le processus d'interactions continues. Cette évaluation fournit des indications sur les comportements du nœud de routage. C'est pourquoi le facteur de confiance dans notre schéma est un facteur qualitatif, décrit par des expressions floues et leurs fonctions d'appartenance.

Pour évaluer la confiance d'une manière compréhensible et claire, nous pouvons utiliser les classes de grappes grises avec une fonction de pondération de blanchiment (*a whitenization weight function*). La fonction de pondération de blanchiment peut être utilisée pour mesurer la valeur d'utilité du revenu attendu [120]. Cela signifie que les fonctions de pondération de blanchiment peuvent décrire les poids d'une valeur dans différentes grappes, ce qui peut être considéré comme le degré d'appartenance de la valeur à une grappe.

2.3.5 *Évaluation de la confiance en utilisant la méthode d'agrégation grise*

Selon [126], l'analyse des grappes grises ($Grey\ clustering\ analysis$) est un processus d'analyse qui consiste à diviser les indicateurs d'observation (facteurs) ou les objets contrôlés en plusieurs catégories définissables (spécifiques). Ce processus est basé sur la matrice des relations grises ou la fonction de blanchiment du nombre gris, afin de vérifier si les objets observés ont été traités différemment ou non. L'analyse des grappes grises a essentiellement servi pour fusionner des facteurs similaires pour simplifier les systèmes complexes.

Définition 5.1 : Supposons qu'il existe "n" objets à regrouper, "m" critères de regroupement et "s" grappes grises différentes. Selon l'échantillonnage x_{ij} $(i = 1, 2, \dots, n;\ j = 1, 2, \dots, m)$ de l'objet "i" en ce qui concerne le critère "j", nous appelons "$Grey\ Clustering$", le classement de l'objet "i" en grappe grise "k" $(k \in \{1, 2, \dots, s\})$.

Dans le modèle de confiance proposé, chaque indicateur de confiance est un objet à regrouper. Les critères de classification sont des critères d'observation utilisés pour juger de la façon dont les

indicateurs de confiance sont contrôlés. Les grappes grises indiquent les catégories grises sélectionnées en fonction de leur degré de confiance. En théorie grise, la fonction de pondération de blanchiment est fréquemment utilisée pour décrire l'étendue de la préférence, lorsqu'un élément gris prend des valeurs différentes dans son champ de valeur, comme le montre la figure 5.6 (a, b, c ou d).

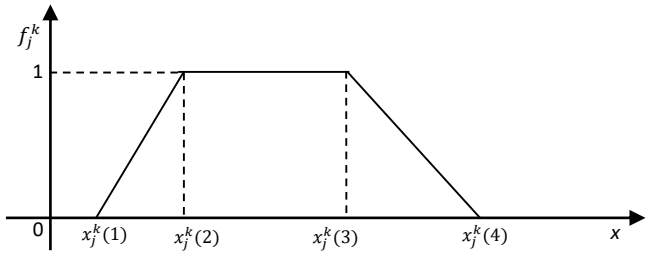


Figure 5.6 (a) : Fonction de pondération de blanchiment typique.

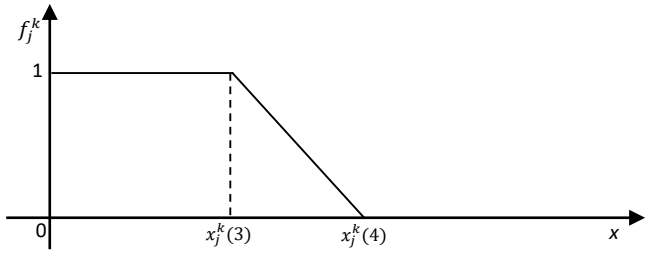


Figure 5.6 (b) : Fonction de pondération de blanchiment de mesure inferieure.

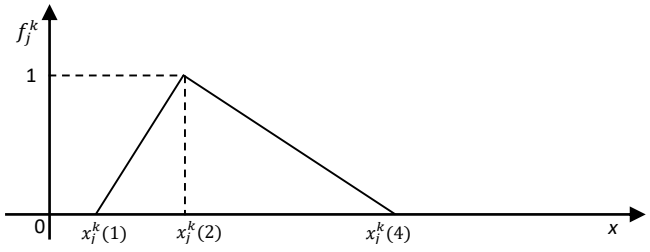


Figure 5.6 (c) : Fonction de pondération de blanchiment de mesure modérée.

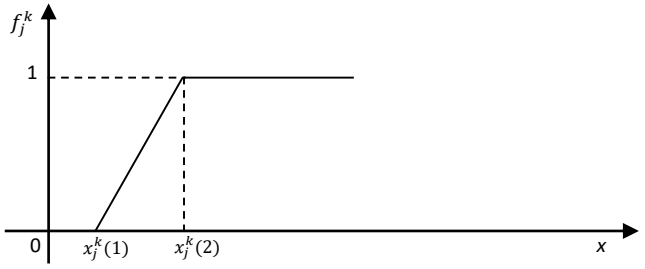


Figure 5.6 (d) : Fonction de pondération de blanchiment de mesure supérieure.

La fonction de pondération de blanchiment est représentée par $f_j^k(*)$. $Si\ f_j^k(*)$ est telle que modelée dans la figure 5.6 (a, b, c ou d), $alors$ elle est représentée comme suit :
$f_j^k[x_j^k(1), x_j^k(2), x_j^k(3), x_j^k(4)]$, $f_j^k[-, -, x_j^k(3), x_j^k(4)]$, $f_j^k\left[x_j^k(1), x_j^k(2), -, x_j^k(4)\right]$ ou $f_j^k[x_j^k(1), x_j^k(2), -, -]$.

La fonction de pondération de blanchiment *typique* telle que représentée dans la figure 5.6 (a), est donnée par l'équation 5.8.

$$f_j^k(x) = \begin{cases} 0 & x \notin [x_j^k(1), x_j^k(4)] \\ \frac{x - x_j^k(1)}{x_j^k(2) - x_j^k(1)} & x \in \left[x_j^k(1), x_j^k(2)\right] \\ 1 & x \in \left[x_j^k(2), x_j^k(3)\right] \\ \frac{x_j^k(4) - x}{x_j^k(4) - x_j^k(3)} & x \in \left[x_j^k(3), x_j^k(4)\right] \end{cases} \qquad (5.8)$$

La fonction de pondération de blanchiment de *mesure inférieure* telle que représentée dans la figure 5.6 (b), est donnée par l'équation 5.9.

$$f_j^k(x) = \begin{cases} 0 & x \notin \left[0, x_j^k(4)\right] \\ 1 & x \in \left[0, x_j^k(3)\right] \\ \frac{x_j^k(4) - x}{x_j^k(4) - x_j^k(3)} & x \in \left[x_j^k(3), x_j^k(4)\right] \end{cases} \qquad (5.9)$$

La fonction de pondération de blanchiment de *mesure modérée* telle que représentée dans la figure 5.6 (c), est donnée par l'équation 5.10.

$$f_j^k(x) = \begin{cases} 0 & x \notin \left[x_j^k(1), x_j^k(4)\right] \\ \frac{x - x_j^k(1)}{x_j^k(2) - x_j^k(1)} & x \in \left[x_j^k(1), x_j^k(2)\right] \\ \frac{x_j^k(4) - x}{x_j^k(4) - x_j^k(2)} & x \in \left[x_j^k(2), x_j^k(4)\right] \end{cases} \qquad (5.10)$$

La fonction de pondération de blanchiment de *mesure supérieure* telle que représentée dans la figure 5.6 (d), est donnée par l'équation 5.11.

$$f_j^k(x) = \begin{cases} 0 & x < x_j^k(1) \\ \frac{x - x_j^k(1)}{x_j^k(2) - x_j^k(1)} & x \in \left[x_j^k(1), x_j^k(2)\right] \\ 1 & x \geq x_j^k(2) \end{cases} \qquad (5.11)$$

Définition 5.2 : Les classes grises "s" d'une division de valeurs d'un index "j" sont appelées les sous-classes "$j-index$".

Définition 5.3 : Supposons que $X=(x_{ij})_{n*m}$ est la valeur d'observation de l'objet "i" par rapport à l'indice "j" et $f_j^k(*)$ est la fonction de pondération de blanchiment des sous-classes "k". x_j^k est déterminé par les normes de classement de l'indice de grappe "j", c'est-à-dire la valeur standard de la fonction de blanchiment $f_j^k(*)$. w_j est le poids de regroupement de l'indice "j" qui est obtenu à partir du calcul du "$poids\ de\ l'entropie$[3]", le coefficient de regroupement est obtenu selon l'équation 5.12.

$$\begin{Bmatrix} \sigma_i^k=\sum_{j=1}^{m} f_j^k\left(x_{ij}\right)*w_j \\ (i=1,2,\dots,n) \\ (j=1,2,\dots,m) \\ (k=1,2,\dots,s) \end{Bmatrix} \qquad (5.12)$$

Dans notre contribution, nous définissons trois grappes grises $Cluster_1$, $Cluster_2$ et $Cluster_3$ comme suit :

$Cluster_1 \rightarrow$ Mauvaise confiance
$Cluster_2 \rightarrow$ Confiance générique
$Cluster_3 \rightarrow$ Bonne confiance

Les fonctions de pondération de blanchiment devraient être comme :

"*Mauvaise confiance*" $\rightarrow f_j^k[-,-,x_j^k(3),x_j^k(4)]$.
"*Confiance générique*" $\rightarrow f_j^k[x_j^k(1),x_j^k(2),-,x_j^k(4)]$.
"*Bonne confiance*" $\rightarrow f_j^k[x_j^k(1),x_j^k(2),-,-]$.

2.3.6 Synthèse globale de la valeur de confiance

Grâce à l'observation et la recommandation des nœuds voisins (comme il a été expliqué dans la figure 5.2), le nœud évaluateur "i" calcule $DT(t)$, $RT(t)$ et $IT(t)$. Supposons que le nœud "i" reçoit diverses valeurs de confiance du nœud "j" {nœud évalué} via différents nœuds voisins "k", le poids de blanchiment d'une valeur de confiance T_{jk} appartenant à la classe "s" est $f_s(T_{jk})$, $s \in \{1,2,3\}$.

[3] Le terme entropie a été introduit en 1865 par Rudolf Clausius à partir d'un mot grec signifiant « transformation ». Il caractérise le degré de désorganisation, ou d'imprédictibilité du contenu en information d'un système.

Pour calculer la valeur de confiance globale dans notre modèle, l'algorithme 5.1 montre le processus d'évaluation de la classe de grappes grises du nœud "j". Par conséquent, nous pouvons donner la valeur de confiance globale du nœud "j" pour le nœud "i" comme suit :

$$T_{Global\,(v_i,v_j)}(t) = w_1 * \max_s f_s(DT_{ij}) * DT_{ij} + w_2 * \max_s f_s(RT_{jk}) * RT_{jk} + w_3 * \max_s f_s(IT_{ij}) * IT_{ij} \quad (5.13)$$

Où :

- $\{0 \leq T_{Global}(i,j) \leq 1\}$.
- w_1, w_2 et w_3 présentent respectivement les valeurs pondérées de la confiance directe, la recommandation et la confiance indirecte, telles que $(w_1 + w_2 + w_3 = 1)$. Les valeurs de w_1, w_2 et w_3 sont déterminées par l'influence sur l'évaluation de la valeur de confiance dans un environnement différent.

Algorithme 1 : Le processus d'évaluation de la classe de grappes grises du nœud "j"

1. $If\ (0 < \max_s f_s(T_{j,\ evaluating\ node}) < \frac{(\sigma_1+\sigma_2)}{3})\ then$
2. $The\ associated\ grey\ cluster\ class\ is\ 'Cluster_1'$
3. $Else$
4. $If\ \frac{(\sigma_1+\sigma_2)}{3} \leq \max_s f_s(T_{j,\ evaluating\ node}) \leq \frac{(\sigma_2+\sigma_3)}{3}\ then$
5. $The\ associated\ grey\ cluster\ class\ is\ 'Cluster2'$
6. $Else$
7. $If\ \frac{(\sigma_2+\sigma_3)}{3} < \max_s f_s(T_{j,\ evaluating\ node}) < 1\ then$
8. $The\ associated\ grey\ cluster\ class\ is\ 'Cluster3'$
9. $Else$
10. $Default$
11. $End\ if$
12. $End\ if$
13. $End\ if$

2.3.7 Analyse de l'efficacité du modèle proposé

Les principales étapes de la méthode proposée sont résumées comme suit :

✓ Les résultats qualitatifs pour les attributs évalués peuvent être obtenus en utilisant le mécanisme de la logique floue.

- ✓ Le demandeur d'un service dans notre modèle de confiance construit des matrices comparatives basées sur le processus *G.R.A* selon le modèle hiérarchique présenté dans la figure 5.2.
- ✓ L'évaluation de la valeur globale de la confiance est basée sur l'équation (5.13).

La complexité du temps est évaluée pour "n" paramètres, nous supposons qu'il y a "m" facteurs principalement pour chaque paramètre. L'inférence floue nécessite un temps *O(m)* et la combinaison de "m" facteurs coûte aussi $O(m)$. La complexité globale du temps de la quantification des paramètres est : $n * (2 * O(m))$.

Dans l'étape de fusion des attributs, la construction et la vérification d'une matrice comparative nécessite le temps t', le nombre de matrices est $(k + 1)$, où "k" est le nombre d'attributs intermédiaires.

Établir des priorités pour chaque couche (layer) a besoin de temps $O(K)$, le temps de calcul du vecteur pondéré est $(n * k)$ et la complexité du temps global du calcul du poids nécessite :

$$((k + 1)(t' + O(k)) + n * k) \qquad (5.14)$$

Calculer la valeur globale d'un modèle de confiance nécessite $O(n)$, alors la complexité du temps de la méthode proposée est :

$$\big(n * 2 * O(m) + (k\ +\ 1)\big(\ t' + O(k)\big) + n * k\big) + O(n) \quad (5.15)$$

En plus de cela, on peut conclure que la complexité du temps est contrôlée et la complexité de l'espace est faible car aucune information supplémentaire n'a besoin d'être stockée, à l'exception des données utilisées pour calculer la valeur globale évaluée de notre modèle de confiance.

3. CONCLUSION

En raison de contraintes causées par la mise en œuvre de la sécurité dans les réseaux mobiles Ad hoc, la sécurité et la confiance doivent être examinées ensemble, où la confiance est l'aspect le plus important de toute communication sécurisée dans ces réseaux.

À travers ce chapitre, nous avons proposé un nouveau modèle de gestion de la confiance pour les réseaux mobiles Ad hoc. Le nouveau modèle utilise plusieurs paramètres et facteurs de décision pour calculer la valeur globale de confiance d'un nœud. Notre approche utilise également la théorie grise "$Grey\ theory$" et les ensembles flous afin d'améliorer les algorithmes de génération de la valeur de confiance, et gérer la méthode de prédiction des règles de la logique floue pour mettre à jour la confiance d'un nœud, ce qui rend notre modèle plus stable, adaptatif et robuste. Ceci fournit un nouvel avantage significatif pour le modèle proposé, car il peut détecter non seulement un comportement égoïste ou

anormal, mais peut également aider à identifier le type de paramètres utilisés dans la stratégie de l'attaquant ou du nœud égoïste.

L'approche proposée ne se proclame pas d'être immunisée contre tout genre d'attaques malicieuses, mais donne certainement une nouvelle direction aux recherches de sécurité dans les réseaux mobiles Ad hoc.

Chapitre 6

Le protocole de confiance proposé

1. INTRODUCTION

Ce chapitre est consacré à la présentation de notre nouveau protocole de sécurité, nommé $Favorite - AODV$ [127], dédié pour la gestion de la confiance ainsi que la sécurisation des communications dans le réseau. La première partie de ce chapitre consacrée à la présentation et la description du fonctionnement de notre protocole ainsi que les différents algorithmes de routage correspondants, l'idée principale est de fournir un protocole capable d'offrir un niveau de sécurité adapté à l'enjeu de la communication dans un environnement hostile, et dont le niveau pourra évoluer dans le temps en fonction du contexte. Ensuite, nous présentons la gestion d'attaques et les étapes d'implémentation des contremesures du comportement malveillant.

2. PROTOCOLE DE ROUTAGE SÉCURISÉ

2.1 Description du protocole

Notre protocole de routage de confiance est conçu selon les hypothèses mentionnées dans la section 5 du chapitre 4. La mise en œuvre de la couche réseau est notre objectif principal et le protocole que nous proposons ici est une extension du protocole AODV[1] [37], car le modèle proposé est inspiré des régimes à la demande.

Dans cette section, nous présentons une description détaillée de notre protocole que nous appelons "Favorite-AODV" qui est basé sur deux paramètres essentiels qui sont : *la valeur de confiance globale* (T_{Global}) et *le nombre de sauts*, afin d'établir une route de confiance libre de toute entité malveillante avec le chemin le plus court.

Les principales propriétés de notre protocole sont :

- ✓ En plus de la table de routage requise dans le protocole AODV standard, nous ajoutons notre modèle d'évaluation de la confiance en incluant $T_{Global(i,j)}$ dans la table de routage de chaque nœud.
- ✓ Toutes les valeurs de confiance sont obtenues à partir du notre modèle d'évaluation de la confiance.
- ✓ Les paquets contenant les valeurs de confiance sont sauvegardés et protégés contre toute modification par des nœuds malveillants.

[1] Ad hoc On demand Distance Victor.

- ✓ Si une valeur de confiance d'un nœud est évaluée comme étant très faible par tous ses voisins, ce nœud est considéré comme un nœud malveillant, ensuite toute réponse aux demandes de routage via ce nœud est rejetée et toute demande initiée est ignorée.
- ✓ Chaque nœud conserve une liste noire locale, un nœud dans une liste noire est exclu par ses voisins. Il ne recevra ni les paquets de son voisin, ni ses paquets transmis.
- ✓ Pendant le processus de découverte de l'itinéraire (Route discovery), il est important de sélectionner un nœud de saut suivant sécurisé, afin d'éviter de transmettre un paquet à un nœud malveillant (en utilisant l'algorithme 5.1).
- ✓ Nous évaluons la fiabilité d'un itinéraire par la valeur de confiance des nœuds le long du chemin. Lorsque les messages $RREQs$ sont diffusés, chaque nœud intermédiaire qui n'a pas de route directe vers la destination, transfère le paquet $RREQ$ après avoir ajouté sa valeur de confiance à la pile de confiance dans le paquet, qui est désignée par "$Trust - stack\_Path$". "Équation (6.1)"

$$\text{Trust-stack}_{\text{Path}}(t) = \frac{\sum T_{Global\,(v_i, v_k)}(t)}{\text{Hop\_count}} \qquad (6.1)$$

$$v_i \in Path,\ v_k \in Path, v_i \rightarrow v_k \text{ et } (v_k \neq v_{destination})$$

Où, v_i et v_k sont deux nœuds voisins quelconques parmi la route, $v_i \rightarrow v_k$ signifie que v_k est le prochain nœud de v_i. "Hop_count" indique le nombre de sauts inclus dans le message de requête.

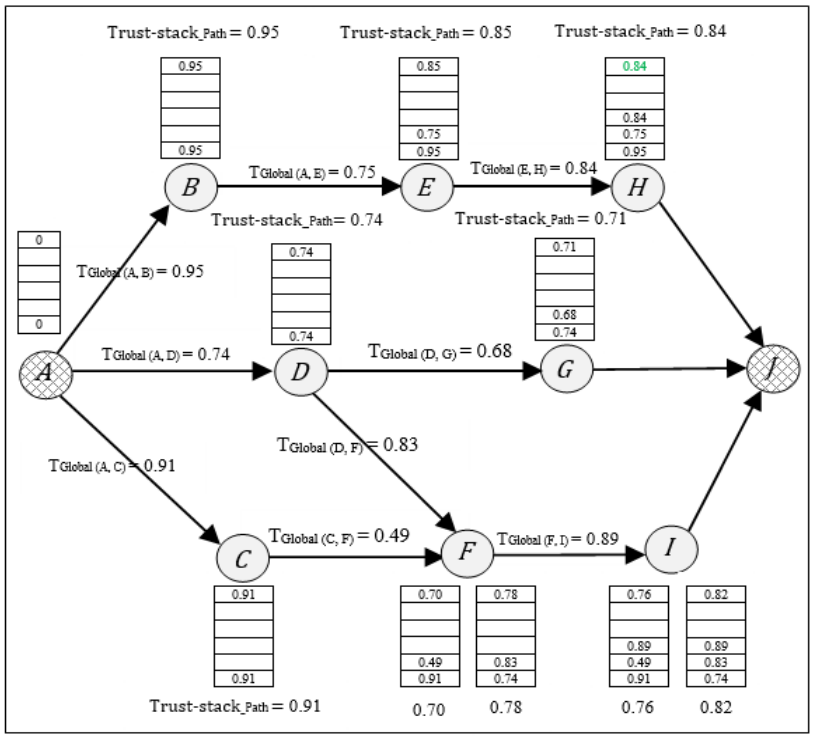


Figure 6.1 : Calcul de la valeur de confiance des chemins.

Comme le montre la figure 5.6, les valeurs de la pile de confiance de chaque chemin (du nœud source "A" au nœud destination "J") sont présentées par le tableau suivant (Tableau 6.1). Parmi les quatre chemins possibles du nœud "A" au nœud "J", le chemin {A→ B→ E→ H→ J} est le chemin le plus confiant.

Table 6.1 : Exemple illustratif de calcul de la pile de confiance.

Trust-Stack	B	C	D	E	F	G	H	I
A	0.95	0.91	0.74	0.85	0.70 / 0.78	0.71	0.84	0.76 / 0.82
B				0.75			0.79	
C					0.49			0.69
D					0.83	0.68		0.86
E							0.84	
F								0.89

Dans notre protocole de routage de confiance "Favorite-AODV", le processus de configuration du routage comprend essentiellement la découverte et la maintenance du routage. "Favorite-AODV" est principalement utilisé pour le processus de découverte, et aucun changement majeur ne devrait avoir été fait pour maintenir l'AODV.

2.2 *Structures étendues par chaque nœud pour "Favorite-AODV"*

2.2.1 *Table de routage favorite*

La gestion d'une table de routage s'impose puisqu'il s'agit d'un protocole de routage. Les informations sur les routes doivent être conservées même pour les liaisons de courtes durées. La structure de cette table dans notre protocole est appelée "$Table\ de\ routage\ favorite$" qui est présentée dans la figure 6.2.

$Des_{\_Addr}$	$Des_{\_Seq\#}$	$Valid_{\_Seq\#}$	$State$	$Interface$	$Hop_{\_count\_Des}$	$Next\_H_{\_count}$	P_List	$Lifetime$	$Trust\_stack_{\_Path}$	$Trusted_{\_Path}$

Figure 6.2 : Table de routage favorite.

Où :

- $Des_{\_Addr}$: L'adresse IP de la destination.
- $Des_{\_Seq\#}$[2] : Numéro de séquence de la destination qui permet d'éviter des boucles de routage.

[2] Sequence Number of destination.

- $Valid\ _{Seq\#}$: Drapeau indiquant la validité du numéro de séquence.
- $State$: Drapeau indiquant l'état de l'entrée (Valide, Invalide, réparable, étant réparé).
- $Interface$: Interface réseau.
- $Hop_{\_count\_Des}$: Est la distance entre le nœud local et la destination mesurée en nombre de sauts.
- $Next\_H_{\_count}$: Prochain saut en direction de la destination.
- P_List[3] : C'est la liste des voisins auxquels une réponse de route est générée ou transférée.
- $Lifetime$: Temps au bout duquel l'entrée est expirée et doit être supprimée.
- $\text{Trust\_stack}_{\text{Path(t)}}$: Indique la valeur de la pile de confiance d'une route à un moment donné "t". (Équation 6.6).
- $Trusted_{\_Path(t)}$: Indique la route la plus fiable et la plus confidente entre la source et le nœud local à un moment donnée "t". (Équation 6.7).

2.2.2 *Table d'Historique Favorite 'THF'*

Pour diminuer le nombre de messages qui circulent dans le réseau, "AODV" ne traite qu'une seule fois un message de demande de route (RREQ). Ainsi, il garde trace des demandes de route déjà traitées en les stockant dans une structure appelée table d'historique. Donc, pour notre protocole nous proposons d'étendre cette table afin d'enrichir la connaissance et détecter les comportements malhonnêtes.

➢ Pour les demandes de route (RREQ), les champs identifiant "$id\#$", $Time_{\_Del}$ et adresse de la source "$Source_{\_Addr}$" sont seulement enregistrés. Nous ajoutons les champs suivants :

- $id\#$: Identifiant de la demande de route RREQ.
- $Source_{\_Addr}$: Adresse de la source.
- $Time_{\_Del}$: Temps au-delà duquel l'entrée sera effacée.
- $Source_{\_Seq\#}$: Numéro de séquence de la source qui peut être modifié par un nœud malhonnête qui projette de s'insérer sur la route entre une source et une destination [57].
- $Hop_{\_count\_Src}$: Nombre de sauts pour atteindre la source "$Source\_Addr$" qui, de la même façon que "$Source_{\_Seq\#}$", peut être modifié par un nœud malhonnête pour s'insérer sur une route ou pour retarder la découverte de route.

[3] Precursor List.

- Src_{Ip} : Adresse IP du nœud qui a envoyé la RREQ et qui peut être différent de l'adresse du nœud ayant initié la demande de route "$Source_{Addr}$". Cette information servira à traquer les nœuds rejouant des messages.

➢ Pour les réponses de route (RREP), AODV ne prévoyait pas d'enregistrer les informations les concernant. Pour cela, nous gardons trace de ces messages dans la table d'historique et nous stockons les champs suivants :

- Des_{Addr} : Adresse de la destination de la RREQ dont cette RREP est une réponse.
- $Source_{Addr}$: Adresse de la source de la RREQ dont cette RREP est une réponse.
- $Des_{Seq\#}$: Numéro de séquence de la destination qui peut être modifié par un nœud malhonnête pour forcer un nœud honnête à le choisir dans le chemin vers la destination.
- Hop_{count_Des} : Nombre de sauts pour atteindre la destination "Des_{Addr}" qui peut être modifié par un nœud malhonnête pour faire croire qu'il a le chemin le plus court.
- Des_{Ip} : l'adresse IP de la destination d'un message RREP. Cette adresse combinée à l'adresse de la source Src_{Ip} permettent de trouver les nœuds rejouant des messages RREP.

2.3 *Structures des messages échangés*

2.3.1 *Demande de route RREQ (Route REQuest)*

Ce message est diffusé lorsqu'un nœud détermine qu'il a besoin d'une route vers une destination et ne dispose pas d'une route disponible. C'est le cas lorsque la destination est inconnue ou lorsqu'une route précédemment valide dans sa table de routage expire ou est marquée invalide. Le nœud créé le paquet présenté dans la figure 6.3.

1 2 3 4 5 6 7 8	9 10 11 12	13 14 15 16 17 18 19 20 21 22 23 24	25 26 27 28 29 30 31
Type =1	J R G D U	Reserved	Hop_{count}
Broadcast Id# (RREQ Identifier)			
Des_{Addr}			
$Des_{Seq\#}$			
$Source_{Addr}$			
$Source_{Seq\#}$			
$Trust\text{-}stack_{Path(t)}$			
$Trusted_{Path(t)}$			

Figure 6.3 : Format de la RREQ dans Favorite-AODV.

Où :

- Type : 1.

- J[4] : réservé au multicast.
- R[5] : réservé au multicast.
- G[6] : indique la nécessité de générer une réponse de route vers la destination en plus de celle générée vers la source pour l'informer qu'une telle source cherche à la joindre et ainsi un chemin bidirectionnel est établi. Une RREP de ce genre (*gratuitous*) est généré seulement lorsqu'il s'agit d'un nœud intermédiaire qui répond.
- D[7] : indique que seulement la destination doit répondre à cette demande de route.
- U[8] : indique que le numéro de séquence de la destination est inconnu.
- Reserved : mis à zéro lors de l'envoi et ignoré à la réception.

Quand un nœud source "S" veut envoyer de manière sécurisée un message vers un nœud de destination "D" pour lequel il a encore un chemin, il initie une découverte de route (Figure 6.4).

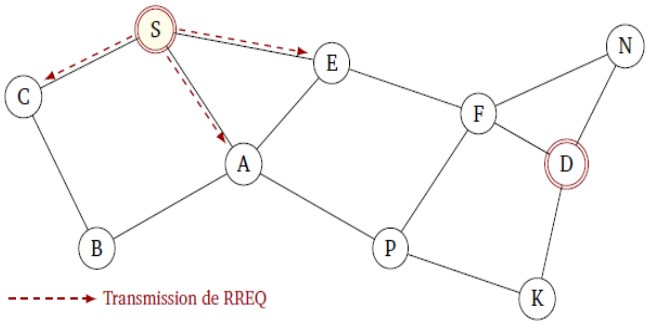


Figure 6.4 : Initialisation de la demande de route par S.

Le nœud source "S" diffuse un message de découverte de route $RREQ$ à ses voisins contenant :

"S" diffuse :

$$RREQ :< Source\_Addr, Source\_Seq\#, Broadcast\ id\#, Des\_Addr,\ Des\_Seq\#,\ Hop\_count,\ \text{Trust} - \text{stack}_{\text{Path(t)}},\ Trusted\_Path(t) >.$$

Lorsqu'un nœud reçoit le $RREQ$, il vérifie s'il s'agit de la destination ou non, si c'est le cas, il doit attendre un certain temps, car le nœud de destination peut recevoir de nombreux paquets différents "$RREQs$" de la source. Ensuite, il met à jour la table de routage et génère une réponse d'itinéraire ($RREP$), mais si le nœud récepteur est un nœud intermédiaire, il vérifie s'il a reçu ou non un même

[4] Join flag.
[5] Repair flag.
[6] Gratuitous RREP flag.
[7] Destination only flag.
[8] Unknown Sequence Number.

$RREQ$. Si oui, ce dernier est rejeté. Sinon[9], le nœud insère la valeur de confiance (T_{Global}) de sa table de routage à la pile de confiance "Trust − stack$_{\text{Path(t)}}$" dans le message du saut précédent, puis le compare au champ de la pile de confiance du $RREQ$ nouvellement reçu, et augmente "$Hop_{\_count}$" par "1" dans le message $RREQ$.

Une fois le message reçu, la destination peut déterminer le chemin de confiance. Un nœud vérifie le chemin de confiance dans sa table de routage avec la nouvelle valeur de chemin de confiance spécifiée dans le message, si le nouveau chemin de confiance est plus fiable que celui de la table de routage, le nœud met à jour sa table de routage. Ensuite, le nœud de destination envoie un paquet de réponse d'itinéraire ($RREP$) le long de la meilleure route qui a un maximum de Trust-stack$_{\text{Path}}$.

Dans cette section, il est important de savoir que "$Hop_{\_count}$" joue un rôle dans le choix de la route finale, lorsqu'un nœud reçoit un message $RREQ$, il rétablit un chemin inverse vers la source en enregistrant le voisin ($id\#$) à partir duquel il a reçu le $RREQ$. Ainsi, à tout moment, le paquet $RREQ$ contient une liste de tous les nœuds visités avec leur valeur ajoutée à $Trust - stack$. Chaque fois qu'un nœud reçoit un paquet $RREQ$, il doit vérifier les mises à jour de l'itinéraire vers le nœud source. Il vérifie également le chemin de confiance pour les nœuds intermédiaires, cela est calculé par l'Equation (6.2).

$$\text{Trusted}_{\text{Path}}(t) = \max_{i} \ (Trust - Stack_{\_Path(i)}) \qquad (6.2)$$

Le pseudo-code du $RREQ$ est montré dans l'algorithme 6.1.

Algorithm 6.1: The RREQ delivery algorithm

To source node:

1. {*When a node receives a Route Request Packet*}
2. *Receive_RREQ* ( );
3. *Check whether it is the destination of the route request*;
4. *If destination Then*
5. *Waits for a specified period* ;

   Compute trust − stack_path between 'S' and 'D';

   Update the routing table for this node;

   In the case where more than one RREQ has same trust − stack_Path;

[9] Le cas où le $RREQ$ n'est pas rejeté.

Selects on the basis of lowest hop − count;

Calculate the trusted_path;

Calls the process of Route Reply;

'D' sends RREP to 'S' with trusted_path;

6. *Else {not destination}*
7. *Checks whether one copy of the same RREQ has been received;*
8. *If duplicate packet Then*
9. *Discards RREQ and the procedure ends;*
10. *Else {not duplicate}*
11. *Calculates the Trust − stack_Path of previous hop;*
12. *Updates the Trust − stack_Path in the message;*
13. *Increments Hop_count by one;*
14. *Calculates the trusted_path;*
15. *Broadcasts the RREQ to all its neighbours;*
16. *End If*
17. *End If*
18. *{End of function receive RREQ}.*

2.3.2 *Réponse de route RREP (Route REPly)*

Lorsqu'une demande de route atteint la destination ou un nœud ayant un chemin valide vers la destination, celui-ci génère une réponse de route qui sera envoyée en unicast d'un nœud à un autre jusqu'à atteindre la source.

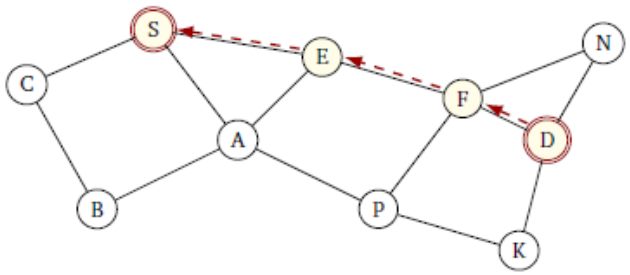


Figure 6.5 : Propagation de la réponse de route RREP.

Le paquet de réponse de route est représenté par la figure 6.6 :

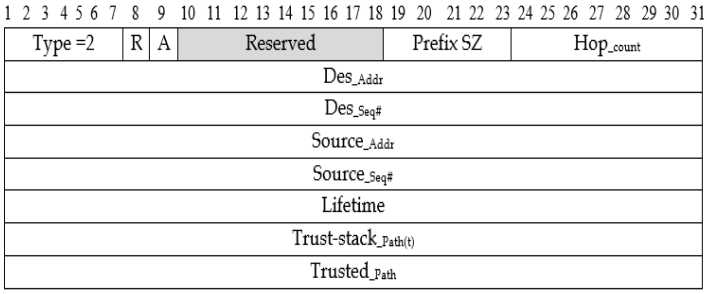


Figure 6.6 : Format de la RREP dans Favorite-AODV.

- Type : 2.
- R : Repair flag, utilisé en multicast.
- A : Accusé de réception requis.
- Reserved : mis à zéro lors de l'envoi et ignorés à la réception.
- Prefix SZ : Si c'est différent de zéro, cela signifie que le prochain saut peut être utilisé pour n'importe quel nœud avec le même préfixe.

Un paquet $RREP$ contient les informations suivantes :

"D" diffuse :

$$RREP : < Dest\_Addr, Dest\_Seq\#, Source\_Addr, Source\_Seq\#, Hop\_count, Lifetime, \text{Trust} - \text{stack}_{\text{Path(t)}}, Trusted\_Path >.$$

Lorsque le nœud source "S" récupère le premier $RREP$, il traite la requête selon la procédure suivante (Algorithme 6.2).

Algorithm 6.2: The RREP delivery algorithm

To destination node:

1. {*When the source node gets back the first RREP*}
2. *Receives_RREP* ( );
3. *Waits for a specified period*;
4. *If receives alternative RREP Then*
5. *Inquiries next hop* ( );
6. *Else*
7. *Sends the RREP with more trusted*$_{path}$ *along the path to the intermediate node*;

8. $End\ If$
9. $\{End\ of\ function\ receive\ RREP\}$.

3. IMPLÉMENTATION DES CONTREMESURES DU COMPORTEMENT MALHONNÊTE

Dans cette section, nous nous intéressons à la détection des comportements malveillants en se basant sur les modules cités dans notre modèle de confiance (section 2.2 du chapitre 5). Pour cela, nous proposons d'enrichir la connaissance de chaque nœud dans le réseau en stockant des informations additionnelles lui permettant de décider de l'honnêteté du voisin.

Ce raisonnement repose sur un ensemble de contrôles qui permettent de déceler les incohérences, signes caractéristiques d'actions malveillantes, après corrélation des informations reçues avec celles stockées. Nous présentons les contrôles que les nœuds doivent intégrer afin de vérifier le comportement des voisins pour détecter des nœuds impliqués dans notre protocole qui essaient de le contourner pour détourner le trafic en $fabriquant$, en $rejouant$, en $supprimant$ ou en $modifiant$ des messages [35].

3.1 Fabrication de messages

La réception d'une réponse de route ($'j' \xleftarrow{RREP_i} 'i'$) implique d'avoir précédemment traité une demande de route et avoir le chemin inverse vers la source. Donc, si un nœud reçoit une réponse de route qui lui est destinée et ne trouve pas d'entrée vers la source dans sa table de routage (RT), $RREP_i.Addr\_Src \notin RT_j$, il déduit que le nœud duquel le message est initié (i) l'a fabriqué. Il ne lui fait plus confiance. L'expression (6.3) présente le critère de contrôle détectant ce genre de fabrication de messages.

$Critère\ 1$:

$$Si \begin{cases} 'j' \xleftarrow{RREP_i} 'i' \\ et \\ RREP_i.Addr\_Src \notin table\ de\ routage_j \end{cases} \quad Alors \quad mettre\ "i"\ dans\ la\ liste\ noire \qquad (6.3)$$

3.2 Rejeu de messages

Le rejeu de messages consiste en la retransmission d'anciens messages par un nœud malveillant. Comme nous l'avons indiqué dans la section (5.1.2 du chapitre 2), il peut être utilisé pour le

détournement du trafic ou la consommation des ressources (batterie, bande passante, etc.). Un ancien message est un paquet de routage reçu depuis un certain temps et qui n'est plus d'actualité puisqu'il a été traité et que d'autres messages plus récents sont reçus depuis. Nous traitons le rejeu d'une demande de route "$RREQ$" et d'une réponse de route "$RREP$".

3.2.1 Rejeu d'une demande de route (RREQ)

Un nœud malveillant peut rejouer des demandes d'itinéraire sans que le voisinage ne s'aperçoit de ce comportement malveillant. Les nœuds cibles (le voisinage) continuent à refaire le traitement prévu pour chaque demande de route reçue puisqu'ils font a priori confiance à cet émetteur pour se comporter normalement. Cependant, l'action de l'attaquant entraîne une consommation de la bande passante par des messages inutiles et un traitement supplémentaire au niveau des nœuds, ce qui dégrade la qualité de services (QoS).

L'action de l'attaquant dans ce genre d'attaques repose sur la retransmission d'anciennes demandes de route. Sachant qu'une demande itinéraire est uniquement identifiée grâce à son identifiant "$Id\#$" et l'adresse de la source "$Source_{\_Addr}$" et qu'on peut facilement obtenir l'adresse de la source de chaque demande de route, nous pouvons introduire un contrôle qui vérifie que cette $RREQ$ a été transmise auparavant par ce nœud ou non.

En effet, en conservant dans la table d'historique favorite (THF) l'adresse IP de la source de la RREQ ($Src_{\_Ip}$) pour chaque demande de route reçue, nous pouvons mettre en place ce contrôle dans l'expression (6.4) :

$Critère\ 2$:

$$Si \begin{cases} 'j' \xleftarrow{RREQ_i} 'i'\ est\ \ déjà\ traitée\ dans\ THF_j. \\ \qquad\qquad et \\ THF_j.RREQ_i.Src_{\_Ip} = RREQ_i.Src_{\_Ip} \end{cases} \quad Alors\ \ mettre\ "i"\ dans\ la\ liste\ noire \quad (6.4)$$

Autrement dit, le nœud émettant la RREQ est considéré comme malveillant puisqu'il vient de rejouer un message et ne devrait plus être considéré comme nœud de confiance.

3.2.2 Rejeu d'une réponse de route (RREP)

Un nœud malveillant peut rejouer des réponses de route impliquant la répétition du traitement au niveau du nœud cible. Toutefois, en utilisant les informations de la table d'historique favorite, à la réception d'une réponse de route ($'j' \xleftarrow{RREP_i} 'i'$), un nœud dispose d'une connaissance suffisante pour vérifier s'il l'a déjà traitée. Il recherche ainsi les réponses de routes similaires reçues

précédemment et vérifie pour chaque occurrence l'adresse IP de la source et de la destination par rapport à celles du paquet en cours de traitement, L'expression (6.5) présente ce critère de contrôle.

$Critère\ 3$:

$$Si \begin{cases} 'j' \xleftarrow{RREP_i} 'i'\ est\ déjà\ traitée\ dans\ THF_j. \\ et \\ THF_j.RREP_i.Src_{\_Ip} = i \\ et \\ THF_j.RREP_i.Des_{\_Ip} = RREP_i.Des_{\_Ip} \end{cases} \quad Alors\ \ mettre\ "i"\ dans\ la\ liste\ noire \quad (6.5)$$

3.3 Modification de messages

Un nœud malveillant peut modifier certains champs en relayant des messages de routage (RREQ et RREP). Pour cela, nous nous basons sur ce principe afin de découvrir les nœuds ayant changé un champ qui est supposé ne pas l'être.

3.3.1 Modification d'une demande de route (RREQ)

D'une manière générale, en relayant une demande de route, un nœud intermédiaire ne peut modifier que le nombre de sauts ($RREQ.Hop_{\_count}$) en l'incrémentant de 'un' et le numéro de séquence de la destination ($RREQ.Des_{\_Seq\#}$) en gardant la plus grande valeur entre celle reçue dans le paquet et celle trouvée dans la table de routage. Tout changement d'un autre champ peut être considéré comme frauduleux.

Afin de détecter ce type de comportement, nous conservons pour chaque demande de route le numéro de séquence de la source ($Source_{\_Seq\#}$) et le nombre de sauts ($Hop_{\_count}$) dans la table d'historique favorite. Ainsi, à la réception d'une demande de route qui a déjà été reçue (d'un autre nœud) est traitée.

$$'j' \xleftarrow{RREQ_i} 'i'\ est\ déjà\ traitée\ dans\ THF_j.$$

Le nœud $'j'$ compare le numéro de séquence de la source reçu dans la RREQ ($RREQ_i.Source_{\_Seq\#}$) avec celui trouvé dans sa table de routage, s'il les trouve différents, il en déduit que cette demande de route a été modifiée. L'expression (6.6) présente le critère de contrôle introduisant la méfiance qu'il faxut développer envers un nœud se comportant de la sorte.

$Critère\ 4$:

$$Si \begin{cases} 'j' \xleftarrow{RREQ_i} 'i' \ est\ déjà\ traitée\ dans\ THF_j. \\ et \\ RREQ_i.Source_{\_Seq\#} \neq (RT_j.(RREQ_i.Addr_{\_Src}).Source_{\_Seq\#}). \end{cases} \quad Alors\ \ mettre\ "i"\ dans\ la\ liste\ noire \quad (6.6)$$

3.3.2 Modification d'une réponse de route (RREP)

D'une façon semblable à une réponse de route, un nœud n'a le droit de modifier que le nombre de sauts en l'incrémentant de 'un' avant de la retransmettre. Ainsi, nous pouvons déduire que tout autre changement est équivalent de la malhonnêteté. En particulier le changement du numéro de séquence de la destination qui donne une indication sur la fraicheur d'une route et la diminution du nombre de sauts pour faire croire en un chemin plus court. Ces modifications introduites par un nœud malveillant provoquent une mise à jour de la table de routage du nœud recevant la RREP par de fausses informations.

L'action malveillante peut être détectée grâce à l'historique des messages enregistrés. Un nœud ayant envoyé une réponse de route et qui entend la retransmission du voisin peut ainsi contrôler ce que le voisin vient de faire sur le message ($'j' \xleftarrow{RREP_j} 'i'$ & $'j' \xleftarrow{RREP_i} 'x'$). Il recherche dans la table d'historique favorite la RREP qu'il a envoyée précédemment et vérifie si le numéro de séquence de la destination n'a pas été altéré (comparer celui reçu ($RREP_N.Des_{\_Seq\#}$) avec celui précédemment envoyé ($THF.RREP_N.Des_{\_Seq\#}$). L'expression (6.7) formalise ce cas et incite à se méfier de ces nœuds.

$Critère\ 5$:

$$Si \begin{cases} ('j' \xleftarrow{RREP_j} 'i' \ \&\ 'j' \xleftarrow{RREP_i} 'x')\ est\ déjà\ traitée\ dans\ THF_j. \\ et \\ RREP_N.Des_{\_Seq\#} \neq THF.RREP_N.Des_{\_Seq\#} \end{cases} \quad Alors\ \ mettre\ "i"\ dans\ la\ liste\ noire \quad (6.7)$$

Le raisonnement adopté dans cette implémentation est introduit au niveau de chaque nœud de telle sorte qu'il puisse porter un jugement sur le comportement des voisins et décider de sa validité.

Pour y parvenir, Nous avons présenté comment les nœuds malveillants opèrent pour aboutir à leur objectif. Cette analyse nous a permis de montrer que les actions malveillantes se basent essentiellement sur la combinaison d'actions élémentaires ($modification$, $rejeu$, $suppression$, $fabrication$) portant sur les messages de routage en exploitant la confiance que les nœuds se vouent.

4. CONCLUSION

Dans ce chapitre, un protocole de routage sécurisé a été présenté pour les réseaux mobiles ad hoc, le protocole proposé repose sur le protocole $AODV$, et repose sur la confiance que les nœuds partagent afin de mettre en place des mécanismes permettant de suivre l'exécution du protocole et de déceler d'éventuelles incohérences. Pour cela, nous avons opté pour la mise en place au niveau de chaque nœud d'un raisonnement sur le comportement des voisins en utilisant le recoupement d'informations fraîchement reçues avec celles reçues précédemment et ainsi détecter les incohérences. Afin de confirmer l'efficacité de notre protocole, nous devons analyser leur résistance aux attaques, aussi une évaluation selon des métriques de performance doit être faite. Le chapitre suivant sera consacré à l'évaluation des performances de nos contributions par simulations réalisées sous le simulateur *NS-2*[10].

[10] Network simulator (http://www.isi.edu/nsnam/ns/).

Chapitre 7

Évaluation des performances à travers la simulation

1. INTRODUCTION

Dans les chapitres précédents, nous avons illustré l'efficacité du raisonnement basé sur la confiance pour sécuriser les communications dans les MANETs, en utilisant notamment des approches adaptables pour ce raisonnement. Cependant, une simulation avec des réseaux à grande échelle est nécessaire pour prouver la capacité et la fiabilité de l'approche proposée.

Dans ce chapitre, nous nous intéressons à l'évaluation de notre proposition par l'intermédiaire de mesures de performance sur différents scénarios de mobilité, de mouvement et de trafic. Dans ce qui suit, nous présentons le choix de l'environnement de simulation (Section 2). Enfin, et après avoir décrit les différentes mesures à prendre en considération pour mettre en place des simulations, nous évaluons la performance du protocole proposé à travers les résultats obtenus de la simulation (Section 3) en montrant son efficacité dans l'amélioration de la qualité de l'interaction dans le réseau, notamment la QoS, l'atténuation des nœuds malveillants et donc l'amélioration de la sécurité du système.

2. ENVIRONNEMENT DE SIMULATION

Il existe plusieurs simulateurs de réseau, tels que NS-2 [128], GloMoSim[1] [129] et OPNET[2] [130]. Afin de choisir le simulateur le plus adapté à notre protocole Ad hoc ($Favorite - AODV$), plusieurs critères peuvent être considérés, en particulier : [131]

- Précision des modèles.
- Performance du moteur de simulation.
- Passage à l'échelle.
- Facilité d'utilisation (prise en main, description des scénarios, automatisation, etc.).
- Facilité de l'analyse des résultats.
- Plate-forme d'exécution.

NS-2 reste le simulateur réseau le plus utilisé dans le milieu académique ainsi que dans l'industrie. NS-2 est un outil gratuit de simulation par événements discrets[3] dont le code source est disponible. Il

[1] Global Mobile Simulator.
[2] OPNET (Optimum Network Performance) est un outil de simulation de réseaux très puissant et très complet. Basé sur une interface graphique intuitive, son utilisation et sa prise en main est relativement aisée.
[3] Dans les simulations par événements, les opérations d'un système sont représentées sous forme d'une séquence chronologique d'événements. Chaque événement se produit à un instant donné et marque le changement d'état du système. En d'autres termes, le simulateur va passer d'un événement au suivant en mettant à jour à chaque fois la description de l'état du système. Et puisqu'il ne se passe rien entre les événements (pas d'évolution significative des variables), la date saute de la date courante à celle du nouvel événement. Si un tel système admet une description selon des variables évoluant de manière discontinue alors il est dit système discret.

fournit les mécanismes nécessaires à la mise en œuvre des protocoles et la simulation de leur comportement. NS-2 a gagné en popularité dans la communauté de recherche en réseau depuis sa naissance en 1989 grâce à sa flexibilité et son caractère modulaire. Plusieurs révisions ont vu le jour marquant la maturité croissante de l'outil tout en maintenant sa solidité et sa polyvalence. Il combine le langage de script OTcl et le langage C++. L'interpréteur OTcl sert à exécuter les scripts de commande utilisateur qui permettent la configuration, la description et la mise en place des simulations alors que C++ est utilisé pour l'implémentation du noyau du simulateur ainsi que des protocoles.

Généralement une simulation suit le processus suivant : [35]

- Phase 1 ($Configuration\ du\ réseau$) : il s'agit de créer et de configurer les composants du réseau (nœuds, connexions entre les nœuds, type de connexion, durée d'échange, le début du transfert de données, etc.). Pour ce faire, l'utilisateur créé un script OTcl ($< nom\_script >.tcl$) contenant les commandes nécessaires pour la mise en place de la simulation.

- Phase 2. (Exécution de la simulation) : il s'agit d'exécuter la simulation configurée dans la phase 1. Une horloge de simulation est maintenue pour exécuter les événements chronologiquement jusqu'à atteindre une limite spécifiée précédemment dans la phase de configuration. Pour cela, NS-2 offre aux utilisateurs une commande exécutable (ns) qui prend le nom du fichier contenant le script de simulation ($< nom\_script >.tcl$) comme argument en entrée et produit en sortie deux fichiers de traces :

  - ✓ "$< nom\_script >.tr$" dans lequel sont inscrites toutes les actions observées lors de la simulation.
  - ✓ "$< nom\_script >.nam$" qui sert à la création de l'animation correspondante à la simulation grâce au programme NAM[4] (Network AniMator) inclus dans NS-2 et ce en exécutant la commande "nam" suivie du fichier à visualiser.

- Phase 3. (Évaluation de la performance du réseau simulé) : il s'agit de collecter et de compiler les résultats des simulations. Un parcours du fichier de traces ($< nom\_script >.tr$) s'impose pour dégager les informations voulues. Des langages de script (tel que awk ou $shell$) sont généralement utilisés pour analyser les fichiers et dégager l'information voulue.

La figure 7.1 met en avant de manière schématique les composants les plus importants de NS-2 [35]. Pour conclure, NS-2 offre un compromis entre performance et facilité d'utilisation. Ainsi, nous avons adopté cet outil pour l'implémentation et la simulation de notre proposition.

[4] Nam est un outil d'animation basé sur Tcl / Tk pour visualiser des traces de simulation de réseau et des traces de paquets dans le monde réel. Il prend en charge la disposition de la topologie, l'animation au niveau du paquet et divers outils d'inspection des données.

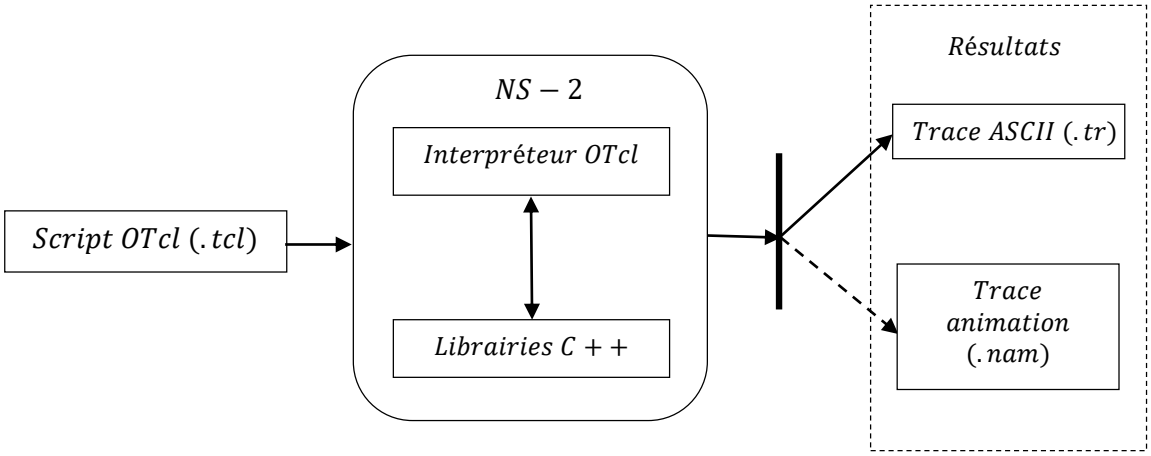


Figure 7.1 : Représentation schématique de NS-2.

Nous présentons dans cette section les différents paramètres à prendre en considération dans la simulation. Ces paramètres sont initialisés dans le script ($< nom\_script.tcl >$). Nous décrivons ainsi le contenu de ce fichier regroupant les paramètres des simulations qui concernent la topologie, le trafic, la mobilité, le timing des événements, etc.

- $Topologie$: Nous considérons un réseau Ad hoc composé de 100 nœuds mobiles répartis aléatoirement sur une surface carrée de $(1000 * 1000)\ m^2$ présentée par la figure 7.2. Les nœuds utilisent le protocole MAC IEEE 802.11, et la bande passante est de $2\ Mbps$.

- $Mobilité$: Les nœuds se déplacent constamment en utilisant le modèle de cheminement "$random\ waypoint$", un scénario de mouvement est généré aléatoirement et les nœuds se déplacent linéairement avec une vitesse comprise entre (0 et 25) m/s. Nous utilisons le script "$setdest$", fourni avec NS-2, qui permet de la génération automatique de ce type de scénario.

- $Structure\ des\ nœuds$: Chaque nœud maintient une file d'attente de type $FIFO$[5]. Ainsi, les messages les plus anciens sont effacés lors d'un débordement dans cette file. La taille de cette file d'attente est fixée à 50 paquets.

- $Modèle\ du\ trafic$: Chaque simulation dure $(1000\ s)$ durant laquelle un certain nombre de pairs de nœuds veulent échanger des paquets de données de type CBR ($Constant\ Bit\ Rate$). La taille des paquets échangés est de $512\ octets$. Ces paquets sont diffusés à intervalles réguliers (tous les $0.25\ s$) entre le début et la fin de la transmission. Les scénarios de transmission sont générés automatiquement en utilisant le script "$cbrgen.tcl$" fourni avec NS-2.

[5] Premier entré, premier sorti.

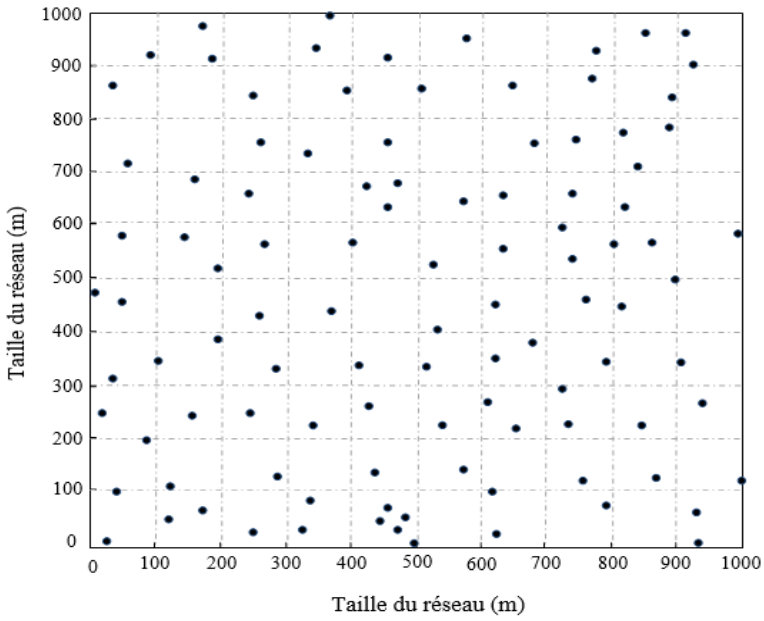


Figure 7.2 : Modèle d'expérimentation.

Nous assumons qu'il existe vingt nœuds attaquants, déployés aléatoirement dans le réseau. Ces derniers passent par une période d'écoute passive (d'une durée aléatoire), et essayent par la suite de se connecter avec des nœuds victimes choisis d'une manière aléatoire (attaquant simple), ou d'une manière plus spécifique (attaquant malicieux). Ainsi, les nœuds malveillants ne vont pas lancer leurs attaques simultanément, mais au fur et à mesure de l'avancement de la simulation. Tous les paramètres de notre simulation sont résumés dans le tableau ci-dessous :

Table 7.1 : Paramètres de simulation dans NS-2.

Paramètre	Valeur
Modèle de propagation sans fil	$Free\ Space$
Simulateur	$NS-2.34$
Protocole d'application	CBR
Protocole MAC	$IEEE\ 802.11$
Protocole de transport	$IEEE\ 802.11b$
Protocole réseau	$IPv4$
Protocole physique	UDP
Type d'antenne	$Omni-directional$
Placement des nœuds	$Random$
Capacité de la file d'attente	$50\ packets$
Nombre de nœuds	$100\ nœuds$
La surface du réseau	$(1{,}000 * 1{,}000)\ m^2$

Modèle de mobilité	*Random Waypoint*[6]
Gestion et ordonnancement de la file d'attente	*DropTail*[7]/*PriQueue*[8]
Vitesse Minimale - Maximale	$0 - 25$[9] m/s
Portée de transmission	$250\ m$
L'énergie initiale des nœuds	2 *Joule*
Capacité d'envoi	2 *Mbps*
Temps de simulation	$1000\ s$
Temps de pause	$10\ s$
Taille du paquet de données	512 *Bytes*
Nombre maximal de nœuds malveillants	20% (20 nœuds)
Type d'attaque	*Attaque coordonnée*

Les valeurs par défaut des paramètres utilisées dans notre protocole pour calculer la valeur de confiance globale d'un nœud sont données dans le tableau (7.2).

Table 7.2 : Valeurs des paramètres utilisés pour calculer la valeur de confiance globale.

μ	θ	w_1	w_2	w_3
$1/2$	3	0.40	0.33	0.27

Pour déterminer les valeurs de pondération des paramètres $w_1, w_2\ et\ w_3$, nous supposons que la valeur pondérée du degré de confiance directe est supérieure aux valeurs de pondération qui sont affectées aux autres types d'interaction de confiance (recommandation de confiance et confiance indirecte).

Une fois que ces scripts de simulations ainsi que les fichiers contenant les scénarios de mobilité et de trafic sont prêts, il suffit de créer un script "*shell*" qui permet d'automatiser le lancement des simulations. Une fois l'exécution des simulations terminée, nous analysons les fichiers de trace (en utilisant des scripts *awk*[10]) afin de filtrer les informations pertinentes et nous synthétisons les résultats obtenus sous forme de courbes. Dans la section quatre, nous présentons ces courbes et nous interprétons leur signification.

[6] Le modèle de point de cheminement aléatoire (Random Waypoint) a été proposé par Johnson et Maltz [39] C'est l'un des modèles de mobilité les plus populaires [132] pour évaluer les protocoles de routage des réseaux ad-hoc mobiles (MANET), en raison de sa simplicité et de sa grande disponibilité.

[7] Tail drop est un algorithme de gestion de file d'attente simple utilisé par les programmeurs réseau dans l'équipement réseau pour décider quand abandonner les paquets.

[8] Une file d'attente prioritaire (PriQueue) est un type de données abstrait qui s'apparente à une structure de file d'attente ou de pile régulière. Dans une file d'attente prioritaire, un élément de haute priorité est servi avant un élément de faible priorité. Si deux éléments ont la même priorité, ils sont servis selon leur ordre dans la file d'attente.

[9] Les nœuds mobiles peuvent être des engins roulants (Quads, tracteurs, etc.).

[10] Est un langage de traitement de lignes, disponible sur la plupart des systèmes Unix et sous Windows avec Cygwin ou Gawk. Il est principalement utilisé pour la manipulation de fichiers textuels pour des opérations de recherches, de remplacement et de transformations complexes.

3. ÉVALUATION DES PERFORMANCES DU PROTOCOLE FAVORITE-AODV

3.1 Mesures de performance

Après la mise en œuvre du protocole proposé, nous avons effectué des simulations respectivement en termes de taux des nœuds malveillant et de la vitesse maximale du nœud. Les résultats de la simulation ont été analysés et comparés avec les protocoles : AODV [37], S-AODV [3] et T-AODV [133].

Nous utilisons différents métriques d'évaluations afin d'évaluer la performance de ces protocoles, dans lequel les deux premières métriques sont les plus importantes pour les meilleurs protocoles d'acheminement et de transmission.

Une étude de simulation a été réalisée pour les mesures suivantes : Taux de paquets reçus avec succès (PDR), Délai moyen de bout-en-bout, Volume de trafic de contrôle (Overhead), Débit, Optimalité du chemin, Consommation moyenne d'énergie, le Taux de détection et le taux de faux-positifs.

3.2 Résultats des simulations

Cette section se compose de trois parties principales. Dans la première, *Nous allons mettre à l'épreuve* les concepts cités dans le modèle de confiance proposé, en prouvant que le cadre présenté peut montrer clairement la différence dans les valeurs de confiance entre un nœud normal et un nœud égoïste sur un paramètre spécifique en définissant un vecteur de poids approprié. De plus, la valeur de confiance globale est calculée en utilisant les facteurs de relation et les poids des nœuds voisins, et pas simplement en prenant en compte une valeur de confiance moyenne. Dans la deuxième partie, nous nous intéressons aux performances de notre protocole, nous montrons ainsi que l'ajout de notre modèle de gestion de la confiance n'a pas d'impact négatif sur les performances du protocole de base (AODV). Enfin dans la troisième partie, nous nous focalisons sur la performance du système de détection en termes de taux de détection d'actions malveillantes.

3.2.1 Mise à l'épreuve du modèle de confiance

3.2.1.1 Valeurs de confiance directes et de recommmandation

Dans cette partie, la simulation définit six nœuds et calcule les valeurs de confiance de 4 nœuds à partir de ceux-ci, (Figure 7.3).

Grace à l'utilisation de la théorie grise, nous pouvons utiliser les paramètres d'entrée d'un nœud pour calculer la valeur de confiance du nœud cible ($le\ nœud\ N_1\ par\ exemple$) à partir du point de vue d'un nœud spécifique comme le $nœud\ N_0$, par rapport aux nœuds voisins du $nœud\ N_0$. C'est-à-dire, nous obtenons la valeur de confiance du $nœud\ N_1$ à partir du $nœud\ N_0$, qui a des nœuds voisins (N_2 et N_3).

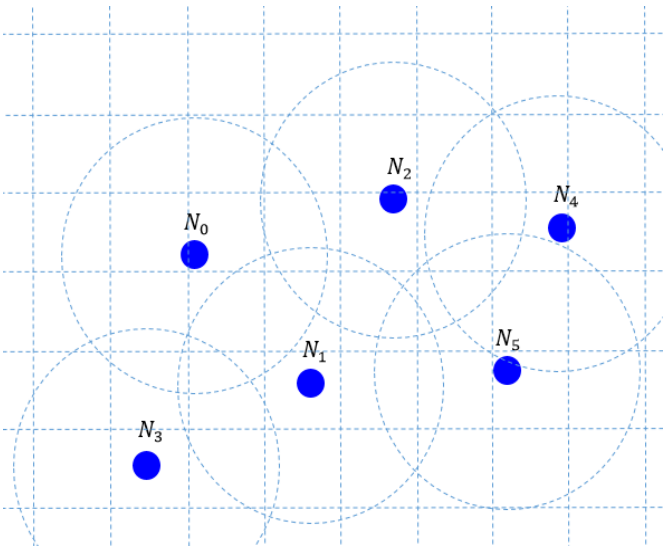


Figure 7.3 : Topologie des six nœuds Ad hoc.

a) *Sélection des attributs clés du nœud*

Avant de confirmer le comportement de confiance des nœuds voisins, nous devons définir les attributs clés qui sont les indicateurs d'évaluation (facteurs). Tous ces attributs devraient bien refléter les caractéristiques du comportement du nœud afin de rendre l'évaluation de confiance objective et efficace. Nous devons prendre en compte les fonctionnalités de communication réseau qui reflètent les performances d'un nœud, telles que la tolérance aux pertes) et les attributs physiques des nœuds, tels que (la mobilité, le signal sans fil, la vitesse de transmission, etc.).

Le taux de perte de paquets[11], est également un facteur clé pour l'évaluation de la fiabilité d'un nœud. En ce qui concerne les attributs physiques d'un nœud, le facteur de mobilité se concentre principalement sur la vitesse du nœud (plus la vitesse est élevée, plus le taux de perte de paquets est élevé). La propriété de la puissance du signal comprend deux attributs, dont l'un est la force du signal, tandis que l'autre facteur est la stabilité des signaux.

En fonction des caractéristiques de la communication réseau, nous pouvons choisir deux attributs de performance et deux attributs de fiabilité. La vitesse de transmission et l'intensité du signal reflètent

[11] Packet loss rate.

la performance, tandis que le taux de perte de paquets et le taux de changement de signal reflètent la fiabilité de la communication.

Selon la définition 4.8 du chapitre 4, les paramètres que nous voulons observer dans notre modèle de confiance sont :

$le\ taux\ de\ perte\ de\ paquets$, $la\ vitesse\ de\ transmission$, $la\ puissance\ du\ signal\ reçu$ et $le\ taux\ de\ changement\ de\ signal$.

$$X = \{Taux\ de\ perte\ de\ paquets,\ Puissance\ du\ signal\ reçu,\ Vitesse\ de\ transmission, Taux\ de\ changement\ de\ signal\}$$

Suivant le principe général d'évaluation du réseau mobile Ad hoc qui se base sur les périphériques compatibles 802.11, un $taux\ de\ perte\ de\ paquets$ inférieur à 10 % est considéré comme "$fiable$", et le taux de (20 à 30) % est un taux de perte "$générique$", alors qu'un taux de 70 % est considéré comme "$mauvais$".

Dans l'aspect de la puissance du signal, selon certaines normes, la puissance d'émission des appareils 802.11 ne doit pas dépasser 100 mW ($milliWatts$), nous pouvons calculer la puissance du signal à différentes distances par le modèle en espace libre. Dans ce modèle, la relation entre l'affaiblissement de puissance et la distance de transmission est mesurée par formule la suivante [136] :

$$L_{bf} = 32{,}4 + 20 * Log\ (d) + 20 * Log\ (f) \qquad (7.1)$$

Où :

L_{bf} représente l'affaiblissement de puissance en espace libre, mesurée en dB ($décibel$), d pour la distance (km), f pour la fréquence (mhz). D'après la formule ci-dessus et la distance de travail et comme la puissance d'émission du réseau est d'environ 20 dBm[12], un nœud devrait être "bon" lorsque la force du signal mesurée est supérieure à $((-65\ dBm) \cong (3{,}36 * 10^{-7} mW))$, et "$générique$" lorsque la puissance du signal est d'environ $((-70\ dBm) \cong (0{,}95 * 10^{-7} mW))$ et "$mauvais$" quand il est $((-75\ dBm) \cong (0{,}28 * 10^{-7} mW))$.

Le taux de variation du signal peut être déduit en fonction de la vitesse de déplacement. Lorsque le taux de variation est supérieur à 2,4 dBm/s, la communication est très "$mauvaise$". De même, si la vitesse de variation est inférieure à 2,1 dBm/s, la communication est "$bonne$", tandis qu'un taux de

[12] $P_{dBm} = 10 * log(P_{mW})$, $1\ dBm = 1{,}25\ mW$, $20\ dBm = 100\ mW$.

variation de 2,2 dBm/s est généralement "$générique$". Si la vitesse maximale du nœud ne dépasse pas la valeur de 25 m/s, le taux maximal de variation du signal est de 3,45 dBm/s.

Quatre nœuds sont disponibles pour l'évaluation et la valeur de l'attribut pour chaque nœud se trouve dans le tableau suivant : (Table 7.3).

Table 7.3 : Valeur d'attribut pour chaque nœud.

$Noeud$	$Taux\ de\ perte$ (%)	$Taux\ de\ variation\ du\ signal\ (dBm/s)$	$Vitesse\ de\ transmission\ (KB/s)$	$Puissance\ du\ Signal\ *10^{-7}\ (mW)$
N_0	20	2,23	280	1
N_1	77	2,58	50	0,5
N_2	9	3,41	235	1,5
N_3	33	2,13	300	2,7

b) *Calcul du poids des attributs du nœud*

Selon le tableau 7.3, la matrice comparative x est constituée par les valeurs d'évaluation de chaque facteur.

$$x = \begin{pmatrix} 0{,}20 & 0{,}77 & 0{,}09 & 0{,}33 \\ 2{,}23 & 2{,}58 & 3{,}41 & 2{,}13 \\ 280 & 50 & 235 & 300 \\ 1 & 0{,}5 & 1{,}5 & 2{,}7 \end{pmatrix}$$

Selon la définition 4.10 du chapitre 4, la suite optimale x^0 est fondée sur la sélection de la valeur maximale de chaque facteur d'évaluation dans la matrice comparative x .

$$x^0 = \begin{pmatrix} 0{,}80 & 0{,}23 & 0{,}91 & 0{,}67 \\ 1{,}22 & 0{,}87 & 0{,}04 & 1{,}32 \\ 280 & 50 & 235 & 300 \\ 1 & 0{,}5 & 1{,}5 & 2{,}7 \end{pmatrix}$$

Pour normaliser les valeurs de la matrice x^0, nous appliquons quelques changements selon les équations $(4.1, 4.2\ ou\ 4.3)$, alors la matrice modifiée est :

$$x^0 = \begin{pmatrix} 0{,}80 & 0{,}23 & 0{,}91 & 0{,}67 \\ 0{,}924 & 0{,}659 & 0{,}030 & 1 \\ 0{,}430 & 0{,}076 & 0{,}361 & 0{,}461 \\ 0{,}297 & 0{,}148 & 0{,}446 & 0{,}803 \end{pmatrix}$$

Les fonctions de blanchiment correspondantes pour chaque facteur (ligne) de la matrice x^0 sont les suivantes :

1. Pour chaque élément x de la ligne 1 dans la matrice x^0.

   a) x appartient à la classe "$Mauvaise$" et sa fonction d'appartenance est :

$$f_{Mauvaise}(x) = \begin{cases} \frac{0,5-x}{0,2} & 0,3 < x \leq 0,5 \\ 1 & x \leq 0,3 \\ 0 & x > 0,5 \end{cases}$$

   b) x appartient à la classe "$Générique$" et sa fonction d'appartenance est :

$$f_{Générique}(x) = \begin{cases} 0 & x \leq 0,5 \\ \frac{x-0,5}{0,2} & 0,5 < x \leq 0,7 \\ \frac{0,9-x}{0,2} & 0,7 < x \leq 0,9 \\ 0 & x > 0.9 \end{cases}$$

   c) x appartient à la classe "$Bonne$" et sa fonction d'appartenance est :

$$f_{Bonne}(x) = \begin{cases} 0 & x \leq 0,7 \\ \frac{x-0,7}{0,2} & 0,7 < x < 0,9 \\ 1 & x \geq 0,9 \end{cases}$$

2. Pour chaque élément x de la ligne 2 dans la matrice x^0.

   a) x appartient à la classe "$Mauvaise$" et sa fonction d'appartenance est :

$$f_{Mauvaise}(x) = \begin{cases} \frac{0,85-x}{0,15} & 0,7 < x \leq 0,85 \\ 1 & x \leq 0,7 \\ 0 & x > 0,85 \end{cases}$$

   b) x appartient à la classe "$Générique$" et sa fonction d'appartenance est :

$$f_{Générique}(x) = \begin{cases} 0 & x \leq 0,85 \\ \frac{x-0,85}{0,05} & 0,85 < x \leq 0,9 \\ \frac{0,9-x}{0,05} & 0,9 < x \leq 0,95 \\ 0 & x > 0.95 \end{cases}$$

   c) x appartient à la classe "$Bonne$" et sa fonction d'appartenance est :

$$f_{Bonne}(x) = \begin{cases} 0 & x \leq 0,95 \\ \frac{x-0,95}{0,05} & 0,95 < x < 1 \\ 1 & x \geq 1 \end{cases}$$

3. Pour chaque élément x de la ligne 3 dans la matrice x^0.

a) x appartient à la classe "$Mauvaise$" et sa fonction d'appartenance est :

$$f_{Mauvaise}(x) = \begin{cases} \frac{0,15-x}{0,07} & 0,08 < x \leq 0,15 \\ 1 & x \leq 0,08 \\ 0 & x > 0,15 \end{cases}$$

b) x appartient à la classe "$Générique$" et sa fonction d'appartenance est :

$$f_{Générique}(x) = \begin{cases} 0 & x \leq 0,15 \\ \frac{x-0,15}{0,2} & 0,15 < x \leq 0,35 \\ \frac{0,7-x}{0,35} & 0,35 < x \leq 0,7 \\ 0 & x > 0.7 \end{cases}$$

c) x appartient à la classe "$Bonne$" et sa fonction d'appartenance est :

$$f_{Bonne}(x) = \begin{cases} 0 & x \leq 0,7 \\ \frac{x-0,7}{0,3} & 0,7 < x < 1 \\ 1 & x \geq 1 \end{cases}$$

4. Pour chaque élément x de la ligne 4 dans la matrice x^0.

a) x appartient à la classe "$Mauvaise$" et sa fonction d'appartenance est :

$$f_{Mauvaise}(x) = \begin{cases} \frac{0,13-x}{0,07} & 0,06 < x \leq 0,13 \\ 1 & x \leq 0,06 \\ 0 & x > 0,13 \end{cases}$$

b) x appartient à la classe "$Générique$" et sa fonction d'appartenance est :

$$f_{Générique}(x) = \begin{cases} 0 & x \leq 0,13 \\ \frac{x-0,13}{0,12} & 0,13 < x \leq 0,25 \\ \frac{0,5-x}{0,25} & 0,25 < x \leq 0,5 \\ 0 & x > 0.5 \end{cases}$$

c) x appartient à la classe "$Bonne$" et sa fonction d'appartenance est :

$$f_{Bonne}(x) = \begin{cases} 0 & x \leq 0,5 \\ \frac{x-0,5}{0,5} & 0,5 < x < 1 \\ 1 & x \geq 1 \end{cases}$$

Sur la base des fonctions d'appartenance citées ci-dessus, nous pouvons transformer la matrice x^0 à la matrice standard H.

$$H = \begin{pmatrix} 0{,}7966 & 0 & 1 & 0{,}5934 \\ 0{,}9124 & 0{,}6033 & 0 & 1 \\ 0{,}7332 & 0 & 0{,}8831 & 1 \\ 0{,}2036 & 0 & 0{,}3156 & 1 \end{pmatrix}$$

En fonction du vecteur de poids d'origine $W = \{0.25, 0.25, 0.25, 0.25\}$ et les fonctions d'appartenance, notre modèle de confiance obtient les valeurs de confiance des trois nœuds (N_1, N_2 et N_3) pour le nœud N_0 sur une période de 10 secondes. Ici, le coefficient d'identification ρ égal à $^1/_2$.

$T_{(N_1,N_0)} = 0{,}4766$, en tant que valeur de $confiance\ directe$ du nœud N_1.

$T_{(N_1,N_2)} = 0{,}2354$, $T_{(N_1,N_3)} = 0{,}2135$ qui sont les valeurs de $recommandation$ des nœuds N_2 et N_3 à propos du nœud N_1.

Selon l'équation (5.11), la valeur globale de la confiance du nœud N_1 pour le nœud N_0 avec 3 nœuds voisins est : $T_{(N_1,N_0)-3\text{nœuds}} = 0{,}3188$.

3.2.1.2 Recommandations et valeurs de confiance indirectes

Ici, le système calcule les valeurs de confiance parmi 6 $nœuds$, en considérant plusieurs liens supplémentaires. À partir des données de simulation[13], nous obtenons :

$T_{(N_1,N_0)} = 0{,}4766$
$T_{(N_1,N_2)} = 0{,}2354$
$T_{(N_1,N_3)} = 0{,}2135$
$T_{(N_1,N_4)} = 0{,}7952$
$T_{(N_1,N_5)} = 0{,}6251$

À partir de ceux-ci, il est possible de calculer la valeur de confiance globale avec 5 nœuds voisins ("$T_{(N_1,N_0)-5\text{nœuds}} = 0{,}2658$"), comme le montre la figure 7.4, nous pouvons savoir que la classe de cluster grise du nœud N_1 est "$Cluster_1$".

[13] **Table 7.4 :** Valeur d'attribut pour les nœuds N_4 & N_5

Noeud	*Taux de perte (%)*	*Taux de variation du signal (dBm/s)*	*Vitesse de transmission (KB/s)*	*Puissance du Signal* $* 10^{-7}$ *(mW)*
N_4	30	2,19	235	0,7
N_5	56	2,44	70	0,45

3.2.1.3 Analyse et discussion

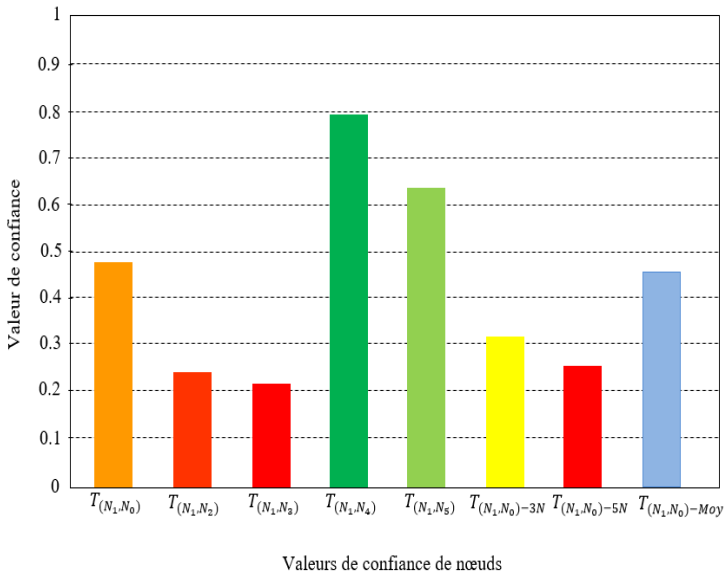


Figure 7.4 : Valeurs de confiance du nœud 1.

De la figure 7.4, les résultats montrent que la prise de différents facteurs de relation affectera la valeur globale de la confiance. Il est évident que la moyenne "$T_{(N_1,N_0)-Moy}$" est supérieure à "$T_{(N_1,N_0)-3\text{nœuds}}$" et "$T_{(N_1,N_0)-5\text{nœuds}}$", en raison de l'exclusion des poids de relation. Si la simulation considère seulement les opinions des nœuds $N_0, N_2\ et\ N_3$, le résultat sera inférieur à celui incluant les nœuds $N_0, N_2, N_3, N_4\ et\ N_5$, car $T_{(N_1,N_5)}$ est supérieur à $T_{(N_1,N_3)}$ et $T_{(N_1,N_4)}$ est supérieur à $T_{(N_1,N_2)}$.

La figure 7.5 montre les valeurs de confiance calculées par la théorie grise et le taux de livraison de paquets (PDR)[14] qui est souvent la métrique la plus sélectionnée par les approches basées sur la confiance [137], car il représente la fiabilité de la transmission de paquets ainsi que d'assurer une bonne qualité de service (QoS) dans le réseau. Elle montre également l'effet de la valeur de confiance sur le PDR, cela implique clairement que la valeur de la confiance décidée augmente le PDR d'une manière presque linéaire.

Les simulations ont été modifiées pour qu'un nœud parmi les six nœuds du réseau se comporte de manière égoïste. Le nœud tend à être normal pendant le temps initial, puis se comporte de manière égoïste, en communiquant uniquement avec ses voisins proches. En utilisant la théorie grise, les valeurs de confiance sont affectées par le changement de la force du signal reçu, bien que le taux de livraison de paquets conserve la même valeur qu'avec les comportements normaux. La raison principale pour laquelle la valeur de confiance du nœud égoïste diminue est que sa force du signal observée par le voisin augmente, ce qui entraîne une baisse de la valeur de gris de la force de son signal.

[14] $PDR = (Nombre\ de\ paquets\ de\ données\ reçus\ par\ la\ Destination / Nombre\ de\ paquets\ de\ données\ émis\ par\ la\ Source) * 100.$

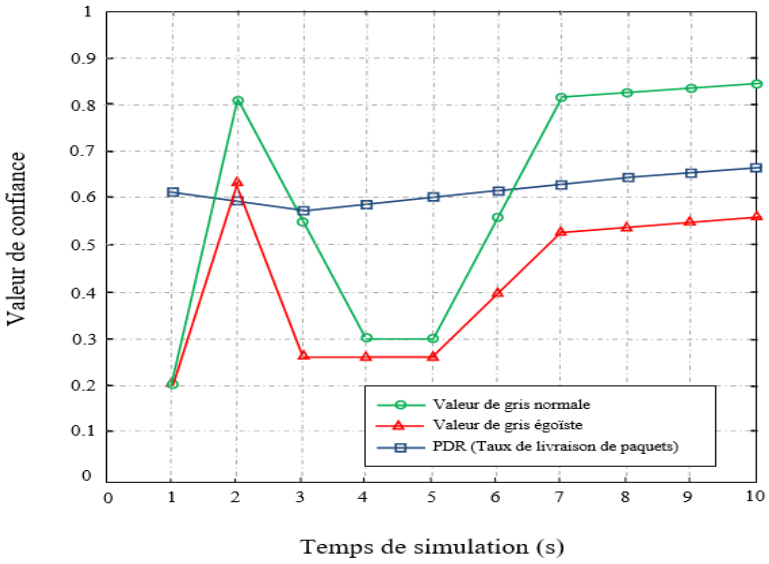


Figure 7.5 : Valeurs de confiance calculées par la théorie grise.

3.2.2 *Performance du protocole*

Pour pouvoir analyser les performances du notre protocole, nous nous basons sur le calcul de métriques citées ci-dessus pour montrer que les performances obtenues par le protocole auquel nous avons ajouté notre modèle de confiance sont au moins améliorées aux performances obtenues par les protocoles AODV "$sans\ modification$", S − AODV et T − AODV. Ainsi, pour chaque métrique, nous effectuons les tests suivants :

- $Test\ 1 :\ Variation\ du\ nombre\ de\ nœuds\ malveillants.$
- $Test\ 2 :\ Variation\ des\ vitesses\ de\ nœud.$

La comparaison des courbes obtenues à l'issue de l'exécution du test 1 et 2, montre qu'en cas d'attaques, l'extension que nous proposons présente de meilleures performances par rapport au protocole de base tout en rendant le routage plus sûr. Nous interprétons dans ce qui suit les résultats obtenus.

La figure 7.6 représente la topologie du réseau pour Favorite − AODV qui est configuré avec 20 % de nœuds malveillants. Il est clair que notre protocole peut interdire les interactions avec des nœuds malveillants, parce que toutes les interactions sont basées sur le modèle de confiance proposé, en prenant en compte la valeur de confiance globale entre les nœuds.

La figure 7.7 montre que l'effet des nœuds malveillants dans notre protocole est inférieur à celui de T − AODV, et que la valeur de confiance du nœud malveillant avec Favorite − AODV est inférieure à celle obtenue par T-AODV. Dans le temps "0 $seconde$", la valeur de confiance d'un nœud malveillant inconnu est initialisée à "0,8". Avec un temps de simulation accru, un nœud malveillant est trouvé et

identifié par le réseau sur la base de différents types de relation de confiance. Ainsi, la valeur de confiance du nœud malveillant diminue progressivement jusqu'à atteindre la valeur "0,1". À partir de cette figure, nous voyons que Favorite − AODV peut efficacement atténuer les risques causés par de tels nœuds par rapport aux valeurs de confiance de nœud basées sur T − AODV.

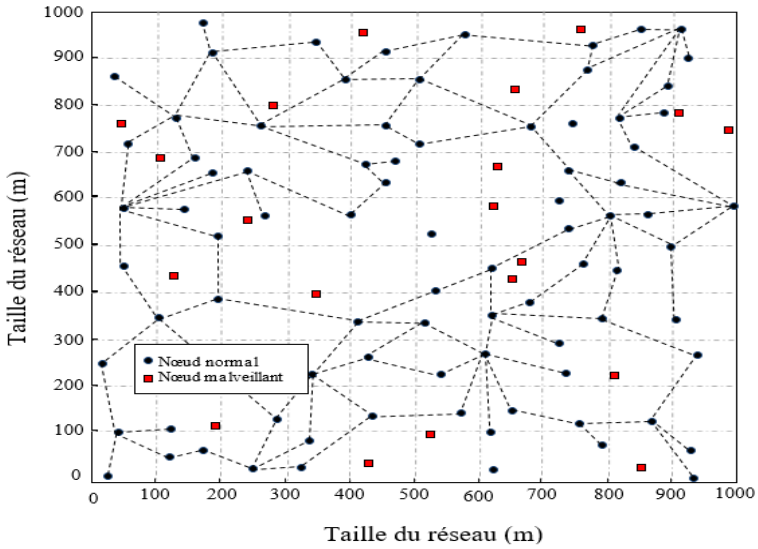


Figure 7.6 : Topologie du réseau de Favorite-AODV.

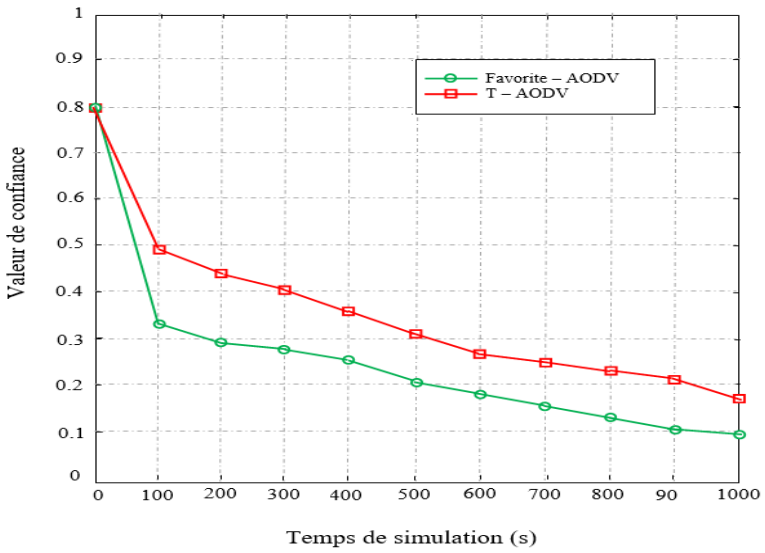


Figure 7.7 : Fluctuation de la valeur de confiance de nœuds malveillants.

La figure 7.8 illustre la comparaison des taux de satisfaction de l'interaction réseau (équation 5.4) avec les protocoles AODV "*sans modification*", S − AODV, T − AODV et notre protocole. Nous constatons que les taux de satisfaction de l'interaction avec les modèles basés sur la gestion de la confiance augmentent avec l'augmentation du temps de simulation, tandis que le taux de satisfaction dans une approche qui ne repose pas sur un modèle de confiance diminue légèrement. En général, les

taux de satisfaction avec les modèles de confiance sont beaucoup plus élevés que ceux sans modèle de confiance.

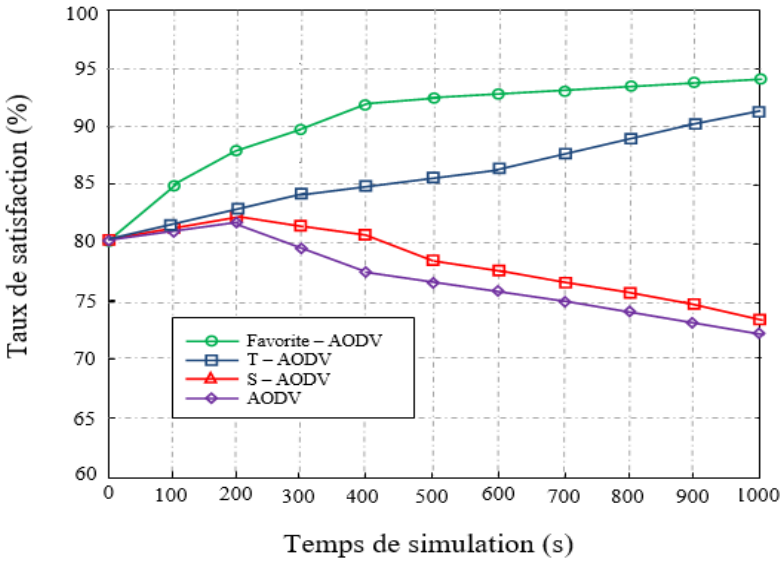


Figure 7.8 : Taux de satisfaction dans différents protocoles.

Cette amélioration avec les modèles basés sur la confiance peut être attribuée au mécanisme de confiance du nœud, ce qui augmente la probabilité de réussite (ou de bon service) à un nœud digne de confiance. Si un nœud est considéré comme un nœud malveillant par ses voisins qui ne poursuivront pas les interactions avec lui, de sorte que les interactions ne se produiront uniquement entre les nœuds normaux, ce qui entraînera une augmentation des taux de satisfaction de l'interaction réseau.

3.2.2.1 Taux de livraison de paquet "Packet delivery ratio – PDR"

Le taux de livraison de paquet de données (PDR) est le nombre de paquets de données reçus avec succès par la destination par rapport au nombre de paquets de données émis par la source.

$$PDR = \frac{Nombre\ de\ paquets\ de\ données\ reçus\ par\ la\ Destination}{Nombre\ de\ paquets\ de\ données\ émis\ par\ la\ Source} * 100\ (\%) \qquad (7.2)$$

PDR Présente une métrique importante pour l'évaluation des performances d'un protocole de routage ad hoc mobile, car le nombre de paquets de données fournis dépend principalement de la disponibilité du chemin, qui dépend de l'efficacité de l'algorithme de routage sous-jacent. Cette métrique nous permet aussi de vérifier si l'extension du protocole a un impact sur le transfert de paquets de données avec succès.

La plus grande valeur du PDR signifie une meilleure performance du protocole.

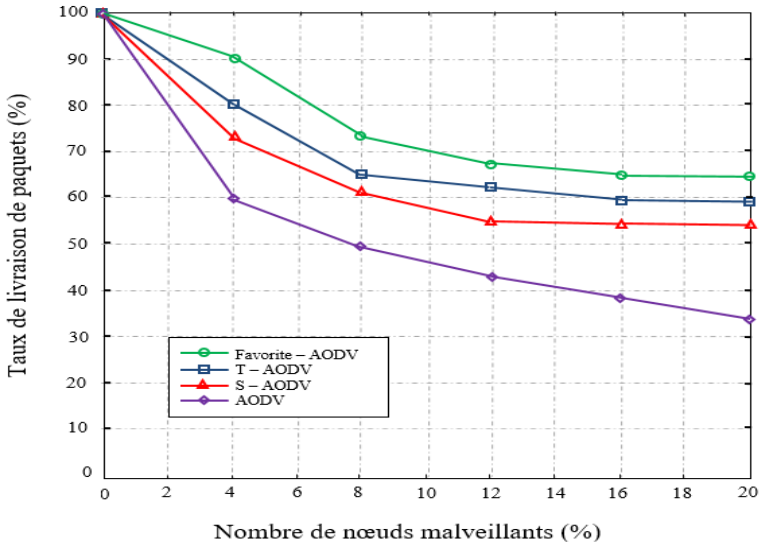


Figure 7.9 : PDR avec différents nombres de nœuds malveillants.

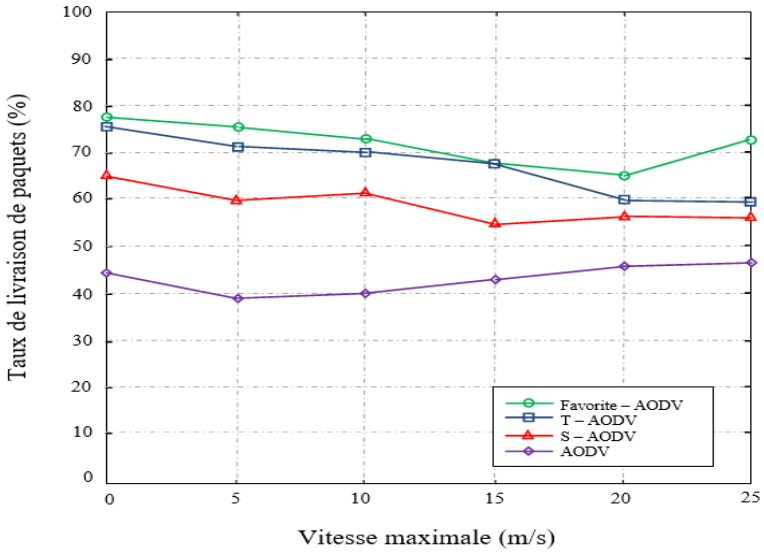


Figure 7.10 : PDR à différentes vitesses.

Comme le montrent les figures (7.9 et 7.10), il est clair que le taux de livraison de paquets de données (PDR) dans le protocole Favorite – AODV est généralement meilleur que celui dans d'autres protocoles. Puisque dans Favorite – AODV, les nœuds intermédiaires sélectionnent des nouvelles routes de confiance dans la procédure de découverte d'itinéraire, en utilisant les chemins approuvés afin de transmettre les paquets de données à la destination. Les résultats démontrent également que l'approche proposée a le plus haut PDR parmi les autres approches.

D'après la figure 7.9, nous pouvons remarquer que le taux de distribution de paquets dans tous les protocoles se dégrade lorsque le nombre de nœuds malveillants augmente. Le taux de livraison d'AODV

passe de 100 % à 35 %, lorsque le nombre de nœuds malveillants varie de 0 à 20 $nœuds$, cela signifie que de nombreux paquets sont perdus et que les performances de communication entre la source et la destination sont faibles. Sur la base de ce résultat, les nœuds malveillants limitent essentiellement les interactions entre les nœuds du réseau. Cependant, notre protocole peut maintenir des communications fiables avec un PDR élevé, en raison de la détection et l'atténuation des attaques pendant le processus de découverte d'itinéraire. Il effectue également la communication d'une manière où le réseau fonctionne sans attaque (Tolérance aux pannes).

En général, le PDR devient plus élevé lorsque la vitesse de simulation augmente. Comme le montre la figure 7.10, nous pouvons remarquer que la valeur de PDR dans notre protocole est presque stable, même dans les conditions de vitesses variées (75 % à 5 m/s − 72% à 25 m/s). Cela signifie que le changement de vitesses n'a aucun effet sur le PDR, en raison du nœud attaquant qui ne peut pas être impliqué dans le réseau.

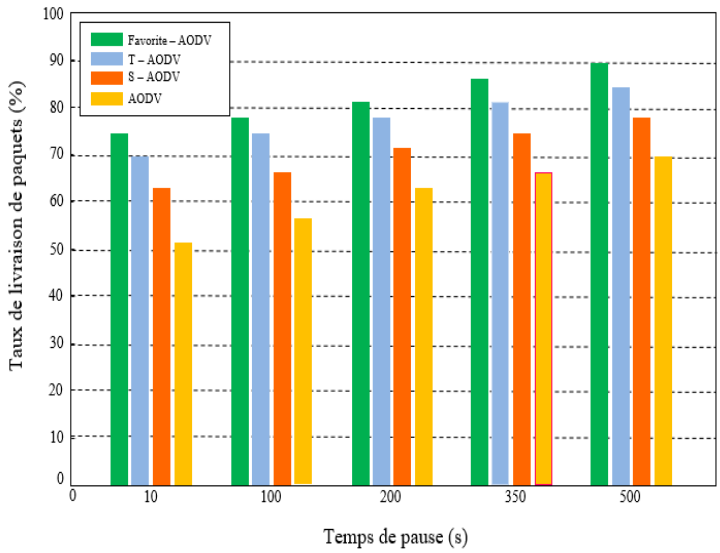


Figure 7.11 : PDR en fonction du temps de pause.

À partir de la figure 7.11, on observe que le protocole proposé fournit un taux de livraison de paquets meilleur par rapport aux autres protocoles. L'analyse comparative entre Favorite – AODV et les approches existantes montre que l'utilisation d'une approche de routage basée sur le calcul de la confiance joue un rôle très clair pour assurer un transfert de données hautement sécurisé et un meilleur taux de distribution des paquets.

3.2.2.2 *Délai moyen de bout − en − bout "Average End − to − End Delay"*

Le délai moyen de bout en bout est le retard subi par les paquets livrés avec succès pour atteindre leurs destinations. Il est dénoté "RAL" pour *Route Acquisition Latency* ou encore End- to- End

Delay [134]. Ce temps inclut le délai de traitement ainsi que le délai d'attente dans les files d'attentes dans chaque nœud intermédiaire. Il est calculé selon la formule suivante :

$$End - to - end\ delay = \frac{\sum(T_R(\mathrm{i}) - T_E(\mathrm{i}))}{Nobmbre\ de\ paquets\ émis} * 1000\ (ms) \qquad (7.3)$$

$T_R(\mathrm{i})$ est l'instant où le paquet "i" est reçu par la destination.

$T_E(\mathrm{i})$ est l'instant où le paquet "i" quitte la source.

La plus faible valeur du délai de bout en bout signifie une meilleure performance du protocole.

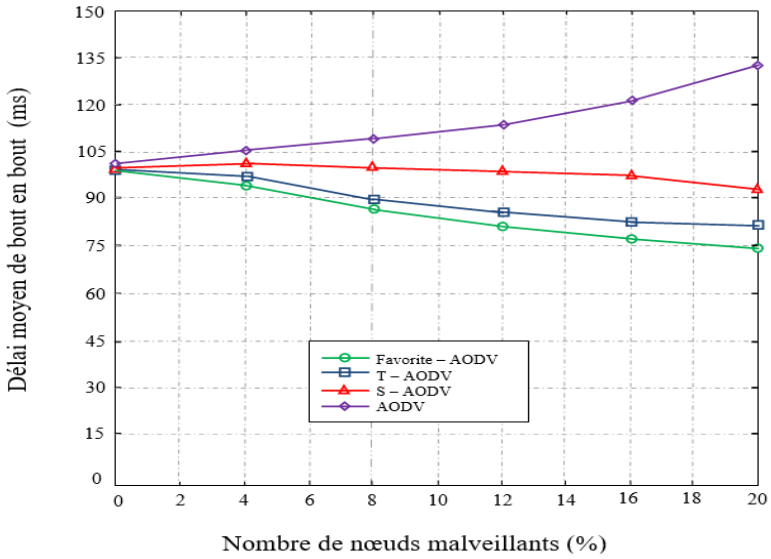


Figure 7.12 : Délai moyen avec différents nombres de nœuds malveillants.

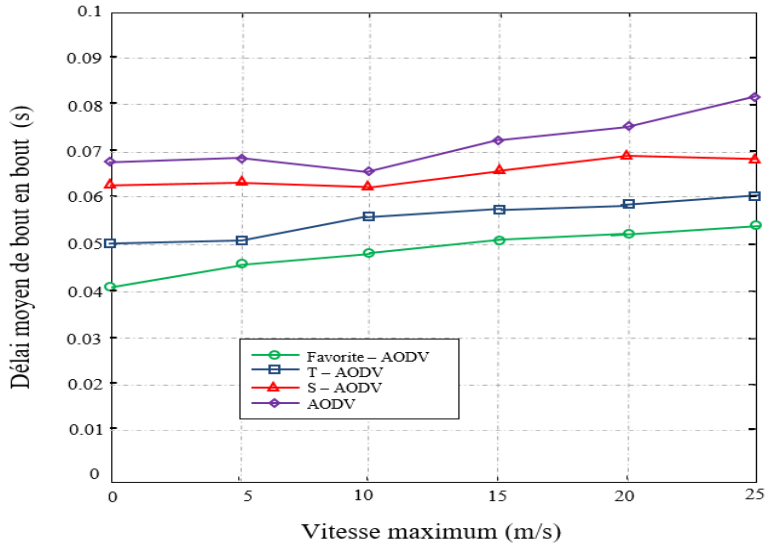


Figure 7.13 : Délai moyen en fonction de la vitesse.

Comme le montre la figure 7.12, le délai moyen dans les protocoles Favorite-AODV, S-AODV et T-AODV diminue lentement avec l'augmentation du nombre de nœuds malveillants, néanmoins, le délai moyen dans le protocole AODV augmente évidemment, ce délai est principalement dû aux délais de la mise en file d'attente et de retransmission. Ceci montre que l'implémentation de l'approche proposée n'influe pas sur le traitement des paquets et n'engendre pas de temps de traitement supplémentaire ralentissant le déroulement normal du protocole. Par conséquent, il y a une réduction évidente du délai moyen avec notre protocole par rapport à S-AODV et T-AODV.

L'observation de la figure 7.13, montre une augmentation du délai moyen de bout en bout dans les quatre protocoles. Effectivement, les approches basées sur la confiance ajoutent un facteur de confiance avec l'augmentation du nombre de nœuds malveillants et que le chemin de routage établi par ces méthodes peut ajouter des sauts, ce qui entraîne un délai plus long. Par conséquent, un retard insignifiant a été introduit dans les quatre approches. Néanmoins, une sécurité accrue est garantie dans le protocole proposé.

3.2.2.3 Volume de trafic de contrôle "Routing Overhead"

Le volume de trafic de contrôle ($Routing\ Overhead$) est le rapport entre le nombre de paquets de contrôle (RREQ, RREP, etc.) transmis par les nœuds émetteurs et le nombre de paquets de données reçus avec succès par les destinations.

Puisque la bande passante dans un réseau Ad hoc est une ressource limitée et partagée, il est crucial de l'économiser au maximum. De ce principe, notre protocole doit être évalué en termes de volume de trafic généré, qui est un facteur important dans la consommation de la bande passante. Il est calculé selon l'équation suivante :

$$Overhead = \frac{Nombre\ de\ Paquets\ de\ contrôle}{Nombre\ de\ Paquets\ reçus} \qquad (7.4)$$

La figure 7.14 illustre les performances de notre protocole avec différents nombres de nœuds malveillants en termes du trafic de contrôle (overhead). Nous pouvons clairement observer que le trafic de contrôle dans les quatre protocoles augmente de manière significative lorsque le pourcentage de nœuds malveillants augmente dans le réseau. L'approche Favorite-AODV a un trafic de contrôle presque équivalent à celle de S-AODV et de T-AODV, mais génère moins de trafic de contrôle par rapport au protocole standard AODV, Ceci à cause des nœuds malveillant qui reçoivent les paquets RREQs et ne les rediffusent pas dans l'AODV. Dans notre approche, les paquets de routage sont réduits car les nœuds ne rediffusent pas les paquets de routage venant des nœuds malveillants, ils abandonnent immédiatement toutes les communications reçues des nœuds malveillants. Nous notons que les trois autres protocoles gardent la découverte de la route par inondation à tous les voisins, cela entraîne de nombreuses transmissions de paquets de contrôle.

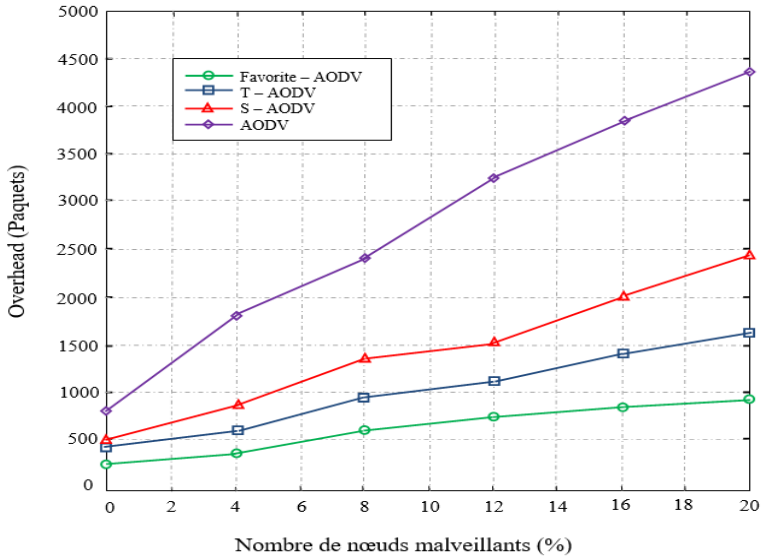


Figure 7.14 : Overhead versus nœuds malveillants.

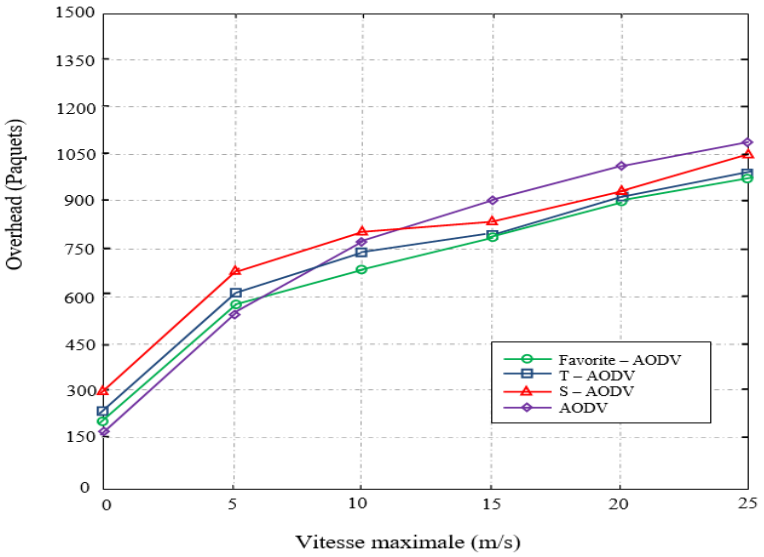


Figure 7.15 : Overhead en fonction de la vitesse.

Sur la figure 7.15, le trafic de contrôle dans ces protocoles augmente avec l'augmentation de la vitesse maximale. Lorsque la vitesse est inférieure à $7\ m/s$, le trafic de contrôle dans : (1) notre protocole, (2) T-AODV et (3) S-AODV reste comparativement plus élevé que celui dans AODV. Avec la croissance de la vitesse, il y a un impact opposé, la raison en est que davantage de paquets de RREQ et de RREP doivent être envoyés pour que les itinéraires éligibles satisfassent à l'exigence de confiance définie dans les protocoles Favorite-AODV, S-AODV et T-AODV, et qu'entre-temps, l'exigence de confiance n'est pas prise en compte dans le protocole AODV.

La figure 7.16 montre le volume de trafic de contrôle de messages importés par rapport à la version originale d'AODV. Comme les valeurs de confiance sont intégrées aux messages de contrôle, il n'est

pas nécessaire d'envoyer plus de messages contenant ces valeurs. Toutefois, lorsque le nombre de nœuds malveillants augmente, le pourcentage de temps système dans les messages, diminue considérablement. En analysant les résultats de la figure 7.16, nous obtenons pratiquement les mêmes effets que pour les figures (7.14 et 7.15), ceci donne une idée sur le nombre de bits de routage nécessaires envoyé avec succès, ce qui nous donne une idée sur la bande passante utile.

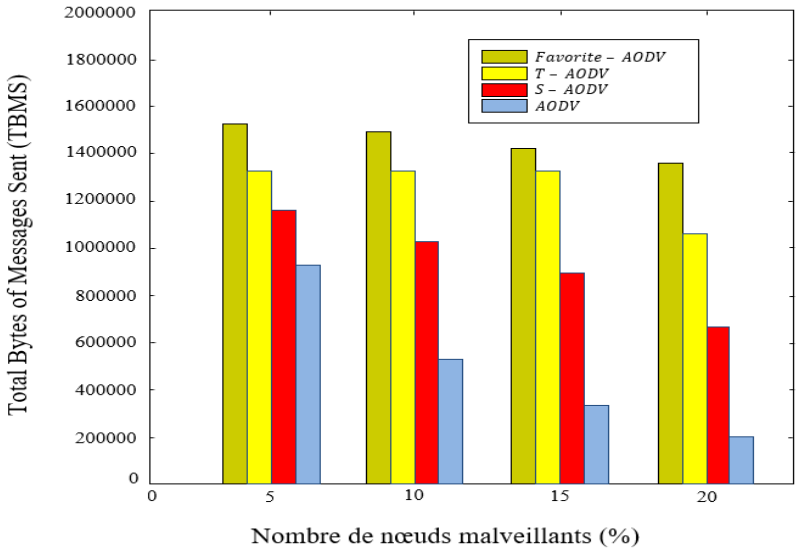


Figure 7.16 : TBMS avec différents nombres de nœuds malveillants.

3.2.2.4 Débit "Throughput"

Le débit indique la quantité de données numériques transmises par unité de temps de la source à la destination [135]. Ces données peuvent être transmises via une liaison physique ou logique ou passer par un certain nœud de réseau. Le débit est généralement la somme des débits de données délivrés à tous les terminaux d'un réseau qui peuvent être analysés mathématiquement au moyen de la théorie de la file d'attente. Cette mesure est calculée selon la formule suivante :

$$Débit = \frac{Nombre\ de\ Paquets\ reçus}{Nombre\ de\ Paquets\ orientés} \qquad (7.5)$$

Les figures (7.17 et 7.18) montrent clairement que notre approche peut générer un débit nettement supérieur à celui du S-AODV et T-AODV, car Favorite-AODV peut détecter efficacement les nœuds malveillants et éviter ainsi l'encombrement du canal. Nous pouvons également noter que le débit est presque stable dans différents scénarios, les vitesses maximales indiquant que notre méthode a une bonne dynamicité, comme illustrée à la figure 7.18.

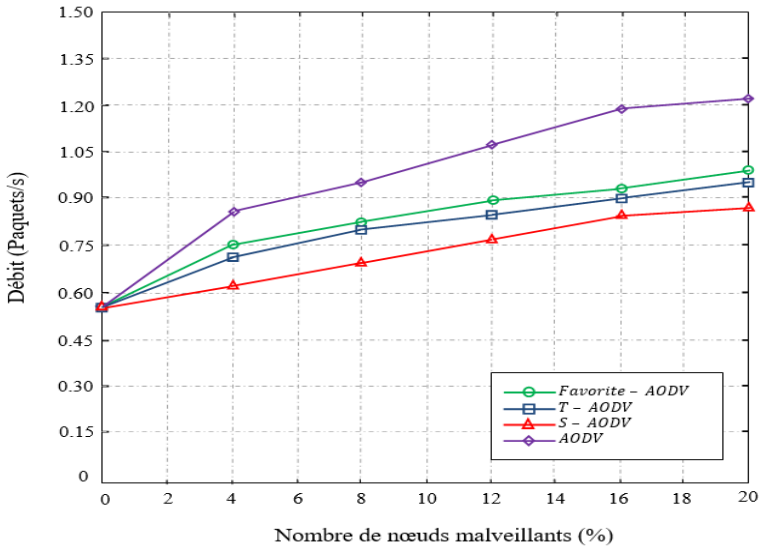


Figure 7.17 : Variation du Débit versus nœuds malveillants.

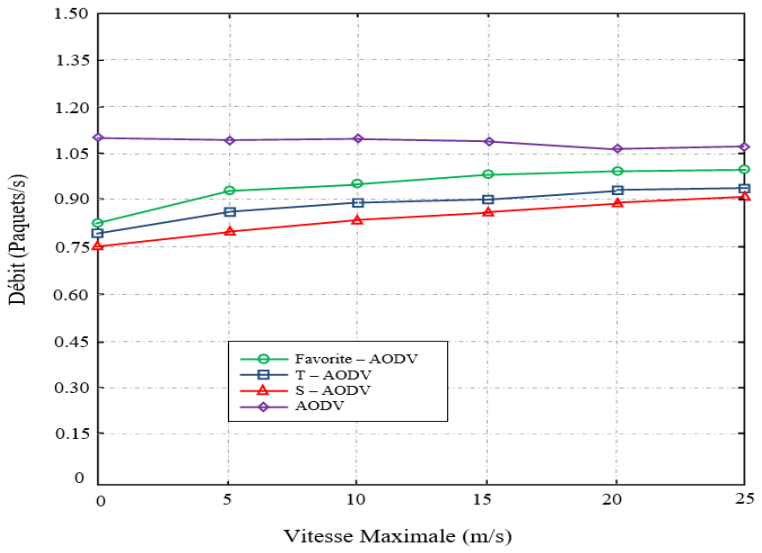


Figure 7.18 : Variation du Débit en fonction de la vitesse.

3.2.2.5 *Optimalité du chemin "Path Optimality"*

L'optimalité du chemin est la proportion du nombre total de sauts dans le chemin le plus court par rapport à celle des sauts dans le chemin sélectionné par les paquets de données. Ce paramètre est mesuré par l'équation suivante :

$$Optimalité\ du\ chemin = \frac{Nombre\ de\ sauts\ dans\ le\ chemin\ le\ plus\ court}{Nombre\ de\ sauts\ dans\ le\ chemin\ sélectionné} \qquad (7.6)$$

À travers les figures (7.19 et 7.20), le protocole AODV présente la meilleure optimalité du chemin (85 % − 68 %) lorsqu'il n'y a pas de nœuds malveillants. À mesure que les nœuds malveillants

augmentent, l'optimalité du chemin des quatre protocoles diminue. Généralement, l'optimalité du chemin dans S-AODV et T-AODV est inférieure que celui dans AODV et la valeur de Favorite-AODV est la plus petite.

Notons que, S-AODV et T-AODV sont capables de détecter et de filtrer les nœuds malveillants. Cependant, Favorite-AODV consiste à obtenir une valeur de confiance plus précise pour les nœuds, ce qui nécessite des recherches plus poussée afin de trouver les chemins de confiance optimaux vers la destination.

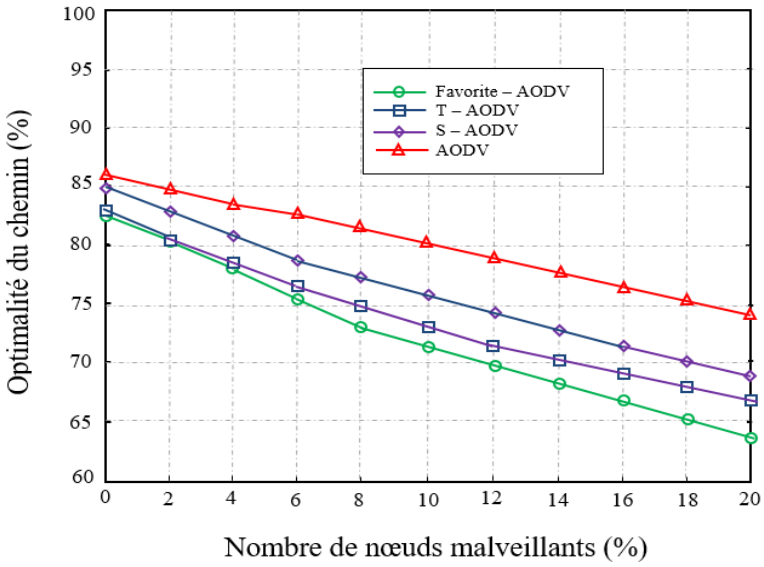


Figure 7.19 : Optimalité du chemin versus nœuds malveillants.

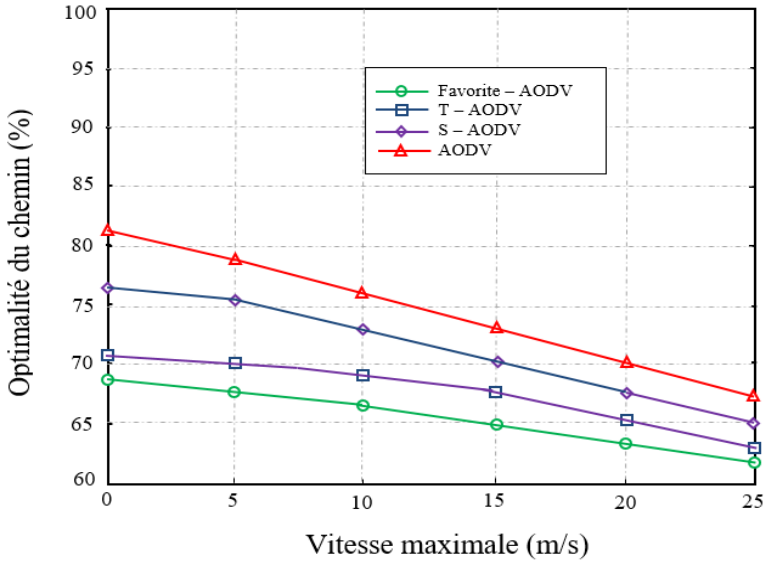


Figure 7.20 : Optimalité du chemin en fonction de la vitesse.

3.2.2.6 *Consommation moyenne d'énergie "AEC"*[15]

En particulier, l'efficacité énergétique peut être le critère de conception le plus important pour les MANETs puisque les nœuds mobiles sont alimentés par des batteries de capacité limitée. La panne d'énergie d'un nœud mobile n'affecte pas seulement le nœud lui-même, mais aussi sa capacité à transmettre des paquets pour le compte d'autrui et donc la durée de vie globale du réseau. Lorsque les nœuds mobiles du réseau partagent les services de transmission réseau, ils fournissent également des services de transmission réseau à d'autres nœuds. Notez que le coût énergétique du calcul des valeurs de confiance n'est pas pris en compte dans ce travail. Par conséquent, la métrique d'AEC est introduite pour montrer le rapport entre la consommation d'énergie et les services offerts par le réseau à travers la couche d'application. Cette métrique est définie comme suit :

$$ACE_1 = \frac{\sum_{i=1}^{m} Nconsommer_i}{\sum_{i=1}^{m}(Nreçus_i + émis_i)} \quad (N = nœuds\ normaux) \tag{7.7}$$

$$ACE_2 = \frac{\sum_{i=1}^{n} Nconsommer_i}{\sum_{i=1}^{n}(Nreçus_i + émis_i)} \quad (N = nœuds\ égoïstes) \tag{7.8}$$

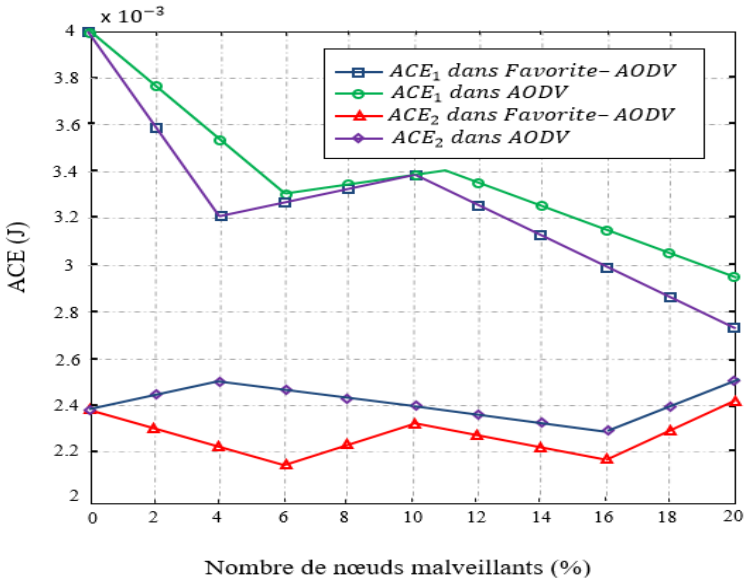


Figure 7.21 : Comparaison de l'AEC avec des nœuds malveillants.

Comme le montre la figure 7.21, nous avons des nœuds malveillants qui varient entre 0 % et 20 % parmi les nœuds mobiles du réseau, tandis que les autres nœuds du réseau se comportent correctement. Parce que les nœuds malveillants ne participant pas à la phase de découverte de la route dans notre protocole ou ne pouvant pas exécuter correctement le transfert des paquets de données, il est évident que l'AEC des nœuds malveillants est inférieure à celle des autres nœuds normaux.

[15] Average Energy Consume.

Les résultats expérimentaux montrent que, même si des nœuds égoïstes individuels affectent sérieusement les performances du réseau, le mécanisme de confiance proposé reste actif afin d'inviter et d'encourager les autres nœuds à transmettre les paquets de données. Selon les résultats de la simulation, notre modèle de confiance peut générer des économies d'énergie et prolonger la durée de vie du réseau. Il est nécessaire alors que le schéma de sécurité adapté aux comportements égoïstes impose l'exécution du transfert de paquets de données sur les protocoles de routage dans les MANETs.

3.2.3 Performance du système de détection

3.2.3.1 Taux de détection "Detection Ratio"

Pour montrer l'efficacité de notre système de détection, nous définissons le ratio de détection, qui est correspondant au nombre de voisins d'un nœud malveillant détectant un comportement frauduleux divisé par le nombre total des voisins de ce nœud. Cette métrique montre combien de nœuds dans le voisinage d'un nœud malveillant peuvent découvrir l'action malveillante [35].

$$Taux\ de\ détection\ (N) = \frac{Nombre\ de\ voisins\ de\ "N"\ détectant\ attaque}{Nombre\ totale\ des\ voisins\ de\ "N"} * 100\ (\%) \qquad (7.9)$$

D'autres formules existent pour mesurer le taux de détection tel que "le nombre d'attaques détectées par au moins un nœud divisé par le nombre total d'attaques". Les résultats obtenus par ces formules sont plus généreux en termes de pourcentage de détection que celle que nous avons adopté. Nous justifions ce choix par le fait que le ratio choisi nous semble plus pertinent par rapport aux autres.

Nous présentons dans un premier temps l'analyse liée à la métrique choisie pour les attaques élémentaires. Ensuite, nous présentons celle correspondant aux attaques complexes. Nous terminons par une discussion concernant les fausses alertes (les faux-positifs).

3.2.3.2 Résultats concernant les attaques élémentaires

Les figures (7.22 et 7.23) montrent l'évolution du taux de détection en fonction du nombre de nœuds. Les résultats montrent un taux de détection supérieur à 90 % pour les attaques contre les demandes de route RREQ. Après analyse, nous avons établi que les cas de non-détection sont dus essentiellement à la mobilité des nœuds. Notamment, les cas de non-détection sont dus au partitionnement du réseau. En effet, les nœuds du réseau sont constamment en mouvement et le système de détection se base sur les observations collectées à propos des nœuds voisins : lorsqu'un nœud malveillant arrive pour la première fois dans le voisinage, il peut ne pas être détecté.

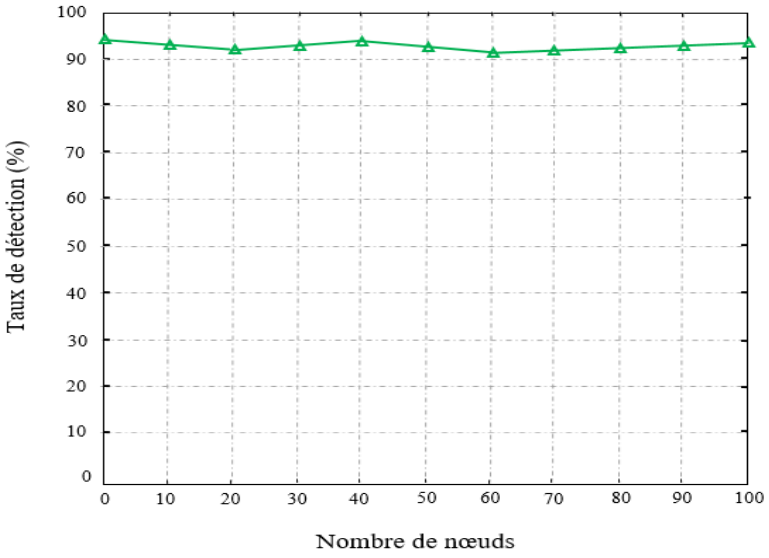


Figure 7.22 : Rejeu de RREQ.

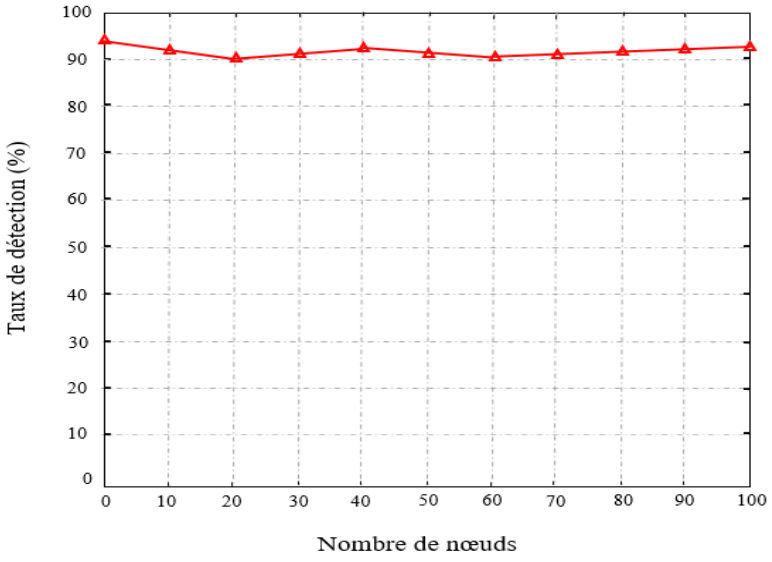


Figure 7.23 : Modification du "$Source_{\_Seq\#}$" dans une RREQ.

Les figures (7.24 et 7.25) présentent l'évolution du taux de détection pour les attaques sur les réponses de route RREP. Pour ces attaques, le taux de détection varie entre 58 % et 95 %. Si les cas de non-détection sont encore une fois dus à la mobilité, la baisse du taux de détection par rapport aux attaques contre les RREQ est ici essentiellement dû à la non-diffusion des RREP dans le réseau qui, sont envoyées en unicast. Ainsi, seuls les nœuds ayant envoyé précédemment la réponse de route ou ceux ayant écouté la transmission du voisin sauront détecter un comportement malveillant.

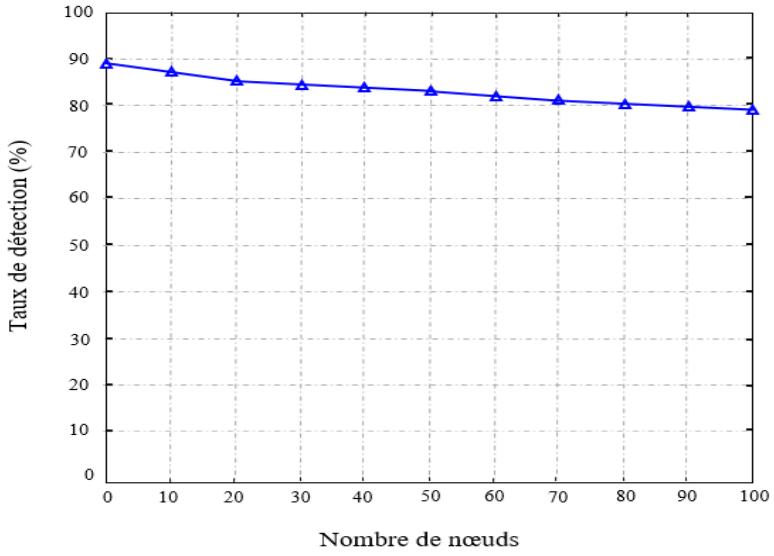


Figure 7.24 : Rejeu de REPP.

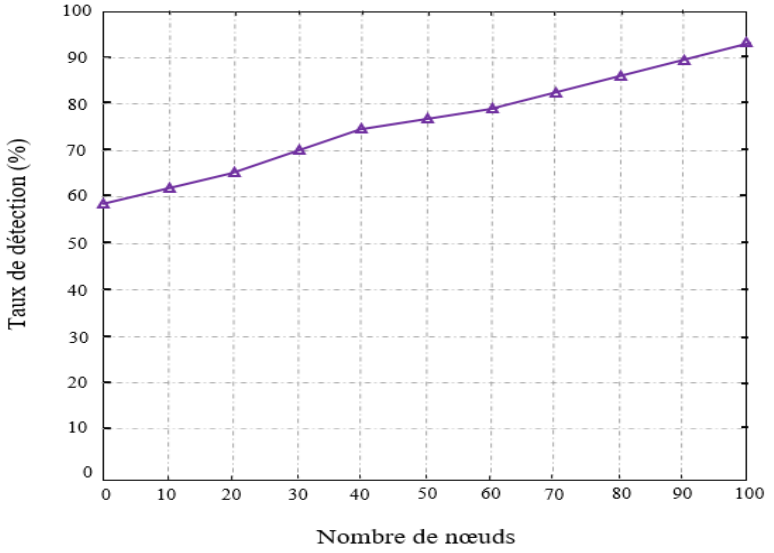


Figure 7.25 : Modification du "$Des\_{seq\#}$" dans une RREP.

3.2.3.3 Résultats concernant les attaques complexes

Dans les figures (7.26 et 7.27), nous notons que le taux de détection est relativement faible. Ceci est dû au fait que plus le réseau est dense, plus le nœud malhonnête a des voisins susceptibles de détecter son comportement. De nouveau, les cas de non-détection sont essentiellement dus au fait que ces attaques reposent exclusivement sur les réponses de routes RREP qui sont moins disséminées dans le réseau par rapport aux demandes de route RREQ.

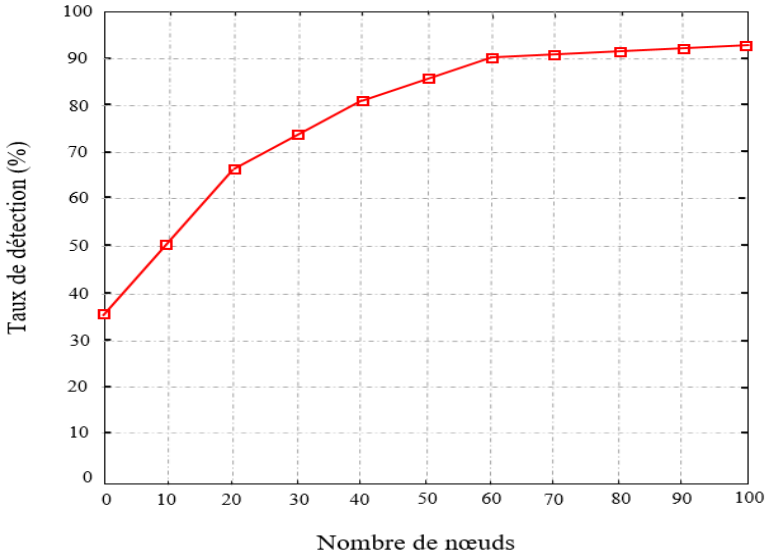


Figure 7.26 : Attaque par fabrication "a".

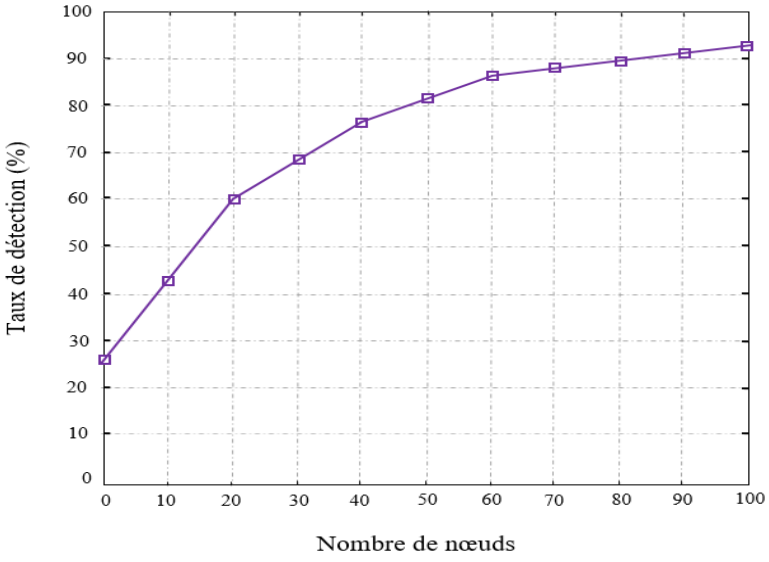


Figure 7.27 : Attaque par fabrication "b".

3.2.3.4 *Résultats concernant les faux − positifs*

Au cours de l'analyse des simulations précédemment effectuées, nous avons noté des cas de faux-positifs : il s'agit des nœuds qui sont placés dans la liste noire (voir chapitre 6, section 3.1) alors qu'ils sont honnêtes. Nous calculons ainsi le taux de faux-positifs (TFP) qui correspond au nombre de nœuds qui apparaissent au moins une fois dans la liste des suspects d'au moins un nœud honnête sur le total des nœuds honnêtes.

$$TFP = \frac{Nombre\ de\ nœuds\ honnêtes\ qui\ apparaissent\ dans\ la\ liste\ noire}{Total\ des\ nœuds\ honnêtes} \qquad (7.10)$$

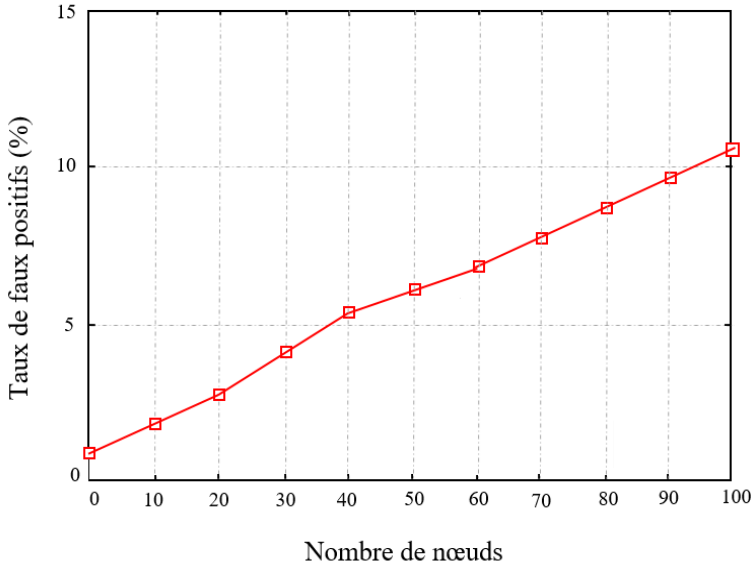


Figure 7.28 : TFP pour l'attaque par modification du "$Source_{Seq\#}$" dans une RREQ.

La figure 7.28 présente la variation du taux de faux-positifs obtenus pour l'attaque par modification du "$Source_{Seq\#}$" dans une RREQ. Il s'agit de la seule attaque où nous avons eu d'aussi grands taux.

Nous avons identifié plusieurs causes à ces faux positifs :

- Le partitionnement du réseau peut induire les nœuds en erreur : certains nœuds suspectent de manière erronée des nœuds innocents qui viennent d'entrer dans leur voisinage. En effet, ces nœuds se déplacent d'une partition à une autre ce qui fait qu'ils n'ont pas la connaissance nécessaire pour juger de la culpabilité ou de l'innocence des nœuds formant leur nouveau voisinage. Étant donné que ces nouveaux arrivants sont honnêtes, ils se comportent normalement et retransmettent les paquets reçus, y compris ceux de l'attaquant, au risque d'être eux même détectés comme malhonnêtes.

- Dans une moindre mesure, les réémissions dues aux collisions peuvent dans certaines situations être détectées incorrectement comme étant des rejeux. Remarquons que ces collisions sont plus nombreuses dans des réseaux denses. Il est important de noter que durant les expérimentations, les nœuds détectés et ajoutés à la liste noire ont été bannis pour une courte période de temps par les nœuds les ayant détectés afin de ne pas perturber le déroulement normal du protocole.

4. CONCLUSION

Tout au long de ce chapitre, nous avons présenté les détails de l'implémentation qui nous ont permis de mettre en place des simulations pour tester l'applicabilité de notre solution. Après la présentation

de l'environnement de simulation, nous avons analysé et évalué les performances de nos contributions en termes de sécurité à travers la description du cheminement des paquets entre les différentes fonctions du système. L'analyse des résultats obtenus après l'exécution des simulations a porté sur trois axes principaux :

Le premier concerne la mise à l'épreuve du notre modèle de confiance. L'approche proposée définit un vecteur de poids pour chacun des paramètres d'entrée, ceci offre un avantage significatif pour notre modèle, car elle peut détecter non seulement un comportement égoïste ou anormal, mais peut également aider à identifier le type de paramètres utilisés dans la stratégie des nœuds attaquants ou des nœuds égoïstes. Les résultats de simulation présentés montrent que le cadre proposé peut montrer clairement la différence entre les valeurs de confiance d'un nœud normal et d'un nœud égoïste sur un paramètre spécifique en définissant un vecteur de poids approprié. De plus, la valeur de confiance globale est calculée en utilisant des facteurs de relation et des poids de nœuds voisins.

Le second montre que le protocole proposé ($Favorite - AODV$) fournit une approche souple et réalisable pour choisir un chemin de routage fiable et sécurisé parmi tous les chemins d'accès, en utilisant la contrainte de confiance. Les résultats expérimentaux obtenus en termes d'amélioration des performances confirment l'efficacité de notre protocole par apport aux travaux proposés dans la littérature. Le mécanisme de sécurité proposé est également plus flexible et évolutif pour son application à grandes échelles.

Le troisième axe teste l'efficacité de notre système de détection. Nous calculons le taux de détection en déterminant le nombre de voisins détectant une attaque par rapport à la totalité des voisins. Les résultats obtenus permettent d'affirmer l'efficacité du système de détection que nous avons mis en place.

Conclusion & Perspectives

Conclusion Générale

Établir des communications sécurisées au sein d'un réseau Ad hoc est un véritable challenge. En effet, un réseau Ad hoc est un environnement hostile, qui apporte plusieurs défis de sécurité, dus à ses caractéristiques et ses spécificités (liens sans fil, capacités limitées, etc.). Dans de tels réseaux, les échanges entre les nœuds s'effectuent de manière dynamique et sans infrastructure fixe. Les nœuds sont donc autonomes et doivent à partir des seules informations localement accessibles prendre de façon sûre la décision d'interagir ou non. De plus, ces nœuds doivent être en mesure de communiquer dans des situations imprévues et avec des nœuds potentiellement hostiles. Il est donc nécessaire que ces nœuds s'appuient sur un modèle de sécurité qui intègre des mécanismes contrant les attaques actives, qui incite à la coopération entre les nœuds, et qui soit en mesure de détecter les comportements malveillants ou défaillants.

Au cours de ce travail, nous avons passé en revue les différentes solutions actuelles de sécurité pour les réseaux Ad hoc en mettant l'accent sur les vulnérabilités, et les solutions de sécurité correspondantes, à savoir :

- Les architectures de gestion de clés.
- Protections utilisant la cryptographie symétrique.
- Protections utilisant la cryptographie asymétrique.
- Protections contre la modification des données.
- Protection contre les attaques de type "tunnel".
- Systèmes à base de conservation de flot.
- Régimes à base d'acquittements.
- Mécanismes basés sur la réputation.
- Systèmes de détection d'intrusion.

À travers ces investigations, nous avons vu que chaque technologie pose ses propres challenges à la conception des solutions de sécurité, mais aucune de ces méthodes ne prétend la toute exhaustivité, ou encore résoudre complètement le problème de la sécurité dans les réseaux Ad hoc, chacune de ces solutions possède ses avantages, et ses inconvénients et chacune s'adapte mieux à un type particulier. Toute la difficulté réside dans la conception des solutions de sécurité qui pourraient répondre à ces challenges, et non seulement d'assurer la robustesse face à des attaques potentielles ou de veiller à ne pas ralentir les communications, mais aussi d'optimiser l'utilisation des ressources en termes de bande passante, de mémoire et de batterie. Le plus important dans ce contexte ouvert, est de garantir

l'anonymat et le secret de la vie privée, tout en permettant la traçabilité pour des raisons légales. En effet, le besoin croissant de traçabilité est aujourd'hui nécessaire pour la lutte contre les organisations de malfaiteurs et de terroristes, ainsi que pour minimiser le pillage des droits d'auteurs. Tous ces éléments influent dans le choix et la mise en place des outils de sécurité qui sont guidés par une évaluation du risque préalable.

Le modèle d'établissement et de gestion de la confiance que nous avons développé [11, 123, 127, 138, 139] est basé sur le concept de la confiance humaine en fournissant aux nœuds les mécanismes nécessaires pour évaluer le niveau de confiance de leurs voisins. L'idée fondamentale est basée sur la combinaison des expériences précédentes et les recommandations des autres voisins pour évaluer le niveau de confiance des nœuds du réseau. Cette solution, qui semble adaptée aux réseaux Ad hoc et à leur topologie dynamique, peut contrer les attaques lors de la phase de découverte du routage et de la phase de transmission des données.

Afin d'établir notre nouveau modèle de gestion de la confiance, nous avons présenté le concept de la théorie des ensembles flous qui permet de quantifier les données imprécises ou de l'incertitude dans les mesures de l'indice de sécurité des nœuds Ad hoc. Nous avons également introduit la méthode de la théorie d'analyse relationnelle grise, dont l'idée essentielle consiste à sélectionner certaines variables d'entrée qui montrent un impact plus fort à la sortie du système afin de prendre la bonne décision. En conséquence, un protocole de routage de confiance basé sur ce modèle avec l'extension du protocole AODV a été proposé, dont le but est d'assurer un chemin sécurisé de bout en bout exempt de tout nœuds malveillants. Ce protocole fournit une approche flexible et réalisable afin de choisir le meilleur itinéraire parmi tous les chemins candidats, en utilisant les nombres de sauts ainsi que la valeur globale de la confiance en tant que valeur de contrainte.

Enfin, nous avons validé l'exactitude et l'efficacité de notre protocole à travers l'exécution de plusieurs simulations. L'analyse des résultats obtenus a porté sur trois axes essentiels qui sont : évaluation de notre modèle de confiance, évaluation des performances du protocole et du système de détection. Les résultats de simulation montrent que le protocole Favorite-AODV est capable de surperformer efficacement les nœuds non fiables tout en obtenant un itinéraire de livraison fiable. Une amélioration est également illustrée, en comparant ses performances avec les protocoles S-AODV, T-AODV et AODV en termes du taux de livraison des paquets, du délai moyen de bout en bout, du volume de trafic de contrôle, du débit du réseau ainsi que le taux de détection. À travers cette simulation, nous avons démontré que le raisonnement basé sur la confiance permet d'assurer le bon fonctionnement des opérations de routage et de garantir la sécurité des communications dans le réseau.

Le travail que nous avons accompli dans cette thèse, ouvre des perspectives laissant envisager un certain nombre d'études complémentaires dans les directions suivantes :

- Il nous parait intéressant de porter ce raisonnement qui repose principalement sur la confiance vers d'autres protocoles réactifs. Nous proposons ensuite de créer un modèle générique pour les protocoles réactifs, auquel il faut associer un ensemble de règles de confiance. Il suffit ensuite de trouver un mappage entre le modèle générique et le protocole lui-même. Ce qui permettra d'adapter le raisonnement à n'importe quel protocole réactif.

- Nous pensons aussi que les mécanismes déployés ne doivent pas se limiter à la couche réseau, mais s'étendre au niveau physique et à la couche MAC (accès au médium) ou encore à une solution inter couches ou 'Cross-layer'. Ainsi, il serait intéressant de mettre en place des mécanismes permettant de suivre l'exécution des protocoles de routage en vue de comparer les messages échangés entre les nœuds du réseau.

- La mise en place d'une politique de gestion des mesures prises à l'encontre des nœuds malveillants peut contribuer dans l'amélioration des performances de détection.

- Nous pouvons aussi envisager d'étendre la connaissance de chaque nœud en stockant localement une vision propre de la table de routage et de la table d'historique des voisins. Il faudra alors démontrer l'utilité de telles informations et justifier leurs ajouts par des règles permettant la détection de nouvelles actions malveillantes.

Postface

Ce travail a été effectué au laboratoire LaSTIC (Laboratoire des Systèmes et des Technologies de l'Information et de la Communication) du département d'informatique de la Faculté des Mathématiques et de l'informatique de l'Université Batna 2, sous la direction de Professeur Azeddine Bilami (directeur du laboratoire).

Ce travail a fait l'objet des publications scientifiques suivantes :

- Conférence Internationale: "RTIC: Reputation and Trust Evaluation Based on Fuzzy LogIC System for Secure Routing in Mobile Ad Hoc Networks ". Abdesselem Beghriche and Azeddine Bilami, Communications in Computer and Information Science, Series of Springer LNCS, Volume 293, Networked Digital Technologies, NDT 2012, Part 8, pp. 620-634.

- Article: "AFIM: A Trusted Routing Algorithm Based on Fuzzy LogIc for Mobile Ad hoc Networks". Abdesselem Beghriche and Azeddine Bilami, AWER Procedia Information Technology & Computer Science, Vol. 03, pp. 1400-1404, 2013.

- Conférence Internationale : "Un modèle de Sécurité basé sur la Confiance Floue pour Assurer la QoS dans les Réseaux Mobiles Ad hoc". Abdesselem Beghriche & Azeddine Bilami, proceeding of 1st International Conference on Artificial Intelligence and Information Technology, Ouargla_Algeria, 2014.

- Article: "A Fuzzy trust-based routing model for mitigating the misbehaving nodes in mobile Ad hoc networks". Abdesselem Beghriche and Azeddine Bilami, International Journal of Intelligent Computing and Cybernetics, Vol. 11, No. 2, pp. 309-340, 2018.

Bibliographie

[Numéro de référence] Auteur(s), "Titre", *Maison d'édition*, Numéro de volume, Numéro d'issue, Année, pages.

[1] S. Marti, T. J. Giuli, K. Lai and M. Baker, "Mitigating routing misbehavior in mobile Ad-hoc networks", *In Proceedings of 6th Annual International Conference Mobile Computing and Networking (MobiCom). ACM Press*, 2000, pp. 255-265.

[2] J. P. Hubaux, L. Buttyán and S. Capkun, "The Quest for security in mobile Ad hoc Networks", *In Proceedings of the 2nd ACM international symposium on mobile Ad hoc Networking and Computing (Mobihoc'01)*, 2001, pp. 146-155.

[3] M. G. Zapata and N. Asokan, "Secure Ad-hoc on-demand distance vector routing", *ACM SIGMOBILE Mobile Computing and Communications Review*, Vol. 3, No. 6, 2002, pp. 106-107.

[4] P. Khatri, "Using identity trust with key management for achieving security in Ad-hoc Networks", *In IEEE International Advance Computing Conference (IACC)*, 2014, pp. 271-275.

[5] H. Y. Lin, M. Y. Hsieh and K. C. Li, "Flexible group key management and secure data transmission in mobile device communications using elliptic curve Diffie-Hellman cryptographic system", *International Journal of Computational Science and Engineering (IJCSE)*, Vol. 12, No. 1, 2016, pp. 47-52.

[6] P. B. Velloso, R. P. Laufer, D. O. Cunha, O. C. M. B. Duarte and G. Pujolle, "Trust Management in Mobile Ad-hoc Networks Using a Scalable Maturity-Based Model", *IEEE Transactions on Network and service management*, Vol. 7, No. 3, 2010, pp. 172-185.

[7] H. Xia, J. Yu, Z. k. Pan, X. G. Cheng and E. H. M. Sha, "Applying trust enhancements to reactive routing protocols in mobile Ad-hoc networks", *Wireless Networks*, 2015, pp. 1-19.

[8] A. Lupia and F. De Rango, "Performance evaluation of secure AODV with trust management under an energy aware perspective", *In International Symposium on Performance Evaluation of Computer and Telecommunication Systems, (SPECTS 2014)*, 2014, pp. 599-606.

[9] F. De Rango and S. Marano, "Trust-based S-AODV protocol with intrusion detection and incentive cooperation in MANET", *In Proceedings of the 2009 ACM International Wireless Communications and Mobile Computing: Connecting the World Wirelessly (IWCMC '09)*, 2009, pp. 1443-1448.

[10] H. Xia, Z. Jia, L. Ju, X. Li and Y. Zhu, "A Subjective Trust Management Model with Multiple Decision Factors for MANET based on AHP and Fuzzy Logic Rules", *In IEEE/ACM International Conference on Green Computing and Communications*, 2011, pp. 124-130.

[11] A. Beghriche and A. Bilami, "RTIC: Reputation and Trust Evaluation Based on Fuzzy LogIC System for Secure Routing in Mobile Ad-hoc Networks", *In Networked Digital Technologies, Communications in Computer and Information Science Series of Springer LNCS,* 2012, Vol. 293, pp. 620-634.

[12] Organisation Mondiale de la Propriété Intellectuelle (OMPI), Convention de Berne pour la protection des œuvres littéraires et artistiques, 9 Septembre 1886. [Consulté le 09 avril 2016]. Accessible sur : http://www.wipo.int/treaties/fr/ip/berne/summary_berne.html

[13] Académie Française, "Dictionnaire de la langue française", 8ème édition, 1932-1935.

[14] H. Chaouchi et M. Laurent-Maknavicius, "La sécurité dans les réseaux sans fil et mobiles 1, concepts fondamentaux", *Lavoisier*, Paris-France, 2007, 239.

[15] J. Hallberg, A. Hunstad and M. Peterson, "A framework for system security Assessment", *In Proceedings from the Sixth Annual IEEE SMC on Information Assurance (IAW '05),* West Point, New-York, 2005.

[16] UIT-T X.800, "Réseaux de communication de données : interconnexion de systèmes ouverts (OSI), sécurité, structure et applications", *Architecture de sécurité pour l'interconnexion en systèmes ouverts d'applications du CCITT*, 1991. [Consulté le 17 septembre 2016]. Accessible sur : https://www.itu.int/rec/T-REC-X.800/fr

[17] Common Criteria, "Common Criteria for Information Technology Security Evaluation", part 1: Introduction and general model, Part 2: Security functional components, Part 3: Security assurance components, version 3.1, September 2012.

[18] C. Burgod, "Contribution à la sécurisation du routage dans les réseaux ad hoc", *Thèse de doctorat, Université de Limoges (France)*, Octobre 2009.

[19] S. Corson and J. Macker, "Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations", *RFC Editor,* United States, 1999.

[20] J. A. Freebersyser and B. Leiner, "A DoD perspective on mobile Ad hoc networks", *Ad Hoc Networking,* Addison-Wesley Longman Publishing, Inc. Boston, MA, USA, 2001, pp. 29-51.

[21] F. Theoleyre, "Une auto-organisation et ses applications pour les réseaux Ad hoc et hybrides", *Thèse de doctorat, Institut national des sciences appliquées de Lyon-France*, Septembre 2006.

[22] S. Chen and K. Nahrstedt, "A distributed quality-of-service routing in ad-hoc networks", *IEEE Journal on Selected Areas in Communications*, Vol. 17, No. 8, 2006, pp. 1488-1505.

[23] C. Chaudet, I. Gu and E. Lassous, "BRuIT: Bandwidth Reservation under InTerferences influence", *In Proceedings of European Wireless (EW 2002)*, Florence, Italy, 2002, pp. 466-472.

[24] R. E. Kahn, "The organization of computer resources into a packet radio network", *In AFIPS'75 National Computer Conference and exposition*, Vol. 44, 1975, pp. 177-186.

[25] D. Estrin, R. Govindan, J. Heidemann and S. Kumar, "Next century challenges: Scalable coordination in sensor networks", *In Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking (MobiCom '99)*, 1999, pp. 263-270.

[26] R. Morris, J. Jannotti, F. Kaashoek, J. Li and D. Decouto, "CarNet: a scalable ad hoc wireless network system", *In Proceedings of the 9th workshop on ACM SIGOPS European workshop*, 2000, pp. 61-65.

[27] M. Weiser, "Some computer science issues in ubiquitous computing", *Communications of the ACM - Special issue on computer augmented environments*, Vol. 36, No. 7, 1993, pp. 75-84.

[28] S. Mann, "Smart clothing: The shift to wearable computing", *Communications of the ACM*, Vol. 39, No. 8, 1996, pp. 23-24.

[29] ReseauCitoyen.be. [Consulté le 02 Mai 2016]. Accessible sur : http://reseaucitoyen.be/

[30] Association Lille Sans Fil. [Consulté le 02 Mai 2016]. Accessible sur : http://www.lillesansfil.org

[31] C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers", *In Proceedings of the conference on Communications architectures, protocols and applications (SIGCOMM'94),* 1994, pp. 234–244.

[32] T. Clausen and P. Jacquet, "Optimized Link State Routing Protocol OLSR", October 2003. [Consulté le 12 Mars 2016]. Accessible sur : https://tools.ietf.org/html/rfc3626

[33] R. Ogier, F. Templin and M. Lewis, "Topology dissemination based on reverse-path forwarding (TBRPF)", February 2004. [Consulté le 15 Mars 2016]. Accessible sur : https://tools.ietf.org/html/rfc3684

[34] D. P. Bertsekas and R. G Gallager, "Data Networks", *(2nd edition) Prentice Hall*, 1992. [Consulté le 01 Juillet 2016]. Accessible sur : http://web.mit.edu/dimitrib/www/datanets.html

[35] M. A. Ayachi, "Contributions à la détection des comportements malhonnêtes dans les réseaux ad hoc AODV par analyse de la confiance implicite", *Thèse de doctorat, Université de Rennes 1 (France)*, Février 2011.

[36] D. Sidhu, T. Fu, S. Abdallah, R. Nair and R. Coltun, "Open shortest path first (OSPF) routing protocol simulation" *ACM SIGCOMM Computer Communication Review*, Vol. 23, No. 4, 1993, pp. 53-62.

[37] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing". *In Proceedings of second IEEE Workshop on Mobile Computing Systems and Applications (WMCSA'99)*, Vol. 2, 1999, pp. 90-100.

[38] C. E. Perkins, E. Belding-Royer and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing", July 2003. [Consulté le 28 Mars 2016]. Accessible sur : https://tools.ietf.org/html/rfc3561

[39] D. B. Johnson and D. A. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks", *Mobile Computing, Kluwer International Series in Engineering and Computer Science*, Vol. 353, 1996, pp. 153-181.

[40] Z. J. Haas, M. R. Pearlman and P. Samar, "The Zone Routing Protocol (ZRP) for Ad Hoc Networks", July 2002, [Consulté le 01 Juillet 2016]. Accessible sur : https://tools.ietf.org/html/draft-ietf-manet-zone-zrp-04

[41] INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS, "IEEE 802.11™ WIRELESS LOCAL AREA NETWORKS", The Working Group for WLAN Standards. [Consulté le 01 Juillet 2016]. Accessible sur : http://www.ieee802.org/11/

[42] H. Chaouchi et M. Laurent-Maknavicius, "La sécurité dans les réseaux sans fil et mobiles 3, technologies émergentes", *Lavoisier*, Paris-France, 2007, 290.

[43] Z. Wei, F. R. Yu and A. Boukerche, "Cooperative Spectrum Sensing with Trust Assistance for Cognitive Radio Vehicular Ad hoc Networks", *In Proceedings of the 5th ACM Symposium on Development and Analysis of Intelligent Vehicular Networks and Applications (DIVANet '15)*, 2015, pp. 27-33.

[44] G. Vigna, S. Gwalani, K. Srinivasan, E. M. Belding-Royer and R. A. Kemmerer, "An intrusion detection tool for AODV-based ad hoc wireless networks", *In Proceedings of the 20th Annual Computer Security Applications Conference (ACSAC'04), IEEE Computer Society,* 2004, pp. 16–27.

[45] Y, Ping, H. Yafei, Z. Yiping, Z. Shiyong and D. Zhoulin, "Flooding attack and defense in ad hoc networks", *Journal of Systems Engineering and Electronics*, Vol. 17, No. 2, 2012, pp. 410-416.

[46] L. Guang, C. Assi and A. Benslimane, "Interlayer Attacks in Mobile Ad hoc Networks", *In Proceedings of the 2nd International Conference on Mobile ad-hoc and sensor networks (MSN 2006)*, 2006, pp. 436-448.

[47] I. Stamouli, P. G. Argyroudis and H. Tewari, "Real-time intrusion detection for ad hoc networks", *In Proceedings of the 6th IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks (WoWMoM'05), IEEE Computer Society*, 2005, pp. 374-380.

[48] F. Stajano and R. Anderson, "The resurrecting duckling: Security issues in ad hoc wireless networks", *In Proceedings of the 7th International Workshop on Security Protocols*, 1999, pp. 172-194.

[49] S. Gupte and M. Singhal, "Secure routing in mobile wireless ad hoc networks", *Ad Hoc Networks*, Vol. 1, No. 1, 2003, pp. 151-174.

[50] P. Obreiter, B. Konig-Ries and M. Klein, "Stimulating cooperative behavior of autonomous devices - an analysis of requirements and existing approaches", *In Proceedings of 2nd International Workshop on Wireless Information Systems (WIS2003)*, 2003, pp. 71-82.

[51] H. Deng, W. Li and D. P. Agrawal, "Routing security in wireless ad hoc networks", *IEEE Communications Magazine*, Vol. 40, No. 10, 2002, pp. 70-75.

[52] T. Condie, V. Kacholia, S. Sankararaman, J. Hellerstein and P. Maniatis, "Induced churn as shelter from routing-table poisoning", *In Proceedings of the 13th Annual Network and Distributed System Security Symposium (NDSS'06)*, 2006.

[53] Y. C. Hu, A. Perrig and D. B. Johnson, "Rushing attacks and defense in wireless ad hoc network routing protocols", *In Proceeding of the 2nd ACM workshop on Wireless security (WiSe'03)*, 2003, pp. 30-40.

[54] Y. C. Hu, A. Perrig and D. B. Johnson, "Wormhole attacks in wireless networks", *IEEE Journal on Selected Areas in Communications,* Vol. 24, No. 2, 2006, pp. 370-380.

[55] K. Sanzgiri, B. Dahill, B. N. Levine, C. Shields and E. M. Belding-Royer, "A Secure Routing Protocol for Ad Hoc Networks", *In Proceedings of 10th IEEE International Conference on Network Protocols (ICNP'02), IEEE Computer Society*, 2002. pp. 78-89.

[56] C. Y. Tseng, P. Balasubramanyam, C. Ko, R. Limprasittiporn, J. Rowe and K. Levitt, "A specification-based intrusion detection system for AODV", *In Proceedings of 1st ACM workshop on Security of ad hoc and sensor networks (SASN'03)*, 2003, pp. 125-134.

[57] V. Akhila and M. E. Meghashri, "Preventing Sybil Attack in Ad-Hoc Networks Using Secret Key Technique", *International Journal of Computer Science and Information Technology Research*, Vol. 3, No. 3, 2015, pp. 325-330.

[58] N. Thakur and A. Sankaralingam, "Introduction to Jamming Attacks and Prevention Techniques using Honeypots in Wireless Networks", *International Journal of Computer Science and Information Technology & Security (IJCSITS)*, Vol. 3, No. 2, 2013, pp. 202-207.

[59] V. Gupta, S. Krishnamurthy and M. Faloutsos, "Denial of service attacks at the mac layer in wireless ad hoc networks", *In Proceedings of MILCOM*, Vol. 2, 2002, pp. 1118-1123.

[60] A. Rachedi, "Contributions à la sécurité dans les réseaux mobiles ad hoc", *Thèse de doctorat, Université de d'Avignon et des Pays de Vaucluse (France),* Novembre 2008.

[61] S. Chokhani, W. Ford, R. Sabett, C. Merrill and S. Wu, "Internet X.509 public key infrastructure certificate policy and certification practices framework". *In acts of Network Working Group Request for Comments: (RFC 3647)*. 2003. [Consulté le 08 septembre 2016]. Accessible sur : https://www.ietf.org/rfc/rfc3647.txt

[62] S. Kent and R. Atkinson, "IP Encapsulating Security Payload (ESP)", *In acts of Network Working Group Request for Comments: (RFC 2406)*, 1998. [Consulté le 17 Septembre 2016]. Accessible sur : https://tools.ietf.org/html/rfc2406

[63] B. Schneier, "Cryptographie Appliquée", *Vuibert Informatique*, Paris-France, 1997, 846.

[64] G. Labouret, "Introduction à la cryptographie", Hervé Schauer Consultants, 1999-2000. [Consulté le 05 Mai 2016]. Accessible sur : www.hsc.fr/ressources/cours/crypto/crypto.pdf

[65] S. Miner and J. Staddon, "Graph-Based Authentication of Digital Streams", *In Proceedings of the IEEE Symposium on Security and Privacy*, 2001, pp. 232-246.

[66] D. Coppersmith and M. Jakobsson, "Almost Optimal Hash Sequence Traversal", *In Proceedings of the 6th international conference on Financial cryptography (FC'02)*, Volume 2357 of the series Lecture Notes in Computer Science, 2002, pp 102-119.

[67] M. Jakobsson, "Fractal hash sequence representation and traversal", *In Proceedings of the IEEE International Symposium on Information Theory (ISIT'02)*, 2002, pp. 437-444.

[68] A. Tsirigos and Z. Haas, "Multipath routing in mobile ad hoc networks or how to route in the presence of frequent topology changes", *In Proceedings of IEEE MILCOM*, Vol. 2, 2001.

[69] S. Lee, B. Han and M. Shin, "Robust Routing in Wireless Ad Hoc Networks", *Computer Science Department, University of Maryland,* 2002.

[70] F. Bennett, D. Clarke, J. Evans, A. Hopper, A. Jones and D. Leask, "Piconet Embedded Mobile Networking", *IEEE Personnel Communications*, Vol. 4, No. 5, 1997, pp. 08-15.

[71] G. Montenegro and C. Castelluccia, "Statistically unique and cryptographically verifiable identifiers and addresses", *In Proceedings of ISOC, Symposium on network and Distributed System Security (NDSS 2002)*, 2002.

[72] L. Zhou and Z. J. Haas, "Securing Ad hoc networks", *IEEE Network: The Magazine of Global Internetworking*, Vol. 13, No. 6, 1999, pp. 24-30.

[73] A. Perrig, R. Canetti, J. D. Tygar and D. X. Song, "Efficient Authentication and Signing Multicasts Streams over Lossy Channels", *In IEEE Symposium on Security and Privacy*, pp. 56-73, 2000.

[74] P. Papadimitratos and Z. J. Haas, "Secure Routing for Mobile Ad Hoc Networks", *In Proceedings of the SCS Communication Networks and Distributed Systems Modelling and Simulation Conference (CNDS 2002)*, 2002.

[75] S. Yi, P. Naldurg and R. Kravets, "A Securing-Aware Ad hoc Routing Protocol for Wireless Networks", *In Proceedings of the 6th World Multi-Conference on Systemics, Cybernetics and Informatics (SCI 2002)*, 2002, pp. 299-302.

[76] Y. Chun Hu, A. Perrig and D. B. Johnson, "Ariadne: A secure on-demand routing protocol for Ad hoc networks", *Wireless Networks*, Vol. 11, No. 1-2, 2005, pp. 21-38.

[77] L. Buttyán and I. Vajda, "Towards provable security for ad hoc routing protocols", *In Proceedings of the 2nd ACM workshop on security of ad hoc and sensor networks (SASN'04)*, 2004, pp. 94-105.

[78] G. ACS, L. Buttyán and I. Vajda, "Provably Secure On-Demand Source Routing in Mobile Ad Hoc Networks", *IEEE Transactions on Mobile Computing*, Vol. 5, No. 11, 2006, pp. 1533-1546.

[79] A. Hafslund, A. Tønnesen, R. B. Rotvik, J. Andersson and Ø. Kure, "Secure Extension to the OLSR protocol", *In OLSR Interop and Workshop*, 2004.

[80] C. Adjih, T. Clausen, P. Jacquet, A. Laouiti, Paul Mühlethaler and D. Raffo, "Securing the OLSR protocol", *In Proceedings of the 2nd IFIP Annual Mediterranean Ad Hoc Networking Workshop*, Mahdia, Tunisia, 2003.

[81] D. Raffo, C. Adjih, T. Clausen and P. Mühlethaler, "An advanced signature system for OLSR", *In Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor networks (SASN'04)*, 2004, pp. 10-16.

[82] C. Adjih, T. Clausen, A. Laouiti, P. Mühlethaler and D. Raffo, "Securing the OLSR routing protocol with or without compromised nodes in the network", *Technical Report INRIA RR-5494, HIPERCOM Project, INRIA Rocquencourt*, 2005.

[83] L. Chen, X. Xue and J. Leneutre, "A lightweight mechanism to secure OLSR", *In International MultiConference of Engineers and Computer Scientists*, 2006, pp. 887-895.

[84] M. G. Zapata and N. Asokan, "Securing Ad hoc Routing Protocols", *In Proceedings of the 1st ACM Workshop on Wireless Security (WiSe'02)*, 2002, pp. 1-10.

[85] Y. Chun Hu, A. Perrig and D. B. Johnson, "Packet leashes: a defense against wormhole attacks in wireless networks", *In Proceedings of IEEE INFOCOM*, 2003.

[86] J. M. Orset, B. Alcalde and A. Cavalli, "An EFSM-Based Intrusion Detection System for Ad Hoc Networks", *In Proceedings of the 3rd international conference on Automated Technology for Verification and Analysis (ATVA'05)*, 2005, pp. 400-413.

[87] B. Alcalde, A. Cavalli, D. Chen, D. Khuu and D. Lee, "Network Protocol System Passive Testing for Fault Management: A Backward Checking Approach", *In Formal Techniques for Networked and Distributed Systems – FORTE,* Volume 3235 of the series Lecture Notes in Computer Science, 2004, pp. 150-166.

[88] M. Wang, L. Lamont, P. Mason and M. G. Orlatova, "An effective intrusion detection approach for OLSR MANET protocol", *In Proceedings of the 1st international conference on Secure network protocols (NPSEC'05)*, 2005, pp. 55-60.

[89] J. M. Orset and A. Cavalli, "A security model for OLSR MANET protocol", *In Proceedings of the 7th International Conference on Mobile Data Management (MDM'06)*, 2006, Page. 122.

[90] F. Cuppens, N. Cuppens-Boulahia, T. Ramard and J. Thomas, "Misbehaviors Detection to Ensure Availability in OLSR", *Mobile Ad-Hoc and Sensor Networks*, Volume 4864 of the series Lecture Notes in Computer Science, 2007, pp. 799-813.

[91] F. Cuppens, N. Cuppens-Boulahia, S. Nuon and T. Ramard, "Property based intrusion detection to secure OLSR", *In Proceedings of the 3rd International Conference on Wireless and Mobile Communications (ICWMC '07)*, 2007, Page. 52.

[92] L. Buttyán and J. P. Hubaux, "Nuglets: a Virtual Currency to Stimulate Cooperation in Self Organized Mobile Ad Hoc Networks", *Technical report No. DSC/2001, Swiss Federal Institute of Technology*, Lausanne, 2001. [Consulté le 15 Mars 2016]. Accessible sur : https://infoscience.epfl.ch/record/274/files/TR01_001.ps

[93] S. Zhong, J. Chen and Y. R. Yang, "Sprite: A simple, cheat-proof, credit-based system for mobile ad-hoc networks", *In Proceedings of the 22nd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2003)*, Vol. 3, 2003, pp. 1987-1997.

[94] H. Yang, X. Meng and S. Lu, "Self-Organized Network-Layer Security in Mobile Ad Hoc Networks", *In ACM MOBICOM Wireless Security Workshop (WiSe'02)*, 2002, pp. 11-20.

[95] L. Anderegg and S. Eidenbenz, "Ad hoc-VCG: A Truthful and Cost-Efficient Routing Protocol for Mobile Ad hoc Networks with Selfish Agents", *In Proceedings of the 9th annual international conference on Mobile computing and networking*, 2003, pp. 245-259.

[96] V. N. Padmanabhan and D. R. Simon, "Secure Traceroute to Detect Faulty or Malicious Routing", *ACM SIGCOMM Computer Communications Review*, Vol. 33, No. 1, 2003, pp. 77-82.

[97] B. Awerbuch, D. Holmer, C. Nita-Rotaru and H. Rubens, "An on-demand secure routing protocol resilient to byzantine failures", *In Proceedings of the 1st ACM workshop on Wireless security (WiSe'02)*, 2002, pp. 21-30.

[98] B. Awerbuch, R. Curtmola, D. Holmer, C. Nita-Rotaru and H. Rubens, "ODSBR: An on-demand secure byzantine resilient routing protocol for wireless ad hoc networks", *ACM Transactions on Information and System Security (TISSEC)*, Vol. 10, No. 4, 2008, pp. 1-35.

[99] L. Buttyán, L. Dóra and I. Vajda, "Statistical Wormhole Detection in Sensor Networks", *In Security and Privacy in Ad-hoc and Sensor Networks*, Volume 3813 of the series Lecture Notes in Computer Science, 2005, pp. 128-141.

[100] K. A. Bradley, S. Cheung, N. Puketza, B. Mukherjee and R. A. Olsson, "Detecting disruptive routers: A distributed network monitoring approach", *IEEE Network: The Magazine of Global Internetworking*, Vol. 12, No. 5, 1998, pp. 50-60.

[101] S. Buchegger and J-Y. Le Boudec, "Performance Analysis of the CONFIDANT Protocol (Cooperation Of Nodes: Fairness In Dynamic Ad hoc NeTworks)", *In Proceedings of the 3rd ACM International Symposium on Mobile Ad hoc networking & computing (MobiHOC 2002),* 2002, pp. 226-236.

[102] P. Michiardi and R. Molva, "Core: A collaborative reputation mechanism to enforce node cooperation in mobile Ad hoc networks", *In Proceedings of the IFIP TC6/TC11 6th Joint Working Conference on Communications and Multimedia Security: Advanced Communications and Multimedia Security*, 2002, Vol. 100, pp. 107-121.

[103] Q. He, D. Wu and P. Khosla, "SORI: a secure and objective reputation-based incentive scheme for Ad-hoc networks", *In Wireless Communications and Networking Conference WCNC 2004 IEEE*, Vol. 2, 2004, pp. 825-830.

[104] R. Molva and P. Michiardi, "Security in Ad hoc Networks", *Personal Wireless Communication*, 2003.

[105] K. Wang and M. Wu, "Improved secure trust-based location-aided routing model for MANETs", *China Communications*, Vol. 8, No. 3, 2011, pp. 154-162.

[106] H. Xia, Z. Jia, L. Ju, X. Li and E. H. M. Sha, "Impact of trust model on on-demand multi-path routing in mobile ad hoc networks", *Computer Communications*, Vol. 36, No. 9, 2013, pp. 1078-1093.

[107] S. L. F. Sirotheau and R. T. De Sousa, "Evaluating trust in Ad Hoc network routing by induction of decision trees", *IEEE Latin America Transactions*, Vol. 10, No. 1, 2012, pp. 1332-1343.

[108] M. Mohanapriya and I. Krishnamurthi, "Trust based DSR routing protocol for mitigating cooperative black hole attacks in ad hoc networks", *Arabian Journal for Science and Engineering*, Vol. 39, No. 3, 2014, pp. 1825-1833.

[109] L. Quéré, "LA CONFIANCE", *Hermès Science Publications*, Vol. 19, No. 108/2001, Paris-France, 2001.

[110] M. Deutsch, "Cooperation and Trust: Some Theoretical Notes", *Nebraska Symposium on Motivation, Nebraska University Press*, 1962, pp. 275-320.

[111] N. Luhmann, "Trust and power", *John Willey & Sons*, 1979.

[112] S. Bok, "Lying: Moral choice in public and private life", *Vintage*, 1999, 368.

[113] D. Gambetta. "Can we trust trust", *Trust: Making and Breaking Cooperative Relations, electronic edition, Department of Sociology, University of Oxford,* chapter 13, 2000, pp. 213-237.

[114] S. P. Marsh, "Formalising Trust as a Computational Concept", *PhD thesis, Department of Computing Science, University of Stirling, UK*, 1994.

[115] S. Marsh and P. Briggs, "Examining trust, forgiveness and regret as computational concepts", *Computing with Social Trust*, *Human–Computer Interaction Series*, 2009, pp. 9-43.

[116] M. Mejia, N. Peña, J. L. Muñoz and O. Esparza, "A review of trust modeling in ad hoc networks", *Internet Research*, Vol. 19, No. 1, 2009, pp. 88-104, DOI 10.1108/10662240910927849.

[117] L. A. Zadeh, "Fuzzy logic, neural networks, and soft computing". *In Communications of the ACM*, Vol. 37, No. 3, 1994, pp. 77-84.

[118] H. Hallani and S. Shahrestani, "Fuzzy Trust Approach for Wireless Ad-hoc Networks", *In Communications of the IBIMA*, Vol. 1, 2008, pp. 212-218.

[119] J. Deng, "Introduction to Grey System", *The Journal of Grey System*, Vol. 1, No. 1, 1989, pp.1-24.

[120] J. Guo, A. Marshall and B. Zhou, "A Multi-Parameter Prediction Model for Misbehaviour Detection in a MANET Trust Framework", *Journal of Applied Science and Engineering*, Vol. 17, No. 1, 2014, pp. 45-58.

[121] H. Liang, W. Chen and Z. Guoqing, "Link reliability assessment based on grey relational analysis for wireless ad hoc networks", *In 29th Chinese Control Conference (ccc),* 2010, pp. 4236-4240.

[122] T. Ghosh, N. Pissinou and K. Makki, "Towards designing a trusted routing solution in mobile ad-hoc networks", *Mobile Networks and Applications*, Vol. 10, No. 6, 2005, pp.985–995.

[123] A. Beghriche et A. Bilami, "De la Sécurité à la E-Confiance dans les Réseaux sans fil Ad hoc", *1st Workshop on Next Generation Networks: Mobility (IEEE WNGN 2008)*, Fès Maroc, 18-19 Juillet 2008. pp. 25-30.

[124] A. Rachedi et A. Benslimane, "Architecture Hiérarchique Distribuée pour sécuriser les Réseaux Ad hoc Mobiles", *8ème journées Doctorales en Informatique et Réseaux*, Marne la Vallée, Janvier 2007.

[125] A. P. Lauf, R. A. Peters and W. H. Robinson, "A distributed intrusion detection system for resource-constrained devices in ad hoc networks", *Ad Hoc Networks*, Vol. 8, No. 3, 2010, pp. 253–266.

[126] H. Hu, B. Liu and T. Shen, "Intelligent reasoning and management decision making with grey rough influence diagrams", *International Journal of Intelligent Computing and Cybernetics (IJICC)*, Vol. 9, No. 4, 2016, pp. 336–353.

[127] A. Beghriche and A. Bilami, "A Fuzzy trust-based routing model for mitigating the misbehaving nodes in mobile Ad hoc networks", *International Journal of Intelligent Computing and Cybernetics*, Vol. 11, No.2, 2018, pp. 309-340.

[128] Information Sciences Institute, "The Network Simulator ns-2", *University of Southern California*, http://www.isi.edu/nanam/ns/

[129] L. Bajaj, M. Takai, R. Ahuja, K. Tang, R. Bagrodia and M. Gerla, "GloMoSim: A scalable network simulation environment", *Technical Report 990027, University of California, Los Angeles, Computer Science Department*, May 1999.

[130] X. Chang, "Network simulations with OPNET", *In Proceedings of the 31st conference on Winter simulation: Simulation - a bridge to the future*, Vol. 1, 1999, pp. 307–314.

[131] H. A. Adnane, "La confiance dans le routage Ad hoc : étude du protocole OLSR", *Thèse de doctorat, Université de Rennes 1* (*France)*, 2008.

[132] T. Camp, J. Boleng and V. Davies, "A survey of mobility models for ad hoc network research". *Wireless Communications and Mobile Computing (WCMC)*, Vol. 2, No. 5, 2002, pp. 483–502.

[133] X. Li, M. R. Lyu and J. Liu, "A trust model based routing protocol for secure ad-hoc networks", *In Proceedings of the IEEE Aerospace Conference*, Big Sky, MT, USA, Vol. 2, 2004, pp.1286–1295.

[134] M. Frikha, "Réseaux ad hoc routage, qualité de service et optimisation", *Collection performance des réseaux. André-Luc Beylot, Germes-Lavoisier*, 2007.

[135] V. V. Mandhare and R. C. Thool, "Evaluating Performance of Reactive and Hybrid Routing Protocol in Mobile Ad Hoc Network", *In International Conference on ICT for Sustainable Development, series of Advances in Intelligent Systems and Computing*, 2016, Vol. 408, pp. 571-580.

[136] F. Zhang, Z. P. Jia, H. Xia, X. Li and E. H. M. Sha, "Node trust evaluation in mobile ad hoc networks based on multi-dimensional fuzzy and Markov SCGM (1,1) model", *Computer Communications,* 2012, Vol. 35, No. 5, pp. 589-596.

[137] J. Li, R. Li and J. Kato, "Future Trust Management Framework for Mobile Ad Hoc Networks", *IEEE Communications Magazine*, 2008, Vol. 46, No. 4, pp. 108-114.

[138] A. Beghriche et A. Bilami, "Modélisation et Gestion de la Confiance dans les Réseaux Mobiles Ad hoc", *Conférence Internationale sur l'informatique et ses applications (CIIA'09), Series of Springer LNCS*, 03-04 Mai 2009.

[139] A. Beghriche et A. Bilami, "AFIM: A Trusted Routing Algorithm Based on Fuzzy LogIc for Mobile Ad hoc Networks", *3rd World Conference on Information Technology, WCIT-2012*, 14-16 November 2012, Barcelona, Spain, Vol. 3, pp. 1400-1404.