وزارة التعليم العالي والبحث العلمي

BADJI MOKHTAR UNIVERSITY – ANNABA

UNIVERSITE BADJI MOKHTAR – ANNABA

جامعة باجي مختار – عنابة

Faculté : Sciences de l'Ingéniorat

Département : Informatique

THESE

Présentée en vue de l'obtention du diplôme de
Doctorat 3ème cycle

Détection d'Intrusion et Sécurisation du Routage dans les Réseaux Ad hoc

Filière : Informatique

Spécialité : Réseaux et Sécurité Informatique

Par
Abdelaziz Amara Korba

DEVANT LE JURY :

Président	Mohamed Tahar KIMOUR	Professeur à l'Université d'Annaba
Examinatrice	Nacira GHOUALMI-ZINE	Professeur à l'Université d'Annaba
Examinateur	Okba KAZAR	Professeur à l'Université de Biskra
Examinatrice	Nouria HARBI	MCA, HDR à Université Lumière Lyon 2
Directeur de thèse	Salim GHANEMI	Professeur à l'Université d'Annaba
Invité	Mehdi NAFA	MCA à l'Université d'Annaba

Année 2016

Remerciement

Je remercie en priorité dieu le tout puissant de m'avoir donné le courage, la force et la volonté d'accomplir ce travail.

Je tiens à remercier mes parents ainsi que toute ma famille pour leurs soutiens et encouragements et sans qui tout cela n'aurait pas pu être possible.

Je tiens à remercier les membres de mon jury : Mr Okba KAZAR , Mme Nacira GHOUALMI-ZINE et Mme Nouria HARBI d'avoir accepté d'être examinateurs de cette thèse, je remercie tout autant Mr Mohamed Tahar KIMOUR d'en avoir accepté la présidence.

Je voudrais aussi remercier mes deux encadrants Mr Salim Ghanemi et Mr Mehdi Nafaa pour toute leurs aides, leurs conseils, ainsi que leurs encouragements.

Je tiens à remercier et à exprimer ma profonde gratitude à Mr Yacine Ghamri-Doudane pour son aide et son soutien durant mon stage au Laboratoire L3I à l'université de La Rochelle.

J'adresse un remerciement à tous ceux qui ont de près ou de loin contribué à la réalisation de ce projet, plus particulièrement à mes amis et mes collègues aux laboratoires L3I et LRS.

Table des matières

Liste des figures

Liste des tableaux

ملخص

االانتشار السريع للأجهزة المحمولة وتطور تكنولوجيا الاتصالات عرف تزايد كبير في العقود الأخيرة. ولا سيما تكنولوجيات الشبكات اللاسلكية التي أحدثت ثورة في أسلوب حياتنا. توافر وتسويق التقنيات اللاسلكية القادرة على إنشاء اتصال مباشر بين أجهزة المستخدم، أفضت إلى ابتكار نموذج الشبكات الاسلكية أدهوك. الشبكات الاسلكية أدهوك هي عبارة عن شبكات مستقلة و بدون بنية تحتية محددة, تتألف من مجموعة من اجهزة الكمبيوتر الموصلة مع بعضها من غير وجود نقطة اتصال مركزية, تدعى العقد. إلى جانب التنقل،العقد قادرة على التنظيم الذاتي وتتميز هذه الشبكات بتنصيب و نشر سريع وسهل لأنها لا تتطلب وجود بنية تحتية قبلية وسلطة مركزية. ومع ذلك فان خدمة تسييرو توجيه المعلومات عبرهدا النوع من الشبكات عرضة لهجمات مختلفة. في حين بذلت جهودا كبيرة لتأمين هذه الشبكات، لكن معظم الآليات المقترحة اقتصرة عموما على معالجة بعض الهجمات وقابلة للتطبيق فقط مع بروتوكول معين. ولذلك، فإن الدافع لهذه الأطروحة ينطلق من الحاجة إلى تأمين الشبكات الاسلكية ادهوك من انواع مختلفة من الهجمات.

الهدف الرئيسي من هذه الأطروحة هو اقتراح آلية عامة لكشف ومنع الاختراق, قادرة على حماية الشبكات الاسلكية أدهوك ضد مجموعة متنوعة من الهجمات وضمان مستوى عال من الأمن. ينبغي ان تكون الآلية الجديدة قادرةعلى حماية الشبكة من الهجمات المختلفة دون التأثير على أداء الشبكة. نقترح نموذجا يتكون من ثلاث وحدات، الوحدة الأولى هي نظام كشف ومنع الاختراق على أساس المواصفات.وتتمثل مهمتها في السيطرة على التفاعلات مع بقية الشبكة. الوحدة الثانية هي عبارة عن موازن للحمولة والذي مهمته الغاء العقد التي تشكل نقاط تركيز حركة مرورالمعلومات وتجنب امتصاص البينات من طرف العقد الخبيثة. الوحدة الثالثة هونظام استجابة ضد الاختراق.مع ان الرد على الاختراق يجب أن يكون جزء من آلية الكشف ومنع التسلل، الا انه لقي دائما أهمية أقل من الكشف ومنع التسلل. لذلك،من أهداف هذه الأطروحة أيضا اقتراح استراتيجية مرنة وقابلة للتكيف و الرد بفعالية ضد محاولات الاختراق. نقترح آلية لكشف ومنع التسلل تتكيف بشكل جيد للغاية مع خصائص الشبكات الاسلكية أدهوك مثل التوزيع، وكثرة تنقل العقد و الموارد المحدودة. قمنا بتقيم الآلية الأمنية المقترحة من خلال محاكاة واسعة النطاق بما في ذلك هجمات مختلفة في إطار سيناريوهات مختلفة. أظهرت نتائج المحاكاة وتقييم الأداء فعالية الالية المقترحة على كشف ومنع مجموعة واسعة من الهجمات على درجة عالية من الدقة ومعدل منخفض جدا من الكشوفات الخاطة.

كلمات البحث: الشبكات الاسلكية أدهوك، الأمن، لكشف ومنع التسلل ،الكشف على أساس المواصفات، الرد على التسلل.

Abstract

The rapid proliferation of mobile devices and communication technologies has been growing in the recent decades, especially wireless network technologies which revolutionized our lifestyle. The availability and marketing of wireless technologies which allow direct connection between user devices, has given rise to the paradigm of wireless multi-hop networks or multi-hop ad hoc networks. Wireless multi-hop ad hoc networks are autonomous network without any predefined infrastructure. They are composed of self-organized mobile entities called mobile nodes. These networks are characterized by a quick and easy deployment because they do not require any pre-existing infrastructure and centralized authority. However data routing service in these networks is vulnerable to various attacks. Although considerable efforts have been made to secure these networks, most of the mechanisms proposed in the literature address particular attacks and are only applicable with a specific protocol. Therefore, the motivation of this thesis emerges from the need to secure multi-hop ad hoc networks from a range of attacks.

In this thesis we address security issues related to the vulnerabilities of routing in ad hoc networks. We secure routing operations through the four contributions of this thesis. The first contribution is providing a systematic analysis of routing attacks based on the concept of basic and compound attacks. The second contribution is proposing a new specification-based intrusion detection and prevention system which protects the network from specification violation attacks, such as modification, fabrication and replay. We proposed a new approach for automatic extraction of specification model of routing protocol, which is expressed through the use of a finite state machine (FSM). An instance of the proposed specification-based intrusion detection is executed within each network node. The third contribution consists on proposing a new anomaly-based intrusion detection and prevention system to protect routing operation from fast forwarding attacks such as Wormhole and Rushing. The proposed system employs a statistical approach that exploits the load balancing concept, in a way that malicious nodes are identified as traffic concentration points due to their high route selection rate. The fourth contribution consists on proposing a hybrid intrusion detection and prevention system which combines the two proposed systems based on a hierarchical architecture. Where cluster nodes run the specification-based intrusion detection system to detect specification violations attacks, while the cluster heads run the anomaly-based intrusion detection system to detect fast forwarding attacks. The performance evaluation of the proposed intrusion

detection and prevention systems has shown that they outperform existing security mechanisms in terms of detected attacks and detection rates without affecting network performance.

Keywords: ad hoc networks, routing, security, intrusion detection and prevention, specification-based detection, anomaly-based detection, intrusion response.

Résumé

La prolifération rapide des dispositifs mobiles et des technologies de communication n'a cessé de croître ces dernières décennies. Particulièrement les technologies des réseaux sans fil qui ont révolutionné notre style de vie. La disponibilité et la commercialisation des technologies sans fil qui permettent l'établissement de connexion directe entre les périphériques utilisateurs, a fait émerger le paradigme des réseaux sans fil multi-sauts ou multi-sauts ad hoc. Les réseaux sans fil multi-sauts sont des réseaux autonomes sans infrastructure prédéfinie, composés par un ensemble d'entités mobiles appelées nœuds. Outre la mobilité, les nœuds sont aptes à s'auto organiser. Ces réseaux sont caractérisés par un déploiement rapide et facile parce qu'ils ne nécessitent pas d'infrastructure préexistante et d'autorité centrale. Cependant le service de routage des données dans les réseaux sans fil multi-sauts est vulnérable à diverses attaques. Malgré les recherches considérables qui ont été réalisées pour sécuriser ces réseaux, la majorité des mécanismes proposés dans la littérature n'adressent généralement que quelques attaques et ne sont applicables qu'avec un protocole spécifique. Par conséquent, la motivation de cette thèse émerge du besoin de mécanisme de sécurité capable de sécuriser les réseaux ad hoc de diverses attaques indépendamment du protocole de routage utilisé.

Dans le cadre de cette thèse nous abordons les problèmes de sécurité liées aux vulnérabilités du routage dans les réseaux ad hoc. Nous proposons une sécurisation du routage à travers les quatre contributions de cette thèse. La première consiste à proposer une analyse systématique pour étudier les attaques de routage en se basant sur le concept d'attaques élémentaires et d'attaques composées. La deuxième contribution consiste à proposer un nouveau système de détection et prévention d'intrusion basé sur la spécification et qui permet de protéger le réseau contre les attaques qui violent la spécification du protocole telles que : modification, fabrication et rejeu. Nous avons proposé une nouvelle méthode qui permet la génération automatique et la modélisation de la spécification du protocole de routage sous forme de machine à états finis. Une instance du système de détection proposé est exécutée par chaque nœud du réseau. La troisième contribution consiste à proposer un nouveau système de détection et prévention d'intrusion basé sur l'anomalie pour protéger le routage contre les attaques à retransmission rapide telles que Trou de ver et Rushing. Le système que nous avons proposé emploie une approche statistique qui exploite le concept d'équilibrage de charge. Une approche qui permet d'identifier les nœuds malveillants comme des points de concentration du trafic dû à leurs taux de sélection élevé. La quatrième contribution

consiste à proposer un système de détection et prévention d'intrusion hybride, qui combine les deux systèmes proposés précédemment sur une architecture hiérarchique. Une détection basée sur la spécification au niveau de chaque membre du cluster, et une détection basée sur l'anomalie au niveau des chefs des clusters. L'évaluation des systèmes de détection et prévention d'intrusion que nous avons proposé a montré que ces derniers surpassent les mécanismes de sécurité existants en terme d'attaques détectées et de taux de détection sans pour autant affecter les performances du réseau.

Mots-clés: réseaux ad hoc, routage, sécurité, détection et prévention d'intrusion, détection basée sur la spécification, détection basée sur l'anomalie, réponse aux intrusions

Chapitre 1 Introduction

La dernière décennie a connu un développement rapide des périphériques mobiles tels que les ordinateurs portables, les PDAs, les smart phones, les tablettes, les systèmes de transport intelligent, et d'autres appareils qui exploitent la communication sans fil. Ce qui a favorisé à promouvoir les recherches dans le domaine des réseaux sans fil. Comme ces derniers sont devenus omniprésent dans notre vie quotidienne, le besoin de fournir les services de connectivité, gestion et sécurité pour ces réseaux s'accroît de façon impressionnante. Le paradigme de réseaux sans fil multi-sauts ou multi-sauts ad hoc [67] a émergé dans le domaine civil dans les années quatre-vingt-dix. Suite à la commercialisation et la disponibilité des technologies sans fil telles que : Bluetooth (IEEE 802.15.1), pour les réseaux personnels sans fil (WPAN), et la famille des normes IEEE 802.11 pour les réseaux locaux sans fil à haut débit (WLAN). Ces technologies permettent de fournir une connexion directe entre les périphériques utilisateurs.

Le paradigme multi-sauts a été conçu pour étendre la possibilité de communiquer entre n'importe quelle paire de nœuds, sans mettre en place une infrastructure omniprésente. Il permet d'interconnecter un ensemble de nœuds mobiles et/ou fixes par une technologie sans fil et/ou filaire pour former un réseau dynamique temporaire avec ou sans faire appel à une administration centralisée ou un support fixe. Les utilisateurs proches (dans la portée radio) peuvent communiquer directement (en exploitant leurs interfaces réseau sans fil en mode ad hoc). Non seulement pour échanger leurs données, mais aussi pour transmettre le trafic d'autres nœuds du réseau qui ne peuvent pas communiquer directement. Ainsi, ils opèrent comme les routeurs dans le réseau Internet. Les réseaux sans fil multi-sauts sont considérablement différent des réseaux filaires. En particulier, la topologie du réseau est dynamique parce que les nœuds du réseau sont libres de joindre et quitter le réseau à n'importe quel moment, puisque ils sont libres de se déplacer arbitrairement.

Les réseaux sans fil multi-sauts s'organisent automatiquement ce qui fait que leurs déploiement est rapide. Cependant l'absence de contrôle central, le routage par coopération et la limitation en termes de ressources posent des problèmes de sécurité supplémentaires. La sécurité dans les réseaux filaire est généralement assurer par l'utilisation des pare-feu pour contrôler le trafic entrant et sortant. Cependant, un pare-feu ne prévient pas les attaques à l'intérieur du réseau. Par conséquent, les systèmes de détection d'intrusion et d'autres mécanismes de sécurité basés sur les techniques de cryptographie sont largement utilisés conjointement avec les pare-feu dans les

réseaux filaires. Puisque la mise en place des pare-feu dans les réseaux sans fil multi-sauts n'est pas possible, due à l'absence de contrôle centralisé, les systèmes de détection et prévention d'intrusion deviennent la première ligne de défense contre les intrusions.

La détection d'intrusion permet de déceler les comportements incorrects, anormaux et malveillants. Bien que le déploiement d'un mécanisme de sécurité dans les réseaux multi-sauts ad hoc n'est pas une tâche facile, la détection et prévention d'intrusion reste l'approche la plus prometteuse pour sécuriser ces réseaux avec un surcoût raisonnable, comme nous allons le montrer à travers les mécanismes de sécurité proposés.

1.1. Motivation

La prolifération rapide des appareils mobiles et leur large utilisation dans notre vie quotidienne, a stimulé la transition des réseaux filaires fixes aux réseaux sans fil mobiles sans infrastructure. Les réseaux sans fil multi-sauts occupent une place importante dans les réseaux pervasifs actuels dus à leurs déploiements rapides et leurs autonomies. Ainsi ils sont très adaptés pour fournir des services réseau dans le cas des urgences et catastrophes naturelles. Comme par exemple dans les cas d'un tremblement de terre ou il n'y a pas d'infrastructure disponible, un réseau ad hoc peut être mis en place pour effectuer les opérations de secours. Cependant le service de routage des données dans ces réseaux est vulnérable aux menaces de sécurité pour les raisons suivantes :

- En l'absence des routeurs, tous les nœuds dans le réseau doivent participer et coopérer pour assurer le service de routage des données
- Due à l'architecture ouverte du réseau où les nœuds sont libres de joindre et quitter à tout moment le réseau, un nœud malveillant peut joindre le réseau pour causer des dommages.
- La plupart des protocoles de routage pour les réseaux sans fil multi-sauts ont été développés avec l'hypothèse que tous les nœuds sont fiables et qu'il n'y a pas des nœuds malveillants dans le réseau.
- La limitation des nœuds en termes de ressources constitue une vulnérabilité que le nœud malveillant peut exploiter pour compromettre les services de routage et transmission des données dans le réseau.
- La topologie dynamique du réseau due à la mobilité des nœuds rend difficile la détection du comportement malveillant.

Par conséquent, la motivation fondamentale de la recherche effectuée dans cette thèse émerge du besoin de sécuriser les réseaux ad hoc contre une grande variété d'attaques. Les attaques exécutées par le nœud malveillant peuvent causer au réseau des dommages à des degrés différents en fonction

du type d'attaque utilisé. Plusieurs mécanismes ont été proposés dans la littérature pour sécuriser les réseaux ad hoc. Cependant, la plupart des mécanismes proposés se sont focalisés sur la prévention et la détection d'une seule attaque. Il y a eu peu de mécanisme générique capable de protéger le réseau d'une grande variété d'attaques.

Nous considérons que la sécurité est un service important pour les réseaux sans fil multi-sauts. Notamment, pour ce type de réseau où il n'y a pas d'administration et de contrôle centraux pour surveiller les activités du réseau et identifier les comportements malveillants. Nous constatons également un manque de recherches en ce qui concerne le développement de mécanismes génériques capables de détecter un large éventail d'attaques (connues et inconnues).

Donc nous pensons que plus d'efforts et de recherches sont nécessaires pour le développement de mécanismes capables de protéger les réseaux ad hoc contre une grande variété d'attaques. Nous pensons également qu'il ne faut pas limiter le mécanisme de sécurité à faire uniquement la détection, mais aussi à mettre en place une stratégie de réponse aux attaques et éventuellement les prévenir. En outre, il est impératif de considérer la faisabilité du mécanisme de sécurité proposé en prenant en compte les contraintes des réseaux ad hoc (distribution, ressources limitées, forte mobilité, etc.).

1.2. Objectifs

L'objectif principal de cette thèse est de sécuriser et protéger le routage dans les réseaux ad hoc de diverses attaques sans affecter les performances du réseau. La plupart des mécanismes de détection d'intrusion proposés dans la littérature n'adressent généralement que quelques attaques et ne sont applicables que sur un protocole spécifique. Bien que la réponse à l'intrusion doit normalement faire partie du mécanisme de détection et prévention d'intrusion, celle-ci a toujours reçu moins d'importance que la prévention et la détection. Par conséquent, il est également dans les objectifs de cette thèse de proposer une stratégie de réponse flexible et adaptative capable de répondre de manière efficace aux intrusions. La distribution, la forte mobilité des nœuds et la limitation en termes de ressources, constituent des contraintes que le mécanisme de sécurité doit prendre en considération. Par conséquent nous proposons des mécanismes de détection et prévention d'intrusion qui s'accommode très bien à toutes ces contraintes.

1.3. Contributions

Les principales contributions de cette thèse sont résumées comme suit :

- Modélisation et analyse systématique des attaques de routage dans les réseaux ad hoc en se basant sur le concept d'attaques élémentaires et composées.

- Une nouvelle méthode qui permet l'extraction automatique et la modélisation de la spécification du protocole de routage sous forme de machine à états finis.
- Un nouveau système de détection et prévention d'intrusion basé sur la spécification et qui permet de protéger le réseau contre les attaques qui violent la spécification du protocole de routage telles que : modification, fabrication et rejeu.
- Un nouveau système de détection et prévention d'intrusion basé sur l'anomalie pour protéger le routage contre les attaques à retransmission rapide telles que trou de ver et Rushing.
- Un système de détection et prévention d'intrusion hybride, qui combine les deux systèmes proposés sur une architecture hiérarchique. Une détection basée sur la spécification au niveau de chaque nœud du cluster, et une détection basée sur l'anomalie au niveau des chefs des clusters.

1.4. Structure de la thèse

La suite de ce manuscrit est structurée comme suit :

Chapitre 2 : ce chapitre présente le paradigme multi-sauts, ses caractéristiques et ses domaines d'application. Il donne un aperçu sur les trois classes émergentes des réseaux sans fil multi-sauts. Il présente également le routage et décrit les différents types de routage en prenant comme exemples les protocoles de routage les plus populaires. Ensuite, il considère la sécurité et présente une étude des vulnérabilités et de toutes les attaques possibles. Enfin, il donne un état de l'art des principales solutions proposées dans la littérature pour détecter et éliminer les attaques les plus populaires.

Chapitre 3 : ce chapitre présente la détection d'intrusion et les différentes techniques de détection. Les réseaux multi-sauts ad hoc ont des caractéristiques différentes de celles des réseaux conventionnels, et soulèvent de nouveaux défis pour les solutions de sécurité existantes. Par conséquent ce chapitre examine les problèmes liés à la détection d'intrusion dans les réseaux sans fils multi-sauts et particulièrement les réseaux ad hoc mobiles MANETs. Ensuite, il présente un état de l'art des principaux mécanismes de détection et prévention d'intrusion. Enfin, il présente une vue d'ensemble sur les futures recherches dans le domaine.

Chapitre 4 : puisque la conception d'un système de détection et prévention d'intrusion efficace, nécessite une compréhension approfondie des attaques. Dans ce chapitre nous nous intéressons de plus près aux attaques internes contre les protocoles de routage ad hoc. Nous présentons une analyse systématique pour étudier et modéliser les attaques de routage dans les réseaux ad hoc mobiles MANETs, en se basant sur le concept d'attaques élémentaires et d'attaques composées.

Chapitre 5 : ce chapitre propose un nouveau système de détection et prévention d'intrusion basé sur la spécification. Le système proposé permet de détecter les attaques qui violent la spécification du protocole de routage. Le système proposé utilise un ensemble de règles qui définissent le comportement normal du nœud du point de vue de l'opération de routage. Les règles de détection sont générées automatiquement sous forme de machine à états finis grâce à une méthode d'extraction automatique qui s'inspire de la programmation logique inductive (PLI). Une instance du système de détection proposé est exécutée sur chaque nœud du réseau afin de contrôler et protéger les interactions du nœud hôte avec les autres nœuds du réseau.

Chapitre 6 : ce chapitre propose un nouveau système de détection et prévention d'intrusion basé sur l'anomalie pour détecter et empêcher les attaques à retransmission rapide telles que trou de ver et Rushing. Le système emploie une approche statistique qui exploite le concept d'équilibrage de charge. Dans ce chapitre nous présentons un système de détection et prévention d'intrusion hybride, qui combine les deux systèmes proposés sur une architecture hiérarchique. Une détection basée sur la spécification au niveau de chaque nœud du cluster, et une détection basée sur l'anomalie au niveau des chefs des clusters.

Chapitre 7 : ce chapitre conclut le manuscrit et donne un résumé de ce qui a été fait dans la thèse. Enfin il donne une vue d'ensemble sur les perspectives et les recherches futures dans le domaine.

Chapitre 2 Réseaux sans fil multi-sauts, routage et sécurité

Dans ce chapitre nous présentons le paradigme multi-sauts, ses caractéristiques et ses domaines d'application. Nous donnons un aperçu sur les trois classes émergentes des réseaux sans fil multi-sauts et nous explorons les menaces de sécurité liés à ces réseaux. Nous décrivons particulièrement les caractéristiques spécifiques des réseaux ad hoc mobiles (MANETs), réseaux maillés (WMNs) et réseaux ad hoc véhiculaires (VANETs). Nous présentons le routage dans les réseaux sans fil multi-sauts, et nous décrivons les différents types de routage en prenant comme exemple les protocoles de routage les plus populaires. Ensuite, nous considérons les aspects sécuritaires et nous étudions les vulnérabilités et les attaques possibles. Enfin nous présentons les solutions proposées dans la littérature pour sécuriser les réseaux multi-sauts contre des attaques particulières.

Dans la section 2.1 nous présentons les réseaux sans fil multi-sauts et les trois classes émergentes de ce type de réseaux. La section 2.2 est consacrée au routage dans les réseaux sans fil multi-sauts. Ensuite, nous nous penchons dans la section 2.3 sur la sécurité des réseaux multi-sauts. Nous examinons toutes les attaques pouvant cibler les réseaux multi-sauts sur les différentes couches de la pile protocolaire. Dans la section 2.4, nous discutons les mécanismes de sécurité proposés dans la littérature pour protéger les réseaux multi-sauts contre les différentes attaques visant à perturber leurs bons fonctionnements. Nous concluons le chapitre dans la section 2.5.

2.1. Les réseaux sans fil multi-sauts

La prolifération des technologies de communication n'a cessé de croître ces dernières décennies. En particulier les technologies des réseaux sans fil qui ont révolutionné notre style de vie. Ces technologies répondent de plus en plus efficacement à nos besoins quotidiens et exigences en termes de communication. Une des avancées majeures dans la technologie des réseaux sans fil est le paradigme multi-sauts ou multi-sauts ad hoc [67]. Les réseaux sans fil multi-sauts attirent l'intérêt des chercheurs depuis le début les années quatre-vingt-dix, dû au développement rapide des technologies sans fil. Ces technologies permettent de fournir des connexions directes entres les équipements, tels que: Bluetooth (IEEE 802.15.1) [68], pour les réseaux personnels PAN (Personal Area Network), et la famille des standards IEEE 802.11 [26] pour les réseaux locaux sans fil WLAN (Wireless Local Area Network).

Le paradigme multi-sauts permet d'interconnecter un ensemble de nœuds mobiles et/ou fixes par une technologie sans fil et/ou filaire pour former un réseau dynamique temporaire avec ou sans faire appel à une administration centralisée ou un support fixe. Dans ce qui suit, nous donnons une brève description des principales classes des réseaux multi-sauts, à savoir les réseaux ad hoc mobiles, les réseaux maillés, les réseaux de capteurs, les réseaux ad hoc véhiculaires.

2.1.1 Les réseaux ad hoc mobiles (MANETs)

Les réseaux ad hoc mobiles ou MANET (Mobile Ad hoc NETwork) sont des réseaux radio autonomes sans infrastructure prédéfinie, composés par un ensemble d'entités mobiles appelées nœuds. Outre la mobilité, les nœuds sont aptes à s'auto organiser, et assurent tous la fonction de routage. Ces réseaux sont caractérisés par un déploiement rapide et facile parce qu'ils ne nécessitent pas d'infrastructure préexistante et d'autorité centralisée. La figure 2.1 montre une topologie standard d'un réseau ad hoc, les nœuds communiquent directement entre eux, sans passer par un point d'accès ou une station de base. Les nœuds comptent sur leurs voisins immédiats pour relayer leurs messages, à chaque fois qu'un nœud souhaite communiquer avec un autre nœud qui n'est pas à sa portée, il envoie son message à ses voisins qui à leur tour l'envoient à leurs voisins et ainsi de suite. Ce processus est répété jusqu'à ce que le message atteigne le nœud destinataire.

2.1.1.1 Caractéristiques des MANETs

Les réseaux ad hoc mobile sont totalement différent des réseaux filaires, cependant ils sont très similaires aux autre réseaux sans fil tels que les réseaux maillés, ou les réseaux de capteurs. Les caractéristiques essentielles [1] des réseaux ad hoc mobile sont :

- **Déploiement rapide :** Les réseaux ad hoc ne nécessitent pas d'infrastructure préexistante, ce qui rend leur déploiement et mise en place facile et rapide. Le minimum pour déployer un réseau ad hoc est d'avoir deux nœuds équipés d'interfaces sans fil, et qui sont à la porte l'un de l'autre. C'est la caractéristique la plus importante des réseaux MANETs, et c'elle qui les a rendu applicable particulièrement dans les cas où le réseau est nécessaire et il n'y a pas d'infrastructure disponible, tels dans les cas de catastrophes naturelles sur des zones dépourvues d'infrastructures.
- **Topologie dynamique:** Les nœuds se déplacent librement et de manière aléatoire, ces derniers peuvent rejoindre ou quitter le réseau de façon dynamique. Ainsi la topologie du réseau qui est typiquement multi sauts peut changer aléatoirement et rapidement à tout moment.

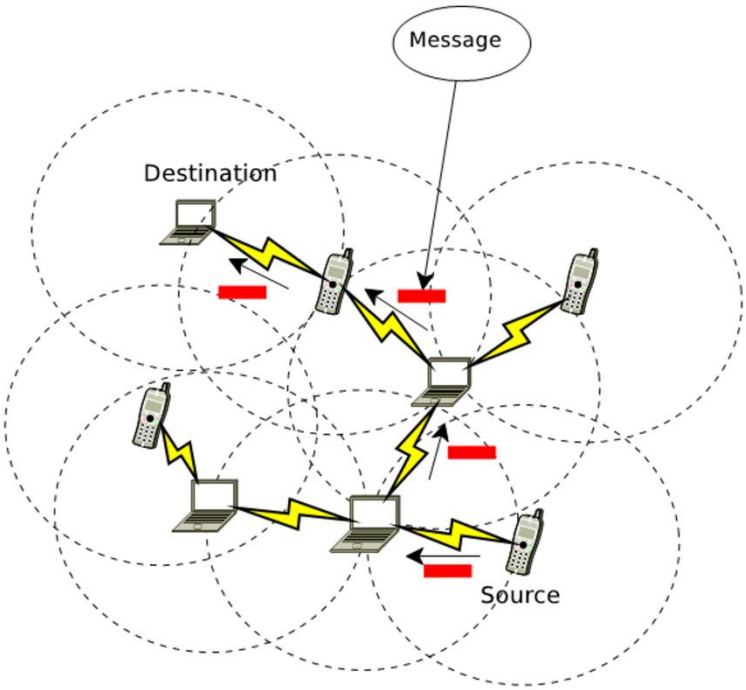


Figure 2.1 Réseaux ad hoc mobiles

- **Auto-organisation:** l'auto-organisation est le processus durant lequel l'organisation interne du système est systématisée sans être géré par un intervenant extérieur [2]. Les nœuds dans les MANETs sont appelés auto-organisés, premièrement parce qu'ils devraient être conscients de leurs états, capacités, et toutes leurs connexions avec les autres nœuds, et deuxièmement parce qu'ils ont besoin de s'auto-configuré automatiquement.
- **Bande passante limitée :** Malgré les énormes progrès des technologies sans fil en termes de bande passante, ils restent toutefois inférieurs à ceux des technologies filaires. Les liens sans fil sont limité et moins fiables que les liens filaire, dû au accès multiples, la perte, le bruit et les interférences.
- **Ressources limitées:** les nœuds d'un réseau ad hoc sont des petits dispositifs électroniques limités terme de calcul, quantité de mémoire et autonomie de batterie.
- **Sécurité physique:** les nœuds mobiles ont une probabilité non-négligeable d'être compromis, et d'être l'objet d'attaques physiques, tels que l'écoute, l'interception, le déni de service, et les attaques de routage. Parce qu'ils se déplacent dans des environnements hostiles sans aucune protection physique, et parce qu'ils ne sont pas géré de manière centralisée.

2.1.1.2 Applications des MANETs

Les réseaux ad hoc sont déployés et utilisés dans divers domaines d'application. Le domaine militaire, gestion de crise et secours, le partage de fichiers entre les participants à un séminaire ou une réunion. Cette section présente quelques domaines d'application des réseaux ad hoc mobiles :

- **Opérations de sauvetage:** Les réseaux ad hoc mobiles sont les plus adaptés aux opérations de sauvetage et de gestion de crises, notamment lors des catastrophes naturelles, ou l'infrastructure de communication est entièrement détruite et l'établissement rapide d'un réseau de communication est crucial. Ces réseaux sont rapidement déployés sur les lieux des sinistres pour assurer le relai et la liaison des communications entre sauveteurs, et secours médicaux.
- **Applications militaires:** Les réseaux mobiles ad hoc sont conçus principalement pour des opérations et des applications militaires. Ces réseaux sont adaptés aux environnements hostiles tels que les champs de bataille, car ils sont autonomes et rapidement déployables. Le réseau ad hoc fournit une plateforme de communication entre les équipements militaires, les véhicules blindés, et les soldats.
- **Applications commerciales:** Il y a un intérêt grandissant pour l'utilisation des réseaux ad hoc mobiles dans le domaine commercial. Les réseaux mobiles ad hoc peuvent être utilisés pour élargir un réseau avec infrastructure afin d'offrir un service tel que l'accès à Internet avec un coût réduit. Ils peuvent être par exemple utilisés par les agences de taxi pour assurer le dispatch des véhicules, la distribution des informations concernant les routes et les conditions météo etc. En outre, ils peuvent être utilisés pour connecter plusieurs ordinateurs pour partager des fichiers, des jeux, l'organisation des réunions, la communication entre agents, etc.

2.1.2 Les réseaux maillés

Les réseaux maillés WMNs (Wireless Mesh Networks) sont composées de trois types d'entités : les routeurs mesh (appelés aussi point d'accès), les routeurs passerelles et les clients, comme le montre la figure 2.2. L'ensemble des routeurs et des passerelles constitue le Backbone du réseau maillé sur lequel tout le trafic est transporté. Les routeurs sans fil sont interconnectés entre eux par des liens radio. Les routeurs passerelles sont souvent équipés de plusieurs interfaces (filaires et sans fil). Ils peuvent être fixes ou mobiles, et servent comme des points d'accès à internet et à d'autres types de réseaux fournissant divers services. Les clients sont connectés au réseau via les points d'accès.

2.1.2.1 Caractéristiques des réseaux maillés

- Topologie du réseau : le Backbone sans fil différencie les réseaux maillés des autres types de réseaux. Contrairement aux réseaux MANETs, la mobilité dans l'infrastructure Backbone n'est pas fréquente. Les nœuds routeurs sont stationnaires.
- Modèle de trafic : la transmission des données est essentiellement entre les nœuds mobiles et la passerelle, le trafic entre les nœuds mobiles est peu fréquent.
- Diversité du canal : les WMNs peuvent bénéficier de la possibilité d'introduire la diversité du canal dans le processus du routage, ce qui est impossible dans les MANETs et VANETs due à la mobilité.

2.1.2.2 Applications des réseaux maillés

Les réseaux maillés sont déployés pour répondre à un ou plusieurs objectifs dont certains sont énumérés ci-dessous :

- L'utilisation par des agences territoriales pour leurs propres besoins : par exemple, relevé des compteurs ou des services d'urgence dans le but d'améliorer le service rendu aux citoyens, de réduire des postes de dépenses (factures de téléphone ou d'accès Internet, réduction des déplacements de personnels).
- La fourniture d'un accès Internet de service public au moyen de Hot Spots, déployés en général dans les centres-villes ou dans les zones commerciales, afin de répondre aux besoins des usagers. Selon une étude réalisée par le SagaTel en Décembre 2006, environ 37 000 Hot Spots sont déployés en France dont la plupart sont pour un usage d'accès Internet. Aux États-Unis, près de 300 projets de déploiement de réseaux ont été menés dont 150 sont opérationnels.
- intégration de différents réseaux sans fil, tels que les réseaux cellulaires, capteurs et WIFI dans le but de permettre aux utilisateurs d'accéder à tous les services fournis par ces réseaux d'une manière transparente.
- Les réseaux maillés peuvent être utilisés pour relier les moyens de transport (bus, tram, train, etc.) par le biais d'un réseau de dorsal maillé (backhauling network). Ainsi, des services pratiques, comme la diffusion des informations de transport aux passagers, la surveillance à distance à l'intérieur des moyens de transport, permettent la communication entre les conducteurs, etc.
- Les réseaux maillés sans fil constituent une solution efficace pour les entreprises. Ceci peut être à petite échelle au sein d'un bâtiment ou bien à grande échelle ; ainsi, comme par exemple relier des bureaux ou des services d'une entreprise qui sont situés dans plusieurs

bâtiments. Les réseaux sans fil 802.11 sont actuellement les plus utilisés. En effet, en plus de leurs difficultés d'installation, les réseaux Ethernet câblés ont un coût élevé. Dans ce cas, les réseaux maillés sans fil permettent de fournir le même service que les réseaux Ethernet mais avec un coût beaucoup plus faible et une certaine facilité d'installation.

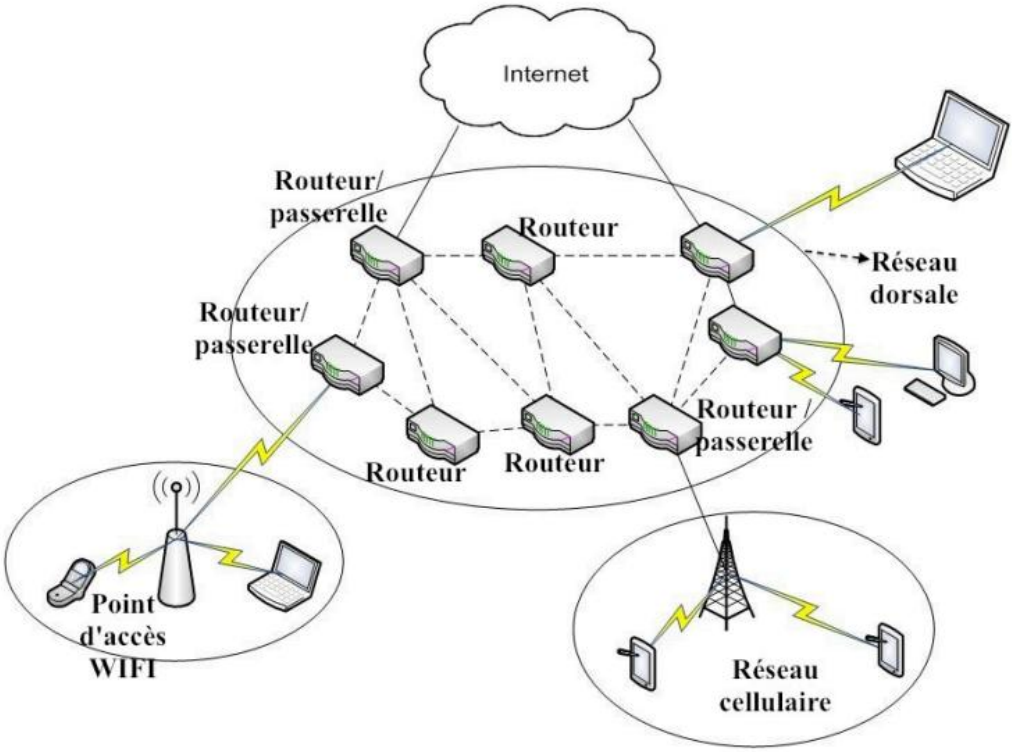


Figure 2.2 Architecture du réseau maillé sans fil

2.1.3 Les réseaux ad hoc véhiculaires (VANETs)

Les réseaux ad hoc véhiculaires VANETs (Vehicular Ad hoc Networks) sont composés de véhicules qui communiquent entre eux en exploitant les technologies sans fil, généralement celles appartenant à la famille 802.11. Dans ce type de réseau, la mobilité des nœuds (véhicules) est contrainte par les caractéristiques des routes et le mouvement des autres véhicules sur la route. Typiquement, la consommation d'énergie n'est pas un problème pour cette classe de réseaux multi-sauts, parce que les batteries des véhicules sont constamment rechargées. Les VANETs font partie de ce que l'on appelle les systèmes de transport intelligent (ITS , Intelligent Transport System) [67] dont le but est de réduire l'embouteillage et les nombre d'accidents routiers, d'améliorer la sécurité, l'efficacité dans les transports routiers. Un réseau VANET est composé de véhicules et d'unités de bords RSU (Road Side Board).

Les véhicules sont équipés de bornes OBU (On Board Unit). La borne OBU est l'interface de calcul, de localisation et d'émission/réception de messages dans le réseau. Le véhicule intelligent et son équipement, ainsi que l'intégralité des protocoles et des normes mise en place pour la communication sont appelés DSRC (Dedicated Short Range Communication) [69]. Les RSUs permettent de retransmettre les données entres les véhicules et vers les point d'entrée du réseau, et aussi diffuser les informations météorologiques et le trafic routier, etc.

Il existe trois modes de communications, véhicule à véhicule (V2V), véhicule à infrastructure (V2I) et le mode hybride. Le mode V2V ne requiert pas d'infrastructure, chaque véhicule représente un nœud et relaye les messages des autres véhicules. On parle de communication véhicule à infrastructure (V2I) lorsque le véhicule échange des informations avec l'infrastructure routière (RSU). Contrairement au mode ad hoc (V2V) le mode V2I offre une connectivité stable. Le mode hybride permet d'échanger des informations sur de longues distances grâce au RSU (V2I) tout en exploitant les forces de la topologie du réseau (V2V).

2.1.3.1 Caractéristiques des réseaux VANETs

Les réseaux ad hoc véhiculaire possèdent des caractéristiques qui les distinguent des autres réseaux sans fil multi-sauts :

- **Topologie hautement dynamique** : le déplacement des véhicules est caractérisé par des vitesses et des directions variantes. Par exemple une voie à grande vitesse (autoroute), une route nationale ou départementale, une localité urbaine (centre-ville). Ceci impacte la qualité et la durée de vie des liens radio entre les véhicules et donc, la topologie du réseau [56].
- **Mobilité prédictive :** le mouvement des véhicules est restreint par la topologie et les caractéristiques routes, ainsi que par les réactions vis-à-vis du mouvement des autres véhicules.
- **Pas de contraintes énergétiques et de puissance de calcul** : étant donné que les que les batteries des véhicules sont constamment rechargées. Ceux-ci peuvent être équipés de capteurs, en nombres et capacités suffisantes.
- **Localisation géographique** : les VANETs supportent les communications basées sur l'acheminement de données vers un groupe de véhicules désigné via sa localisation géographique. Ceci est en effet possible du fait que les véhicules soient équipés le plus souvent de systèmes de localisation plutôt efficaces.

2.1.3.2 Application des réseaux VANETs

- **Sûreté et de sécurité routière** : elles ont pour objectifs de réduire les risques d'accidents routiers [70]. Il existe de nombreux types d'applications qui entrent dans le cadre de la sécurité routière tels que la prévention des risques des collisions, les services de secours, l'avertissement sur les conditions de la route, etc.
- **Gestion du trafic** : l'échange d'information entre véhicules permet d'améliorer le trafic routier, en évitant les embouteillages et couturant les obstacles. Il aussi permet de réduire le temps de voyage et d'économiser la consommation de carburant.
- **Information et divertissement** : aussi appelées applications d'infotainment, dont l'objectif est d'améliorer le confort des conducteurs et des passagers. Elles permettent de fournir des informations d'utilité générale, telles que les informations météorologiques ou de localisation, d'accéder à des services basés sur Internet, comme des jeux en ligne ou de la messagerie instantanée [70].

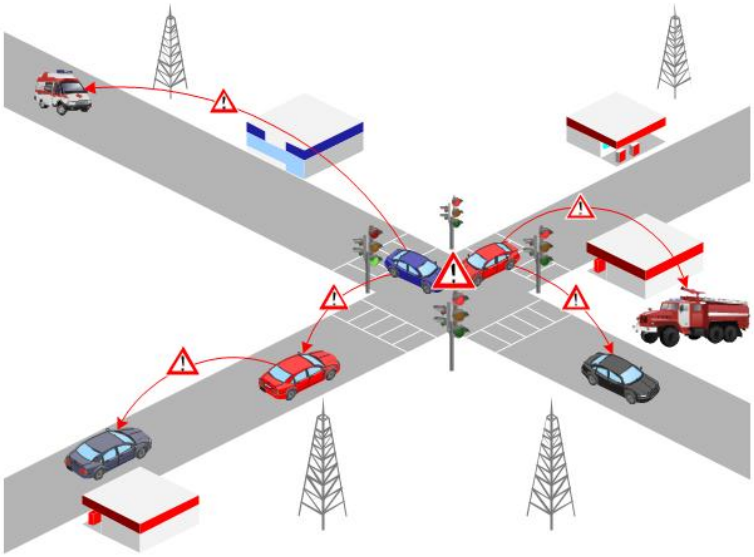


Figure 2.3 Exemple d'un réseau VANET

Le tableau 2.1 résume les caractéristiques et les différences entre les trois classes des réseaux multi-sauts.

2.2. Le routage dans les réseaux sans fil multi-sauts

Les protocoles de routage pour les réseaux filaires et pour internet sont inadéquats pour les réseaux sans fil multi-sauts dues à leurs architectures décentralisées, leurs topologies dynamiques, et le manque d'infrastructure préexistante. Dans [3] les auteurs abordent le routage dans les réseaux sans fil multi-sauts, et considèrent que ces derniers sont mieux adaptés aux communications multicast. Les concepteurs des protocoles de routage doivent prendre en considérations les caractéristiques intrinsèques aux réseaux sans fil multi-sauts. La mobilité des nœuds rend le processus de routage compliqué. Par exemple, les réseaux filaires dépendent de liens fixes qui sont toujours symétriques, alors que les liens sans fils sont parfois asymétriques. Les protocoles de routage pour les réseaux sans fil multi-sauts doivent être capables de refléter les changements topologiques fréquents. Non seulement les protocoles de routages doivent être capables de gérer les mises à jour fréquentes et les routes expirées au niveau de la table de routage, mais aussi considérer le coût du routage.

Les protocoles de routage peuvent être classés selon plusieurs critères. La plupart des protocoles proposés pour les réseaux filaires et sans fil sont des protocoles de routage dynamiques, c'est-à-dire, la table de routage peut changer pour prendre en considération les changements de routes, ce qui est différent du routage statique où la table de routage ne peut pas changer. Depuis l'émergence des réseaux multi-sauts, plusieurs protocoles de routage ont été proposés et évalués par rapport à plusieurs aspects tels que l'évolutivité (scalabilité), la sécurité, la performance, la qualité de service et l'adaptabilité. Dans [4] les auteurs présentent une étude détaillée sur le routage ad hoc, ils classent les protocoles de routages en trois grandes catégories comme le montre la figure 2.4.

Caractéristiques	**MANETs**	**WMNs**	**VANETs**
Mode de communication	Multi-sauts	Multi-sauts	Multi-sauts (V2V) et un saut (V2I)
Composants du réseau	Nœuds mobiles	Nœuds mobiles (clients), routeurs et passerelles mesh	Véhicules et unités de bord (RSU)
Taille du réseau	10^2	10^3	10^3
Mobilité	Moyenne	Moyenne pour les clients, les routeurs et passerelles sont stationnaires	Haute
Trajectoire	Aléatoire	Aléatoire	Prédéfinie
Contraintes	Energie, puissance de calcul et bande passante	Bande passante, déploiement fixe	Bande passante, déconnexions fréquentes et partitionnement du réseau

Tableau 2.1 Caractéristiques des réseaux multi-sauts

Dans les protocoles de routage plats (flat en anglais) tous les nœuds possèdent les mêmes fonctionnalités et peuvent tous participer au processus de routage. Cependant, les protocoles de routage hiérarchiques assignent aux nœuds des rôles différents, et organisent le réseau en groupes (clusters). Le routage hiérarchique permet une plus grande évolutivité, et une réduction du nombre de paquets de contrôle par rapport aux protocoles plats, particulièrement dans le cas des réseaux de grande taille. Cluster gateway switch routing (CGSR) [5] et hierarchical state routing (HSR) [6] sont des exemples de protocoles de routage hiérarchiques. HSR est un protocole de routage à état de lien avec une bonne gestion de la localisation, il combine les concepts de groupes dynamiques et niveaux hiérarchiques. D'abord il partitionne le réseau en un ensemble de groupes suivant un schéma récursif de clustérisassion, où un nœud doit être élu pour représenter le reste des membres. Les représentants des groupes dans un niveau l, deviennent des membres dans le niveau l + 1. Ces nouveaux membres s'organisent de la même manière en un ensemble de groupes, et ainsi de suite pour le reste des niveaux. L'objectif est de réduire le trafic de routage en termes d'échange d'information, de stockage, et de traitement.

Le routage géographique se fait en deux étapes, d'abord la localisation du nœud destinataire, ensuite l'acheminement des paquets vers ce nœud. Dans la première étape, le nœud source doit déterminer la position géographique du nœud destinataire à l'aide d'un service de localisation tel que le système de localisation mondial (GPS). Ensuite le routage se fait essentiellement en se basant sur la position des voisins immédiats et la position du nœud destinataire. Les informations de localisation géographique peuvent améliorer la performance du routage, cependant il arrive parfois que ces informations manquent de précision, ce qui peut générer des routes inexactes. Plusieurs protocole de routage géographique ont été proposés tels que : Geographic addressing and routing [7] ; et location aided routing protocol [8].

2.2.1 Protocoles proactifs

Le routage proactif visent à maintenir une vue globale et cohérente sur la topologie du réseau, l'établissement des routes se fait avant que les demandes en soit effectuées. Dans un protocole proactif, chaque nœud possède à tout instant un chemin vers n'importe quel autre nœud du réseau. Les protocoles proactifs diffusent des messages de contrôle à travers le réseau soit périodiquement, soit suite à des évènements particuliers tels que la perte d'un lien. Dans ce qui suit, nous citons deux exemples de protocole de cette catégorie à savoir DSDV [10] et OLSR [11].

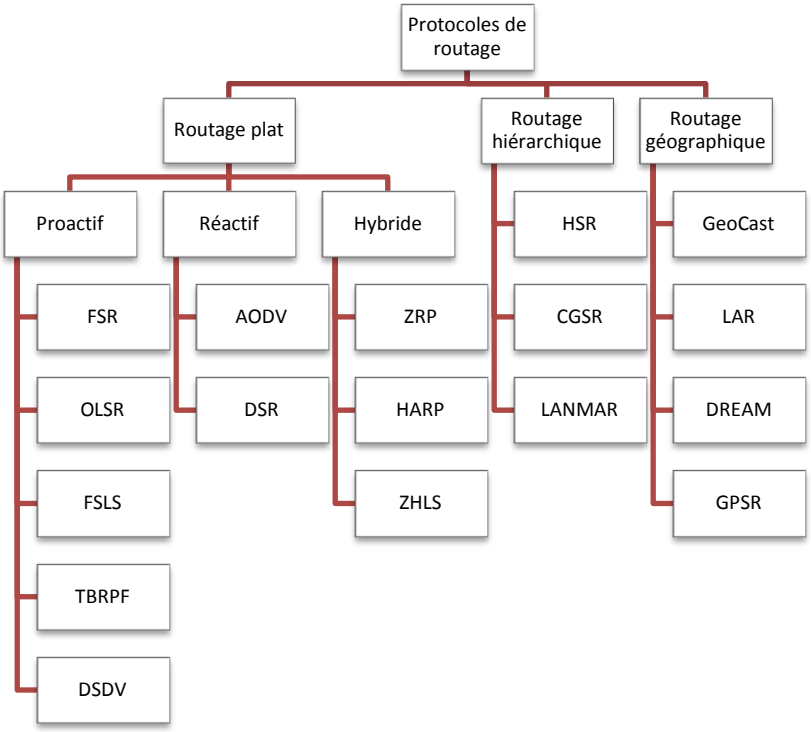


Figure 2.4 Classification des protocoles de routage ad hoc

2.2.1.1 DSDV

Le protocole DSDV (Destination Sequenced Distance Vector Protocol) est un protocole à vecteur de distance adapté aux réseaux multi-sauts. DSDV introduit un attribut supplémentaire appelé numéro de séquence aux champs de la table de routage, afin de permettre aux nœuds d'évaluer la fraicheur des routes et ainsi éviter la formation des boucles de routage. Un nœud incrémente son numéro de séquence à chaque fois qu'il y a un changement au niveau de son voisinage (c.à.d. l'ajout ou la suppression d'un lien). Afin d'avoir une vision complète du réseau et maintenir la cohérence des tables de routage. Les nœuds s'échangent périodiquement ou lors d'un changement de topologie les mises à jour (mises à jour incrémentales), ou la totalité de leurs tables de routage (mises à jour complète). Puisque la fraicheur d'une route est représentée par son numéro de séquence, les nœuds retiennent les routes qui ont les plus grands numéros de séquence lors de la réception des paquets de mises à jour. En cas d'égalité des numéros de séquence, c'est la route la plus courte c.-à-d. celle ayant le moins de sauts qui sera sélectionnée. En dépit des améliorations proposées par DSDV tel que l'élimination des boucles de routage par l'utilisation des numéros de séquence, ce dernier est long à converger, génère un surcout considérable, et requiert des mises a jours régulières même lorsque le réseau est inactif.

2.2.1.2 OLSR

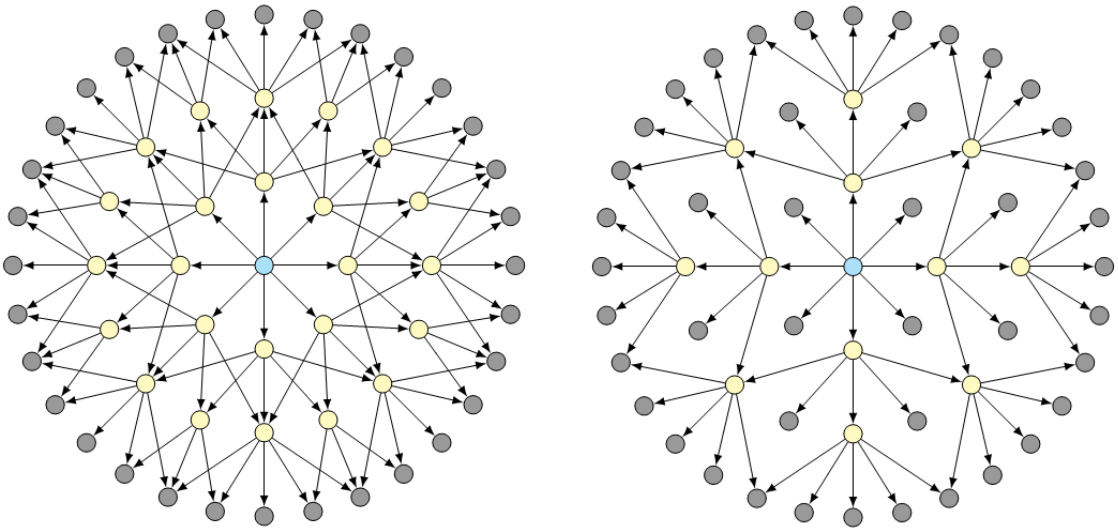
Le protocole de routage OLSR (Optimized link State Routing) proposé par Jacquet et al [11], est une optimisation du routage à état de lien classique, adapté au contexte ad hoc pour fournir une meilleure performance. Le concept clé de ce protocole est l'utilisation des relais multipoints (MPR) pour réduire les diffusions redondantes des paquets de contrôle dans une même région du réseau. Afin d'assurer les fonctions de découverte et de maintenance de la topologie du réseau, OLSR utilise des échanges périodiques de messages de contrôle de taille réduite. Les MPRs d'un nœud N représentent un sous ensemble de ses voisins à un saut qui permet de joindre tous les voisins à deux sauts. Ainsi, lors d'une diffusion, tous les voisins du nœud N peuvent recevoir et traiter le message mais uniquement les nœuds appartenant à l'ensemble MPR du nœud N peuvent le retransmettre ce qui diminue considérablement le nombre de messages diffusés dans le réseau (voir la figure 2.5).

2.2.2 Protocoles réactifs

Contrairement aux protocoles proactifs qui veillent à maintenir constamment un panorama de la topologie du réseau, les protocoles réactifs établissent les routes uniquement à la demande des nœuds ayant des données à transmettre. Ces protocoles sont plus adaptés aux réseaux à forte mobilité, et permettent d'économiser l'utilisation de la bande passante et la consommation d'énergie. Nous présentons dans ce qui suit les protocoles DSR [12] et AODV [13] qui utilisent respectivement les méthodes état de lien et vecteur de distance.

2.2.2.1 DSR

Le protocole DSR (Dynamic Source Routing) proposé par Johnson et Maltz [12], emploie l'approche "routage source". Dans cette approche, le nœud source détermine la séquence complète des nœuds formant le chemin à travers lequel les données seront transmises. Le nœud (source) souhaitant transmettre des données, lance un processus de découverte de routes, en diffusant un message de requête de route à travers tous le réseau. Si la découverte de route est réussie, le nœud source reçoit un message réponse de route qui liste la séquence des nœuds formant le chemin vers la destination. Le message réponse de route contient donc un champ contenant la liste des nœuds visités durant la diffusion de la requête dans le réseau. Pour maintenir les routes et garantir leurs validités, le protocole DSR utilise les messages erreur de route. Lorsqu'un nœud détecte un problème dans la transmission (perte de liens), il envoie un message erreur de route au nœud a partir duquel il a reçu le paquet. Le message d'erreur contient l'adresse du nœud qui a détecté l'erreur et celle du nœud qui le suit dans le chemin. Quand le nœud source reçoit le paquet erreur de route, il supprime le nœud concerné par l'erreur de la route sauvegardée, et toutes les routes qui contiennent ce nœud sont tronquées à ce niveau-là.



a. Inondation pure (24 transmissions pour atteindre tous les nœuds à 3 sauts)

b. Inondation par relais MPR (12 transmissions pour atteindre tous les

Figure 2.5 Avantage de l'utilisation des MPR

2.2.2.2 AODV

AODV (Ad hoc On demand Distance Vector) est un protocole de routage à vecteur de distance inspiré des protocoles DSDV et DSR. Les routes sont établies à la demande, afin de réduire le nombre de messages diffusés, et limiter l'impact des modifications topologiques uniquement aux routes actives. AODV utilise un champ indiquant la fraicheur de la route appelé numéro de séquence, (dans les messages de contrôle et au niveau de la table de routage), afin de maintenir la cohérence des informations de routage, et ainsi prendre toujours les routes les plus récentes. Quand 'un nœud source désire établir une route vers un nœud distant (ex. le nœud A dans la figure 2.6 désire envoyer des données au nœud J) et qu'il ne possède pas de route disponible au niveau de sa table de routage (route expiré ou non existante), il diffuse un message de demande de route RREQ (Route REQuest).

La demande de route contient un identifiant (RREQ ID) associé à l'adresse du nœud source qui servira à identifier la demande de route de façon unique. A chaque fois qu'un nœud reçoit une nouvelle demande de route (qui n'a pas été traité auparavant), il sauvegarde l'identifiant de de la demande de route (RREQ ID) et l'adresse du nœud source dans son historique pour une durée bien déterminé (Path_ Discovery_Time). Lorsqu'un nœud intermédiaire (le cas des nœuds C, B et D dans la figure 2.6) qui ne possède pas de route valide vers la destination reçoit la demande de route RREQ,

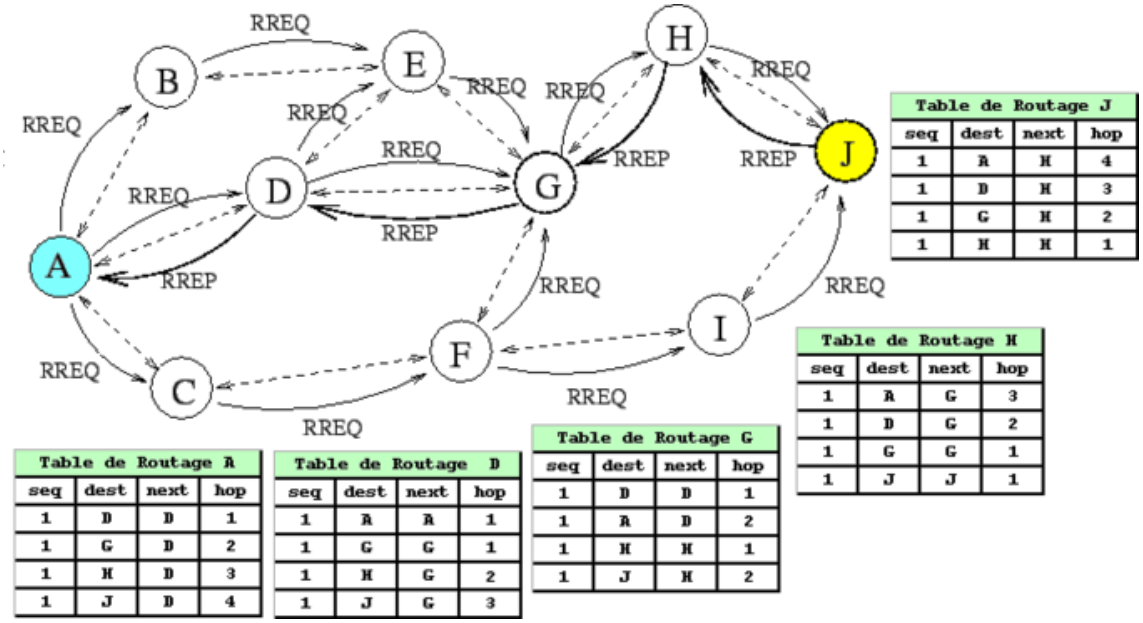


Figure 2.6 Découverte de route en AODV

il ajoute ou met à jour le voisin émetteur au niveau de sa table de routage. Si la RREQ a été déjà traitée, le nœud récepteur l'abandonne et ne la rediffuse pas. Sinon, il ajoute ou met à jour la route vers le nœud source au niveau de sa table de routage. Il incrémente ensuite le nombre de sauts dans la demande de route et la rediffuse. À la réception de la demande de route, la destination (nœud J) ajoute ou met à jour dans sa table de routage une route vers le nœud voisin a partir duquel il a reçu la RREQ (nœud H) ainsi qu'une route vers le nœud source (nœud A). Le nœud destination (nœud J) génère ensuite une réponse de route RREP qu'il transmit en unicast vers le prochain saut en direction du nœud source. En AODV même un nœud intermédiaire peut générer une réponse de route s'il possède une route valide et fraiche vers la destination à condition que le nœud source l'autorise à le faire (bit Destination_only de la RREQ mis à 0). Lors de la réception d'une réponse de route par un nœud intermédiaire, celui-ci ajoute ou met à jour la route vers la destination, ensuite retransmit la réponse de route RREP en unicast (après l'incrémentation du nombre de sauts) via la route inverse qui a été créé lors du passage de la RREQ. Lorsque le nœud source reçoit la RREP, une route bidirectionnelle est établie entre lui et le nœud destination, et ainsi la transmission de paquets de données peut commencer. Pour maintenir les routes, AODV utilise des diffusions périodiques de messages HELLO (une RREP diffusé aux voisins à un saut), et des diffusions (ou des unicasts) des messages d'erreur RERR (Route ERRor) pour signaler les ruptures de liaison et invalider les routes non fiables.

2.2.3 Protocoles de routage hybrides

Afin de profiter des avantages des protocoles proactifs et réactifs, un nouveau type de protocole a été proposé «les protocoles hybrides ». Ce troisième type combine le routage proactif et réactif, en appliquant l'approche proactive pour acquérir à l'avance des routes dans un voisinage de quelques sauts et l'approche réactive au-delà des sauts couvert. L'association des deux méthodes partage le réseau en zones. Lorsqu'un nœud reçoit un message pour une destination qui n'appartient pas à sa zone, il le redirige vers une autre zone. Nous présentons dans ce qui suit le protocole ZRP [13] et ZHLS [20].

2.2.3.1 ZRP

Le protocole ZRP (Zone Routing Protocol) proposé en 1997 par Haas et Pearlman [13] se fût le premier protocole de routage ad hoc hybride. Il spécifie pour chaque nœud une zone de routage incluant tous les nœuds à K sauts. Les nœuds qui sont exactement à distance K sont appelés nœuds périphériques. ZRP se compose d'un protocole de routage proactif (IARP [16] Intrazone Routing Protocol) et d'un protocole de routage réactif (IERP [17] IntErzone Routing Protocol). Le protocole proactif IARP permet de trouver toute les routes à k-sauts (par exemple k=3) et fournir une vue détaillée de la zone de routage. Le protocole réactif IERP permet de trouver des routes vers les nœuds situés à une distance supérieure à K. Afin de créer sa zone de routage, chaque nœud doit disposer du voisinage à un saut qui pourra être obtenu par un protocole de couche liaison ou en utilisant un protocole dédié à cet usage comme le NHDP [18] (Neighborhood Discovery Protocol). La figure 2.7 montre les zones de routage pour les nœuds 1 et 4, les nœuds 3, 4, 7, 8 et 10 sont des nœuds périphériques de la zone de routage du nœud 1. IERP permet d'établir les routes vers les destinations en dehors de la zone de routage du nœud source (à plus de K sauts). Le nœud source envoie une demande de route à tous les nœuds périphériques appartenant à sa zone de routage, en utilisant le protocole BRP (Bordercast Resolution Protocol) [19]. En utilisant la topologie obtenue par l'IARP, le protocole BRP construit un arbre de multicast (bordercast tree) pour fournir les différents chemins pour atteindre les nœuds périphériques. Les nœuds périphériques vérifient à leurs tours si la destination existe au niveau de leurs zones, si c'est le cas, une réponse de route est envoyée à la source. Sinon la demande de route sera diffusée aux nœuds périphériques qui, à leurs tours, effectuent la même opération. Chaque nœud doit faire attention à ne pas retransmettre la demande de route aux nœuds l'ayant déjà traité pour éviter les boucles de routage et optimiser le temps de découverte de route. ZRP à l'avantage de réduire le nombre de messages de contrôle envoyés à travers le réseau par rapport aux protocoles proactifs et réactifs, et de réduire le temps de latence pour la découverte de nouvelles routes.

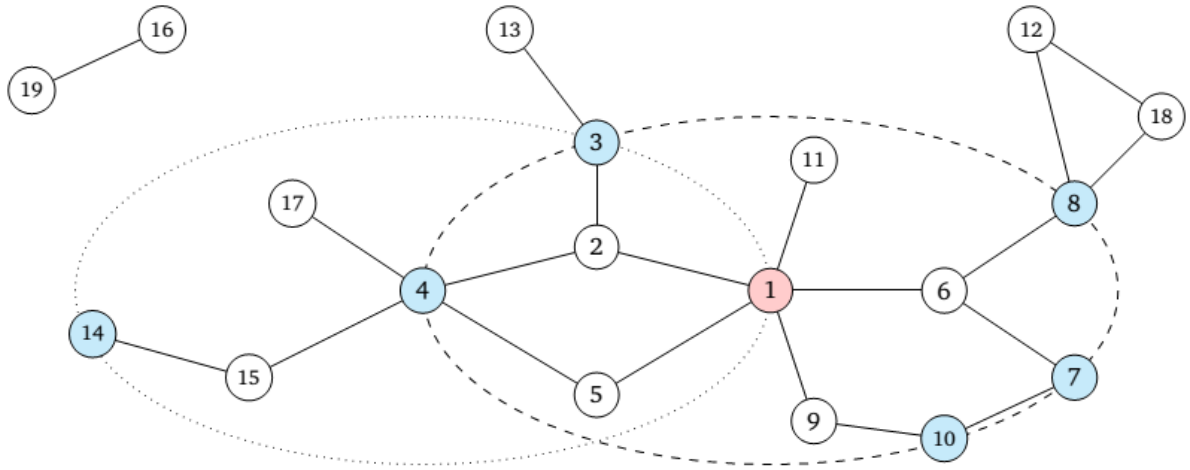


Figure 2.7 Zone de routage (ZRP)

2.2.3.2 ZHLS

Le protocole de routage ZHLS (Zone-Based Hierarchical Link State) est un protocole hiérarchique qui divise le réseau en zones géographiques qui ne se chevauchent pas (figure 2.8), ou chaque nœud est localisé par ses coordonnées GPS (Global Positioning System). Il emploi l'approche réactive pour trouver la zone du nœud destination. Chaque nœud dispose de deux tables de routage, une table intra-zone qui maintient les liens entre les nœuds appartenant à la même zone et les liens vers des nœuds d'autres zones. Et une table inter-zone pour maintenir une carte des zones qui fournit un plan des connexions entre les zones.

2.1. Sécurité, vulnérabilités et attaques dans les réseaux sans fil multi-sauts

2.1.1 Concepts de base en sécurité

Les objectifs de base en sécurité pour les réseaux multi-sauts sont les mêmes que ceux des réseaux filaires: la confidentialité, l'authentification, la disponibilité, l'intégrité, et la non répudiation des utilisateurs.

2.1.1.1 Confidentialité

Consiste à conserver les informations (données) à l'abri de ceux qui ne sont pas autorisé à les connaître (contenu invisible). Ce service garantie une communication privée entre les nœuds, et protège contre la divulgation non autorisée d'informations. Elle est principalement basée sur la cryptographie, en particulier les algorithmes de chiffrement. La confidentialité est un service très important pour les réseaux sans fil multi-sauts, parce que les liens sans fil sont faciles à écouter.

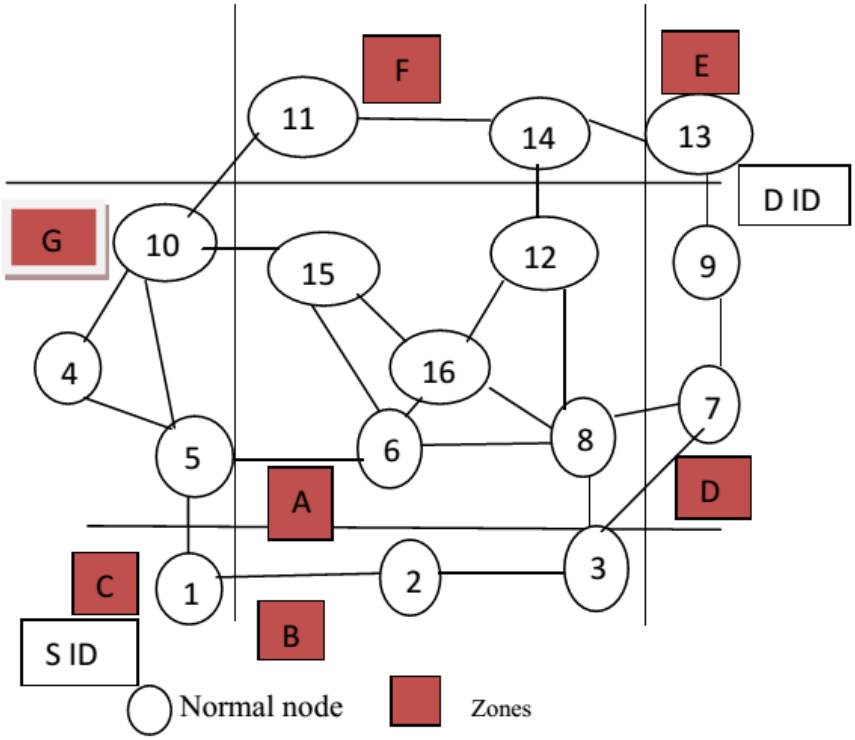


Figure 2.8 Topologie d'un réseau routé par ZHLS

Plusieurs solutions à base de cryptographie ont été proposées, dans [22] un protocole sécurisé pour un transfert fiable des données a été proposé afin d'assurer la confidentialité pour les réseaux ad hoc mobiles. Il propose de décomposer le message crypté en fragments séparés, qu'il transmit à travers plusieurs chemins indépendants. Ainsi l'écoute des données devient difficile, parce que l'attaquant doit collecter et décrypter tous les fragments pour comprendre le message. Dans [23] les auteurs proposent un schéma de sécurité à base de clé partagé pour assurer la confidentialité des échanges. Bien que les algorithmes de chiffrement symétrique et asymétriques puissent être utilisés pour assurer la confidentialité, la gestion de clés dans le contexte ad hoc constitue un vrai défi.

2.1.1.2 Authentification

L'authentification est le processus permettant la vérification de l'identité d'un nœud dans le réseau. On peut distinguer deux types d'authentification, l'authentification des identités qui assure l'intégrité de l'identité de l'interlocuteur, et l'authentification des données [21] qui fournit une garantie que les données sont vraiment envoyées par l'interlocuteur.

2.1.1.3 Disponibilité

Elle consiste à assurer la continuité du service fourni et la disponibilité des ressources pour les utilisateurs autorisés du réseau, même en présence d'une attaque. Les nœuds doivent assurer la continuité des services fournit par le réseau quelle que soit l'ampleur du déni de service. Les réseaux

multi-sauts sont particulièrement vulnérables aux différentes formes du déni de service, dues à leurs caractéristiques inhérentes.

2.1.1.4 Intégrité

Ce service assure que le trafic n'a pas été altéré ou modifié sans autorisation préalable durant sa transition de la source à la destination. Ce service constitue une protection contre l'insertion, la substitution, la suppression, et la fabrication du trafic (messages de contrôle ou de données). L'intégrité vise à protéger le trafic contre la modification intentionnelle (par un nœud malveillant) ou accidentelle (erreurs liées à la propagation radio). Les mécanismes de chiffrement et de signature numérique sont généralement utilisés pour garantir l'intégrité, cependant leurs application dans le contexte ad hoc est difficile. D'abord parce que les nœuds sont limités en énergie et en puissance de calcul, de plus des vérifications d'intégrité au niveau de chaque saut s'imposent.

2.1.1.5 Non-répudiation

La non-répudiation assure que les nœuds ne peuvent pas nier leurs actions ou rejeter la validité de leurs échanges. Par exemple, pendant la transmission d'un message, la non-répudiation assure que l'émetteur ne peut pas nier l'envoi et le destinataire ne pas nier la réception du message. Généralement, la non-répudiation est assurée en utilisant les certificats numériques, une technologie qui permet de vérifier l'identité du nœud en utilisant sa clé privée.

2.1.2 Vulnérabilités des réseaux sans fil multi-sauts

Vulnérabilités, menaces, et attaques sont des termes souvent utilisés dans le domaine de la sécurité des réseaux. Nous définissons ces termes d'après [21] comme suit:

- **Vulnérabilité :** est n'importe quel défaut matériel ou logiciel qui laisse le réseau ouvert pour une potentielle exploitation.
- **Menace :** toute intention ou méthodes utilisées pour exploiter une faiblesse dans un système, ça peut être une opération, un équipement ou une fonction.
- **Attaque :** tentative de contourner le contrôle de sécurité d'un système ou d'un réseau avec l'intention de nuire.

Les réseaux multi-sauts sont vulnérables dans leurs fonctionnements, un nœud malveillant peut compromettre les opérations du réseau à n'importe qu'elle couche : physique, MAC (Medium Access Control), réseau, transport, ou application. Les réseaux multi-sauts sont exposés aux attaques tels que : l'écoute (ou l'espionnage), les interférences et le brouillage fréquentiel au niveau de la couche physique, l'égoïsme (selfishness) au niveau de la couche MAC, ou le déni de service au niveau de la

couche réseau. Le fonctionnement fragile des réseaux multi-sauts est dû aux vulnérabilités suivantes :

- **Medium sans fil:** l'utilisation des liaisons sans fil rend le réseau vulnérable à des attaques telles que l'écoute du canal (l'espionnage) et le brouillage.
- **Utilisation d'algorithmes de routage coopératifs :** chaque nœud dans le réseau acte comme un routeur, envoie les paquets des autres nœuds, et participe dans la découverte et la maintenance des routes. Le nœud malveillant peut exploiter la propriété de coopération pour troubler et perturber le routage.
- **Limitation en énergie:** les nœuds dépendent d'une batterie limitée, ainsi leurs capacités de traitement et de calcul sont limitées. Le nœud malveillant peut exploiter cette propriété en forçant les nœuds à traiter et à transmettre des paquets inutiles, afin de consommer leurs énergies.
- **Bande passante limitée :** typiquement les réseaux sans fil ont une bande passante limitée, ce qui permet aux nœuds malveillants de la consommer facilement et d'empêcher ainsi les communications légitimes.
- **Capacité de calcul limitée:** généralement dans ce type de réseau les nœuds ont une capacité de calcul limitée, des fréquences d'horloge réduites et des mémoires de petite taille, ce qui peut être ajouté aux vulnérabilités existantes.
- **Interception physique:** les nœuds changent de position fréquemment et bougent aléatoirement, ce qui les rendent facilement capturables.
- **Service transitoire :** puisque les nœuds changent de positions fréquemment, tout service fournit par le réseau est éphémère. Ce qui rend la distinction entre le comportement malveillant et le comportement normal une tache très difficile.
- **Pas de frontières physiques:** la topologie du réseau est définit par les positions géographiques des nœuds mobiles, rendant ainsi la définition des frontières, et des points d'accès au réseau difficile. Par conséquent, le contrôle d'accès aux réseaux multi-sauts est plus difficile a géré comparé aux réseaux filaires.

2.1.3 Attaques contre les réseaux sans fil multi-sauts

Les attaques dans les réseaux sans fil multi-sauts peuvent être classées selon plusieurs critères, nous citons dans ce qui suit les plus importants entre eux:

- **Position de l'attaquant:** Selon la position du nœud malveillant par rapport au réseau, on peut distinguer deux type d'attaques :

- Les attaques externes : effectuées par des nœuds qui ne font pas partie du réseau. Ce type d'attaques peut etre eviter en utilisant les techniques de chiffrement.
- Les attaques internes : effectuées par des nœuds appartenant au réseau et disposant de l'ensemble des connaissances associées à ce statut (matériel cryptographique, table de routage, etc).

- **Objectif de l'attaque:** l'objectif de l'attaque est etroitement lié au profil de l'attaquant. De ce point de vue on peut distinguer deux types d'attaquants : l'attaquant rationnel qui aspire tirer un profit direct ou indirect suite a son attaque. L'attaquant irrationnel qui vise juste a perturber les services du réseau sans tirer aucun profit, tel qu'il est le cas pour le déni de services (DoS) [28].
- **Impact de l'attaque :** une attaque passive se limite à l'écoute du trafic des communications pour obtenir des informations importantes sur le réseau. L'attaque active peut supprimer, modifier ou injecter les paquets, il s'agit de modifier le comportement du réseau de façon arbitraire par rapport au comportement normal [28].

Afin de citer et présenter toutes les attaques qui existent dans la litterature, nous proposons une classification de l'ensemble des attaques selon l'impact et la couche concernée comme le montre la figure 2.9.

2.1.3.1 Attaques passives

L'attaquant se contente de capturer le trafic pour l'analyser et extraire des informations sensibles. Ce qui peut conduire à la divulgation d'informations confidentielles et critiques sur le réseau tels que : la topologie du réseau, et la position des nœuds.

a Ecoute indiscrète

Lors de la transmission d'un message sur le medium sans fil, chaque nœud équipé d'émetteur-récepteur peut écouter le message, et extraire les informations contenues si le message n'est pas chiffré. Il est impossible aux nœuds émetteur et récepteur de détecter cette attaque [24].

b Analyse du trafic

Le nœud malveillant observe le trafic et analyse le modèle de communication notamment: le volume de données transmises et les caractéristiques de la transmission sur les différentes liaisons du réseau. L'analyse du trafic permet au nœud malveillant d'obtenir des informations critiques comme : les nœuds qui ont un rôle important dans le réseau, les identités et les positions des nœuds. Même si le trafic est chiffré, cette attaque peut être utilisée pour extraire des informations sur le réseau [24].

c Divulgation d'emplacement

Il s'agit de révélation d'informations sensibles sur le réseau telles que la position des nœuds, ou même la topologie complète du réseau. Ce qui peut causer de sérieux dégâts dans les réseaux exigeant un niveau de sécurité très élevé tels que les réseaux militaires [30].

2.1.3.2 Attaques actives

Contrairement aux attaques passives, les attaques actives tentent d'injecter des informations dans le réseau et/ou d'interagir avec les nœuds du réseau. Dans ce qui suit nous présentons les attaques en les classant selon la couche sur laquelle elles interviennent (figure 2.9).

a Couche physique

a.1 *Interception physique*

L'attaquant intercepte physiquement le nœud et extrait des informations tels que les clés de chiffrement, et gagner ainsi un accès illimité aux services du réseau. Il peut même détruire le nœud définitivement en l'endommageant matériellement [24].

a.2 *Brouillage et interférences*

Il s'agit de transmettre un faux signal sur des radios fréquences utilisées par les nœuds du réseau. Si le signale généré est assez puissant il pourra submerger le signal cible et interrompre la communication. Cette attaque se manifeste dans sa forme la plus typique sous forme de bruit aléatoire. Le brouillage peut être lancé à partir de location distante, et peut mettre N nœud hors service avec k nœud attaquants distribués aléatoirement, ou K est très petit par rapport à N [25].

b Couche MAC

La majorité des réseaux sans fil multi-sauts (particulièrement les MANETs) utilisent au niveau de la couche MAC le standard 802.11 [26] en mode DCF (distributed coordination function) qui gère et maintient la communication entre les nœuds par la coordination d'accès au canal radio partagé. Un nœud malveillant peut exploiter les vulnérabilités de ce protocole pour effectuer les attaques suivantes [27] :

b.1 *Détournement d'accès*

- A chaque fois qu'un signal RTS (Request To Send) est reçu, le nœud malveillant génère un signal qui entre en collision avec le signal CTS (Clear To Send), pour empêcher les autres nœuds de transmettre leurs données.
- L'envoi continu des faux RTS et CTS avec des paramètres prétendants une transmission d'un grand volume de données, afin de laisser les autres nœuds attendre indéfiniment la libération du canal.

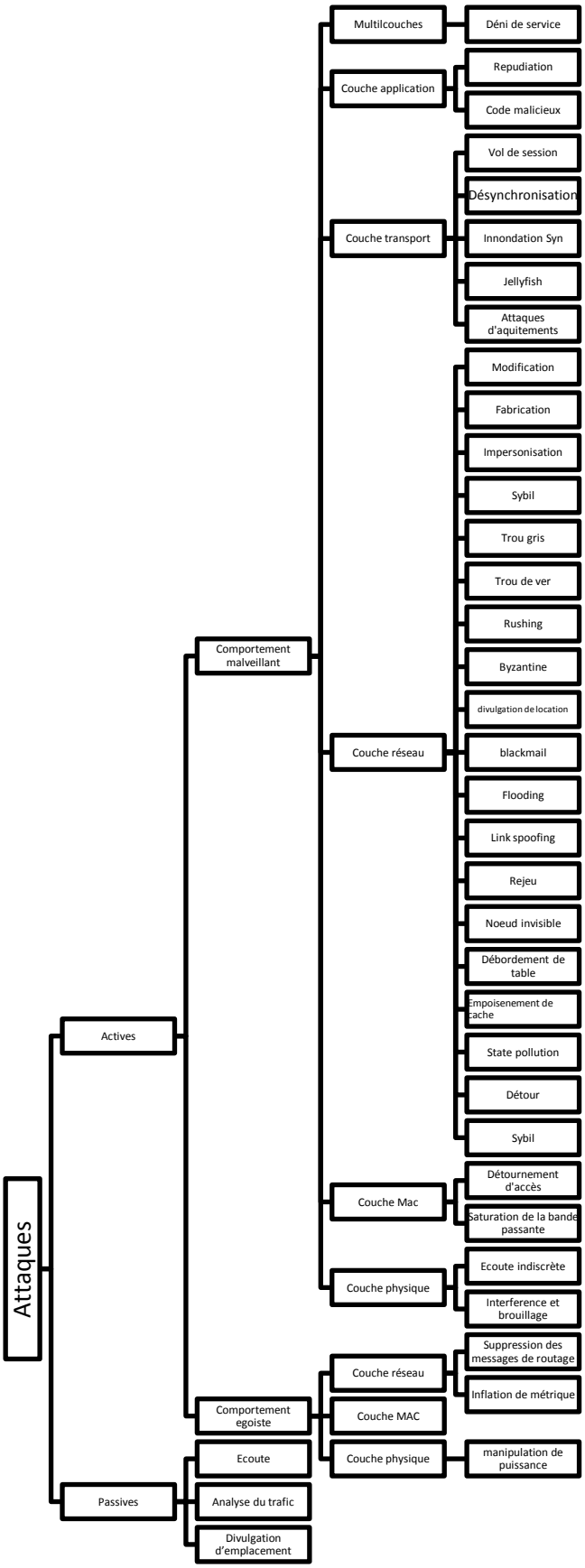


Figure 2.9 Classification des attaques contre les réseaux multi-sauts ad hoc

b.1 *Saturation de la bande passante*

Transmission d'un grand volume de données inutiles entre les nœuds malveillants pour consommer la bande passante dans leur proximité et empêcher ainsi les transmissions légitimes.

c Couche réseau

c.1 *Suppression*

Cette attaque consiste tout simplement à supprimer les messages de contrôle ou de données.

c.2 *Modification*

La modification illégitime du contenu des messages reçus, est l'attaque la plus fréquente dans les réseaux multi-sauts. Le nœud malveillant peut facilement modifier le contenu des messages qui transitent par lui, il peut par exemple : mettre de fausses adresses ou manipuler les métriques de routage pour rediriger le trafic. Cette attaque peut causer de sévères perturbations dans le réseau tels que: l'utilisation de routes non optimales ou erronées, partitionnement du réseau, ou la perte de connectivité [31].

c.3 *Rejeu*

Consiste à enregistrer une séquence de trafic et la réinjecter ensuite dans le réseau.

c.4 *Fabrication*

Le nœud malveillant fabrique un message et l'injecte dans le réseau, dans le but d'attirer le trafic, perturber le fonctionnement du réseau, ou consommer la bande passante et l'énergie des nœuds.

c.5 *Usurpation d'identité*

Elle constitue généralement la première étape de la majorité des attaques. Le nœud malveillant tente de cacher sa vrai identité en se faisant passer pour un autre nœud, pour pouvoir ainsi recevoir les messages destinés à ce dernier, accéder au réseau, ou lancer des attaques.

c.6 *Trou noir & trou gris*

Ce type d'attaques se fait en deux étapes. D'abord le nœud malveillant exploite les vulnérabilités du protocole de routage pour attirer le trafic en annonçant des informations de routage attractives. Ainsi il sera sélectionné lors du processus de découverte de route et fera partie de plusieurs routes. Ensuite, il supprime tous les paquets reçus de la part des autres nœuds. Une version plus sophistiqué de cette attaque appelée trou gris. Dans cette attaque le nœud malveillant supprime sélectivement les paquets reçus pour éviter d'être détecter par les mécanismes de sécurité [31]. Sur la figure 2.10 le nœud 3 attire le trafic vers lui en envoyant une fausse réponse de route au nœud source 1.

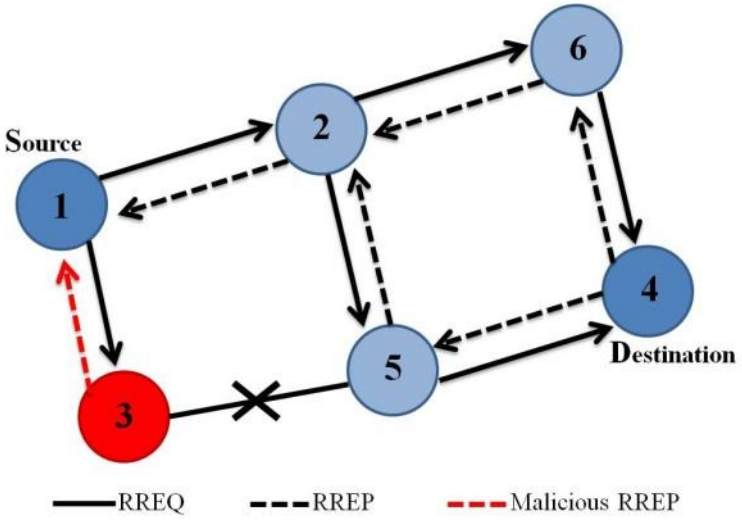


Figure 2.10 Attaque du trou de vert

c.7 *Trou de ver*

Le trou de ver est une attaque qui peut être lancée par un attaquant externe et être réussie même en présence d'un système d'authentification et de chiffrement [33]. Elle nécessite la coopération d'au moins deux nœuds se trouvant sur deux zones différentes du réseau et connectés via un tunnel de communication privé. Le tunnel est établi de telle façon que chaque paquet intercepté par le nœud malveillant à une des extrémités du tunnel soit retransmis vers le nœud malveillant se trouvant à l'autre extrémité du tunnel (figure 2.11). Il existe plusieurs méthodes d'établir un tunnel telles que [29]:

- *Liaison virtuelle* : les paquets interceptés par un des nœuds malveillants sont encapsulés, chiffrés et envoyés à travers une route, préalablement établie par le protocole de routage utilisé, vers le deuxième nœud malveillant distant, qui à son tour déchiffre les paquets reçus et les relaie à ses voisins.
- *Liaison physique directe*: les nœuds malveillants sont reliés par une liaison rapide (p. ex. liaison filaire ou une liaison sans fil longue distance).

c.8 *Rushing*

Cette attaque concerne les protocoles de routage réactifs, elle exploite leurs procédures de découverte de route, qui consiste à ne traiter que la première demande de route reçue, et à supprimer toutes les autres copies de la même demande. Le nœud malveillant tente de devancer les autres nœuds lors de la diffusion de la demande de route pour atteindre la destination en premier, et ainsi faire partir de la route. Afin de faire parvenir sa demande de route avant les autres, le nœud malveillant peut par exemple la transmettre avec une puissance de signal élevée, ou ignorer les

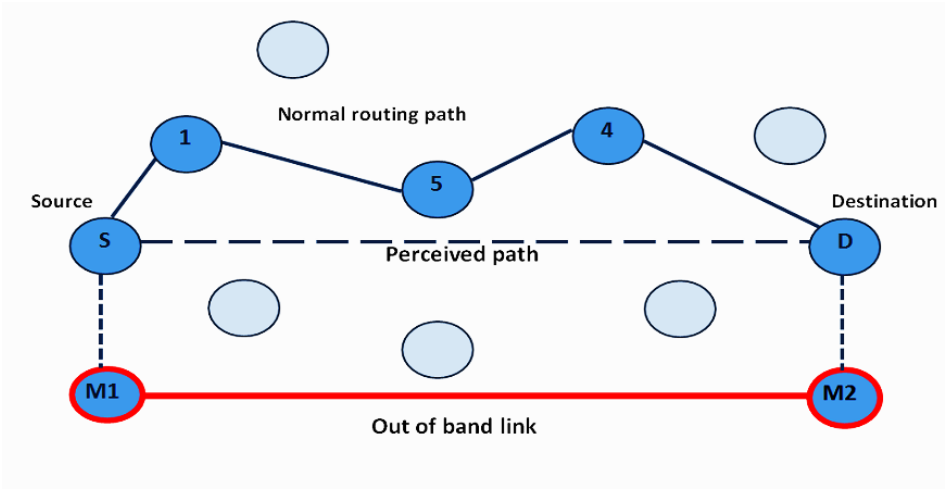


Figure 2.11 Attaque du trou de ver

délais de transmission spécifiés par le protocole de routage et aussi les délais d'accès au medium de communication [34].

c.9 *Sybil*

L'attaque Sybil (en anglais Sybil) a été introduite et décrite par Douceur [35] dans le contexte des réseaux pair-à-pair. Chaque nœud dans un réseau multi-sauts doit être identifié par une adresse unique pour participer dans le routage, cependant il n'y pas d'autorité centrale pour vérifier les identités. Le nœud malveillant peut exploiter cette vulnérabilité, en usurpant plusieurs identités et en simulant l'ensemble des nœuds associés a à ces identités [36]. Le nœud malveillant peut utiliser des identités aléatoires ou les identités des autres nœuds pour créer des confusions dans le processus de routage, ou pour servir comme étape préliminaire pour d'autres attaques (figue 2.12). Par exemple, le nœud malveillant peut donner l'illusion à un nœud de posséder plusieurs routes vers une destination tandis que toutes ces routes passent par la même entité physique [29]. Dans la figure 2.13, le nœud A est connecté aux nœuds B et C, et au nœud malveillant M1. Si M1 se fait passé pour d'autres nœuds (M2, M3 et M4) cela donnera l'illusion au nœud A d'avoir six voisins au lieu de trois.

c.10 *Débordement de la table de routage*

Cette attaque concerne les protocoles de routage proactifs. Le nœud malveillant crée constamment des demandes de routes et/ou des messages de mise à jour pour des nœuds inexistants jusqu'à déborder la table de routage du nœud(s) victime(s). Par conséquent le routage des données et l'établissement de routes ne sera plus possible [40].

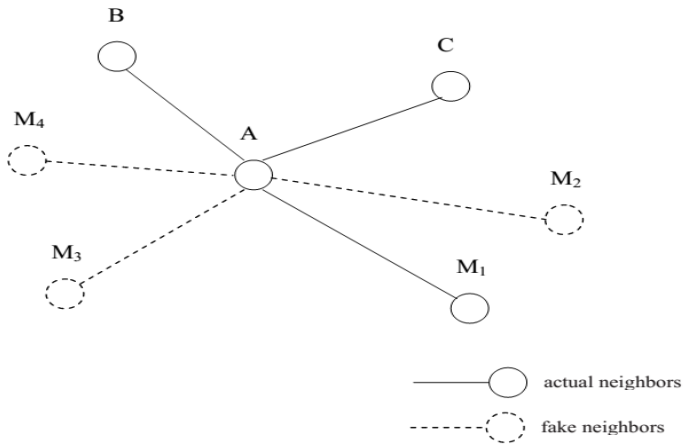


Figure 2.12 Attaque Sybil

c.11 *Privation du sommeil*

Ou consommation des ressources, dans cette attaque le nœud malveillant s'attaque à la disponibilité des nœuds. Une technique consiste à inonder le réseau avec des messages fabriqués afin de générer un trafic inutile et provoquer ainsi de sévères congestions. Le nœud malveillant peut se contenté de modifier les messages reçus pour que ces derniers circulent dans le réseau le plus longtemps possible, et provoque des traitements additionnels consommant ainsi la bande passante du réseau et les ressources des nœuds.

c.12 *Comportement égoïstes.*

Il s'agit des nœuds qui s'abstiennent de participer au routage pour conserver leurs ressources (particulièrement leurs batteries). Le comportement égoïste compromet la disponibilité du routage parce que dans certains cas il peut empêcher l'établissement des routes. Bien que le nœud égoïste n'attaque pas activement le réseau, mais l'effet de son comportement se fait ressentir en terme d'efficacité et de fluidité de communication [25].

d Couche transport

Les objectifs du protocole TCP comme un protocole de couche transport dans les réseaux sans fil multi-sauts, inclus l'établissement des connexions de bout en bout, la livraison fiable des données, le contrôle de flux, le contrôle de congestion, la libération des connexions. Comme sur Internet, les nœuds dans un réseau multi-saut sont vulnérables aux attaques tels que : l'inondation par SYN ou le détournement de session. Cependant, les réseaux multi-sauts sont caractérisés par un taux d'erreur assez élevé comparés aux réseaux filaires. Le protocole TCP ne dispose d'aucun mécanisme pour distinguer entre les pertes liées à la congestion, une panne, ou une attaque, TCP réduit sa fenêtre de congestion quand il détecte des pertes, ce qui peut dégrader significativement les performances du réseau [27].

d.1 *Détournement de session*

Dans cette attaque, le nœud malveillant usurpe l'identité du nœud victime, déduit le numéro de séquence en cours d'utilisation, et ensuite il l'écarte du réseau en épuisant ses ressources. Ainsi le nœud malveillant peut poursuivre la session en cours avec le nœud cible. Par la suite le nœud malveillant peut injecter des données erronées pour provoquer des conflits de numéro de séquences, causant l'échange d'un grand volume de paquets d'acquittement TCP ACK [42].

d.2 *Désynchronisation*

Le nœud malveillant envoie des faux messages aux deux nœuds en communication, ou à l'un d'eux, pour demander la retransmission des messages manquants. Ainsi les messages demandés seront retransmis, et si le nœud malveillant arrive à maintenir un bon timing de transmission, il peut empêcher l'établissement de toute communication utile entres les deux nœuds [43].

d.3 *Inondation des SYN*

Pour que deux nœuds puissent communiquer en utilisant le protocole TCP, ils doivent d'abord établir une connexion sur trois étapes (Handshake) comme le montre la figure 2.13. D'abord le nœud A envoie une demande de connexion SYN, le nœud B répond avec un segment SYN/ACK, et enfin le nœud A accuse réception avec un segment SYN. Le nœud malveillant peut exploiter les étapes décrites précédemment, en ouvrant un grand nombre de connexions TCP avec le nœud cible, mais sans finaliser l'établissement de ces connexions, dans le but de déborder les buffers de ce dernier, de telle sorte qu'il ne pourra plus accepter d'autre demande de connexions. Le nœud malveillant peut maintenir ces connexions demies ouvertes, en envoyant de façon répétée des demandes de connexions avant leurs expirations [25].

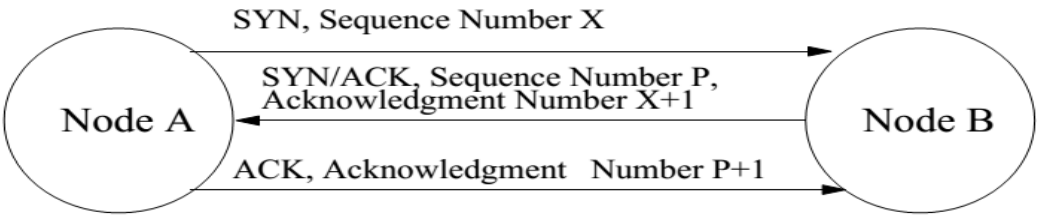


Figure 2.13 Etablissement de connexion (Handeshake) TCP

e Couche application

La couche application est aussi vulnérable aux attaques que les autres couches. La couche application contient les données utilisateur, et normalement supporte plusieurs protocoles comme HTTP, SMTP, TELNET, FTP, ce qui fournit plusieurs vulnérabilités et points d'accès aux attaques.

e.1 *Attaque par code malveillant*

Les codes malveillants tels que les virus, les vers, les spywares, et les Trojan, peuvent attaquer les systèmes d'exploitation et les applications. Généralement ces programmes malveillants se propagent à travers le réseau, et lui cause une baisse de performance, et parfois le mettent hors service. Le nœud malveillant peut effectuer des attaques similaires dans les réseaux multi-sauts [25].

e.2 *Attaque par répudiation*

Au niveau de la couche réseau, les pare-feu sont utilisés pour surveiller le trafic entrant et sortant. Au niveau de la couche transport le chiffrement est utilisé pour protéger les connexions. Cependant ces solutions ne résolvent pas les problèmes liés à l'authentification et la non-répudiation. La répudiation réfère à la dénégation de participation dans toutes ou une partie de la communication. Par exemple, une personne mal intentionnée peut nier d'avoir fait une opération d'achat avec sa carte de crédit, ou nier n'importe qu'elle transaction bancaire en ligne, ce qui représente l'attaque par répudiation typique dans un système commerciale [25].

f Multicouche

f.1 *Déni de service*

Le déni de service DoS (Denial Of Service) est une attaque multicouche qui cible la disponibilité des nœuds et des services fournis par le réseau, la majorité des attaques abordées précédemment peuvent être considéré comme des dénis de service. Wood et Stankovic [38] définissent l'attaque DoS comme étant n'importe qu'elle action qui diminue les capacités du réseau et l'empêche d'accomplir ses fonctions correctement ou dans les délais. Cette attaque peut avoir plusieurs formes, par exemple un nœud malveillant peut modifier et fabriquer un grand volume de messages inutiles et les envoyer au nœud victime afin de consommer son énergie et le mettre hors service. Cette forme est connue dans la littérature comme la privation de sommeil (sleep deprivation) [39], elle est aussi appelée consommation d'énergie. La perturbation du routage est une autre forme de l'attaque DoS. Elle consiste à perturber le fonctionnement du protocole de routage en disséminant des fausses informations de routages (métriques manipulées, faux paquets de contrôle, etc), ou en créant des boucles de routage. DDoS (Distributed Denial of Service) est une version plus sophistiquée du DoS, dans laquelle l'attaque est exécutée de manière distribué par un groupe de nœuds malveillants dans le but de paralyser tout le réseau [15].

2.2. Solutions et mécanismes de sécurité

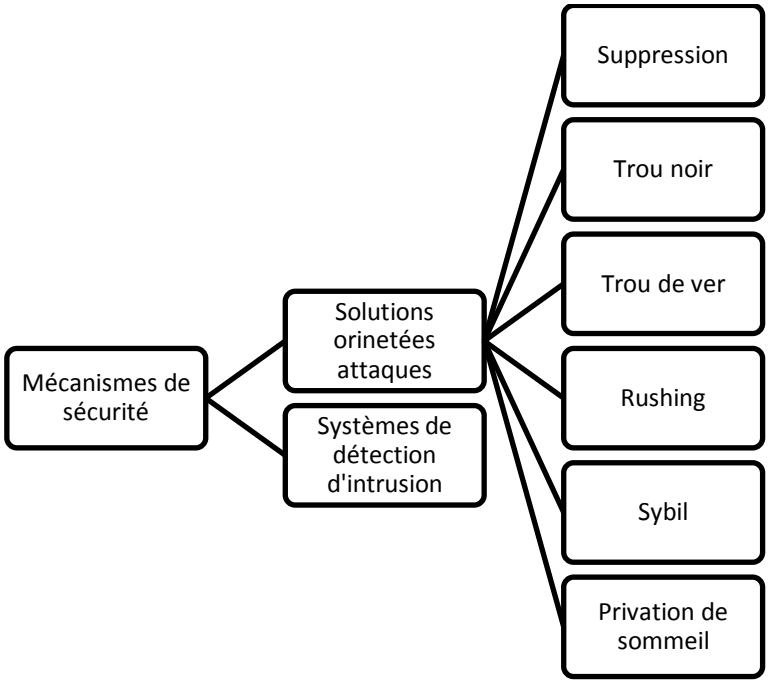
Dans cette section nous allons présenter un état de l'art des solutions et des mécanismes qui ont été proposés dans la littérature pour protéger les réseaux multi-sauts et particulièrement les réseaux MANETs contre les attaques abordées précédemment. Selon le nombre d'attaques considérés et identifiées par le mécanisme, nous classons les solutions proposées dans la littérature en deux classes : solutions orientées attaque et systèmes de détection d'intrusion (figure 2.14). Nous appelons solutions orientées attaque ou solutions dédiés, les mécanismes conçus pour détecter et résoudre un seul type d'attaque. Les systèmes de détection d'intrusion quant eux sont capables de détecter plusieurs types d'attaques. Comme le montre la figure 2.14, les solutions dédiées sont classés selon le type d'attaque qu'elles traitent. Les systèmes de détection d'intrusion seront discutés en détails dans le chapitre 3.

2.2.1 Solutions orientées attaque

2.2.1.1 Suppression des paquets

Beaucoup de recherches ont été faites pour détecter et contrer la suppression des paquets. Par exemple dans [55], les auteurs ont proposé un mécanisme de protection basé sur la coopération des nœuds participants dans le réseau. Chaque nœud du réseau doit surveiller les comportements de ses voisins. Quand le mécanisme détecte une suppression, il demande l'opinion de ses voisins en invoquant une approche distribuée. Ensuite il collecte les valeurs de confiance attribuées au nœud suspect par ses voisins. Si la majorité des voisins attribuent au nœud suspect un niveau de confiance réduit, alors une alarme générale sera soulevée pour informer tous les autres nœuds du réseau. Comparer au Watchdog [56], le mécanisme proposé réduit considérablement le nombre de fausses alarmes.

D'autres approches à base de systèmes de surveillance de voisin (Neighbour Watch System) ont été proposées pour détecter les nœuds malveillants qui suppriment les paquets de données [57] [51]. Des mécanismes de détection à base de conservation de flux ont été proposés dans [58] et [59]. Dans lesquels les nœuds surveillent continuellement leurs voisins, maintiennent une liste des nœuds qu'ils écoutent et examinent périodiquement leurs comportements. Les nœuds malveillants sont détectés, en comparant le taux de paquets supprimés avec un seuil préétabli. Une version adaptative de cet algorithme a été proposée dans [60]. Pour assurer l'adaptabilité de leurs l'algorithme ils proposent une méthode qui calcule le seuil de malveillance et des stratégies qui considèrent les changements du réseau (conditions et objectifs de gestion).



.

Figure 2.14 Taxonomie des mécanismes de protection de la couche réseaux

Dans [61] les auteurs ont proposé un protocole auto-organisé SCAN pour sécuriser la retransmission des paquets de routages et des données. Il est supposé qu'un nœud peut écouter les paquets reçus par son voisin, et qu'il possible au nœud d'avoir une copie de la table de routage de son voisin. Un paquet est considéré comme supprimé, si le nœud superviseur n'a pas écouté la retransmission du paquet de la part du nœud voisin. SCAN exploite la collaboration localisée et les informations de validation croisée pour protéger le réseau. Pour mitiger l'effet de cette attaque, Marti et al [56] ont proposé un mécanisme de détection composé d'un Watchdog, et d'un évaluateur de chemin Pathrather. Le Watchdog utilise l'écoute passive des transmissions (interfaces en mode promiscuité) pour vérifier si les paquets sont correctement relayés. Lorsque le nombre des paquets supprimés dépasse un certain seuil, une notification est envoyée au nœud source pour lui prévenir de la suppression. Le Pathrater utilise les informations collectées par le Watchdog pour évaluer la fiabilité des routes. Il attribue à chaque nœud un score qui représente son degré de fiabilité. D'autres propositions comme WATCHERS [62], utilisent le principe de conservation du flux, d'autres utilisent les paquets d'acquittement ou de test.

2.2.1.2 Trou noir

Plusieurs mécanismes ont été proposés pour résoudre cette attaque. Dans [47] le mécanisme TOGBAD propose l'utilisation d'un graphe de topologie pour détecter le trou noir. L'algorithme compare le nombre des voisins qu'un nœud déclare avoir et le nombre réel des voisins dans le graphe. TOGBAD a été développé pour le protocole proactif OLSR, dans lequel les informations sur la

topologie peuvent être obtenues. Cependant ce mécanisme ne sera pas efficace pour les protocoles réactifs, où l'acquisition d'informations sur la topologie complète n'est pas possible. Dans [48] les auteurs ont proposé une méthode de détection du trou noir dans le protocole AODV. Chaque fois qu'une réponse de route est reçue, le nœud récepteur initie un processus d'évaluation de l'émetteur. Les voisins échangent et partagent leurs opinions sur l'émetteur, et la décision est faite à base d'un seuil fixe. Un nœud est considéré comme malveillant si le nombre de paquets reçus par ce dernier est disparate du nombre de paquets qu'il a envoyé. Cependant cette proposition souffre d'un taux élevé de faux positifs, parce qu'elle se base sur un seuil fixe et ne possède aucun moyen pour s'adapter aux changements liées à la mobilité.

Dans [49] Zhang et al, proposent un mécanisme de détection basé sur la vérification des numéros de séquence des réponses de route. Ils considèrent le scénario où l'attaquant est un nœud intermédiaire, et suggèrent, qu'à chaque fois qu'un nœud envoie une RREP au nœud source, le nœud intermédiaire devrait aussi générer une requête de numéro de séquence au nœud destination. Le nœud destination répond en envoyant un paquet contenant son numéro de séquence au nœud source. Le nœud source ensuite vérifie la fraicheur de la route en comparant le numéro de séquence de la RREP reçue de la part du nœud intermédiaire (le suspect) avec le numéro de séquence de la réponse de route reçu de la part de la destination. Bien que cette proposition détecte la modification du numéro de séquence, l'introduction de deux nouveaux paquets avec chaque réponse de route, augmentera considérablement le nombre de paquets de contrôle. De plus les nœuds doivent s'assurer que le nœud malveillant n'a pas supprimé ou modifié les messages de demande et de réponses de séquence [53].

2.2.1.3 Trou de ver

La majorité des solutions proposées dans littérature utilisent des techniques de calcul de distances, afin de repérer les paquets qui transitent au-delà de la portée de transmission. En général ces techniques exigent du matériel spécial et utilisent des informations géographiques ou temporelles. Dans [44] la position géographique (Geaographic packet leash) du nœud est ajoutée dans le paquet (protégé par un chiffrement) pour estimer la distance parcourue par le paquet depuis l'émetteur. Il est aussi possible d'utiliser des informations temporelles (Temporel packet leash) tel que la date de création du paquet, si les horloges des nœuds sont synchronisées. Grace aux informations temporelles le récepteur peut estimer la distance parcourue par le paquet en examinant la durée de sa transmission. Il est évident que rien n'empêche un nœud malveillant possédant le matériel cryptographique de falsifier les informations géographiques et temporelles pour éviter la détection. Une autre proposition pour empêcher le trou de ver, consiste à utiliser les antennes directionnelles [45]. Afin d'assurer une découverte stricte du voisinage, la direction de transmission est ajoutée dans

les paquets de découverte de voisinage. A la réception des transmissions, les antennes directionnelles permettent aux nœuds de déterminer la zone à partir de laquelle la transmission est reçue. Les nœuds acceptent uniquement les voisins pour lesquels la direction de transmission est opposée à la zone directionnelle dans laquelle la transmission est reçue. Les paquets reçus à partir des directions inattendues sont ignorées. Cette solution peut être utile pour les applications qui utilisent déjà des antennes directionnelles, autrement la complexité associée à cette solution ne peut être justifiable [53].

2.2.1.4 Rushing

Dans cette attaque le nœud malveillant peut utiliser plusieurs techniques afin de faire parvenir sa demande de route en premier (voir section 2.3). Hu et al [34] proposent un protocole d'authentification mutuelle qui permet la détection des voisins. Le protocole proposé utilise un timing bien synchronisé pour vérifier si un nœud X qui prétend être le voisin d'un nœud Y, se trouve vraiment à la portée du nœud Y. Avant de retransmettre la demande de route, le nœud doit effectuer une de détection sécurisé de voisin avec le nœud qui représente le saut précèdent. En ce qui concerne le nœud malveillant qui ignore le Backoff pour atteindre la destination en premier. Les auteurs proposent la retransmission aléatoire des demandes de route à retransmettre. En d'autre terme la demande de route qui arrive en premier n'est pas forcement celle qui sera retransmise. Dans [50] les auteurs ont proposé le protocole SMT (Secure Message Transmission), qui garantit un transfert sécurise des données de bout en bout. Ils proposent SMT principalement pour protéger les opérations de transfert des données, en ce qui concerne la procédure de découverte de route qui est vulnérable aux attaques de routage tels que Rushing, ils suggèrent le protocole de routage sécurisé SRP (Secure Routing Protocol) [51]. Dans [52] Rawat et al ont évalué et tester SRP contre les attaques de routages, et ils ont conclue qu'il résiste efficacement aux attaques [53].

2.2.1.5 Sybil

L'utilisation des certificats de confiance via une autorité des certificats est de loin la solution la plus cité dans la littérature pour résoudre l'attaque Sybil. Douceur [35] considère les certificats de confiance comme la seule solution pour éliminer complètement l'attaque Sybil. Piro et al [36] ont démontré que la mobilité peut être utilisée pour détecter les identités multiples dans les réseaux multi-sauts. Un nœud peut détecter l'attaque Sybil en gardant et en suivant les identités (c.-à-d. adresse IP or MAC) des nœuds dont il écoute la transmission. Les groupes de nœuds qui sont écoutées ensemble peuvent être détectés comme des nœuds malveillants. Ils proposent aussi un échange d'opinion entre les nœuds de confiance afin d'augmenter la précision de la détection. Dans [54] Monica et al utilisent les tests radio pour détecter les fausses identités dans le réseau. Ils évaluent la puissance et la performance des différents tests radio incluant : le test des émetteurs

simultanés, test optimisé des émetteurs simultanés, récepteurs simultanés et test des collisions forcés.

2.2.1.6 Privation de sommeil

Yi et al [63] proposent un mécanisme de prévention basé sur la supervision des nœuds voisins, où chaque nœud maintient une file d'attente capable de gérer la priorité des demandes de routes reçus. Le mécanisme diminue la priorité des demandes de route générées par un nœud particulier, s'il observe qu'un grand nombre de demandes de route a été reçu de la part de ce dernier. Cependant cette proposition ne prend pas en considération le cas des nœuds qui relayent un nombre important de paquets, dues à leurs positions stratégiques dans le réseau. Dans [64] les auteurs ont proposé un modèle analytique qui utilise des caractéristiques de détection à base de flux. Dans [65] les auteurs considèrent différents scénarios, tels que : l'envoi répétitif des demandes de routes ; et l'incitation des nœuds à faire des tâches gourmandes en énergie. Ils proposent une architecture sécurisée capable de garantir un minimum de batterie même en présence d'attaques. L'architecture utilise deux caractéristiques : suivi de la signature d'énergie ; et l'authentification multicouche. Yu et al [66] considèrent deux scénarios de privation de sommeil : l'inondation par demandes de route ; et l'inondation par injection de paquets de données. Ils analysent théoriquement les deux scénarios du point de vue du nœud malveillant pour évaluer la probabilité de succès de l'attaque. En supposant que les nœuds peuvent s'authentifié via un mécanisme d'authentification à clé publique, ils proposent un mécanisme basé sur la supervision des voisins. A la réception d'une demande de route, chaque nœud doit vérifier un ensemble de conditions telles que : la loyauté de la source et de la destination ; le RREQ ID de la demande de route ; le timing ; et aussi si la route comporte un nœud marqué comme malveillant.

2.2.1.7 Comparaison

Maintenant nous examinons les mécanismes de sécurité orientés attaques abordés précédemment. D'abord nous les analysons en se basons sur des paramètres tels que : la technique de détection, le type d'attaque ciblé, le protocole de routage utilisé, l'architecture, et la réponse à l'attaque. Le tableau 2.2 présente un résumé des principales solutions orientées attaque. L'architecture de la majorité des mécanismes proposés est distribué ou hiérarchique. L'architecture hiérarchique permet de fournir un certain niveau d'évolutivité (scalabilité) au réseau et au mécanisme. Dans une architecture hiérarchique les nœuds sont organisés en tiers, et à chaque niveau de tiers est assigné un rôle. Dans les réseaux multi-sauts, la technique du Clustering est largement employée pour organiser les nœuds en une structure hiérarchique. L'architecture hiérarchique réduit la charge de routage, comparé aux architectures distribués plates, parce que le nœud communique avec un petit groupe de nœuds, et non pas avec la totalité du réseau.

En examinons La colonne « source des données » dans le tableau 2.2, nous remarquons que l'écoute en mode promiscuité des transmissions des nœuds voisins est un mécanisme particulièrement populaire pour la collection des données. Il s'agit d'écouter les transmissions à partir ou à destination d'un nœud particulier se trouvant dans la portée radio, sans même être impliqué dans la transmission. L'alternative principale à l'écoute des transmissions des voisins, est le Reporting direct, où les données sont collectées directement à partir des nœuds. Dans ce cas, les nœuds s'échangent les données d'observation entre eux en échangeant des paquets spéciaux. L'avantage du monitoring des voisins, c'est qu'il fournit une source de données indépendante pour le mécanisme de sécurité, tandis que les données rapportées directement peuvent être compromises. Cependant, due à la nature du médium sans fil, certaines transmissions ne peuvent pas être écoutées, même si les nœuds sont très proches. Ainsi les mécanismes qui utilisent l'écoute en mode promiscuité doivent prendre en considération cette contrainte, et gérer l'incertitude des données qu'ils utilisent. Il faut noter aussi que la plupart des mécanismes orientés attaque utilisent des techniques de détection applicables à un protocole particulier, et contre une attaque particulière. Par exemple dans [63], ils proposent l'utilisation d'une file d'attente à priorité pour gérer les RREQ reçus, uniquement pour contrer l'inondation par RREQ dans le protocole AODV. Evidemment, cela limite l'applicabilité de leur mécanisme.

Dans la plupart des solutions proposées la réponse à l'attaque n'est pas considérée. Bien que certains mécanismes la considèrent, ils répondent généralement en isolant le nœud malveillant, en le contournant et en supprimant tout trafic reçu de sa part. Le fait qu'un mécanisme ne peut détecter qu'un seul type d'attaque, signifie que : (1) il faut implémenter et combiner plusieurs mécanismes pour détecter un grand nombre d'intrusions ; (2) lors de la combinaison de plusieurs mécanismes pour détecter plusieurs attaques, l'interaction entre ces mécanismes doit être prise en considération ; (3) la combinaison de plusieurs mécanismes impose une charge importante sur le réseau en terme de trafic, puissance de calcul et énergie; et (4) la détection de nouvelles type d'attaques (ou attaques inconnues) n'est pas possible. Dû aux limites des mécanismes orientés attaque, les recherches se sont orientées vers les systèmes de détection et prévention d'intrusion générique qui peuvent fournir une meilleure protection contre plusieurs types d'attaques, qu'elles soient connues à l'avance ou non.

2.2.2 Détection d'intrusion

Il est presque impossible de concevoir un système assez correctement pour que les intrusions ne puissent pas avoir lieu. Les techniques de prévention d'attaque, mises en place à l'intérieur ou à l'extérieur du système, tels que l'authentification et le chiffrement, peuvent être utilisés comme une première ligne de défense pour réduire la possibilité d'attaques. Cependant ces techniques sont

Mécanisme	architecture	Attaques considérées	Technique de détection	Réponse	Protocole de routage	Source des données	Contributions
Sen et al. [55]	Distribuée	Suppression de paquets de données	Approche basée sur la confiance	Isolation des nœuds malveillants	Non précisé	Ecoute en mode promiscuité	améliorations par rapport au mécanisme proposé dans [56]
Yang et al. [61] SCAN	Distribuée	Suppression de paquets de données	Validation croisée d'information	Isolation des nœuds malveillants	AODV	Surveillance collaborative	Livraison sécurisé des paquets
Gonzalez et al. [58] [59]	Hiérarchique	Suppression de paquets de données	Conservation du flux	Isolation des nœuds malveillants	AODV	Ecoute en mode promiscuité	Application de la conservation du flux
Marti et al. [56]	Distribuée	Suppression de paquets de données	Watchdog et Pathrater	Isolation des nœuds malveillants	DSR	Ecoute en mode promiscuité	Détection par Watchdog et évaluation de route par pathrater
Yi et al. [63]	Distribuée	Privation du sommeil	File d'attente prioritaire pour gérer les demandes de routes	Isolation des nœuds malveillants	AODV	Ecoute en mode promiscuité	Prévention de privation de sommeil par inondation de RREO
Yu & Ray [66]	Distribuée	Privation du sommeil	Vérification de certaines conditions	Non-considérée	DSR	demandes de route des nœuds voisins	Prévention de privation de sommeil par injection de trafic
Martin et al. [65]	Non spécifié	Privation du sommeil	Authentification multi niveau	Non-considérée	Non précisé	Requêtes de service à un serveur SSH via ipaq	Analyse l'impact de la privation de sommeil en temps réel
Padillia et al. [47]	Hiérarchique centralisée	Trou noir	Compare le nombre de voisins déclarés avec le graphe de topologie	Non-considérée	OLSR	Graphe de topologie	Détection du trou noir dans le protocole OLSR
Zhang et al. [49]	Distribuée	Trou noir	Vérification du numéro de séquence de la réponse de route RREP	Non-considérée	AODV, SAODV	Réponse de route issue des nœuds intermédiaires	Détection du trou noir
Medadian et al. [48]	Distribuée	Trou noir	Trouver une route fiable	Non-considérée	AODV	Surveillance des voisins	Détection du trou noir via la surveillance des voisins
Hu et al. [34]	Distribuée	Rushing	Protocole d'authentification mutuelle	Non-considérée	DSR	Demandes de route transmises dans la portée radio	Prévention de l'attaque Rushing dans les MANETs
Douceur [35]	Centralisée	Sybil	Certificats de confiance	Non-considérée	Non précisé	Certificats gérés par une autorité de certification	Démontrer que sans autorité centrale, il est difficile d'éliminer les identités Sybil
Piro et al. [36]	Distribuée	Sybil	Enregistrement des identités et le modèle de mobilité	Non-considérée	AODV	Ecoute en mode promiscuité	Utilisation de la mobilité pour identifier les identités Sybil

Tableau 2.2 Mécanismes de sécurité orientés attaque

conçues pour prévenir des attaques connues. De plus, elles ne sont pas efficaces contre les attaques internes. Quelles que soient les techniques de prévention mises en place contre les intrusions, il existe toujours des failles qui pourront être exploitées par le nœud malveillant. La détection d'intrusions peut donc être considérée comme une action complémentaire à la mise en place des mécanismes de préventifs. L'idée de système de détection d'intrusion a été introduite en 1980 par James Anderson [71]. Cependant le concept n'a pas eu beaucoup d'intérêt. C'est la publication du modèle de détection d'intrusion de Denning en 1987 [72] qui a marqué réellement le lancement du domaine. Les systèmes de détection d'intrusion (souvent abrégé IDS, traduction de Intrusion Detection System) permettent de déceler un comportement malveillant de quelque nature que ce soit dans le réseau. Dans les réseaux sans fil multi-sauts, les IDSs analysent les paquets entrants et sortants dans le réseau afin d'identifier les intrusions. Dans le chapitre prochain, nous étudions en détails la détection d'intrusion, ainsi que les différentes techniques et mécanismes existants dans la littérature.

2.1. Conclusion

Dans ce chapitre nous avons d'abord présenté les réseaux sans fil multi-sauts, et particulièrement les trois classes émergentes : MANETs, WMNs et VANETs. Nous avons examiné les caractéristiques et les domaines d'application de chaque classe. Ensuite, nous avons abordé le routage dans les réseaux sans fil multi-sauts, où nous avons décrit le mode de fonctionnement de différents protocoles de routage. Puis nous nous somme penché sur l'aspect sécuritaire des réseaux ad hoc. Nous avons introduit les concepts basiques de la sécurité, et passé en revue les différentes vulnérabilités des protocoles de routage. Après, nous avons examiné toutes les attaques pouvant cibler les réseaux multi-sauts sur les différentes couches de la pile protocolaire. Finalement, nous avons discuté les mécanismes de sécurité proposés dans la littérature pour protéger les réseaux multi-sauts contre les différentes attaques.

Chapitre 3 Détection d'intrusion dans les réseaux ad hoc mobiles

Les mécanismes sécuritaires préventifs tels que l'authentification et le chiffrement, peuvent être utilisés comme une première ligne de défense pour réduire la possibilité d'attaques. Cependant ces techniques sont conçues pour prévenir des attaques connues. De plus, elles ne sont pas efficaces contre les attaques internes. Quelles que soient les techniques de prévention mises en place contre les intrusions, il existe toujours des failles qui pourront être exploitées par le nœud malveillant. La détection d'intrusion peut donc être considérée comme une action complémentaire à la mise en place des mécanismes préventifs. Les systèmes de détection d'intrusion sont capables d'améliorer la sécurité et de protéger le réseau contre un large éventail d'attaques externes et internes.

Dans la section 3.1 nous présentons la détection d'intrusion et les techniques de détection d'intrusion. Les réseaux ad hoc mobiles ont des caractéristiques différentes des réseaux conventionnels, et soulèvent de nouveaux défis pour les solutions de sécurité existantes. Les problèmes liés à la détection d'intrusion dans les réseaux MANETs sont présentés dans la section 3.2. Dans la section 3.3 nous présentons un état de l'art des principales solutions proposées dans la littérature. Nous conclurons ce chapitre par un bref résumé sur les futures recherches dans le domaine de la détection d'intrusion.

3.1. Systèmes de détection d'intrusion (IDS)

Une intrusion peut être définit comme n'importe qu'elle action qui tente de compromettre l'intégrité, la confidentialité, ou la disponibilité d'une ressource [73], le système de détection d'intrusion (IDS) est un système qui détecte de telles intrusions. Le développement d'un IDS est motivé par les facteurs suivants :

- La majorité des systèmes existants ont des failles (limites ou défauts) de sécurité ce qui les rendent susceptible aux intrusions, et trouver et régler toutes ces défaillances n'est pas possible (faisable) [74].
- Les techniques de sécurité préventive ne sont pas suffisantes. Il est presque impossible d'avoir un système totalement sécurisé [74].
- Même le système le plus sécurisé est vulnérable aux attaques internes [74].
- Il y a toujours de nouvelles intrusions qui émergent, et il faut des nouvelles techniques pour défendre contres ces intrusions.

Puisque il y a toujours de nouvelles intrusions qui ne peuvent pas être évitées, l'IDS est introduit pour détecter les éventuelles violations de la politique de sécurité en surveillant les activités du système. Les systèmes de détection d'intrusion constituent une deuxième ligne de défense, parce que l'IDS entre en jeux une fois que l'intrusion s'est produite. Si on détecte l'attaque dès qu'elle survienne dans le réseau, une réponse peut être initiée pour prévenir ou minimiser son impact sur le système.

3.1.1 Taxonomie des systèmes de détection d'intrusion

Il y a trois principaux composants dans un IDS : le module de collection des données, le module de détection, et le module de réponse. Le module de collection des données assure les tâches de collecte et du prétraitement des données comme : le stockage et le transfert des données au module de détection [75]. Un IDS peut utiliser différentes sources de données telles que: les logs du système, les paquets du réseau, etc. Un IDS qui surveille uniquement les activités de l'hôte et détecte les intrusions uniquement au niveau hôte, est appelé host-based IDS (abrégé HIDS, traduction de système de détection d'intrusion basé sur l'hôte). Si l'IDS surveille les activités et détecte les intrusions au niveau réseau, alors il est appelé network-based IDS (abrégé NIDS, traduction de système de détection d'intrusion basé sur le réseau). Dans les réseaux sans fil multi-sauts les NIDS utilisent généralement le mode promiscuité pour écouter et collecter des données dans un segment du réseau, et ainsi détecter les attaques distribuées. Il existe aussi les systèmes de détection d'intrusion hybrides qui surveille les activités au niveau réseau aussi bien qu'au niveau hôte pour assurer une meilleure détection [76]. En fonction de l'architecture du système on peut distinguer deux types d'IDS : centralisés et distribués. L'étude réalisé par Puttini et al [77] démontre que la majorité des IDSs distribués sont hiérarchiquement organisés autour d'un nœud central et que peu d'entre eux sont complètement distribués. Généralement, seule la collecte de données est distribuée dans un IDS distribué.

Au niveau du module de détection les données sont analysées pour détecter les intrusions ou les tentatives d'intrusion. Dans la littérature, trois techniques de détection d'intrusion sont utilisées. La première est la détection à base d'anomalie (anomaly based detection) qui modélise le comportement normal du système, puis identifie toute déviation par rapport à ce modèle comme étant une anomalie (voir figure 3.1). Plusieurs techniques ont été proposées dans la littérature telles que : le plus proche voisin, réseaux de neurones, machines à vecteurs de support (SVM), et les méthodes statiques [78]. Définir le comportement ou profil normal est un défi majeur. Le profil normal peut changer avec le temps et le système de détection d'intrusion doit être mis à jour périodiquement. L'avantage de cette technique est sa capacité à détecter des attaques inconnues, cependant elle induit un coût de calcul et une consommation d'énergie élevés dues aux mises à jour

périodiques du profil. D'autant plus elle génère un taux élevé de faux positifs c.-à-d. les activités normales détectées par l'IDS comme anomalies.

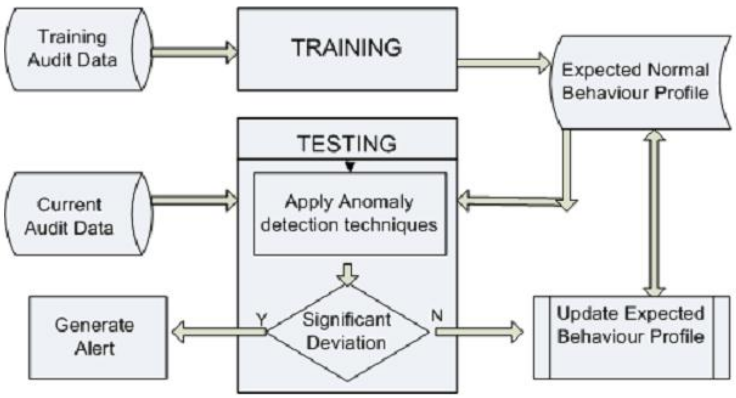


Figure 3.1Détection d'intrusion à base d'anomalie

La détection d'intrusion à base de signature (Signature-based detection) compare les activités observées du système avec un ensemble de signatures d'attaques (voir figure 3.2). Cette approche est généralement préférée pour les IDSs commerciaux parce qu'elle est précise et génère un faible taux de faux positifs. L'inconvénient de cette approche est qu'elle ne peut pas détecter de nouvelles attaques (ou attaques inconnues). L'efficacité de cette technique s'appuie sur la mise à jour continue de sa base de signatures, par conséquent ceci induit à d'importante consommation des ressources du réseau. Les deux techniques de détections (à base d'anomalie et à base de signature) ont leurs points forts et point faibles. Par conséquent, les deux techniques sont généralement employées conjointement pour assurer une détection efficace.

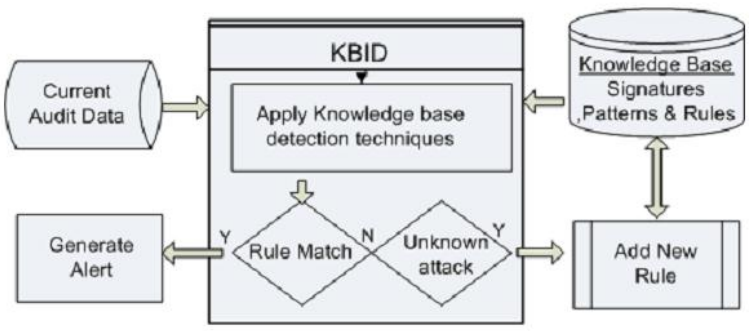


Figure 3.2 Détection d'intrusion à base de signatures

La troisième technique est la détection à base de spécification. Dans cette approche, un ensemble de contraintes de fonctionnement du protocole (ou programme) est spécifié, les intrusions sont détectées comme des violations de la spécification. Cette technique se présente comme une alternative prometteuse qui combine les avantages des deux techniques précédentes, une détection

des attaques connues et inconnues avec un faible taux de faux positifs (voir figure 3.3). Cette technique est capable de détecter de nouvelles attaques tant que ces dernières violent la spécification du protocole. De plus, cette technique ne déclenche pas de fausses alarmes quand le réseau présente un comportement inhabituel mais légitime, parce qu'elle se réfère à la spécification du protocole [79]. Cette technique a été appliquée au protocole DHCP (Dynamic Host Configuration Protocol), ARP (Adresse Resolution Protocol) [80], et quelques protocoles de routage ad hoc tels qu'AODV et OLSR. Définir une spécification détaillée pour chaque protocole est une tâche qui nécessite de l'expertise et qui peut prendre beaucoup de temps. En outre, ils existent certaines attaques qui ne violent pas la spécification du protocole telles que trou de ver et Rushing.

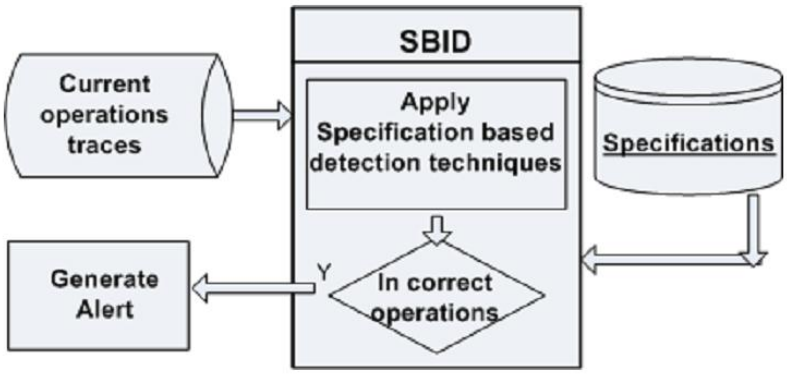


Figure 3.3 Détection d'intrusion à base de spécification

Lorsqu'une intrusion est détectée, le système lance la réponse appropriée à l'intrusion suivant sa stratégie de réponse. Les réponses aux intrusions peuvent être passives ou actives. Une réponse passive se limite au soulèvement d'une alarme et à la notification de l'autorité concernée. La réponse active quant à elle tente d'éliminer l'attaque et/ou d'arrêter l'attaquant, ou du moins minimiser l'impact de l'attaque. D'après l'objectif on peut classer les réponses aux intrusions en deux types : une réponse qui cherche à contrôler le système attaqué, et une réponse qui tente de contrôler le système attaquant [81]. Le premier type de réponse cherche à réparer le système endommagé en arrêtant l'attaque, et en terminant les connexions suspectes, etc. Le deuxième tente d'empêcher les futures tentatives de l'attaquant, ce qui est nécessaire pour les applications miliaires.

3.2. Problèmes liés à la détection d'intrusion dans les réseaux sans fil multi-sauts

Bien que plusieurs systèmes de détection d'intrusion ont été proposés pour les réseaux filaires, ces IDSs ne peuvent pas être appliqués dans les réseaux ad hoc dues à leurs caractéristiques spécifiques. Par conséquent, plusieurs recherches ont été réalisées pour le développement de nouveaux IDSs ou l'adaptation des IDSs existants aux caractéristiques des réseaux ad hoc. Il est important de prendre en considération les contraintes suivantes lors du développement d'IDSs pour les réseaux sans fil multi-sauts.

Manque des points centraux : les réseaux sans fil multi-sauts (particulièrement les MANETs et VANETs) n'ont pas des points d'entré tels que les routeurs, passerelles, etc. Ces points typiquement présents dans les réseaux filaires permettent de contrôler tout le trafic réseau qui transite par eux. Un nœud dans un réseau sans fil multi-sauts ne peut voir qu'une portion du réseau : les paquets transmis dans sa portée radio. Puisque les réseaux sans fil multi-sauts sont distribués et coopératifs, le système de détection d'intrusion devrait être distribué et coopératif [83]. Ce qui introduit quelques difficultés. Par exemple, la distribution et la coopération des agents de détection est difficile dans un environnement limité en terme de ressources telles que : bande passante, puissance de calcul, et l'énergie. De plus, dans le cas des IDSs à base de signatures, stocker les signatures d'attaques dans une base de données centrale et les distribuer aux agents de détection ne convient pas à un tel environnement

Mobilité : Les nœuds dans un réseau sans fil multi-sauts peuvent quitter et joindre le réseau et se déplacer aléatoirement, donc la topologie du réseau peut changer fréquemment. Une telle topologie rend les techniques de détection traditionnelles peu fiables. Par exemple, il est difficile pour les approches à base d'anomalie de déterminer si un nœud émettant des informations de routage expirées, est malveillant ou s'il s'agit simplement d'un nœud honnête qui n'a pas reçu de mises à jour [84]. Un autre effet de la mobilité, est que l'architecture de l'IDS peut changer avec les changements de la topologie.

Liaisons sans fil : dans les réseaux sans fil la bande passante est limitée et les coupures de liaisons sont fréquentes. Les agents de détection ont besoin de communiquer entre eux pour échanger des données ou alertes, ils ont besoin d'être au courant de l'état des liaisons. Les agents de détection doivent minimiser leurs échanges de données pour ne pas congestionner les liaisons et restreindre le trafic normal. Par exemple, l'IDS peut ne pas répondre en temps réel à une attaque dû à un retard de communication. De plus, les agents de détection peuvent perdre la connexion entre eux dues aux

ruptures de liaisons. Un IDS doit être capable de tolérer les pertes de messages tout en assurant un taux de détection raisonnable.

Ressources limitées : la limitation des nœuds en termes de calcul et stockage affecte l'efficacité des agents de détection d'intrusion. L'algorithme de détection d'intrusion doit prendre en considération la limitation en ressources. Par exemple, les IDSs à base de signature doivent prendre en considération la contrainte d'espace mémoire pour le stockage des signatures, les IDSs à base d'anomalie doivent optimiser leurs usages en mémoire et en calcul.

Manque de ligne de défense et de communication sécurisée : les réseaux sans fil multi-sauts ne possèdent pas une ligne de défense claire. Dans cet environnement le trafic de l'IDS doit être chiffré pour empêcher les attaquants de découvrir le mode opératoire de l'IDS [83]. Cependant, la cryptographie et l'authentification sont des tâches difficiles à implémenter dans un environnement sans fils mobile parce qu'elles sont très gourmandes en ressources.

Coopération : les protocoles de routage pour les réseaux sans fil multi-sauts sont généralement très coopératifs, ce qui les rendent cibles aux nouvelles attaques. Par exemple, un nœud peut se faire passer comme voisin d'autres nœuds et participer dans le mécanisme de décision, et ainsi affecter d'importantes parties du réseau.

3.3. Systèmes de détection d'intrusion proposés pour les réseaux MANETs

Nous examinons les IDSs proposés dans la littérature afin de découvrir comment ils adressent les problèmes discutés précédemment. Nous étudions particulièrement les IDSs proposés pour les réseaux MANETs, dues aux caractéristiques génériques et représentatives de ces derniers. Les critères suivants seront considérés dans cet état de l'art :

Les données entrées : les systèmes de détection d'intrusion peuvent utiliser les données de l'hôte, les paquets du réseau ou les statistiques sur ces données (ex. les statistiques des mises à jour de la table de routage et le nombre des paquets reçus dans les dix dernières secondes).

Collecte des données : outre l'utilisation des données locales, un nœud dans un réseau sans fil multi-sauts peut écouter ses voisins pour collecter leurs données. Puisque les réseaux multi-sauts utilisent des liaisons sans fil, un nœud peut être surveillé par ses nœuds voisins en utilisant le mode promiscuité.

Architecture : les architectures des IDSs dans les réseaux sans fil multi-sauts peuvent être classées en trois catégories : autonomes, distribués et coopératives, et hiérarchiques [86]. Dans un IDS à

architecture autonome, chaque nœud dans le réseau possède un agent de détection d'intrusion qui détecte les attaques tout seul sans collaborer avec d'autres nœuds. Dans la plupart des cas, cette architecture ne peut pas détecter les attaques réseau (scan réseau, attaques distribuées, etc.) en se référant uniquement aux données réseau partielles se trouvant au niveau de chaque nœud. Dans une architecture distribuée et coopérative, chaque nœud possède un agent de détection locale qui communique avec les agents d'autres nœuds pour échanger les informations et pour répondre aux intrusions. Les deux architectures (autonome, distribués et coopératives) conviennent plus aux infrastructures réseau plates. Une architecture hiérarchique est une sorte d'architecture distribuées et coopérative adaptée aux réseaux multicouches [86]. Dans cette architecture, le réseau est divisé en groupes appelés clusters. Un chef de cluster (Cluster head) gère la communication à l'intérieur du cluster, et avec d'autres clusters. Tandis que chaque nœud dans le cluster effectue une détection locale, les chefs de clusters eux effectuent une détection globale.

Interopérabilité : cet aspect se réfère aux moyens par lesquels les agents de détection d'intrusion communiquent entre eux (dans les architectures distribuées et hiérarchiques). Cela peut être réalisé par une communication traditionnelle à base de paquets réseau ou encore par le biais d'agents mobiles. Un agent mobile est une composition de logiciel et de données capable de migrer de manière autonome d'un hôte à un autre et de terminer son exécution sur l'hôte de destination [86]. Ce qui permet de réduire la charge du réseau en déplaçant le calcul aux données, et ainsi conserver la bande passante dans certain cas.

Méthode de détection : la méthode de détection d'intrusion la plus proposée dans la littérature est la détection à base d'anomalie. Cette méthode permet de détecter les attaque connues et inconnues avec un bon taux de détection. Cependant, elle génère un taux élevé de faux positifs dans un environnement à forte mobilité. Il existe aussi quelques IDSs à base de spécification, qui sont capables de détecter des attaques connues et inconnues avec un faible taux de faux positifs. Il y a eu peu de d'IDSs à base de signature pour les réseaux sans fil multi-sauts, et peu de recherches ont été menées sur les signatures des attaques. La mise à jour des signatures des attaques constitue un problème important pour cette technique.

Prise de décision : dans les IDSs distribuées et coopératifs, deux type de prise de décision sont utilisés. Prise de décision collaborative, où chaque nœud participe dans le processus de détection d'intrusion. Prise de décision indépendante, où quelques nœuds particuliers sont responsables de la prise de décision [87]. Les deux mécanismes ont leurs avantages et inconvénients. Les systèmes a prise de décision collaboratif sont plus fiables. Si tous les nœuds contribuent à la décision, quelques nœuds malveillants ne peuvent pas perturber la prise de décision. Cependant, si n'importe quel

nœud dans le réseau peut initier une réponse globale, cela peut affecter le réseau entier et le rendre vulnérable aux attaques [87]. La prise de décision collaborative est aussi plus tolérante aux pannes des nœuds. D'un autre côté, l'échec ou la compromission des nœuds particuliers dans les systèmes de prise de décision indépendante peut avoir des effets drastiques. Cependant, ces systèmes sont moins exposés aux fausses accusations que les systèmes de prise de décision collaborative [87].

Réponses aux intrusions: le système peut avoir une réponse passive ou une réponse active par rapport aux intrusions détectées. Les systèmes de réputation cités dans [88] sont des exemples de méthodes utilisées dans les réseaux MANETs comme réponses actives.

Sécurité de l'IDS : Il peut y avoir des attaques contre le système de détection d'intrusion lui-même, cette caractéristique concerne le niveau de sécurité de l'IDS contre ce type d'attaques.

Tous les critères cités ci-dessus sont résumés dans le tableau 3.1. Les principaux systèmes de détection d'intrusion proposés pour les réseaux MANETs dans la littérature sont décrits ci-dessous.

3.3.1 Les Systèmes de détection d'intrusion à base de spécification

Généralement, les systèmes de détection d'intrusion à base de spécification, définissent d'abord explicitement la spécification comme un ensemble de contraintes. Ensuite, ils utilisent la spécification pour surveiller les opérations du protocole afin de détecter les attaques dans le réseau. La première étape réalise une extraction de la spécification qui définit le fonctionnement correct du protocole via un ensemble de contraintes. Ensuite le système surveille l'exécution du protocole par rapport à la spécification donnée, toute déviation par rapport à la spécification sera considérée comme une intrusion [57]. Des approches syntaxiques et sémantiques ont été proposées pour les systèmes de détection d'intrusion dans les réseaux filaires [58].

Caractéristique	**Explication**	**Classification**
Données entrées	Données contrôlées	Paquets réseau, données MIB, données hôte, statistiques, etc.
Collecte des données	D'où les données sont collectées	Hôte, ou écoute en mode promiscuité
Architecture	Structure et organisation des agents de détection d'intrusion	Autonome, distribuée et coopérative, hiérarchique
Groupement	Comment les agents sont 'ils groupés	Clusters, zones, à un saut
Interopérabilité	Comment les agents communiquent entre eux	Paquets réseau ou agents mobiles
Méthode de détection	Méthode employée pour détecter les intrusions	à base d'anomalie, à base de spécification, à base de signature
Prise de décision	comment prendre des décisions concernant les intrusions	Locale, collaborative ou indépendante
Mécanisme de réponse	comment réagir aux intrusions détectées	Réponse passive ou active
Sécurité de l'IDS	vulnérabilités des agents de détection	un seul point de défaillance, attaques contre les agents mobiles, attaques DoS, etc.

Tableau 3.1 Caractéristiques des IDSs

3.3.1.1 Approches à base de machine à états finis

Le premier IDS à base de spécification pour les réseaux MANETs a été proposé par Tseng et al. [89]. Ils appellent moniteurs réseau (Network monitor) l'ensemble des nœuds chargés de surveiller les actions des autres nœuds du réseau. Les moniteurs réseau sont supposés couvrir tout le réseau afin de surveiller tous les nœuds. Les auteurs supposent que : i) les moniteurs réseau connaissent toutes les adresses IP et MAC, et que les adresse MAC ne peuvent pas être usurpée ; ii) les moniteurs réseau et leurs messages sont protégé et sécurisé ; iii) si certains nœuds ne répondent pas aux messages de diffusions, cela ne causera pas de sérieux problèmes. Le processus de découverte de route dans le protocole AODV est modélisé sous forme de machine à états finis (Finite state machines FSM). Chaque message de demande et de réponse de route dans la portée du moniteur réseau est contrôlé sous forme de flux demande-réponse. Lorsqu'un moniteur a besoin d'une information sur un message précédent ou sur d'autres nœuds qui ne sont pas à sa portée, il peut demander ces informations aux moniteurs réseau voisins. Sous une forte mobilité, la communication entre les moniteurs réseau augmente considérablement vu les fréquents déplacements des nœuds surveillés qui quittent et rejoignent la portée du nœud moniteur. Les auteurs ajoutent un nouveau champ à la demande de route: l'adresse du nœud précédent. Puisque les demandes de routes sont des messages de diffusion, il est nécessaire de garder la trace du chemin emprunté par la demande de route. L'adresse du nœud précédent est nécessaire pour détecter certains types d'attaques telles que l'envoi de réponse de route à un nœud qui ne figure pas sur le chemin inverse [89]. Les auteurs proposent une approche prometteuse qui permet de détecter les attaques connues et inconnues contre les protocoles de routage qui ont des spécifications clairement définies. Ils prétendent que leur approche détecte la majorité des attaques en temps réel et avec une faible charge de communication. Cependant, certaines hypothèses acceptées dans ce papier ne sont pas très réalistes. Par exemple, supposer que les moniteurs réseau possèdent les adresses IP et MAC de tous les nœuds, et qu'ils peuvent surveiller tous les nœuds du réseau. L'évolutivité est une des caractéristiques les plus importantes de plusieurs applications MANET où les nœuds peuvent joindre et quitter le réseau librement et se déplacer fréquemment. Supposer que les adresses MAC ne peuvent pas être usurpées facilement est irréaliste. De plus, la suppression de certains messages de diffusion peut affecter les services du réseau, si le nœud supprimant les messages se trouve dans un emplacement critique du réseau. En outre, les détails de l'architecture de l'IDS ne sont pas abordés, tels que le positionnement des moniteurs dans le réseau.

Tseng et al [90] proposent un IDS à base de spécification pour le protocole OLSR (optimized link state routing). Ils définissent le fonctionnement normal du protocole en utilisant un automate à états finis qui spécifie un ensemble de contraintes décrivant comment le protocole OLSR gère le trafic de

contrôle. Dans une architecture distribuée, les nœuds surveillent le comportement de leurs voisins en se basant sur la spécification du protocole. Les nœuds détectent les intrusions en comparant le comportement du nœud voisin avec la définition de la spécification. Orset et al [93] proposent l'utilisation de machine à états finis étendus (EFSM) pour représenter la spécification du protocole OLSR. Les auteurs extraient manuellement un ensemble de contraintes à partir de la spécification IETF du protocole sous forme de traces de messages envoyés et reçus. Ensuite la machine à états finis étendus compare les traces du réseau avec la spécification en utilisant le rétro traçage (backward tracking) pour identifier les intrusions.

Il existe dans la littérature d'autres IDSs à base de spécification proposés pour le protocole AODV [90] [91]. Une approche à base de spécification combinée à la cryptographie a été proposée dans [90]. Dans cette approche la spécification est représentée sous forme de machine à états finis étendues (Extended Finite State Machine EFSM). Stakhanova et al [94] proposent une méthode d'extraction et de synthétisation de spécification afin de modéliser et analyser les protocoles de routage pour les réseaux MANETs. Ils se focalisent sur le flux de trafic pour extraire la spécification sous forme de graphe orienté où les nœuds représentent la configuration du protocole et les arcs montrent comment le protocole évolue d'une configuration à une autre. Ils utilisent cette spécification pour détecter les comportements anormaux qui violent la spécification. Ils valident la méthode proposée en la simulant sur le protocole DSR et AODV.

3.3.1.2 Approche à base d'agents mobiles

Panos et al. [95] proposent une architecture à base d'agents mobiles, et un IDS multicouche à base de spécification pour surveiller les opérations des protocoles des couches transport, réseau, et liaison de données. Un ensemble de détecteurs autonomes RWDs (Random Walk Detectors) se déplacent aléatoirement à travers le réseau d'un nœud à l'autre pour contrôler le comportement du nœud visité. Les agents RWDs qui s'exécutent sur les nœuds visités peuvent détecter les attaques qui violent les spécifications des trois couches. Cependant le processus de migration des RWDs induit d'importants transferts de données et une charge de communication supplémentaire qui augmente proportionnellement avec le nombre d'agents RWDs. Les spécifications des protocoles sont superficielles et incomplètes, elles ne considèrent que quelques opérations des protocoles. De plus, les auteurs ne considèrent pas les lourdes conséquences qui peuvent résulter de laisser les nœuds sans protection pendant la migration des agents mobiles RWDs.

3.3.1.3 Modèle de détection d'intrusion à base de spécification

Lin et al [96] proposent un modèle de détection d'intrusion à base de spécification qui utilise les messages du nœud précédent pour assurer l'intégrité du routage. Le modèle est conçu pour pallier à

certaines faiblesses du protocole AODV. Cette version sécurisée du protocole AODV utilise des techniques telles les chaines de hachage et les signatures digitales pour sécuriser le routage. Le modèle suppose qu'une autorité centrale existe et disponible pour authentifier les nœuds et leurs fournir les clés avant de joindre le réseau. Les signatures digitales sont utilisées pour vérifier l'authenticité des messages. Chaque nœud sauvegarde les informations de routage de son prédécesseur. Ensuite les informations sont comparées à l'aide de sept règles de spécification. Si un nœud malveillant modifie les informations de routage, l'IDS détecte l'altération et notifie le nœud récepteur pour qu'il supprime le message. L'approche proposée aborde certains challenges qui n'ont pas été considérés par les solutions précédentes tels que la détection des attaques en temps réel, cependant cette solution ne convient pas aux architectures a forte mobilité ou les nœuds voisins changent fréquemment.

3.3.1.4 Système de détection d'intrusion & plateforme matérielle de confiance

Panos et al. [97] proposent un mécanisme de détection d'intrusion appelé SIDE (Specification-based Intrusion Detection) pour surveiller le comportement du nœud hôte. SIDE protège les opérations du protocole AODV en utilisant une machine à états finis qui définit le comportement normal du protocole. SIDE comprend deux fonctionnalités complémentaires : (i) un IDS à base de spécification pour le protocole AODV, et (ii) une procédure d'attestation à distance pour garantir l'intégrité de l'instance SIDE en exécution. Le mécanisme proposé fonctionne sur une plateforme d'informatique de confiance qui fournit une base matérielle de confiance et d'accélération cryptographique. La plateforme est utilisée par la procédure d'attestation à distance, elle assure la protection contre les attaques d'exécution. L'avantage clé du mécanisme proposé est sa capacité à détecter efficacement et en temps réel des attaques connues et inconnues. Cependant, SIDE dépend de support matériel, et utilise des fonctions de cryptographie et d'authentification qui sont très coûteuses en termes de ressources.

3.3.2 Systèmes de détection d'intrusion à base d'anomalie

Appelés aussi systèmes de détection basés sur le comportement, parce que le système définit le comportement normal du réseau, ensuite compare le comportement du réseau périodiquement avec le comportement normal. La détection à base d'anomalie se déroule typiquement sur deux phases : l'apprentissage et le test. L'apprentissage est le processus de modélisation du comportement normal ou du comportement attendu du réseau (ou du nœud). Le modèle peut servir également comme profile du nœud ou du réseau. Pour qu'un IDS à base d'anomalie soit efficace, il faut qu'il ait un profile consistent et stable qui caractérise ce comportement. Un profile consiste en un ensemble d'information concernant une liste de paramètres qui sont spécifiquement axés sur la cible à surveiller. La construction d'un profile efficace implique la collection de données sur le

comportement et l'activité qui sont considérés comme acceptables pour le réseau. La phase de test implique la comparaison du comportement normal (ou attendu) obtenu durant la phase d'apprentissage avec le comportement actuel du réseau ou des nœuds. Les techniques de détection utilisent souvent des approches statistiques ou mathématiques pour signaler n'importe qu'elle divination significative entre les deux comportements.

Les systèmes à base d'anomalie utilisent généralement des méthodes statistiques [98] [99] afin d'estimer l'écart entre le comportement attendu et le comportement actuel du réseau pour détecter les intrusions. Les techniques statistiques et probabilistes telles que le test chi-square, le test T2 de Hotelling, arbres de décision et les chaînes de Markov sont utilisées dans les systèmes de détection à base d'anomalie. Les réseaux de neurones [100] ont été également utilisés pour l'apprentissage et la modélisation du comportement des utilisateurs dans le réseau.

3.3.2.1 Matrice de similarité

Cretu et al [101] proposent une approche de détection à base d'anomalie pour les réseaux MANETs. Ils modélisent le comportement du nœud, que l'IDS peut ensuite utiliser pour déterminer la fiabilité des nœuds. L'objectif de l'approche est de détecter la charge anormale. Dans la phase d'apprentissage l'IDS observe les charges de différents nœuds et ensuite les agrège pour construire les profiles qui seront comparés ensuite en formant les matrices de similarités. Cependant, les échanges de modèles entre tous les nœuds du réseau peuvent générer une importante charge de communication et de traitement. Il est important de prendre en considération que les nœuds peuvent avoir des comportements différents, en fonction de l'application du réseau MANET.

3.3.2.2 Théorie des jeux

Liu et al [102] proposent un framework pour la détection d'intrusion dans les réseaux MANETs. En utilisant la théorie des jeux, ils modélisent l'intrus et le défendeur comme deux joueurs dans un jeu bayésien. Ils proposent une approche bayésienne hybride qui surveille le réseau de deux manières différentes : surveillance légère et surveillance lourde. La surveillance légère du système consomme moins d'énergie et ainsi elle est toujours active, tandis que la surveillance lourde utilise la détection à base d'anomalie pour construire le profile normal et ensuite le comparer avec les données testées pour détecter les intrusions.

3.3.2.3 Chaines de Markov

Les classeurs à base de chaines de Markov sont aussi utilisés dans la détection à base d'anomalie. Par exemple Jiang et Wang [104] proposent un algorithme de détection à base des chaines de Markov pour les réseaux sans fils ad hoc. L'algorithme est constitué de deux parties : la construction de la table de chaine de Markov, et la construction du classeur en utilisant le modèle chaine de Markov.

D'abord les traces des données d'audit sont convertit en séquences de symboles et une table de chaine de Markov avec transitions d'états. La deuxième partie consiste à construire un classeur en utilisant le modèle chaine de Markov. Le modèle vérifie si la transition en cours est conforme à la propriété de Markov en utilisant une distribution uniforme pour calculer les valeurs des traces et fixer un seuil pour la détection d'anomalies. Sun et al [105] proposent un classeur à base de chaines de Markov pour qu'il puisse détecter la fabrication des messages RREP dans le protocole AODV. De la même façon, dans [106], les auteurs proposent une approche à base de chaine de Markov et du test T2 de Hotelling pour détecter les intrusions locales dans un réseau MANET. Ils considèrent que la vitesse moyenne (vélocité) du nœud ne reflète pas précisément le dynamisme du réseau, et ainsi il ne représente pas un paramètre efficace. Ils proposent que chaque agent local de de détection utilise périodiquement le taux de changement de liens pour évaluer l'effet de la mobilité.

3.3.2.4 Les techniques probabilistes statistiques

Les techniques probabilistes statistiques sont largement utilisées pour la détection à base d'anomalies. Ye et al [107] ont mené une étude sur la fréquence et la propriété de l'ordre dans les données d'audit en appliquant des techniques probabilistes incluant : le test T2 de Hotelling ; le test chi-square multi variable ; les chaines de Markov et les arbres de décision comme techniques de reconnaissance pour la détection d'intrusion dans les réseaux filaires. Afin de tester les performances de ces techniques, ils ont collecté deux échantillons d'activités normales et intrusives. Le premier contient des données collectées d'une station Sun SPARC exécutant Solaris avec un module de sécurité basique BSM (Basic Security Module). Le deuxième échantillon (d'activités normales et intrusives) est issue du MIT Lab et contient les données du projet de L'US Air Force. L'étude a démontré que lorsque la propriété de fréquence est considérée, le test Chi-square donne une bonne performance. Et dans le cas où la propriété d'ordonnancement est considérée le modèle de Markov permet de de fournir des avantages supplémentaire pour la détection d'intrusion. Les mêmes auteurs proposent dans [108] une approche basée sur le test Chi-square pour la détection d'intrusion dans les réseaux filaires. Ils ont utilisé le même échantillon de données d'audit utilisé dans [107]. Ils partagent l'échantillon de données en deux, une partie pour l'apprentissage du profile normal et une autre pour le test. Ils ont considéré quelques scénarios, créé des données d'activités intrusives et appliqué le test Chi-square.

Nadeem et al [109] proposent un IDS à base d'anomalie AIDP (Anomaly-based Intrusion Detection Protocol) qui combine la qualité d'ajustement du test Chi-square et carte de contrôle. L'IDS détecte et élimine l'attaque déni de service (Denial of Service DoS) de type inondation de RREQ (RREQ Flooding) exécutées par un attaquant interne dans un réseau hiérarchique. AIDP s'exécute uniquement au niveau des chefs de groupes (cluster heads), les membres de chaque groupe ne font

qu'envoyer les données à leurs chefs de groupes. AIDP se compose de deux modules: un module d'apprentissage et un module de test. Le chef du groupe collecte continuellement les données d'audit (statistiques) sur le réseau (nombre de RREQ), et ensuite applique le module d'apprentissage afin d'établir le profile initial du réseau (Initial Training Profil ITP) qui représente le profile normale des nœuds dans le réseau. Dans la phase de test le chef du groupe applique le module de test après chaque intervalle de temps TI. AIDP utilise le test chi-square pour examiner le comportement global du réseau, et ensuite identifie et isole l'attaquant en utilisant la carte de contrôle (control chart). Bien qu'AIDP présente un bon taux de détection, il génère un nombre important de faux positifs et induit une charge de calcul non négligeable.

3.3.2.5 Réseaux de neurones

Mitrokosta et al [110] proposent un IDS qui utilise les réseaux de neurones et les techniques du tatouage (watermarking). Ils utilisent les cartes auto adaptatives en combinaison avec les techniques de tatouage et d'apprentissage automatique. D'abord le système extrait les caractéristiques de la couche MAC, ensuite il fait la collection des données et la détection d'intrusion. Enfin, il applique la réponse appropriée contre l'intrusion. Chaque nœud crée une carte qui représente son statut de sécurité, ensuite il la distribue avec ses voisins. Une fois que le nœud reçoit toutes les cartes de ses voisins, il génère une carte globale qui l'aide à : (i) estimer le niveau de sécurité du réseau; (ii) effectuer un routage sécurisé et efficace en évitant les routes qui contiennent les nœuds malveillants. Le tatouage est utilisé pour maintenir l'intégrité et l'authenticité des cartes auto adaptatives. Les auteurs prétendent que l'IDS proposé peut détecter divers attaques, cependant ils ne spécifient pas les scénarios d'attaques contre lesquelles il pourra être testé. De même, Jabbehdari et al [111] proposent un IDS qui utilise les réseaux de neurones pour détecter les attaques par déni de service dans les réseaux MANETs.

3.3.2.6 Algorithmes évolutionnaires

Il s'agir d'algorithmes s'inspirant de la théorie de l'évolution pour résoudre des problèmes divers. Ils font ainsi évoluer un ensemble de solutions à un problème donné, dans l'optique de trouver les meilleurs résultats. Ce sont des algorithmes stochastiques, car ils utilisent itérativement des processus aléatoires. Sen et al [112] proposent l'utilisation des techniques d'intelligence artificielle, ils utilisent les techniques de calcul évolutionnaire, particulièrement la programmation génétique et l'évolution grammaticale, pour évoluer les programmes de détection d'intrusion dans les réseaux MANETs. Ils appliquent les techniques d'optimisation multi-objective (MOO) afin de trouver un compromis entre la capacité à détecter les intrusions et la consommation des ressources.

3.3.2.7 Approches bio-inspirées

Barani et al [113] proposent BeeID un IDS à base d'anomalie en utilisant l'algorithme d'optimisation par colonie d'abeilles artificielles (ABC), et l'algorithme de sélection négative (NS). L'approche consiste en trois étapes: l'apprentissage, la détection, et la mise à jour. Dans la phase d'apprentissage, l'algorithme de colonie d'abeilles artificielles nommé NicheNABC, exécute l'algorithme de sélection négative plusieurs fois afin de générer un ensemble de détecteurs négatifs matures. Dans la phase de détection, les détecteurs négatifs matures sont utilisés pour différencier entre les activités normales et malveillantes. Dans la phase de mise à jour, l'ensemble de détecteurs négatifs matures est mis à jour partiellement ou totalement. Les auteurs utilisent l'intégration de Monte Carlo pour estimer la quantité de l'espace non-self couvert par les détecteurs, et pour déterminer quand est-ce que une mise à jour totale est nécessaire. Les mêmes auteurs [114] proposent une approche à base d'algorithmes génétiques (GA) et de système immunitaire artificiel (AIS), appelée GAAIS pour une détection d'intrusion dynamique dans un réseau MANETs à base d'AODV. GAAIS s'adapte aux changements topologiques du réseau en utilisant deux méthodes de mises à jour: partielle et totale. Chaque vecteur d'attributs normales extrait du trafic réseau est représenté par un hyper sphère avec un radius fixe. Un ensemble de détecteurs sphériques est généré en utilisant l'algorithme NicheMGA. Les détecteurs sphériques sont employés pour détecter les anomalies dans le trafic réseau.

3.3.3 Systèmes de détection à base de signatures

Dans certaine littérature systèmes de détection à base de connaissance. Cette technique est basée sur la comparaison du comportement observé d'un nœud avec un ensemble de signatures d'attaques stockées. Si une correspondance est trouvée, l'activité observée est classée comme intrusion. L'avantage de cette technique est sa précision dans la détection des attaques connues, et son faible taux de faux positifs par rapport aux autres techniques (basée sur l'anomalie et/ou la spécification). La fiabilité de cette technique dépend fortement de la mise à jour continue de la base des signatures. Les systèmes de détection d'intrusion à base de signatures utilisent différentes approches pour modéliser la connaissance concernant les attaques. Les approches proposées diffèrent dans la représentation des intrusions, ainsi que dans les algorithmes employés pour détecter et reconnaitre les signatures des attaques. Parmi les approches proposées dans la littérature, il y les réseaux de Petri colorés, systèmes expert, machine à états finis, reconnaissance des formes.

Anjum et al [124] ont mené une étude afin d'analyser l'efficacité des IDSs à base de signatures dans les réseaux MANETs. Les auteurs supposent que la signature de l'attaque est connue, et que le nœud peut exécuter le processus de détection d'intrusion pour détecter l'attaque contre le protocole de

routage. Ils considèrent un scénario très simple avec un nœud malveillant appartenant à la route initiale entre la source et la destination, et ils estiment la probabilité de détection de l'intrusion avec et sans mobilité. Ils concluent que la détection dans les protocoles de routage réactifs est moins efficace comparé aux protocoles de routage proactifs. Kong et al proposent [125] un IDS à base de signature avec une architecture paire à paire, l'IDS distribué utilise une base statique. L'architecture est composée de deux composants. Le premier, un agent mobile de détection d'intrusion qui réside dans chaque nœud du réseau, et qui est responsable sur la détection des intrusions locales à partir des données d'audit locales, et qui participe aussi dans l'algorithme de détection coopérative avec les autres agents de détection pour prendre les décisions concernant les intrusions potentielles. Le deuxième composant consiste en une base stationnaire et sécurisé qui contient les signatures d'attaques connues. Cependant, dans une architecture MANET décentralisée, il est difficile d'accomplir l'hypothèse qui suppose que tous les nœuds du réseau peuvent accéder à la base des signatures.

3.3.3.1 Système expert

Certains systèmes de détection d'intrusion à base de signatures utilisent des systèmes experts [115] [116] pour la détection d'intrusion. Le système expert maintient la connaissance concernant les attaques connues dans une base de connaissances sous forme d'ensemble de règles. Les données d'audit collectées suite à la surveillance du réseau sont traduites en un ensemble de faits, ensuite le moteur d'inférence utilise ces faits et l'ensemble de règles dans la base de connaissance pour détecter les intrusions dans le réseau. Porras et al [121] proposent EMERALD une extension du système IDES [122] et du système de détection d'intrusion de nouvelle génération (NIDES) [123]. EMERALD utilise un système expert à base de règles pour la détection à base de signatures. Le système expert emploie un ensemble de règles qui décrivent les faits concernant les attaques et les inférences qui peuvent être faites à partir de ces règles. Une règle est déclenchée lorsque les conditions spécifiées sont satisfaites. Les systèmes experts sont généralement très lents, parce que toutes les données d'audit doivent être importées comme des faits. Par conséquent, les systèmes experts sont rarement utilisés dans les IDS commerciaux.

3.3.3.2 Machine à états finis

La modélisation par transition d'états peut aussi être utilisée pour la détection d'intrusion, où l'attaque est représentée comme une série de transitions d'états. Les modèles de transition d'état qui représentent les attaques sont normalement maintenues dans une base de connaissances et le modèle de transition d'état est appliqué en temps réel pour identifier une intrusion dans le réseau. L'analyse de signatures est également utilisée par les systèmes à base de signatures, où les attaques sont modélisées par une séquence d'événements, qui sont ensuite comparées avec l'audit généré

pour détecter les intrusions. Llgun et al [118] modélisent la connaissance sur les attaques dans les réseaux filaires sous forme de machine à états finis. Cette approche représente l'intrusion comme une série de changements d'états qui part d'un état sécurisé initial et termine dans un état final compromis. Vgina et al [119] proposent AODVSTAT, en reprenant le concept proposé dans [118], pour détecter la suppression des paquets et l'usurpation dans les réseaux MANETs. Les signatures sont représentées en utilisant un modèle d'évènements, où les évènements sont des échanges soit de paquets de données ou des paquets de routage AODV. Ils supposent que chaque nœud est équipé d'un détecteur AODVSTAT, qui fait l'analyse du flux des paquets que le nœud a observé soit en surveillant ses voisins ou via les mises jour obtenus à partir des observations de ses voisins pour détecter les signes d'intrusions.

Certains systèmes de détection à base de signatures utilisent des approches à base de règles [117] pour modéliser la connaissance sur les attaques connues sous forme d'un ensemble de règles qui est obtenu par l'observation ou la considération des scénarios d'attaques. L'IDS vérifie les données d'audit en appliquant des règles d'attaques connues, en utilisant les techniques de chaînage pour chercher la preuve d'une attaque. Les systèmes de détection à base de signatures sont plus adaptés à des scénarios où le réseau est vulnérable à certaines attaques connues.

3.3.3.3 Les réseaux de Petri

Les signatures d'attaques peuvent être représentées avec les réseaux de Petri colorés (Colored Petri Networks CPNs). Cette approche est plus générique et permet d'écrire des scénarios d'intrusion complexes. Cependant, elle est relativement gourmande en termes de calcul, et donc moins utilisée dans la pratique. Les méthodes de reconnaissance des formes encodent la connaissance concernant les intrusions existantes avec les signatures comme des modèles, et compare les modèles d'intrusion aux données d'audit. Cette méthode est efficace avec les représentations concises des règles, et donc elle est largement utilisée dans le domaine commercial.

3.3.3.4 Extraction de signature à base de Logs

Alattar et al proposent dans [126] IDAR, un système de détection d'intrusion à base de signatures pour protéger le protocole de routage OLSR. IDAR extrait les preuves d'intrusion à partir des Logs OLSR. D'après le niveau de suspicion de l'activité, l'IDS initie une investigation coopérative profonde pour confirmer l'intrusion. Afin d'identifier le motif de l'intrusion, IDAR compare les journaux de Logs avec un ensemble de signatures prédéfinies, où la signature est définie comme une séquence partiellement ordonné d'événements qui caractérise une activité malveillante. Bien qu'IDAR présente un taux de détection élevé et un faible taux de faux positifs, ce dernier ne peut détecter que les attaques dont les signatures existent dans sa base de signature. En outre, IDAR est vulnérable aux

nœuds malicieux qui pourraient transmettre des opinions falsifiées pendant le processus d'investigation coopérative (attaque Blackmail). De plus, des tâches telles que la collecte et l'analyse des Logs consomment considérablement les ressources du réseau (mémoire, et bande passante).

3.3.4 Systèmes de détection d'intrusion hybrides

Nadeem et al [127] proposent d'étendre AIDP [109] pour pouvoir détecter plusieurs types d'attaques. Le mécanisme général de détection et de prévention d'intrusion (Generalised Intrusion Detection & Prevention GIDP) combine les deux techniques de détection : à base de signatures et d'anomalie, afin de tirer profit des avantages des deux techniques. GIDP collecte les données d'audit (statistiques sur le réseau) sous forme de deux matrices, la première appelée matrice des caractéristiques du réseau (Network Characteristic Matrix NCM) concerne les informations sur le protocole de routage telles que le nombre de RREQ et RREP reçus durant un intervalle de temps TI. La deuxième appelée matrice dérivée, elle comporte les paramètres qui représentent la performance du réseau tels que la charge de routage, nombre de paquets supprimés, bande passante, etc. Si une intrusion est détectée le chef du groupe identifie l'attaque grâce aux informations dans sa base de connaissance, sinon il met à jour le profile initial. GIDP permet de détecter et d'éliminer divers types d'attaques, cependant son déploiement est difficile, et son approche d'identification et de réponse aux attaques est inflexible. Afin de pallier aux limites de GIDP, et minimiser l'impact négatif des réponses fixes aux intrusions sur la performance du réseau. Nadeem et al. [128] proposent IDAR (Intrusion Detection & Adaptive Response) un mécanisme de réponse flexible et adaptatif. Ce nouveau système de réponse choisit l'action de réponse en se basant sur les paramètres suivants : la gravité de l'attaque, la dégradation des performances du réseau, et l'impact attendu de la réponse sur la performance du réseau. IDAR permet de répondre efficacement aux intrusions tout en préservant la performance du réseau. Bien que les mécanismes décrits ci-dessus permettent de détecter divers types d'attaques connues et inconnues, ces derniers n'interviennent qu'après l'occurrence de l'attaque et ne parviennent pas à la prévenir. Des tâches telles que : la collecte continue des données, l'apprentissage répété, l'inférence d'attaques, et la gestion de la base des signatures, consomment considérablement le temps et les ressources du réseau. De plus, la construction et l'ajout de nouvelles règles pour les nouvelles attaques est susceptible de générer des signatures erronées.

3.3.5 Autres approches

Shakshuki et al. [129] proposent un système de détection d'intrusion local, nommé Enhanced Adaptive ACKnowledgment (EAACK) ce qui peut être traduit en «acquittement adaptatif amélioré». EAACK est développé spécialement pour les réseaux MANETs, afin de pallier à trois faiblesses du mécanisme de protection Watchdog, à savoir, fausse accusation (false misbehavior report), limitation

de la puissance de transmission (limited transmission power), collisions de réception (receiver collision). EAACK est un schéma à base d'accusés de réception, qui exige un acquittement de bout en bout pour chaque transmission de paquet. Afin d'empêcher la modification et la fabrication des paquets d'acquittement et garantir leurs authenticité et leurs validité, EAACK propose d'utiliser la signature numérique. Le schéma d'acquittement constitue le processus primaire dans lequel la source transmet le paquet d'acquittement (ACK) à la destination. Bien que le mécanisme EAACK offre un taux de détection élevé contre un type particulier d'attaques, les transmissions des paquets d'acquittement induit une importante charge de communication. En outre, EAACK ne peut pas détecter les attaques inconnues, ou même la plupart des attaques connues telles que le trou noir, privation de sommeil, etc… Malgré que les auteurs aient soulevé la question liée au coût de traitement supplémentaire induit par la signature numérique en termes d'utilisation des ressources, ils ne proposent aucune solution pour le minimiser.

Pnaouins et al [103] proposent une stratégie de défense optimale contre les nœuds malveillants dans les réseaux MANETs. Le protocole qu'ils proposent GTMR (Game Theoretic MANET Routing) est basé sur un modèle de jeux non-coopératif. GTMR maximise l'utilité du réseau MANET à l'équilibre de Nash. Le modèle permet d'optimiser l'effort de détection d'intrusion (probabilité de surveillance) que chaque nœud du réseau doit dépenser. Ce qui permet d'atteindre le meilleur équilibre entre le coût et l'efficacité de la détection proposant ainsi une stratégie de défense optimale. Le tableau 3.2 présente un résumé des principaux mécanismes de détection d'intrusions proposés dans la littérature.

3.4. Recherches futures

Beaucoup de recherches se sont émergées dans le domaine de la détection d'intrusion. Les principaux domaines de recherche [130] sont les suivants :

- **Les bases :** recherches sur l'intrusion, l'intrus et les vulnérabilités
- **Collection des données :** sélection des sources de données et les propriétés qui caractérisent le système, la collection et le formatage des données.
- **Rapportage et réponses :** comment répondre aux intrusions détectées, représentation et rapportage des intrusions détectées à l'autorité concernée.

Mécanismes	Architecture	Technique de détection	Attaques considérées	Protocoles	Réponse	Sources de données	Contributions
Tseng et al. [89]	Distribuée & coopérative	Spécification	Suppression, modification, fabrication	AODV	Alarmes	Paquets de routage (RREQ & RREP)	Premier IDS à base de spécification
Kachirski et al [87]	Distribuée	Anomalie	Non-spécifiées	Non-spécifié	Non-considérée	Agents mobiles Surveillance Evènements	Architecture distribuée d'agents mobiles
Ndeem et Howarth [109]	Hiérarchique, Clusters	Anomalie	DoS	AODV	Isoler le nœud malveillant	Information de routage	Détection de DoS dans les MANETs
Cretu et al [101]	Distribuée Paire à paire	Anomalie	Non-spécifiées	Non-spécifié	Ne pas coopérer avec le nœud malveillant	Données expérimentales	Détection avec échange de modèle pour les MANETs
Panos et al. [95]	Agents mobiles	Spécification	Empoisonnement de table de routage, trou noir, DoS	AODV	Alarmes, écarter le nœud malveillant	Données locales (hôte)	Détection d'intrusion multicouche à base de spécification
Liu et al [102]	Distribuée	Anomalie	DoS	Non-spécifié	Non-considérée	Surveillance	Modélisation de l'intrusion avec la théorie des jeux (jeu Bayésien)
Sun et al [105]	Distribuée	Anomalie	Perturbation de routage	AODV	Non-considérée	Données d'audit	Détection d'anomalie avec les chaines de Markov
Mitrokosta et al [110]	Distribuée coopérative	Anomalie	Divers attaques	Non-spécifié	Contourner le nœud malveillant	Caractéristiques de la couche MAC	Détection à base de réseaux de neurones & tatouage
Vgina et al [119]	Distribué coopérative	Signature	Suppression de paquets, consommation des ressources	AODV	Non-considérée	Paquets de routage & données	Modélisation d'attaque avec machine à états finis
Panos et al. [97]	Autonome	Spécification	Modification, fabrication, privation de sommeil, trou noir & gris	AODV	Non-considérée	Données locales (hôte)	Spécification complète ; détection temps réel
Shakshuki et al. [129]	Autonome	A base d'acquittement	Suppression, fausse accusation, transmission à puissance limitée	DSR	Alarmes	Surveillance	Résout les faiblesses du Watchdog
Nadeem et al. [127]	Hiérarchique organisée en clusters	Hybride (à base d'anomalie & signature)	Privation de sommeil, Trou noir& gris, Rushing	AODV	Isoler le nœud malveillant	Information de routage	IDS général ; taux de détection élevé; isolation du nœud malveillant

Alattar et al. [126]	Distribuée et coopérative	signature	Suppression, Modification, fabrication	OLSR	Non-considérée	journaux de Logs	Adaptatif ; investigation coopérative
Orset et al [93]	Distribuée	Spécification	Fabrication, modification, Sybil	OLSR	Non-considérée	Spécification IETF du protocole OLSR	Modélisation de la spécification avec machine à états finis étendus
Barani et al [113]	Autonome	Anomalie	Privation de sommeil, Trou noir& gris, Rushing, Trou de ver	AODV	Alarmes	Données locales	Adaptation rapide aux changements de topologie

Tableau 3.2 Comparaison des mécanismes de détection d'intrusion

- **Environnement et architecture de l'IDS :** comment distribuer les agents de détection d'intrusion et faciliter l'interopérabilité entre les agents de détection, les problèmes liées aux IDSs dans différents systèmes, les réseaux avec chiffrement, etc.
- **La sécurité de l'IDS :** protection de l'IDS et de son trafic.
- **Aspects opérationnels :** maintenance, portabilité, évolutivité, etc.
- **Aspects sociaux :** problèmes de confidentialité.

La plupart des domaines de recherche cités ci-dessus sont encore immatures. La majorité des recherches sur les systèmes de détection d'intrusion se sont focalisées sur les techniques de détection. Il y a aussi des recherches sur le développement des standards pour les IDSs tel que le format d'échange de détection d'intrusion IDEF (Intrusion Detection Exchange Format). L'objectif du format IDEF est de définir les formats de données et les procédures d'échanges pour le partage d'information dans les systèmes de détection et de réponses, et pour les systèmes de gestion qui pourraient avoir besoin d'interagir avec eux [130].

Les réseaux sans fil multi-sauts sont un nouveau type de réseau distribué dont les caractéristiques sont complexes et incomprises. La détection d'intrusion dans les réseaux sans fil multi-saut est encore un domaine de recherche immature. Il y a beaucoup moins d'IDS proposés pour les réseaux sans fil multi-sauts qu'il en est pour les réseaux filaires. Les futures recherches devraient se focaliser soit sur la proposition de nouveaux IDSs qui prennent en considération les caractéristiques des réseaux ad hoc, soit à l'adaptation des IDSs existants. La majorité des IDSs proposés dans la littérature et présentés dans ce chapitre, couvrent un ensemble restreint d'attaques et se limitent à un protocole spécifique. Plusieurs architectures proposées ne sont pas adaptées à la nature

dynamique des réseaux sans fil multi-sauts. De plus, certains IDSs ne prennent pas en considération la mobilité du réseau [130].

Les systèmes proposés cherchent à pallier aux problèmes liés à l'absence d'entité centrale en proposant des IDSs à architectures distribuées et coopératives. Cependant ces architectures soulèvent d'autres problèmes liées à la sécurité, la communication et les aspects de gestion. L'adaptation de l'architecture à l'environnement est un point important dans la conception de système de détection d'intrusion. L'architecture ne doit ni introduire de nouvelles vulnérabilités (failles), ni générer un surcoût supplémentaire.

Nous pensons que les IDSs à base d'anomalie doivent attribuer plus d'importance à la mobilité. Le taux de faux positif élevé est fortement affecté par la mobilité des nœuds. L'IDS doit disposer d'informations sur la mobilité et la topologie du réseau. Ainsi les attributs contenant des informations sur la mobilité devraient être inclus dans l'IDS lors de sa conception [130].

Puisque les nœuds sont les seules sources de données dans le réseau, plusieurs nœuds doivent faire de la surveillance (généralement en mode promiscuité), la collecte des données et la détection. Cependant, les nœuds n'ont pas tous la même capacité de calcul. En outre, certains d'entre eux ne sont pas assez puissants pour exécuter des algorithmes de détection d'intrusion complexes. l'IDS doit aussi minimiser la communication entre les agents de détection due à la bande passante limité des liaisons sans fils. Il semble y avoir peu de recherches sur la gestion des ressources dans le domaine de la détection d'intrusion dans les réseaux ad hoc.

3.5. Conclusion

Dans ce chapitre nous avons donné un état de l'art sur les systèmes de détection d'intrusion dans les réseaux ad hoc. Nous avons présenté plusieurs mécanismes de détection d'intrusion qui ont été proposés dans la littérature, avec différentes techniques de détection, architectures, et mécanismes de réponse. Nous nous somme focalisé sur la contribution et la nouveauté qu'apporte chaque IDS. Nous avons identifié les limites de certaines propositions. Les systèmes proposés n'adressent généralement que quelques problèmes des réseaux ad hoc. Les réseaux ad hoc ont la plupart des problèmes des réseaux filaires et bien d'autres encore. Par conséquent la détection d'intrusion dans les réseaux ad hoc reste un sujet de recherche compliqué et difficile pour les chercheurs en sécurité.

Chapitre 4 Modélisation et analyse des attaques de routage

La conception d'un mécanisme de sécurité efficace, nécessite une compréhension approfondie des attaques. Dans ce chapitre nous nous intéressons de plus près aux attaques internes qui ciblent le routage ad hoc. Nous présentons une analyse systématique pour étudier les attaques de routage dans les réseaux ad hoc mobiles, en se basant sur le concept d'attaques élémentaires et d'attaques composées. Ce chapitre est organisé comme suit: dans la section 4.1 nous présentons une analyse systématique et une nouvelle taxonomie des attaques. Dans la section 4.2 nous prenons comme exemple le protocole AODV pour examiner comment le nœud malveillant exécute les différentes attaques en exploitant les vulnérabilités du protocole de routage. Finalement la section 4.3 conclut ce chapitre.

4.1. Concepts et taxonomie des attaques

Les attaques se basent essentiellement sur une ou plusieurs actions élémentaires. Les travaux de Ning et Sun [131], Huang et Lee [132], et Ayachi et al [30] consolident cette perspective et proposent principalement deux approches d'analyse d'attaques basées sur la décomposition d'attaques. La première [131] décompose l'attaque en un ensemble d'abus atomique (atomic misuse). Alors que la deuxième [132] considère l'attaque comme un ensemble d'évènements basiques anormaux (anomalous basic events). La décomposition d'attaques permet de déceler l'incohérence de l'action du nœud malveillant vis-à-vis la spécification du protocole. Nous classifions les attaques en deux catégories: élémentaires et composées. Une attaque élémentaire peut être définie comme une succession indivisible d'opérations non conformes à la spécification du protocole sur un seul message. Les attaques composées peuvent être définies comme étant une collection d'attaques élémentaires.

4.1.1. Attaques élémentaires

Nous pouvons distinguer dans la littérature deux types d'attaques élémentaires. Les attaques qui violent la spécification du protocole de routage de manière directe, souvent appelée violation de la spécification (en anglais : specification violation). Les attaques qui ne violent pas la spécification du protocole de routage directement, mais qui violent la spécification des protocoles d'autres couche dont dépend le routage tels que les protocoles des couches MAC et physique.

4.1.1.1. Attaques élémentaires qui violent la spécification

- **Suppression**: effacer illégitimement un message reçu.
- **Modification**: modifier illégitimement un ou plusieurs champs du message avant de le retransmettre.
- **Fabrication**: fabriquer illégitimement un message et l'injecter dans le réseau.
- **Rejeu**: retransmission non autorisé d'un message.

4.1.1.2. Attaques élémentaire qui ne violent pas la spécification

Dans cette catégorie nous distinguons principalement deux attaques : trou de ver et Rushing, que nous appelons attaques à retransmission rapide. Les deux attaques sont basées sur la retransmission rapide des messages de routage (en anglais: fast forwarding).

- **Trou de ver:** (wormhole attack [37]) nécessite la coopération de deux ou plusieurs nœuds situés à des zones différentes du réseau. Les nœuds malveillants mettent en place un tunnel de communication privé entre eux où les paquets reçus d'un côté sont retransmis de l'autre côté. Le tunnel créé peut être une liaison physique directe ou une liaison virtuelle où les nœuds malveillants utilisent l'encapsulation des messages.
- **Rushing:** dans cette attaque le nœud malveillant exploite la propriété du routage à la demande (réactif), qui consiste à retenir la route par laquelle la demande de route est parvenue en premier. Ainsi, en exécutant la Rushing attack [37], le nœud malveillant retransmet plus rapidement les messages pour que la route qui passe par lui soit choisie.

4.1.2. Attaques composées

Combinaison d'attaques élémentaires de différents types, ou répétition d'un seul type d'attaque élémentaire. Les attaques composées permettent au nœud malveillant d'avoir un impact plus important et persistant sur le réseau. Pour analyser les attaques composées, nous utilisons la méthode semi-formelle «arbre d'attaque» proposée dans [132], et appliqué ensuite sur les réseaux MANETs par Ebinger et al. [133]. Dans cette méthode l'attaque est graphiquement représentée par un arbre, la racine représente le principal objectif de l'attaque. Les branches représentent les sous-objectifs de l'attaque ou les attaques qui la composent. Les sous objectifs sont reliés par les opérateurs logiques "AND" et "OR". Les feuilles représentent les attaques élémentaires. La figure 4.1 représente la structure schématique de l'arbre d'attaque. Seul les nœuds "AND" sont mis en évidence dans le schéma, tous les autres nœuds sont considérés comme "OR". Nous citons ci-dessous une liste non exhaustive des attaques composées les plus cités dans la littérature.

4.1.2.1. Invasion de route

Le nœud malveillant diffuse dans le réseau des messages de routage modifiés ou fabriqués proposant des routes plus fraiches et plus courtes, ou de nouvelles routes non-existantes. Le but de cette attaque est d'attirer et de rediriger tout le trafic vers le nœud malveillant. Elle permet au nœud malveillant d'être sélectionné sur la majorité des routes découvertes.

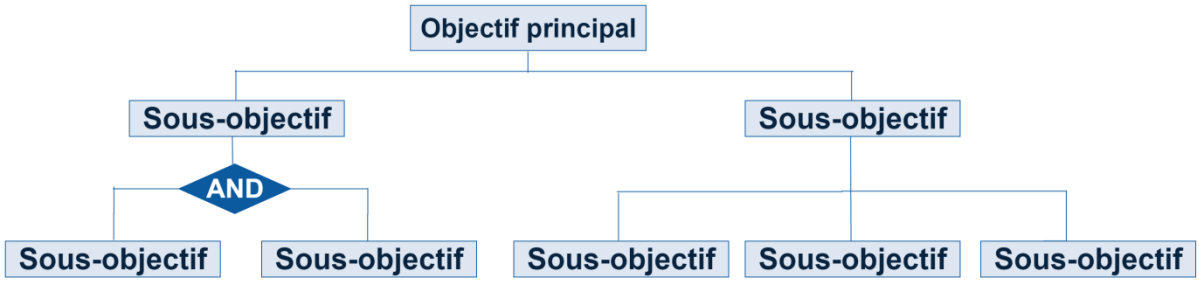


Figure 4.1 Structure schématique de l'arbre d'attaque

4.1.2.2. Isolation des nœuds

Dans cette attaque le nœud malveillant tente d'isoler un nœud ou un groupe de nœuds, en leurs fournissant des informations de routages incorrectes telles que des routes non-existantes. Le nœud malveillant peut même partitionner le réseau s'il est placé dans une position stratégique.

4.1.2.3. Usurpation d'identité et Sybil

L'usurpation d'identité peut être vue comme une modification non autorisée de l'identité (adresse IP ou MAC), ou la fabrication d'une nouvelle identité (adresse IP qui n'existe pas). Dans l'attaque Sybil [37] le nœud malveillant usurpe plusieurs identités en même temps et se comporte comme s'il était un groupe de nœuds.

4.1.2.4. Privation de sommeil

Ou consommation des ressources, dans cette attaque le nœud malveillant s'attaque à la disponibilité des nœuds. Une technique consiste à inonder le réseau avec des messages fabriqués afin de générer un trafic inutile et provoquer ainsi de sévères congestions. Le nœud malveillant peut se contenté de modifier les messages reçus pour que ces derniers circulent dans le réseau le plus longtemps possible, et provoque des traitements additionnels consommant ainsi la bande passante du réseau et les ressources des nœuds.

4.1.2.5. Trou noir et trou gris

Dans cette attaque le nœud malveillant attire d'abord le trafic vers lui en utilisant les techniques décrites précédemment dans l'attaque invasion de route. Ensuite il supprime tous les paquets de

données reçus. Le trou gris est une version plus sophistiqué du trou noir, où le nœud malveillant effectue une suppression sélective pour ne pas être détecté par les mécanismes de sécurité.

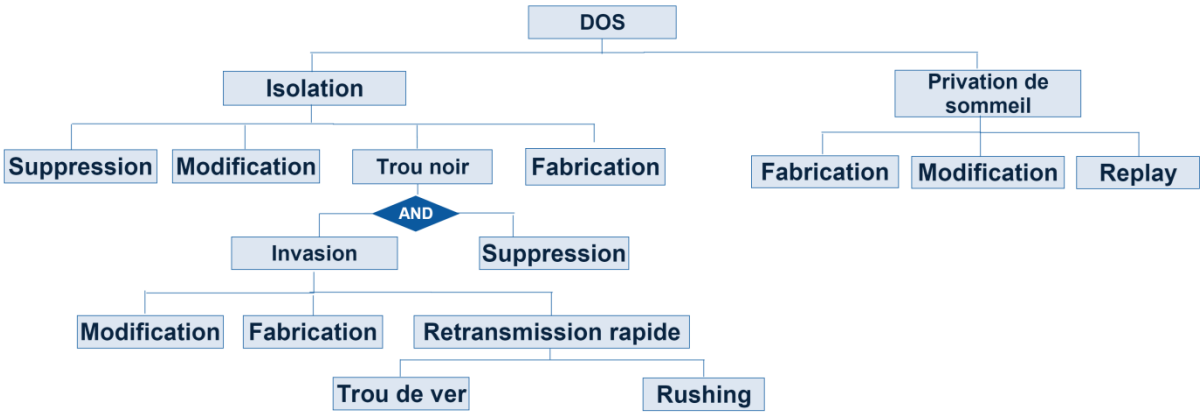


Figure 4.2 Arbre d'attaque de l'attaque DoS

4.1.2.6. Déni de service DoS

Wood et Stankovic [23] définissent DoS comme tout événement qui diminue la capacité d'un réseau à remplir sa fonction correctement ou à la remplir dans les délais. DoS s'attaque à la disponibilité des services de routage fournis par le réseau. Pour atteindre son objectif le nœud malveillant tente d'isoler le nœud cible du réseau, ou consomme les ressources de ce dernier pour l'exclure du réseau. La figure 4.2 montre l'arbre d'attaque DoS, ainsi que toutes les attaques qui la compose.

4.2. Attaques contre le protocole AODV

Parmi les nombreux protocoles de routage qui ont été développés pour permettre l'auto configuration et le routage dans les réseaux MANETs, le protocole AODV (Ad-hoc On-demand Distance Vector) a émergé comme l'un des plus populaires. Il a été le sujet de beaucoup de recherches universitaires et il a été ratifié comme RFC expérimentale (Perkins et al.) par l'Internet Engineering Task Force (IETF). Pour ces raisons, nous avons choisi comme cas d'étude le protocole AODV. Ce protocole a été développé avec l'hypothèse que tous les nœuds du réseau sont honnêtes et coopératifs. Par conséquent, il existe de nombreuses vulnérabilités dans l'AODV qui permettent aux nœuds malveillants de perturber son fonctionnement.

Une analyse compréhensive des attaques possibles contre le protocole AODV a été présentée par Ning et Sun dans [131]. Ils classent les attaques en deux catégories : (i) atomiques, qui représentent des manipulations indivisibles sur un seul message de routage et (ii) composés, qui représentent une collection d'attaques atomiques. Ils identifient quatre objectifs principaux que le nœud malveillant

peut tenter d'atteindre, à savoir: (1) perturbation de route, (2) invasion de route, (3) isolation du nœud, ou (4) consommation des ressources. Ils décrivent en détails les scénarios dans lesquels le nœud malveillants exécute des attaques atomiques et composés sur les messages de routage. Ils identifient les quatre attaques atomiques suivantes:

- Suppression illégitime de messages ;
- Modification illégitime d'un ou plusieurs champs d'un message avant de le retransmettre ;
- Fabrication d'une réponse de route suite à la réception d'une demande de route RREQ ;
- Fabrication active, ou la fabrication des messages n'est pas conditionné par la réception d'un message de routage ;

Le tableau 4.1 synthétise les objectifs du nœud malveillant et les attaques possibles sur les messages de routage du protocole AODV. Les colonnes sont basées sur les attaques exposées par Guerrero-Zapata et Asokan dans [134], les lignes spécifient les objectifs du nœud malveillant, identifiés par Ning et Sun dans [131].

4.2.1. Attaques élémentaires qui violent la spécification du protocole AODV

4.2.1.1. Suppression

a Suppression de demande de route RREQ

Le nœud malveillant pourrait simplement supprimer toutes les demandes de route reçues. Par conséquent il ne fera pas partie des routes découvertes, et il n'aura aucun impact sur le routage. Dans certain cas le nœud malveillant supprime certaines demandes de routes pour ne pas consommer ses ressources dans le processus de routage des données appartenant aux autres nœuds, ce comportement est appelé dans la littérature égoïsme (selfishness).

b Suppression de réponse de route RREP

Une fois le nœud malveillant est sélectionné sur la route entre la source et la destination, il peut supprimer la réponse de route qu'il reçoit pour empêcher l'établissement de la route vers la destination. La suppression de la réponse de route engendre du trafic supplémentaire, et occasionne un délai plus long pour l'établissement de routes.

c Suppression d'erreur de route RERR

Le nœud malveillant supprime les messages d'erreur de route RERR qu'il reçoit pour retarder la détection des liaisons défaillantes et par conséquent engendrer des retards de livraison.

4.2.1.2. Modification

a Modification de demande de route

Dans une demande de route, il y a des champs mutables (changeables), à savoir le nombre de saut et le numéro de séquence de la destination, et des champs non mutables à savoir l'identifiant, l'adresse

	Type d'attaque			
Objectifs de l'attaque	**Suppression:** des paquets de routage et de données	**Modifier et retransmettre:** Le nœud malveillant modifie incorrectement les messages de routage avant de les transmettre	**Fabrication de RREP:** Le nœud malveillant répond avec une RREP fabriquée en réponse à une demande de route	**Fabrication active:** Le nœud malveillant fabrique des messages de routage
Perturbation de route: Perturbation des tables de routage et rupture des liaisons	Supprimer RREQ, RREP ou les paquets de données			Diffusion des RERRs fabriqués avec des DSN élevés causant l'invalidation des routes et le rejet des RREQ
Invasion de route: Le nœud malveillant attire les routes vers lui pour intercepter les paquets		Manipulation du nombre de sauts, ou du DSN dans la RREP rend la route plus fraiche et plus courte	Le nœud malveillant se fait passer pour la destination en envoyant une RREP fabriquée	Le nœud malveillant se fait passer pour le nœud destination en usurpant son adresse IP
Isolation de nœud: empêcher la cible de communiquer avec le reste du réseau	Empêcher l'envoi et la réception des paquets de routage et des données	Attirer les routes, ensuite supprimer les paquets de routage et des données	Le nœud malveillant se fait passer pour la destination en envoyant une RREP fabriquée, ensuite supprime les paquets des données	Les routes vers un nœud peuvent être perturbées en envoyant une RERR fabriquée
Consommation des ressources: Consommer les ressources du réseau				Inonder le réseau avec des RREQs et RERRs

Tableau 4.1 Attaques basiques contre le protocole AODV

de la source, l'adresse de la destination, le numéro de séquence de la source, et les drapeaux D et G. Pour atteindre ses objectifs le nœud malveillant tente de modifier les champs non mutables, ou modifier illicitement les champs mutables en violant la spécification. L'identifiant le nombre de sauts, et le numéro de séquence de la source sont les champs les plus critiques dans le message de demande de route. Par exemple le nœud malveillant peut augmenter la valeur de l'identifiant et le numéro de séquence de la source et diminuer le nombre de sauts. En agissant ainsi la demande de route sera considérée et traitée même par les nœuds qu'ils l'ont déjà traitée. Par conséquent ces derniers mettront à jour leurs tables de routage avec des fausses informations (numéro de séquence et nombre de sauts).

b Modification de réponse de route

Dans la réponse de route, le nombre de sauts est le seul champ mutable. Le nœud malveillant diminue le nombre de saut pour que la réponse de route qu'il retransmet soit retenue par le nœud source. Il peut aussi modifier certains champs non mutables tels que le numéro de séquence de la destination. Par exemple quand le nœud malveillant augmente le numéro de séquence de la destination et diminue le nombre de sauts, il assure ainsi sa place sur la route découverte. Cela est dû au fait que tous les nœuds qui reçoivent cette réponse de route sont obligés de mettre à jour leurs tables de routage.

c Modification d'erreur de route

Le nœud malveillant peut remplacer ou ajouter une ou plusieurs destinations qui sont joignables et actives à la liste des destinations non-joignables. Il peut aussi supprimer des destinations non-joignables pour faire croire qu'elles sont encore joignables.

Nous résumons dans le tableau 4.2 les modifications que peut faire le nœud malveillant sur les champs d'une demande de route, une réponse de route, et une erreur de route.

Champs	Messages	Modifications
Type	Tout	Changer le type de message
Drapeau	Tout	Changer la valeur
Nombre de sauts	RREQ, RREP	Réduire pour mettre à jour la route inverse (RREQ), ou augmenter pour annuler la mise à jour (RREP)
RREQ ID	RREQ	Augmenter pour rendre la RREQ plus fraiche, ou réduire pour la rendre obsolète
@ destination	RREQ, RREP	Changer pour fausser la découverte de route
DSN	RREQ, RREP	En cas de RREP multiples, augmenter le DSN permet de mettre à jour la route avec le plus grand numéro de séquence, le réduire permet d'annuler la mise à jour
@ source	RREQ, RREP	Changer pour fausser la découverte de route
SN	RREQ	Augmenter pour mettre à jour la route vers la source, diminuer pour annuler la mise à jour
Durée de vie	RREP	Augmenter ou diminuer pour prolonger ou réduire la durée de vie d'une route
Nb dest	RERR	Incrémenter ou diminuer pour ajouter ou supprimer une destination non joignable
Un_Dest IP	RERR	Changer pour faire croire qu'une autre destination est non joignable
Un_Dest_Seq	RERR	Augmenter pour invalider l'entrée correspondante, réduire invalider l'erreur de route

Tableau 4.2. Modifications possibles sur les champs des RREQ, RREP, RERR

4.2.1.3. Fabrication

a Fabrication de demande de route

Le nœud malveillant utilise les informations disponibles dans sa table de routage, ou des informations qu'il a collectées pour fabriquer des demandes de routes, et les diffuser ensuite dans le réseau. Dans ce cas-là le nœud malveillant peut appliquer sur la demande fabriquée toutes les

manipulations qu'on a décrites antérieurement dans le cas de la modification de demande de route. Le nœud malveillant peut aussi inonder le réseau en fabriquant des demandes de route de manière répétitive pour consommer l'énergie des nœuds et congestionner le réseau.

b Fabrication de réponse de route

Dès que le nœud malveillant reçoit une demande de route, il fabrique une réponse de route même s'il n'a pas de route valide vers la destination. Ou même s'il possède une route valide mais qu'il n'est pas autorisé à répondre (drapeau D activé signalant que seule la destination doit répondre). Le nœud malveillant peut fabriquer des réponses de route sans même recevoir des demandes de route. Dans ce cas-là il fabrique des réponses de route avec des numéros de sauts réduit et des numéros de séquences élevés proposant ainsi des routes plus courte et plus fraiche. Ce qui lui permet de s'insérer sur plusieurs routes soit en mettant à jour des routes existantes ou en créant des nouvelles entre le(s) nœud(s) cible(s) et les différentes destination.

c Fabrication d'erreur de route

Le nœud malveillant peut fabriquer un message d'erreur de route et annoncer autant de routes non-joignables provoquant l'invalidation de plusieurs routes dans les tables de routage des nœuds récepteurs.

4.2.2. Attaques élémentaires qui ne violent pas la spécification du protocole AODV

4.2.2.1. Trou de ver

Le nœud malveillant encapsule la demande de route reçue en paquet de données, ensuite il la transmit via le tunnel à son partenaire (se trouvant à plusieurs sauts de lui). Cela signifie, du point de vue du nombre de sauts que les nœuds sont voisins, et que le tunnel est invisible pour le protocole AODV. Par conséquent la route la plus courte entre les nœuds dans le voisinage du premier nœud malveillant et ceux dans le voisinage du deuxième nœud malveillant sera toujours via le tunnel. De la même façon la réponse de route sera transmise à travers le tunnel (voir figure 4.3).

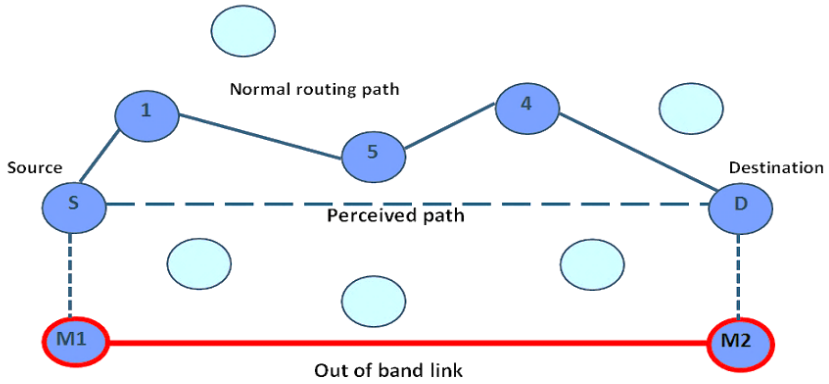


Figure 4.3 Trou de ver

4.2.2.2. Rushing

Dans cette attaque le nœud malveillant exploite la propriété du protocole AODV, qui consiste à retransmettre la demande de route qu'une seule fois par chaque nœud. Le nœud malveillant retransmet la demande de route rapidement pour qu'elle arrive au nœud destination avant les demandes de route légitimes, causant ainsi le rejet de ces dernières. Par conséquent, le nœud source sera incapable de trouver une route plus longue que deux sauts qui n'inclut pas le nœud malveillant. Pour faire parvenir sa demande de route avant les autres, le nœud malveillant ignore les délais d'attentes spécifiées par le Backoff (IEEE 802.11) au niveau de la couche MAC. Une autre stratégie que le nœud malveillant utilise est de diffuser la demande de route avec haute puissance, la transmettant ainsi sur une plus grande distance pour sauter plusieurs nœuds et atteindre la destination avant les autres (voir figure 4.4). Cependant la réponse de route ne pourra pas emprunter la route prise par la demande de route. Par conséquent les demandes de route légitimes seront bloquées parce que les nœuds sur la route ont déjà retransmit la demande de route transmise par le nœud malveillant.

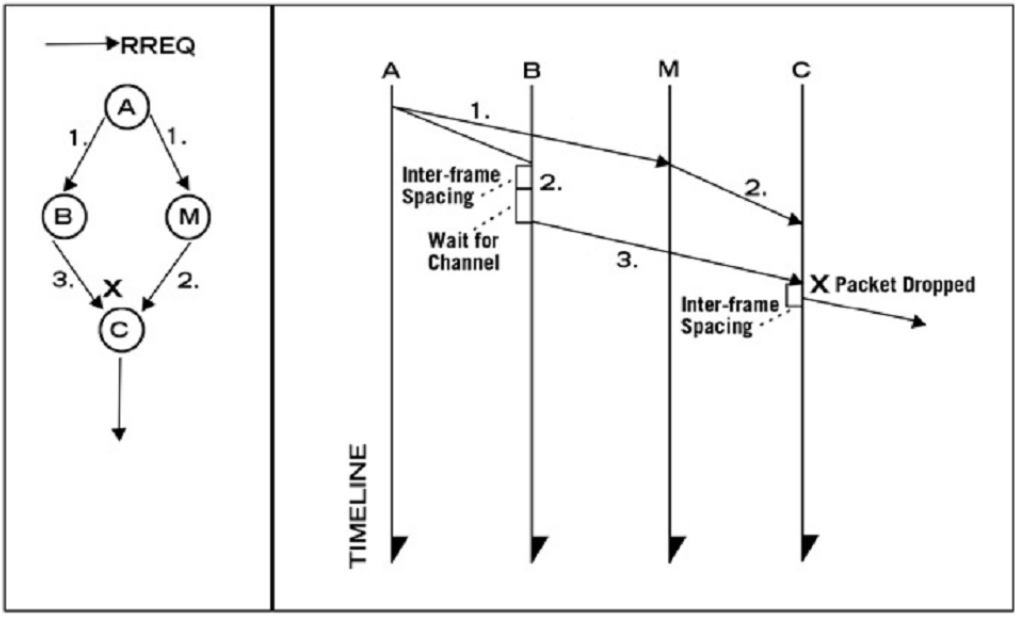


Figure 4.4 Rushing de la demande de route

4.2.3. Attaques composées

Comme nous l'avons expliqué dans la section 4.2, le nœud malveillant peut combiner des attaques élémentaires pour exécuter des attaques plus sophistiqués qui auront un impact plus important et persistant sur le réseau. Nous présentons dans ce qui suit des exemples et des scénarios des attaques composées citées dans la section 4.2.2 dans le cas du protocole AODV.

4.2.3.1. Invasion de route

Le nœud malveillant peut effectuer les attaques suivantes pour s'insérer dans une route existante, ou dans une route en cours de découverte:

- **Modification des demandes de routes**: l'incrémentation du RREQ ID et du numéro de séquence, et la diminution du nombre de sauts dans les demandes de routes reçues. Permet au nœud malveillant de mettre à jour les tables de routage de ses voisins, et d'être leur prochain saut pour ces routes.
- **Fabrication des demandes de routes**: contenant des RREQ ID et des numéros de séquence élevés, et des nombre de sauts réduits. Cette attaque provoque le même effet que la modification des demandes de routes.
- **Retransmission rapide**: des demandes de routes par précipitation (Rushing). Retransmission rapide des demandes et des réponses de routes en utilisant un tunnel (Trou de ver).

4.2.3.2. Isolation des nœuds

Dans cette attaque le nœud malveillant tente d'isoler un nœud ou un groupe de nœuds, en leurs fournissant des informations de routages incorrectes telles que des routes non-existantes. Le nœud malveillant peut même partitionner le réseau s'il est placé dans une position stratégique.

- **Modification des demandes et réponses de routes**: si le nœud malveillant est le voisin de la source de la demande de route, alors il peut l'isoler en incrémentant le RREQ ID et le numéro de séquence de la source, et en remplaçant l'adresse source dans l'entête IP par une adresse non existante. Lorsque les voisins du nœud malveillant reçoivent la demande de route, ils mettent à jour leurs prochains sauts vers le nœud victime avec l'adresse source dans l'entête IP. Par conséquent toutes les routes inverses crées ou qui ont été mises à jour pointeront vers une adresses non-existante, ce qui empêchera le nœud victime de recevoir les messages de routage et les données. La répétition de cette attaque permet d'isoler le nœud partiellement pendant une durée limité.

- **Fabrication des demandes et réponses de routes**: le nœud malveillant opère de la même manière que la modification. Il fabrique des demandes de routes avec des RREQ ID et numéros de séquence élevés, ou des réponses de routes avec des numéros de séquences de destinations élevé et des nombres de saut réduits. Il met une adresse IP non-existante dans l'entête IP. La répétition de cette attaque permet dans certains cas d'isoler le nœud complètement mais pour une durée limité.

4.2.3.3. Privation de sommeil

Pour consommer l'énergie des nœuds du réseau le nœud malveillant, exécute les attaques suivantes:

- **Modification répétée des demandes de route**: chaque fois que le nœud malveillant reçoit une demande de route il incrémente son RREQ ID, provoquant ainsi sa rediffusion à travers tous le réseau. La répétition de cette attaque génère une boucle de diffusion dans le réseau.
- **Fabrication répétée des demandes de routes**: le nœud malveillant fabrique de manière répétée des demandes de route en incrémentant à chaque fois le RREQ ID, provoquant une boucle de diffusion dans le réseau.
- **Fabrication répétée d'erreurs de routes**: consomme la bande passante et le temps de calcul des nœuds. D'abord, lorsqu'un un nœud reçoit une erreur de route fabriquée il la diffuse au voisins présents dans la liste des précurseurs, qui a leurs tour feront la même chose. Ensuite si le nœud à des données à envoyer mais que la route a été invalidé par l'erreur de route fabriquée, le nœud victime sera obligé d'initier une découverte de route provoquant la diffusion de plusieurs demandes de routes.
- **Création de boucle de routage**: le nœud malveillant peut créer des boucles de routage dans des routes existantes en fabriquant des demandes et des réponses de routes. Ning et Sun présentent dans [131] un scénario permettant la formation d'une boucle de routage dans une route déjà établie en fabriquant deux réponses de routes.

4.2.3.4. Trou noir et trou gris

Cette attaque est effectuée en combinant les attaques décrites dans l'invasion de route et la suppression.

4.2.3.5. Déni de service

Pour atteindre son objectif le nœud malveillant tente d'isoler le nœud cible du réseau, ou consommer ces ressources pour l'exclure du réseau. Cette attaque est le résultat de toutes les combinaisons possibles des attaques précédentes.

4.3. Conclusion

Dans ce chapitre nous avons présenté une analyse systématique des attaques de routage dans les réseaux ad hoc. En se basant sur le concept d'attaques élémentaires et d'attaques composées, nous avons modélisé et donné une nouvelle taxonomie des différentes attaques de routage. L'application de notre analyse sur le protocole AODV a permet d'examiner comment le nœud malveillant exécute les différentes attaques en exploitant les vulnérabilités du protocole de routage.

Chapitre 5 : Un système de détection et prévention d'intrusion basé sur la spécification

Dans ce chapitre nous proposons un nouveau système de détection et prévention d'intrusion pour prévenir et détecter les attaques et les comportements malveillants contre le routage dans les réseaux ad hoc. Le système que nous proposons est conçu pour détecter les attaques qui violent la spécification du protocole de routage telles que : modification, fabrication et rejeu. Nous proposons une technique de détection qui grâce un ensemble de règles définit le comportement normal du nœud du point de vue de l'opération de routage. Les règles de détection sont générées automatiquement sous forme de machine à états finis grâce à une méthode d'extraction automatique de la spécification. La méthode d'extraction que nous proposons s'inspire de la programmation logique inductive (PLI) et prend avantage des points communs entre les protocoles de routage ad hoc. Une instance du système de détection proposé est exécutée sur chaque nœud du réseau afin de contrôler et protéger les interactions du nœud hôte avec les autres nœuds du réseau. Les résultats des simulations montrent la capacité du système à détecter et empêcher la majorité des attaques qui violent la spécification avec un taux de faux positifs avoisinant les 2%.

Ce chapitre est organisé comme suit : La section 5.1 présente une nouvelle méthode d'extraction et de synthétisation automatique de la spécification du protocole de routage. Dans la section 5.2 nous discutons et annotons le modèle de spécification généré et représenté sous forme de machine à états finis. La section 5.3 décrit comment le système proposé détecte et empêche les différentes attaques de routage. La section 5.4 présente le mécanisme de réponse aux attaques. Dans la section 5.5 nous évaluons les performances du système proposé en considérons divers attaques et scénarios, et en présence de plusieurs nœuds malveillants. Finalement, la section 5.6 conclut ce chapitre.

5.1. Extraction automatique de la spécification du protocole de routage

Un système de détection d'intrusion basé sur la spécification est composé de deux éléments, le modèle de spécification et le mécanisme de contrôle (en anglais : monitoting mechanism). Le modèle de spécification définit le comportement valide du protocole. Le mécanisme de surveillance utilise le modèle de spécification pour contrôler les interactions du nœud hôte avec les autres nœuds. Toute action non-conforme à la spécification du protocole est détectée comme une violation de la spécification. Le bon fonctionnement du protocole de routage est bien défini dans les documents

RFCs (en anglais Requests For Comments). Les RFCs, littéralement «demande de commentaires», sont une série numérotée de documents officiels qui décrivent et spécifient la majorité des normes et protocoles liés à Internet et aux réseaux en général [140]. Cependant l'exploitation de la spécification dans le contexte de détection d'intrusion, nécessite la représentation de cette dernière par un modèle facilement vérifiable. La représentation de la spécification par un ensemble de règles facilement vérifiables assure une détection efficace et rapide.

L'extraction manuelle de la spécification constitue un des principaux inconvénients de la détection d'intrusion basée sur la spécification. L'extraction d'un modèle de spécification correct n'est pas une tâche facile, il s'agit d'un processus qui demande de l'expertise et du temps et qui est sujet aux erreurs. Par conséquent, nous pensons que l'utilisation d'une méthode d'extraction automatique permettrait de remédier à ces contraintes de conceptions. Bien que l'extraction automatique ne pourra pas inclure et porter toutes les propriétés et les fonctions du protocole, l'annotation manuelle du modèle et plus facile qu'une extraction complètement manuelle. Ainsi nous minimisons l'intervention de l'expert dans le processus de conception à l'annotation du modèle généré automatiquement.

Dans ce contexte, nous proposons une méthode qui permet d'extraire et générer automatiquement le modèle de spécification à partir des traces d'exécution du protocole de routage. Notre méthode est similaire à la programmation logique inductive (PLI) [142], qui à partir d'exemples et de connaissances développe une hypothèse. La figure 5.1 présente le processus de conception que nous proposons pour la détection d'intrusion à base de spécification. Le modèle de spécification généré automatiquement est représenté par un ensemble d'automates à états finis et transitions étendues EFSM (Extended Finite State Machine). Un EFSM est comme un automate à états finis, sauf que les états et les transitions peuvent porter un ensemble fini de paramètres que nous appelons ; conditions de transition et variables d'états. Nous proposons un algorithme générique d'extraction qui exploite les points communs entre les protocoles de routage ad hoc pour extraire automatiquement le modèle de spécification. L'algorithme d'extraction que nous proposons modélise la spécification du protocole sous forme d'ensemble d'automates à états finis (en anglais : Finite State Machine). Les états représentent la configuration du protocole, et les transitions entre les états montrent comment le protocole progresse d'une configuration à une autre. Un tel modèle de spécification couvre explicitement toutes les interactions possibles et légitimes entre les nœuds du réseau. La définition du comportement correct permet de détecter tout comportement différent de ce dernier comme violation de la spécification.

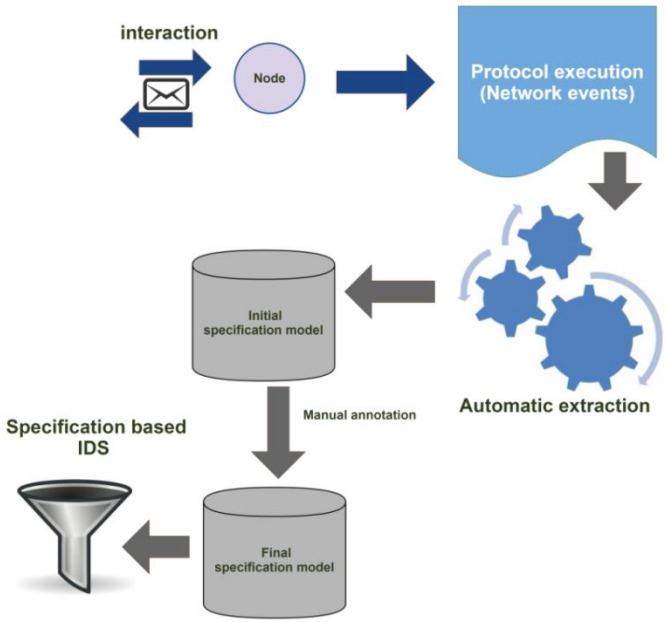


Figure 5.1 Processus de conception du système de détection d'intrusion

Nous développons un modèle abstrait de la spécification du protocole à partir de son exécution, c.-à-d. les séquences des messages de routage échangés pendant une session de découverte de route. Notre proposition s'inspire des travaux de Ko et al. [141] et Stakhanova et al. [94]. Dans [94] les auteurs ont proposé une méthode de synthétisation de la spécification du protocole de routage ad hoc. Ils ont modélisé la spécification sous forme de graphe orienté dans lequel les nœuds représentent la configuration du protocole et des arêtes indiquent comment le protocole évolue d'une configuration à une autre. La modélisation proposée dans [94] ne ressort pas le séquencement des actions et des évènements. Pour une meilleure modélisation du séquencement des évènements notre algorithme d'extraction modélise la spécification par un automate à états finis. La transition d'un état à un autre est déclenché par l'échange (l'envoie ou la réception) d'un message de routage. L'idée de notre méthode d'extraction de spécification est essentiellement similaire à la programmation logique inductive que nous discuterons dans la section suivante.

5.1.1. Programmation logique inductive

La programmation logique inductive PLI (en anglais : Inductive Logic Programming) a été définie comme l'intersection de la programmation logique et l'apprentissage automatique. Elle utilise la programmation logique comme une représentation uniforme des exemples, des connaissances et des hypothèses. La PLI construit des connaissances à partir d'exemples, tous deux représentés dans la logique du premier ordre, en inversant le processus d'inférence déductive. Contrairement à l'inférence déductive, la PLI tire une croyance générale des croyances spécifiques.

- Faits (Background knowledge) B:
  parent_de(charles,george).
  parent_de(george,diana).
  parent_de(bob,harry).
  parent_de(harry,elizabeth).

- Exemples positifs E^+:
  grandparent_de(charles,diana).
  grandparent_de(bob,elizabeth).

- Générer une hypothèse H:
  grandparent_de(X,Y) :- parent_de(X,Z),
  parent_de(Z,Y).

Nous considérons l'exemple ci-dessus *grandparent_de(bob,elizabeth)*. Nous essayons de trouver le sous-ensemble de connaissances relatif à cet exemple: parent_de (bob, harry), parent_de (harry, elizabeth). Former une règle à partir de ces faits: grandparent_de (bob, elizabeth): - parent_de (bob, harry), parent_de (harry, elizabeth). Ensuite, nous généralisons la règle: grandparent_de (X, Y): - parent_de (X, Z), parent_de (Z, Y). Enfin, nous vérifions si cette règle est valable en ce qui concerne les exemples positifs et négatifs. Formellement, si B désigne les connaissances de base, E désigne l'ensemble des exemples et H est l'hypothèse générée.

La propriété de complétude (1) signifie que l'hypothèse couvre tous les exemples positifs. La complétude est définie par rapport à l'ensemble des exemples positifs dans E^+. En plus des exemples positifs, un ensemble d'exemples négatifs E^- peut également être fourni, l'hypothèse ne doit pas couvrir les exemples négatifs. La propriété de cohérence (2) est définie uniquement par rapport aux exemples négatifs dans E^-.

$$\forall e \in E : B \wedge H \models e \qquad (1)$$

$$\forall \bar{e} \in \bar{E} : B \wedge H \not\models \bar{e} \qquad (2)$$

Dans le contexte d'extraction de spécification, le modèle de spécification représente l'hypothèse selon laquelle il sera démontré pour vérifier la complétude et la cohérence par rapport au comportement correct du nœud (exemples positifs) et le comportement qui viole la spécification (exemples négatifs).

5.1.2. Extraction de la spécification à partir des sessions de découvertes de routes

Nous modélisons la spécification du protocole considéré en utilisant l'ensemble des sessions de découvertes de routes. Une découverte de route entre deux nœuds (dans une direction) est représentée par une séquence de messages de routage initiés par une demande de route. Chaque message dans la séquence représente un état et comprend des informations correspondant au message de routage (ça dépend du protocole de routage utilisé, par exemple : adresse IP de la source et la destination, le nombre de sauts, les numéros de séquence, etc.). Les messages de routage sont mappés en fonction de la session de découverte de route à laquelle ils appartiennent. Une session de découverte de route peut être identifié de manière unique par les adresses source et destination. Les messages de routage : demande de route RREQ (Route REQuest) ; réponse de route (Route REPly) ; et erreur de route (Route ERRor) peuvent être mappé à leurs sessions de découverte de route par les champs suivants : adresse source et destination dans le cas du RREQ et RREP), et la destination injoignable dans le cas du RERR. Nous définissons formellement le modèle de spécification comme un automate à états finis: $\boldsymbol{SM} = (\boldsymbol{S}, \boldsymbol{S_T}, \boldsymbol{T}, \boldsymbol{A})$ où $\boldsymbol{S}$ est l'ensemble d'états, $\boldsymbol{S_T} \in \mathbf{S}$ est l'état initial, $\boldsymbol{T}$ est l'ensemble des transitions ($\boldsymbol{S} \times \boldsymbol{C} \times \boldsymbol{S}$) et $\boldsymbol{C}$ est l'ensemble des conditions de transition (contraintes). La transition d'un état à un autre est déclenchée par l'envoi ou la réception d'un message de routage. Dans ce contexte la méthode d'extraction construit le modèle de spécification (l'hypothèse) à partir des sessions de découvertes de routes (les exemples). La méthode d'extraction doit vérifier la complétude et la consistance.

- **Complétude** : s'assurer que toutes les sessions de découvertes de routes appartenants à l'ensemble des exemples existent dans le modèle de spécification :

$$\begin{aligned} &\forall\, s_1^e \xrightarrow{x1} s_2^e \xrightarrow{x2} s_3^e \dots \xrightarrow{x_{n-1}} S_n^e \in E: \\ &\exists\, s_1 \xrightarrow{y1} s_2 \xrightarrow{y2} s_3 \dots \xrightarrow{y_{n-1}} s_n \in SM: \\ &S_1 = S^0 \wedge \forall i: s_i^e = s_i \wedge x_i = y_i \end{aligned} \qquad (\mathbf{3})$$

- **Consistance** : s'assurer que le modèle de spécification ne comprend aucun exemple négatif, soit les états sont différents ou les conditions de transition sont inconsistantes:

$$\begin{aligned} &\forall\, s_1^{\bar{e}} \xrightarrow{x1} s_2^{\bar{e}} \xrightarrow{x2} s_3^{\bar{e}} \dots \xrightarrow{x_{n-1}} S_n^{\bar{e}} \in \bar{E}: \\ &\forall\, s_1 \xrightarrow{y1} s_2 \xrightarrow{y2} s_3 \dots \xrightarrow{y_{n-1}} s_n \in SM: \\ &S_1 = S^0 \wedge \exists\, i: s_i^e \neq s_i \vee x_i \neq y_i \end{aligned} \qquad (\mathbf{4})$$

5.1.1.1 Algorithme d'extraction de spécification

Nous adoptons la notation dans le tableau 5.1 pour le reste de ce document.

Notations	Définitions
RTE	Entrée dans la table de routage
BID	Identificateur de la demande de route
Orig	Adresse de la source
Dst	Adresse de la destination
DSN	Numéro de séquence de la destination
OSN	Numéro de séquence de la source
ip_src	Adresse source au niveau de l'entête IP
ip_dst	Adresse destination au niveau de l'entête IP
TTL	Valeur du champ TTL
rcv_	Désigne la réception
New	Désigne une nouvelle valeur du champ précédé
TTL_initial	L'ensemble des valeurs possible du champ TTL
Type-du-message. Champ	Désigne un champ dans le message
Stat	Désigne l'état d'une entrée dans la table de routage
my_addr	Désigne l'adresse IP du nœud récepteur
dst_count	Nombre de destination injoignable dans une erreur de route RERR
unr_dst	Destination injoignable

Tableau 5.1 Notation et abréviation

L'algorithme dans la figure 5.2 présente la procédure d'extraction du modèle de spécification à partir de l'ensemble des sessions de découvertes de routes. Dans le contexte de notre travail, nous définissons un évènement comme l'envoi ou la réception d'un message de routage. L'envoi et la réception d'un message de routage est effectué soit pour découvrir une nouvelle route ou maintenir une route existante. L'établissement d'une nouvelle route se fait via le processus de découverte de route, alors que la maintenance est effectuée via les échanges périodiques (message Hello) et l'envoi des messages d'erreur de route RERR. La découverte de route entre deux nœuds X et Y est initiée par l'envoi de demande de route du nœud source X vers le nœud destination Y. L'envoi de demande de route est suivi soit par l'envoi d'une réponse de route ou par d'autres envois de demandes de routes (plusieurs tentatives de découvertes de routes). Puisque le processus de maintenance de route dépend de celui de la découverte, tous les messages de routage échangés peuvent être mappés et groupés selon la session de découverte de route à laquelle ils appartiennent. L'algorithme parcourt la liste des événements en les examinant un par un. Une fois la session de découverte de route à laquelle appartient l'événement est identifié, la fonction ***Build_FSM*** est appelée. En fonction

```
Compare_State_Variables (Last_Message, Current_Message)
Begin
        if (Last Message.type = Current Message.type) then
                Compare field by field
        Else
                Compare common fields
        EndIf
End

FSM Build_FSM (Event e, Stat_Set)
Begin
If (e.stat ∉ Stat_Set) then
        Create new Stat S;
        Add S to Stat_Set;
Else
        S = e.stat;   // assign to S the existing state
EndIf

if(last_stat ≠ null) then
        Create transition (Last_stat, S);
        Compare_State_variables ( );
        Set_Transition_conditions ( );    // from compare_state_variables
EndIf
Last_stat = S;
Return FSM
End

FSM Specification_extraction (Event_list)
Begin
while (Event_list not end)
        for each (Route_discovery_session ) do
                if (event ∈ Route_discovery_session) then
                        if (node = originator) then
                                originator_FSM = Build_FSM (e, orig_stats);
                        Else if (node = destination) then
                                destination_FSM = Build_FSM (e, dst_stats);
                            Else
                                intermediate_FSM = Build_FSM (e, inter_stats);
                        EndIf
        Endfor

       if (event = first route discovery) then
                Create new Route_discovery_session;
```

Figure 5.2 Algorithme 1 Extraction Automatique de la spécification

du statut ou du rôle du nœud vis-à-vis l'évènement (source, intermédiaire, destination) dans la session de découverte de route, l'algorithme construit l'automate correspondant. La fonction ***Build_FSM*** vérifie d'abord si l'état correspondant à l'événement examiné appartient à l'ensemble des états ***Stat_Set.*** Si ce n'est pas le cas, un nouvel état correspondant à l'évènement est créé et ajouté à l'ensemble ***Stat_Set***. Ensuite ***Build_FSM*** crée une nouvelle transition entre le dernier état et le nouvel état, et détermine les conditions de transition qui sont généré en comparant les variables d'états. L'algorithme compare les variables d'états correspondants aux champs communs entre l'état courant et l'état précédent. Par exemple dans le cas du protocole AODV, tous les champs de la transition entre l'état ***RREQ_rcv*** et ***RREQ_send***, seront comparés parce qu'il s'agit du même type de message. Dans le cas où les deux états concernent différents type de messages, seuls les champs

communs seront comparés. Par exemple dans la transition entre l'état **RREQ_rcv** et **RREP-rcv**, seul le numéro de séquence de destination sera comparé.

La fonction ***set_conditions*** génère les conditions de transition à partir des résultats retournés par la fonction **Compare_State_variables**. Si l'événement examiné ne correspond à aucune session en cours, et qu'il s'agit de la première demande de route entre les deux nœuds, alors une nouvelle session de découverte de route sera créée. Nous appliquons l'algorithme d'extraction automatique sur les traces d'une exécution correcte du protocole de routage AODV obtenues à partir du simulateur ns-2 [136] (voir figure 5.3). L'algorithme d'extraction ne considère que les paramètres relatifs au protocole AODV et les champs IP suivants : adresse IP source, adresse IP destination, et le TTL.

event	time	from node	to node	pkt type	pkt size	flags	fid	src addr	dst addr	seq num	pkt id

r : receive (at to_node)
+ : enqueue (at queue)
- : dequeue (at queue)
d : drop (at queue)

src_addr : node.port (3.0)
dst_addr : node.port (0.0)

```
s 1.388112419 _2_ AGT  --- 0 cbr 512 [0 0 0 0] ------- [2:0 3:0 32 0] [0] 0 1
r 1.388112419 _2_ RTR  --- 0 cbr 512 [0 0 0 0] ------- [2:0 3:0 32 0] [0] 0 1
s 1.388112419 _2_ RTR  --- 0 AODV 48 [0 0 0 0] ------- [2:255 -1:255 5 0] [0x2 1 1 [3 0] [2 4]] (REQUEST)
r 1.389100724 _8_ RTR  --- 0 AODV 48 [0 ffffffff 2 800] ------- [2:255 -1:255 5 0] [0x2 1 1 [3 0] [2 4]] (REQUEST)
r 1.389100801 _3_ RTR  --- 0 AODV 48 [0 ffffffff 2 800] ------- [2:255 -1:255 5 0] [0x2 1 1 [3 0] [2 4]] (REQUEST)
s 1.389100801 _3_ RTR  --- 0 AODV 44 [0 0 0 0] ------- [3:255 2:255 30 2] [0x4 1 [3 4] 10.000000] (REPLY)
s 1.389981951 _8_ RTR  --- 0 AODV 48 [0 ffffffff 2 800] ------- [8:255 -1:255 4 0] [0x2 2 1 [3 0] [2 4]] (REQUEST)
r 1.391087037 _3_ RTR  --- 0 AODV 48 [0 ffffffff 8 800] ------- [8:255 -1:255 4 0] [0x2 2 1 [3 0] [2 4]] (REQUEST)
r 1.391087225 _2_ RTR  --- 0 AODV 48 [0 ffffffff 8 800] ------- [8:255 -1:255 4 0] [0x2 2 1 [3 0] [2 4]] (REQUEST)
r 1.391087691 _7_ RTR  --- 0 AODV 48 [0 ffffffff 8 800] ------- [8:255 -1:255 4 0] [0x2 2 1 [3 0] [2 4]] (REQUEST)
s 1.393433709 _7_ RTR  --- 0 AODV 48 [0 ffffffff 8 800] ------- [7:255 -1:255 3 0] [0x2 3 1 [3 0] [2 4]] (REQUEST)
r 1.394939518 _2_ RTR  --- 0 AODV 44 [13a 2 3 800] ------- [3:255 2:255 30 2] [0x4 1 [3 4] 10.000000] (REPLY)
s 1.394939518 _2_ RTR  --- 0 cbr 532 [0 0 0 0] ------- [2:0 3:0 30 3] [0] 0 1
r 1.400740665 _3_ AGT  --- 0 cbr 532 [13a 3 2 800] ------- [2:0 3:0 30 3] [0] 1 1
r 1.402243766 _14_ RTR  --- 0 AODV 48 [0 ffffffff 7 800] ------- [7:255 -1:255 3 0] [0x2 3 1 [3 0] [2 4]] (REQUEST)
r 1.402244059 _17_ RTR  --- 0 AODV 48 [0 ffffffff 7 800] ------- [7:255 -1:255 3 0] [0x2 3 1 [3 0] [2 4]] (REQUEST)
r 1.402244062 _3_ RTR  --- 0 AODV 48 [0 ffffffff 7 800] ------- [7:255 -1:255 3 0] [0x2 3 1 [3 0] [2 4]] (REQUEST)
```

Figure 5.3 Fichier trace généré par ns-2

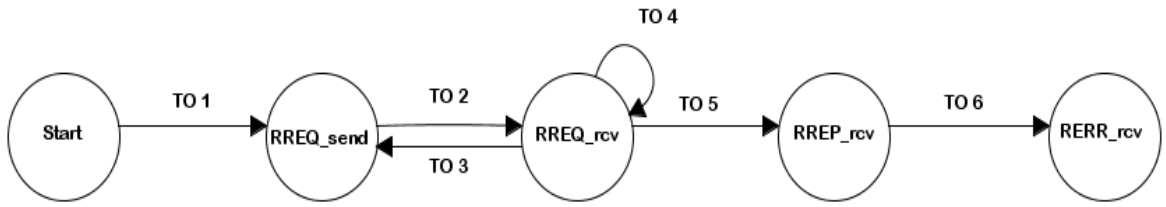


Figure 5.4 Automate à etats finis du noeud source

La figure 5.4 présente l'automate (généré automatiquement par l'algorithme) correspondant à l'exécution du protocole AODV par le nœud source. Il contient cinq états: RREQ_send, RREQ_rcv, RREP_rcv, RERR_rcv. L'état de de départ représente l'étape initiale où le nœud ne participe encore pas dans la session de découverte de route. Une fois que le nœud source génère et envoie la

demande de route, l'automate se met à l'état RREQ_send. Ensuite le nœud source reçoit sa propre demande de route diffusée par ses nœuds voisins (état RREQ_rcv). Si le nœud source au bout d'un intervalle d'attente donné (timeout) ne reçoit pas de réponse de route, il génère et envoie une nouvelle demande de route et ainsi de suite jusqu'à qu'il atteint le nombre maximal de tentatives de découverte de route (RRER_retries). Sinon, à la réception de la réponse de route l'automate se met à l'état RREP_rcv et ainsi la découverte de route se termine avec succès.

Dans le cas d'une perte de liaison (causé par la mobilité des nœuds) à un point de la route, le nœud précurseur tente de réparer la route en initiant la procédure de réparation locale. Si la réparation ne réussit pas, alors le message d'erreur est transféré au nœud source à travers la route inverse (RERR_rcv). Les contraintes de transition suivantes sont retournées par la fonction de comparaison *Compare_State_Variables* :

- **TO 1:** pas de comparaison possible
- **TO 2:** $\boldsymbol{C1}$: $(RREQ.HC > HC\ ) \wedge (RREQ.BID = BID) \wedge (RREQ.OSN = OSN) \wedge (RREQ.DSN \geq DSN) \wedge (RREQ.ip\_src \neq ip\_src) \wedge (RREQ.TTL < TTL)$
- **TO 3**: $\boldsymbol{C2}$: $(RREQ.HC = HC\ ) \wedge (RREQ.BID = BID + 1) \wedge (RREQ.OSN = OSN + 1) \wedge (RREQ.DSN \geq DSN) \wedge (RREQ.ip\_src = ip\_src) \wedge (RREQ.TTL > TTL)$
- **TO 4:** ***C1***
- **TO 5:** $\boldsymbol{C3}$: $RREP.DSN \geq rcv\_RREQ.DSN$
- **TO 6:** $\boldsymbol{C4}$: $(RERR.DSN > RREP.DSN) \wedge (RERR.ip\_src = RREP.ip\_src)$

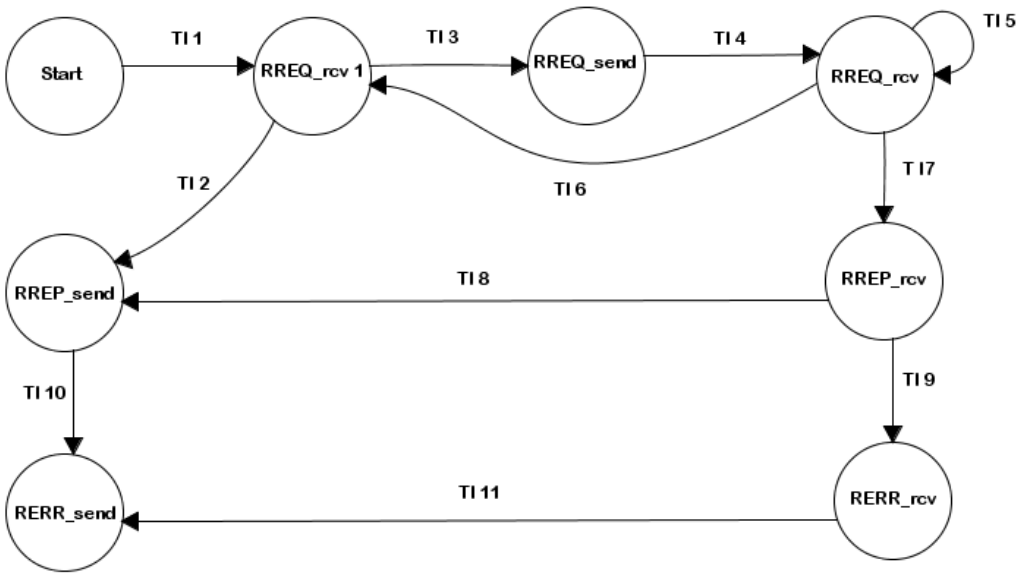


Figure 5.5 Automate à états finis du nœud intermédiaire

La figure 5.5 présente l'automate (généré automatiquement par l'algorithme) correspondant à l'exécution du protocole AODV par un nœud intermédiaire. L'automate contient huit états: Start, RREQ_rcv_1 ; RREQ_send ; RREQ_rcv ; RREP_send ; RREP_rcv ; RERR_send ; RERR_rcv. L'état RREQ_rcv_1 décrit la première réception de la demande de route, alors que l'état RREQ_rcv décrit la réception d'une demande de route qui a été déjà reçue. Quand le nœud intermédiaire reçoit la demande de route pour la première fois, d'abord il consulte sa table de routage pour vérifier s'il dispose d'une route vers la destination demandée. S'il dispose d'une route valide et fraiche (le numéro de séquence de destination dans le demande de route est inférieur à celui détenu par le nœud) et le champ D_flag n'est pas activé, alors il envoie une réponse de route (l'état RREP_send). Sinon le nœud intermédiaire diffuse la demande de route au niveau de son voisinage (état RREQ_send). L'automate se met à l'état RREQ_rcv lors de la réception de la demande de route diffusée par son voisinage. Ensuite le nœud intermédiaire peut recevoir soit une réponse de route et ainsi l'automate se met à l'état RREP_rcv, ou reçoit une nouvelle demande de route (dans le cas d'une nouvelle tentative de découverte de route) et dans ce cas-là l'automate se met à l'état RREQ_rcv_1. Le nœud intermédiaire transfert la réponse de route à son prochain saut vers le nœud source (état RREP_send). Le nœud intermédiaire envoie une erreur de route si : (i) il a envoyé une réponse de route, ensuite il a détecté une perte de liaison (RERR_send) ; (ii) ou il reçut une erreur de route de la part de son prochain saut envers la destination (la transition de RERR_rcv to RERR_send). Les contraintes de transition suivantes sont générées par la fonction de comparaison Compare_State_Variables :

- **TI 1:** pas de comparaison possible
- **TI 2:** $\boldsymbol{C5}$: $(RREP.DSN > rcv\_RREQ.DSN)$
- **TI 3:** $\boldsymbol{C6}$: $(RREQ.HC = rcv\_RREQ.HC + 1) \wedge (RREQ.BID = rcv\_RREQ.BID) \wedge (RREQ.OSN = rcv\_RREQ.OSN) \wedge (RREQ.DSN \geq rcv\_RREQ.DSN) \wedge (RREQ.ip\_src \neq rcv\_RREQ.ip\_src) \wedge (RREQ.TTL = rcv\_RREQ.TTL - 1)$
- **TI 4:** $\boldsymbol{C7}$: $(RREQ.BID = rcv\_RREQ.BID) \wedge (RREQ.OSN = rcv\_RREQ.OSN) \wedge (RREQ.DSN \geq rcv\_RREQ.DSN)$
- **TI 5:** $\boldsymbol{C7}$
- **TI 6:** $\boldsymbol{C8}$: $(RREQ.BID = BID + 1) \wedge (RREQ.OSN = OSN + 1) \wedge (RREQ.DSN \geq DSN)$
- **TI 7:** $\boldsymbol{C3}$
- **TI 8:** $\boldsymbol{C9}$: $(RREP.orig = rcv\_RREP.orig) \wedge (RREP.dst = rcv\_RREP.dst) \wedge (RREP.HC = HC + 1) \wedge (RREP.DSN = DSN) \wedge (RREP.lifetime = lifetime) \wedge (RREP.TTL = TTL - 1)$
- **TI 9:** $\boldsymbol{C10}$: $RERR.DSN > DSN$
- **TI 10:** $\boldsymbol{C10}$
- **TI 11:** $\boldsymbol{C11}$: $(RERR.unr\_dst = unr\_dst) \wedge (RERR.DSN = DSN) \wedge (RERR.TTL = TTL)$

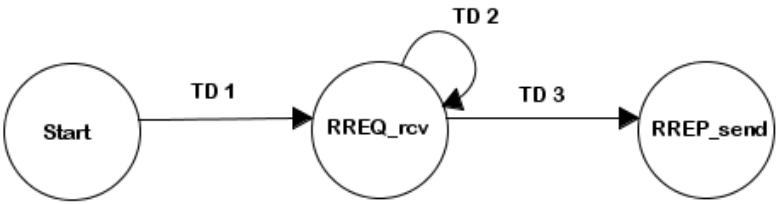


Figure 5.6 Automate à états finis du nœud destination

La figure 5.6 présente l'automate (généré automatiquement par l'algorithme) correspondant à l'exécution du protocole AODV par le nœud destination. Il contient trois états: start, RREQ_rcv, RREP_send. Le nœud de destination peut recevoir plusieurs copies de la même demande de route RREQ, vue que les demandes de routes empruntent des routes différentes avant d'atteindre le nœud destination. Cependant le nœud destination ne répond qu'à la première demande de route, ainsi l'automate se met à l'état RREP_send. Puisque le nœud destination ne peut pas savoir s'il n'est plus joignable pour l'un de ses précurseurs (suite à une perte de liaison par exemple) il ne génère et transfert pas d'erreur de route. Les contraintes de transition suivantes sont générées par la fonction de comparaison Compare_State_Variables :

- **TD 1**: pas de comparaison possible
- **TD 2:** $\boldsymbol{C6}$
- **TD 3:** $\boldsymbol{C5}$

5.2. Annotation manuelle

La spécification générée automatiquement ne peut pas être complète en raison de certaines règles et propriétés du protocole, qui ne se manifestent pas explicitement pendant l'exécution du protocole et ainsi ne peuvent pas être extraites. Par conséquent l'annotation manuelle du modèle est nécessaire pour fournir un modèle de spécification complet et correct. Nous nous référons à la RFC du protocole AODV [13] pour compléter les contraintes et conditions de transition générées par la fonction ***Set_Transition_conditions*** et ainsi compléter le modèle de spécification.

5.2.1 Règles et propriétés du protocole AODV

5.2.1.1 Rayon de diffusion

Le protocole AODV emploie la technique de recherche à anneau d'expansion afin d'éviter une large diffusion de la demande de route à travers le réseau. Au départ, le nœud source utilise TTL_START comme valeur du champ TTL_value au niveau de l'entête IP. Il attend la réception de réponse de route pour un intervalle de temps appelé RING_TRAVERSAL_TIME qui est calculé en utilisant la valeur

du TTL_value. Si le nœud source ne reçoit pas de réponse de route au-delà de cet intervalle, il génère et envoie une nouvelle demande de route dont la valeur du TTL_value (au niveau de l'entête IP) est incrémentée avec l'intervalle TTL_increment et ainsi de suite. Lorsque la valeur du champ TTL_value atteint le seuil TTL_THRESHOLD le nœud source utilisera désormais le diamètre du réseau NET_DIAMETER comme valeur du champ TTL_value [13]. Nous représentons le processus de recherche à anneau d'expansion par l'équation 5 (***n*** représente le nombre de tentatives de découverte de route).

$$\forall\,(\boldsymbol{n}\in\ \boldsymbol{N})\wedge(\boldsymbol{n}\ \geq \mathbf{1})\ \wedge \boldsymbol{TTL\_value} < \boldsymbol{Threshold}\,, \qquad (\mathbf{5})$$
$$\boldsymbol{TTl\_value}\ = \boldsymbol{TTL\_start}\ +\ (\boldsymbol{n}-\mathbf{1})\,\boldsymbol{TTL\_increment}$$

5.2.1.2 Temps d'attente entre les demandes de route consécutives

Après l'émission d'une demande de route le nœud source doit attendre la réception de la réponse de route pour un intervalle de temps déterminé (timeout). Si la réponse de route n'est pas reçue au bout de cet intervalle, le nœud source envoie une nouvelle demande de route. Le nœud source continue à envoyer des demandes de routes jusqu'à atteindre le nombre maximal de tentatives RREQ_RETRIES. L'équation 6 décrit comment le nœud source calcule le temps d'attente (timeout) entre les tentatives de découverte de route. Quand la valeur du champ TTL atteint le diamètre du réseau Network_diameter, le protocole AODV emploie le mécanisme du Backoff binaire exponentiel (binary exponential Backoff) pour éviter et diminuer les congestions dans le réseau. C'est à partir de la deuxième tentative de découverte de route avec un TTL égale au diamètre du réseau que le mécanisme du Backoff est employé. Il s'agit de multiplier par deux le temps d'attente pour chaque nouvelle tentative. Nous proposons l'équation (7) pour calculer le temps d'attente lorsque le Backoff est employé, ***m*** représente le nombre de tentative de découverte de route avec un TTL égale au diamètre du réseau.

$$\boldsymbol{TTL}\ < \boldsymbol{Threshold}\,,$$
$$\boldsymbol{Timeout} = \mathbf{2}\ *\ \boldsymbol{node\_traversal\_time}\ *\ (\boldsymbol{TTL\_value}\ +\ \boldsymbol{timeout\_buffer}) \qquad (\mathbf{6})$$

$$\boldsymbol{TTL} \geq \boldsymbol{Threshold}, \forall\,(\boldsymbol{m}\in\ \boldsymbol{N})\wedge(\boldsymbol{m}\ \geq \mathbf{1})$$
$$\boldsymbol{Timeout_m}\ = \mathbf{2}^{\boldsymbol{m}} * \boldsymbol{node\_traversal\_time}\ *\ (\boldsymbol{Network\_diameter}\ +\ \boldsymbol{timeout\_buffer}) \qquad (\mathbf{7})$$

5.2.1.3 Retransmission de la demande de route

Lorsqu'un nœud génère une nouvelle demande de route, il met le nombre de sauts à zéro et calcule la valeur du TTL (selon la procédure de recherche à anneau d'expansion). A chaque fois que la demande de route est retransmise à travers le réseau, la valeur du champ TTL_value (au niveau de l'entête IP) est décrémentée, alors que le nombre de sauts (au niveau de l'entête RREQ) est

incrémenté pour le compte du nouveau saut. Ce qui fait que la somme des valeurs des champs TTL et nombre de sauts dans la demande de route est toujours égale à la valeur initiale du champ TTL_value calculée par le nœud source. Donc en sachant le nombre de tentatives de découverte de route ***n***, on peut calculer la valeur initiale du champ TTL. Nous représentons cette corrélation par l'équation (8), qui permet de vérifier si le nombre de sauts a été modifié ou non. Même si le nombre de tentatives de découverte de route ***n*** est inconnu on peut détecter les incohérences qui résultent de la manipulation du nombre de sauts en utilisant l'équation (9) ou (10).

$$((RREQ.HC + RREQ.TTL < TTL\_threshold)\ is\ True) \wedge (\exists\ n \in connaissance\ locale) \Rightarrow ((\boldsymbol{RREQ.HC + RREQ.TTL}) - \boldsymbol{TTL\_start})/\boldsymbol{TTL\_increment}) + \mathbf{1} = \boldsymbol{n} \quad (\mathbf{8})$$

$$((RREQ.HC + RREQ.TTL) - TTL\_start)/TTL\_increment) + 1 \in N \Leftrightarrow ((\boldsymbol{RREQ.HC + RREQ.TTL}) - \boldsymbol{TTL\_start}) \ \%\ \boldsymbol{TTL\_increment} = \mathbf{0} \quad (\mathbf{9})$$

$$(RREQ.HC + RREQ.TTL \geq TTL\_threshold)\ est\ Vrai \Rightarrow \boldsymbol{RREQ.HC + RREQ.TTL} = \boldsymbol{Network\_diameter} \quad (\mathbf{10})$$

5.2.2 Annotation des conditions de transition

5.2.2.1 Nœud source

- **TO 1:** $(\exists\ data) \wedge \Big(\big(RTE_{RREQ.dst} \notin RT\big) \vee \big(\big(RTE_{RREQ.dst} \in RT\big) \wedge \big(RTE_{RREQ.dst}.stat = down\big)\big)\Big) \wedge (RREQ.HC = 0) \wedge (RREQ.BID = BID + 1, RREQ.OSN = OSN + 1, RREQ.ip\_src = orig = node\_ip)$
- **TO 2:** $((C1) \wedge (ip\_src \notin Senders\_list))$
- **TO 3:** $(C2) \wedge (RREQ\_atemp < RREQ\_retries) \wedge (RREQs\_interval \geq Timeout)$
- **TO 4:** $((C1) \wedge (ip\_src \notin Senders\_list))$
- **TO 5:** $(C3) \wedge (dst = rcv\_RREQ.dst) \wedge (orig = rcv\_RREQ.orig) \wedge (lifetime = My\_Route\_Timeout)$
- **TO 6:** $(dstCount \geq 1) \wedge (TTL = 1) \wedge (RERR.ip\_src = RTE_{unr\_dst}.NH) \wedge (RERR.DSN > RTE_{unr\_dst}.DSN)$

5.2.2.2 Nœud intermédiaire

- **TI 1:** $((HC + TTL - TTL_{start})\ \%\ TTL\_incr = 0) \wedge ((HC + TTL - TTL\_start)\ /TTL\_incr > 0) \wedge (((ip\_src = orig) \wedge (HC = 0)) \vee ((ip\_src \neq orig) \wedge (HC > 0)))$
- **TI 2:**
  $(\exists\ RTE_{RREQ.dst}) \wedge (RTE_{RREQ.dst}.DSN \geq rcv\_RREQ.DSN) \wedge (rcv\_RREQ.D\_flag \neq 1) \wedge (RREP.HC = RTE_{RREQ.dst}.HC) \wedge (RREP.dst = rcv\_RREQ.dst) \wedge (RREP.DSN = RTE_{RREQ.dst}.DSN) \wedge (RREP.orig = rcv\_RREQ.orig) \wedge (RREP.ip\_dst = rcv\_RREQ.ip\_src))$
- **TI 3:** $(((RTE_{RREQ.dst} \notin RT) \vee ((RTE_{RREQ.dst} \in RT) \wedge (RTE_{RREQ.dst}.DSN < RREQ.DSN)) \vee ((RTE_{RREQ.dst} \in RT) \wedge (RTE_{RREQ.dst}.DSN \geq RREQ.DSN)) \wedge (RREQ.D\_flag = 1))) \wedge (C7)$

- **TI 4:** $(C7) \wedge (RREQ.ip\_src \notin senders\_list)$
- **TI 5:** $(C7) \wedge (RREQ.ip\_src \notin senders\_list)$
- **TI 6:**
  $((RREQ\_attempts < RREQ\_retries) \wedge (RREQ\_interval \geq timeout)) \wedge ((HC + TTL) - TTL\_start)/(TTL\_increment)) + 1 = RREQ\_attempts) \wedge (C8)$
- **TI 7:**
  $(RREP.dst = rcv\_RREQ.dst) \wedge (RREP.DSN \geq rcv\_RREQ.DSN) \wedge (RREP.orig = rcv\_RREQ.orig) \wedge (RREP.lifetime = My\_Route\_Timeout)$
- **TI 8:** $\big(if\ (\exists\ RTE_{RREP.dst}) \wedge (RREP.DSN \geq RTE_{RREP.dst}.DSN)\big) \wedge (ip\_dst = RTE_{orig}.NH) \wedge (C9)$
- **TI 9:** $(C10) \wedge (dstCount \geq 1) \wedge (TTL = 1) \wedge (RERR.ip\_src = RTE_{unr\_dst}.NH)$
- **TI 10:** $\big((link\ break) \wedge (no\ local\ reparir)\big) \wedge (C10) \wedge (TTL = 1) \wedge (RERR.ip\_src = RTE_{unr\_dst}.NH)$
- **TI 11**: $(C11) \wedge (RERR.ip\_src = RTE_{unr\_dst}.NH)$

5.2.2.3 Nœud destination

- **TD 1:** $((HC + TTL - TTL_{start})\ \%\ TTL\_incr = 0) \wedge ((HC + TTL - TTL\_start)/TTL\_incr > 0)$
- **TD 2:** $(C7) \wedge (RREQ.ip\_src \notin senders\_list)$
- **TD 3:** $(RREP.dst = RREP.ip\_src = node\_ip) \wedge (RREP.DSN \geq rcv\_RREQ.DSN) \wedge (RREP.orig = rcv\_RREQ.orig) \wedge (RREP.HC = 0) \wedge (RREP.ip\_dst = rcv\_RREQ.ip\_src) \wedge (lifetime = My\_Route\_Timeout)$

5.3. Détection des violations de la spécification

La transition d'un état à un autre est déclenchée par l'envoie ou la réception d'un message de routage. Les attaques de routage qui violent la spécification sont considérées comme des violations de l'automate qui modélise la spécification du protocole. Dans ce contexte nous classons les violations de spécification en deux catégories : (1) transition inexistantes ; (2) transition incorrecte : qui ne satisfait pas les conditions de transition. Le système de détection et prévention d'intrusion SIDPS contrôle l'exécution du protocole en examinant les états et les transitions. L'état initial ***start*** désigne le début de la session (état inactif), où le système SIDPS est en attente d'évènement pour commencer la session. SIDPS commence la surveillance des interactions du nœud hôte, si ce dernier s'apprête à envoyer ou vient de recevoir un message de routage. SIDPS commence d'abord par vérifier la validité de la transition en s'assurant de son existence. Ensuite il vérifie si les conditions de transition ont été satisfaites. Dans ce qui suit nous montrons comment le système de détection d'intrusion détecte les différentes attaques de routage.

5.3.1 Modification

5.3.1.1 Demande de route

- **Modification du nombre de sauts** : le nœud malveillant manipule le nombre de sauts pour qu'il soit sélectionné lors de la découverte de route. Cette modification est détectée par SIDPS comme une transition incorrecte : une violation des conditions de la transition TI1 dans le cas du nœud intermédiaire, et de la transition TD1 dans le cas du nœud destination. Le nombre de sauts modifié ne satisfait pas l'équation suivante :

  $((HC + TTL - TTL_{start}) \% TTL\_incr = 0) \wedge ((HC + TTL - TTL\_start) / TTL\_incr > 0)$

- **Modification du format** : le nœud malveillant peut créer une demande de route valide mais sans respecter les règles du protocole, par exemple il peut incrémenter le numéro de séquence et l'identificateur de la requête (BID) par 2 au lieu de 1. La modification est détectée par SIDPS (du nœud hôte) comme une violation des conditions de la transition TO1. L'IDS du nœud recevant la demande de route, détecte la modification comme une violation des conditions des transitions suivantes : TI1, TI4, TI5 et TI6 (si le nœud récepteur est un nœud intermédiaire), ou comme TD1, TD2 (si le nœud récepteur est la destination).
- **Retransmission incorrecte** : pour être sélectionner lors de la procédure de découverte de route, le nœud malveillant augmente le numéro de séquence de la source ou diminue le nombre de sauts. Le nœud malveillant peut aussi faire absolument le contraire (c.-à-d. diminuer le numéro de séquence de la source et augmenter le nombre de sauts) pour faire rejeter la demande de route et ainsi déstabiliser l'opération de routage. Au niveau du nœud hôte (celui qui envoie la demande) SIDPS détecte la modification comme une violation des conditions de la transition TI3. Dans le cas du nœud récepteur, SIDPS détecte la modification comme une violation des conditions des transitions TO2 et TO4 (si le nœud récepteur est la source), et TI4 et TI5 (si le nœud récepteur est un nœud intermédiaire).

5.3.1.2 Réponse de route

- **Modification du format** : si le format de la réponse de route n'est pas conforme à la spécification, la modification sera détectée comme une violation des conditions de la transition TD3 (par l'IDPS du nœud destination), et TI2 (par l'IDPS du nœud intermédiaire). Au niveau du nœud récepteur SIDPS détecte la manipulation des champs de réponse de route comme une violation des conditions des transitions TO5 et TI7.
- **Retransmission incorrecte** : le nœud malveillant peut changer l'adresse du nœud destination ou diminuer le numéro de séquence de la destination. Au niveau du nœud hôte SIDPS

détecte la modification comme une violation des conditions de la transition TI8 grâce à la règle suivante : $RREP.dst \neq rcv\_RREP.dst$ and $RREP.DSN \neq rcv\_RREP.DSN$. Au niveau du nœud récepteur SIDPS détecte la modification comme une violation des conditions des transitions TO5 et TI7.

5.3.1.3 Erreur de route

- **Modification du format** : Le changement ou l'ajout d'une destination injoignable permet au nœud malveillant de perturber plusieurs routes. Au niveau du nœud hôte SIDPS détecte la modification comme une violation des conditions des transitions TI10 et TI11, alors qu'au niveau du nœud récepteur SIDPS la détecte comme une violation des conditions des transitions TO6 et TI9.

5.3.2 Fabrication

- **Demande de route** : la transition TO1 représente la création de demande de route quand le nœud émetteur (source ou intermédiaire) a des données à transmettre mais qu'il ne possède pas de route vers la destination. Le nœud malveillant peut créer une demande de route même s'il possède une route vers la destination demandée ou sans même avoir des données à transmettre. Dans ce cas-là l'IDPS du nœud hôte détecte la fabrication comme une violation des conditions de la transition TO1, alors qu'au niveau du nœud récepteur la détecte comme une modification du nombre de saut (comme il a été expliqué dans la section précédente).
- **Réponse de route** : à la réception de la demande de route le nœud malveillant fabrique une réponse de route prétendant qu'il possède une route vers la destination demandée. La fabrication est détectée au niveau du nœud hôte comme une violation des conditions de la transition TI2. Dans le cas d'une fabrication active (non contrainte par la réception d'une demande de route), l'IDPS du nœud récepteur détecte la fabrication comme une transition inexistante. Un nœud ne peut pas faire partie de la route découverte sans participer dans le processus de diffusion de la demande de route, autrement dit un nœud ne peut pas recevoir de réponse de route sans envoyer préalablement la demande de route correspondante. Ce qui est bien représenté par notre modèle de spécification : la transition vers l'état RREP_rcv se fait à partir de de l'état RREQ_rcv qui est précédé par l'état RRQ_send (dans le cas du nœud source et intermédiaire), et ne peut en aucun cas être le premier état d'une session de découverte de route.
- **Erreur de route** : la fabrication d'erreur de route est détectée au niveau du nœud hôte comme une violation des conditions des transitions T10 et T11. Si la destination injoignable n'existe pas dans la table de routage du nœud récepteur, SIDPS détecte la fabrication comme

transition inexistante. Autrement SIDPS détecte la fabrication comme une violation des conditions des transitions TO6 et TI9.

5.3.3 Rejeu

- **Demande de route** : le rejeu d'anciennes demandes de routes est détecté au niveau du nœud récepteur comme une violation des conditions de la transition TO2 dans le cas du nœud source, et TI4, TI5 dans le cas du nœud intermédiaire, et TD2 (nœud destination). SIDPS détecte que l'adresse IP du nœud émetteur existe dans la liste des émetteurs *senders_list*, ce qui ne satisfait pas la condition : $ip\_src \notin Senders\_list$.
- **Réponse de route** : au niveau du nœud récepteur SIDPS détecte le rejeu de la réponse de route comme une transition inexistante, comme le montre les figure 5.4, 5.5 et 5.6 il n'y a pas de transition boucle dans l'état RREP_rcv.
- **Erreur de route** : L'IDPS du nœud récepteur détecte le rejeu come une transition inexistante, il n'y a pas de transition qui boucle sur l'état RERR_rcv.

5.3.4 Suppression

Le nœud malveillant peut supprimer les demandes et réponses de routes qu'il reçoit afin d'interrompre leurs transfert. Un nœud malveillant qui supprime toutes les demandes de routes qu'il reçoit n'aura aucun un effet négatif sur le routage, c'est comme s'il n'existait pas dans le réseau. Au niveau du nœud émetteur SIDPS détecte la suppression de demande de route comme une violation des conditions de la transition TI3. Au niveau du nœud récepteur SIDPS détecte la suppression de la réponse de route comme une violation des conditions de la transition TI8, et la suppression d'erreur de route comme une violation des conditions de la transition TI11.

5.3.5 Violation du timing

Le nœud malveillant ne respecte pas le temps d'attente entre les demandes de routes consécutives (section 5.2.1.2). Il peut également excéder le nombre maximal de tentatives de découvertes de routes spécifié par le protocole de routage. SIDPS détecte cette attaque comme une violation de la condition $(RREQs\_interval \geq Timeout)$ exigée par la transition TO3 (nœud hôte), et la transition TI6 (nœud récepteur).

5.4. Mécanisme de réponse

Le mécanisme de réponse que nous proposons permet de punir les violations de spécification commises par le nœud qui a généré le message plus que celles commises par le nœud qu'il a retransmit. Cela revient à isoler le générateur du message violant la spécification pour une période plus longue que le nœud qui le retransmet. Cette stratégie permet d'éviter l'isolement pour une longue période des nœuds honnêtes qui retransmettent les messages des nœuds malveillants à

défaut de connaissance locale. Le mécanisme de réponse que nous proposons prend en considération la récurrence des violations commises par le nœud malveillant. De telle sorte que les nœuds récidivistes seront punis plus que les autres. Nous employons le *Backoff* exponentiel binaire afin de considérer les intrusions répétés lors de l'estimation de la durée d'isolement. L'utilisation du *Backoff* permet d'isoler les nœuds récidivistes pour une période plus longue. La première fois qu'un nœud commet une violation de la spécification il sera isolé pour une durée de temps égale à l'intervalle *long_isolation* s'il s'agit du générateur du message ou à l'intervalle *short_isolation* s'il s'agit d'un nœud retransmeteur. La deuxième fois la durée d'isolement est le double de la durée d'isolement précédente, et ainsi de suite pour chaque violation supplémentaire le temps d'isolement est multiplié par deux. Pendant la durée d'isolement les messages de routage reçus de la part d'un nœud retransmetteur seront directement rejetés, alors que ceux reçus de la part d'un nœud générateur malveillant seront traités. Si le nœud malveillant commet une violation durant la période d'isolement, alors sa période d'isolement sera étendue. Dans notre expérimentation (section 5.3) nous fixons l'intervalle *short_isolation* à *NET_TRAVERSAL_TIME*, et l'intervalle *long_isolation* à *3* NET_traversal_time*. L'utilisation de différentes périodes d'isolement permet d'éviter les déconnexions réseau qui peuvent résulter suite aux longues périodes d'isolement des nœuds légitimes qui retransmettent les messages des nœuds malveillant à défaut de connaissance locale. Pendant l'isolement le nœud légitime continue à recevoir les messages de routage de la part du nœud malveillant, et ainsi il enrichit sa connaissance locale et pourra détecter par la suite les violations commises par ce dernier. Par conséquent il ne transmettra plus les messages du nœud malveillant.

5.5. Simulation et évaluation des résultats

Dans cette section, nous considérons les attaques de routage discutées dans la section 3 pour évaluer l'applicabilité et la performance du système de détection d'intrusion que nous proposons. Nous présentons plusieurs scénarios d'attaque ainsi que nous analysons et discutons les résultats obtenus. Nous supposons que SIDPS s'exécute dans un environnement protégé de telle sorte que son fonctionnement ne pourra pas être manipulé par le nœud hôte.

5.5.1 Connaissance locale

L'application du modèle de spécification que nous proposons nécessite la conservation d'informations de routage supplémentaires concernant l'historique des interactions. SIDPS sauvegarde ces informations que nous appelons connaissance locale au niveau de la table de routage et la table d'historique du nœud hôte. Nous ajoutons à la table d'historique les champs suivants (voir figure 5.7): la liste des expéditeurs de la demande de route, l'adresse de la destination, le numéro de séquence de la source et de la destination. La liste des expéditeurs contient les adresses IP des

nœuds qui ont envoyé la demande de route RREQ. L'adresse de destination, et les numéros de séquences de la source et de la destination sont nécessaires pour la comparaison lors de la réception de la même demande de route (adresse source, RREQ ID). Le numéro de séquence de la destination est utilisé uniquement si la demande de route a été reçue de la part du nœud source, afin d'éviter de prendre comme référence une information incorrecte ou manipulée. La figure 5.8(b) montre l'ensemble des champs d'une entrée dans la table de routage. Nous avons ajouté deux champs additionnelles **rd_list** et **RREQ_count**. Le champ **rd_list** sert à maintenir la liste des adresses IP cherchées par l'adresse de destination de l'entrée. Le champ **RREQ_count** permet d'accumuler le nombre de demandes de routes (non répétées) envoyées par un nœud voisin (plus de détails seront données dans la section 5.4). Nous présentons dans la figure 5.8(a) les informations qui doivent être sauvegardées sur la destination cherchée. Elle contient l'adresse IP de la destination cherchée **rd_add**. Le champ **rd_count** enregistre le nombre de demandes de routes émises par l'adresse destination de l'entrée de la table de routage vers l'adresse IP que contient le champ **rd_add**. L'indicateur (ou drapeau) **rd_flag** quand il est activé indique que le champ **rd_count** représente le nombre des demandes de routes RREQ reçues avec un TTL égale au diamètre du réseau. La date d'arrivé de la prochaine demande de route est enregistrée dans le champ **next_rd_time**. La date d'expiration d'une destination recherchée est enregistrée dans **rd_expire.** Sa valeur est calculée en prenant le maximum entre la valeur du temps nécessaire pour traverser le réseau (network traversal time) et la valeur du champ **next_rd_time**.

5.1.3. Environnement de simulation

La simulation est largement utilisée dans les recherches académiques. Elle permet de modéliser le réseau et ensuite de récupérer des données et des statistiques sur le réseau au cours de la simulation afin de mesurer les performances des protocoles. Cela est possible parce que les simulateurs intègrent un grands nombre d'outils permettant de réaliser des simulations assez réalistes. La communauté de recherche adopte plusieurs simulateurs pour étudier et simuler les protocoles dans les réseaux ad hoc tels que : ns-2 [136] (*Network Simulator 2*), OPNET [137], OMNET++ [138], GloMoSim [139]. Ns-2 est le simulateur le plus populaire dans le milieu académique et industriel. Il est considéré par beaucoup de spécialistes comme le meilleur logiciel de simulation, en raison de son modèle libre, permettant l'ajout très rapide de modèles correspondant à des technologies émergentes. Ns-2 est un outil logiciel de simulation par événements discrets. Il est principalement bâti avec les idées de la conception par objets, de réutilisabilité du code et de modularité. Il est devenu aujourd'hui un standard de référence en ce domaine grâce à sa flexibilité et sa modularité. Son utilisation est gratuite et son code source est disponible. Le logiciel est exécutable tant sous Unix que sous Windows. Ns-2 est basée sur le langage de script OTCL et le langage C++. Les scripts

utilisateurs permettant la configuration et la description des simulations sont écrits en en langage OTCL, alors que l'implémentation du noyau et des protocoles est écrite en C++. Le simulateur contient des bibliothèques pour la génération de topologies réseau, des trafics ainsi que des outils

Adresse de la source
Adresse de la destination
RREQ ID
Numéro de séquence de la source
Numéro de séquence de la destination
Liste des expéditeurs
Date d'expiration

Figure 5.7 Les champs de la table d'historique

rd_add
rd_count
rd_flag
next_rd_time
rd_expire

(a) Destination cherchée

Adresse de destination
Numéro de séquence de destination
Nombre de sauts
Liste des précurseurs
Date d'expiration
rd_list
RREQ_count

(b) Table de routage

Figure 5.8 Les champs de la table de routage

de visualisation tels que l'animateur réseau NAM (network animator). Le simulateur 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.

Nous évaluons notre proposition en utilisant ns-2 version 2.35. Nous supposons que la couche liaison de données et la couche physique sont fiables. Dans toutes nos expériences, nous considérons un réseau ad hoc composé de 50 nœuds mobiles placés aléatoirement sur un terrain dégagé de 1000 m × 1000 m. Les nœuds utilisent le protocole MAC IEEE 802.11 et une bande passante de 2 Mbps. Les nœuds se déplacent en utilisant le modèle de mobilité RWP (Random Wayoint). Chaque nœud se déplace avec une vitesse comprise entre 0 et 20 m/s à partir de sa position vers la destination choisie. Une fois arrivé à la destination, il y reste pendant 2 secondes avant de se déplacer vers la destination suivante. Chaque simulation dure pour une période de 300 secondes durant laquelle 50 connexions sont établis entre les nœuds. Les nœuds échangent des paquets de données de type CBR (Constant Bit Rate). Le nœud source envoie 5 paquets de données par seconde simulée, la taille du paquet est de de 512 octets. Le tableau 5.2 résume les paramètres utilisés. Pour chaque vitesse de mobilité, nous effectuons 10 simulations avec la même configuration mais avec des scénarios

aléatoirement générés (en utilisant le script *setdest* fourni avec ns-2) pour obtenir une valeur moyenne. Afin d'évaluer notre mécanisme et quantifier sa performance, nous avons utilisé les métriques suivantes : **(i)** le pourcentage des paquets de données transmit à travers les nœuds malveillants; **(ii)** le coût de routage, qui représente le nombre total des paquets de routage; **(iii)** le taux de livraison, qui représente le pourcentage des paquets transmis et bien reçus par le nœud de destination ; **(iv)** le taux de détection; qui représente le rapport entre le nombre d'intrusion détectées correctement et le nombre totale d'intrusions ; **(v)** le taux de faux positifs, représente le taux des nœuds détectés par erreur comme étant malveillants alors qu'ils ne le sont pas.

Paramètres de simulation	**Valeurs**
Modèle de propagation	Free Space
Type d'antenne	Omni-directional
Modèle de mobilité	Random waypoint (RWP)
Protocole MAC	802.11
Dimension du terrain	1000 m x 1000 m
Nombre de nœuds	50
Temps de simulation	300 s
Temps de pause	2 s
Type du trafic	CBR (Constant Bit Rate)
Volume du trafic	5 paquets/ seconde
Nombre de connexions	50

Tableau 5.2 Paramètres de simulation

5.1.4. Attaques et scénarios

Pour l'implémentation des attaques de violation de spécification, nous reprenons l'implémentation de Ning et Sun [131] disponible sur http://discovery.csc.ncsu.edu/software/MisuseAODV/. Les auteurs ont présenté une analyse systématique des attaques contre le protocole AODV, ils ont utilisé la définition d'attaques atomiques, qui est similaire à notre définition d'attaques élémentaire. Ils ont identifié quatre objectifs que le nœud malveillant tente d'atteindre, à savoir: perturbation de route RD (*Route disruption*) ; invasion de route RI (Route invasion) ; isolement du nœud NI (*Node isolation*) ; consommation des ressources RC (*Resources consumption*). Ils ont décrit comment le nœud malveillant atteint les objectifs cités précédemment en appliquant les attaques atomiques suivantes : suppression DR (*Drop*) ; modification et retransmission MF (*Modify and forward*) ; fabriquer et répondre FR (*Forge and reply*) ; Fabrication active AF (*Active forge*). Nous recommandons au lecteur de se référer à [131] pour plus de détails sur les attaques ainsi que les scénarios.

5.1.4.1. Invasion des routes

Pour cette attaque nous considérons les quatre scénarios suivants:

- ***Scénario 1*** *:* un nœud malveillant se trouvant à la portée du nœud victime, tente de s'insérer dans les routes entre ce dernier et les autres nœuds du réseau. Pour s'insérer dans la route le nœud malveillant incrémente la valeur du RREQ ID et du numéro de séquence des demandes de routes issues du nœud victime.
- ***Scénario 2 :*** un nœud malveillant fabrique une demande de route avec un nouveau RREQ ID et un numéro de séquence élevé, ensuite il la diffuse en prétendant qu'il la reçu de la part du nœud victime (en augmentant le nombre de sauts).
- ***Scénario 3 :*** une autre technique qui permet au nœud malveillant de s'insérer facilement dans la route est de fabriquer une réponse de route à chaque fois qu'il reçoit une demande de route de la part du nœud victime. Dans ce cas, il augmente le numéro de séquence et réduit le nombre de sauts pour annuler toute les mises à jour possibles qui peuvent résulter de la réception des réponses de route légitimes.
- ***Scénario 4 :*** le nœud malveillant peut aussi fabriquer une réponse de route sans même qu'il y ait une demande de route. La réponse de route fabriquée contient un numéro de séquence élevé et un nombre de sauts réduit, ce qui permet au nœud malveillant de mettre à jour une route existante.

La figure 5.9 montre un faible nombre de paquets de données transmis à travers le nœud malveillant ce qui est similaire à l'état normal du réseau (pas de nœuds malveillants). Ce qui prouve l'efficacité de SIDPS à détecter et à éviter les tentatives d'invasion de routes sous différents scénarios. Dans le premier scénario la tentative d'invasion de route est détectée comme une modification du format de la demande de route (transitions incorrectes : TO1, TI1, TI4, TI5, TI6, TD1, TD2). Dans le scénario 2, l'invasion de route est détectée comme une modification du nombre des sauts (transitions incorrectes : TI1 et TD1). En ce qui concerne le scénario 3 et 4 outre que l'attaque est détectée comme modification du nombre de sauts elle est aussi détectée comme fabrication de la réponse de route. Puisque que le nœud malveillant n'existe pas dans la liste des expéditeurs de la demande de route correspondante.

5.1.4.2. Privation du sommeil

Pour cette attaque nous considérons les deux scénarios suivants:

- ***Scénario 1*** : chaque fois que le nœud malveillant reçoit une demande de route, il incrémente son RREQ ID pour la rendre plus fraîche, et ainsi consommer l'énergie des autres nœuds dans une boucle de retransmission.

- ***Scénario 2*** : le nœud malveillant fabrique de manière répétitive des demandes de route avec des RREQ ID incrémentés tout en mettant le nombre de sauts au moins à 1 (prétendant retransmettre la demande de route).

Puisque dans les deux scénarios le nœud malveillant ne respecte pas le temps d'attente entre les demandes de routes successives, SIDPS détecte l'attaque dans les deux scénarios comme une violation du timing (transitions incorrectes TO3, TI6). Comme le montre la figure 5.10, dans le scénario 1 SIDPS maintient le nombre de paquet de contrôle presque similaire à celui du cas normal. Ce qui n'est pas le cas pour le scénario 2, cela s'explique par le fait que SIDPS ne détecte l'attaque qu'après la deuxième demande de route fabriquée. Malgré cela, SIDPS arrive à minimiser l'impact de l'attaque et à réduire le nombre de paquets de contrôle par 90 % (Figure 5.10).

5.1.4.3. Isolation des nœuds

Afin d'isoler le nœud victime, le nœud malveillant tente de l'empêcher de recevoir les paquets de données de la part des autres nœuds pendant une période plus ou moins courte. Pour cette attaque nous considérons les trois scénarios suivants:

- ***Scénario 1*** : nous supposons que le nœud malveillant est le seul voisin du nœud victime. Chaque fois que le nœud malveillant reçoit une demande de route de la part du nœud victime, il remplace l'adresse de destination avec une adresse non-existante, incrémente le RREQ ID et le numéro de séquence de la source, avant de retransmettre le message.
- ***Scénario 2*** : nous supposons que le nœud malveillant est le seul voisin du nœud victime. Il fabrique itérativement des demandes de routes avec l'adresse du nœud victime comme adresse source. Il met une adresse non-existante comme adresse de destination, augmente le RREQ ID et le numéro de séquence de la source, et met le nombre de sauts au moins à 1 (prétendant retransmettre la demande de route).
- ***Scénario 3*** : le nœud malveillant peut aussi partiellement isoler le nœud victime en répondant à toutes ses demandes de routes par des réponses de routes fabriquées, portants des numéros de séquences élevés et des nombres de sauts réduits. Une autre technique qui permet au nœud malveillant d'isoler le nœud victime complètement du réseau (pour une courte durée), consiste à devenir son prochain saut envers tous les autres nœuds du réseau. Dans ce cas le nœud malveillant envoie au nœud victime plusieurs réponses de route fabriquées avec différentes adresses de destination, et des numéros de séquences de destination élevés.

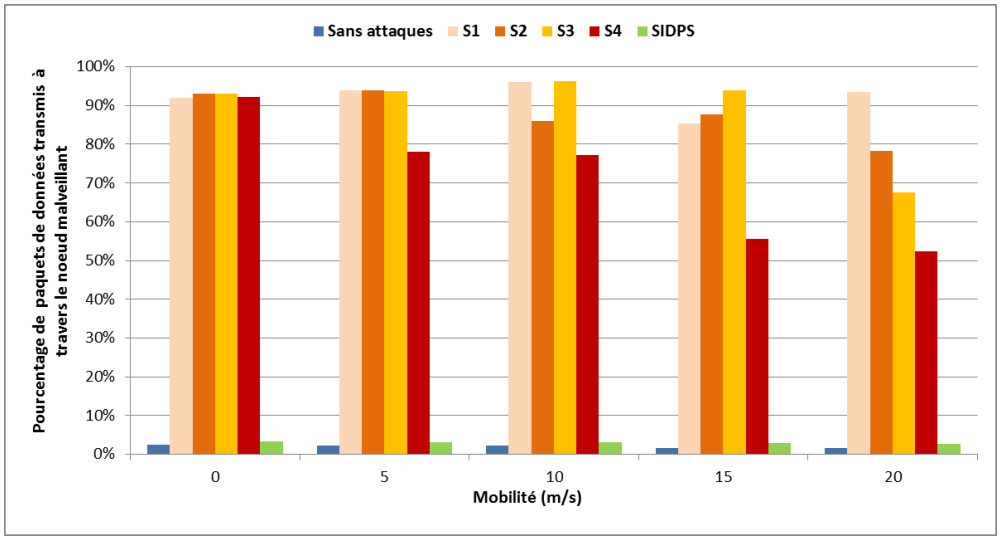


Figure 5.9 Invasion des routes

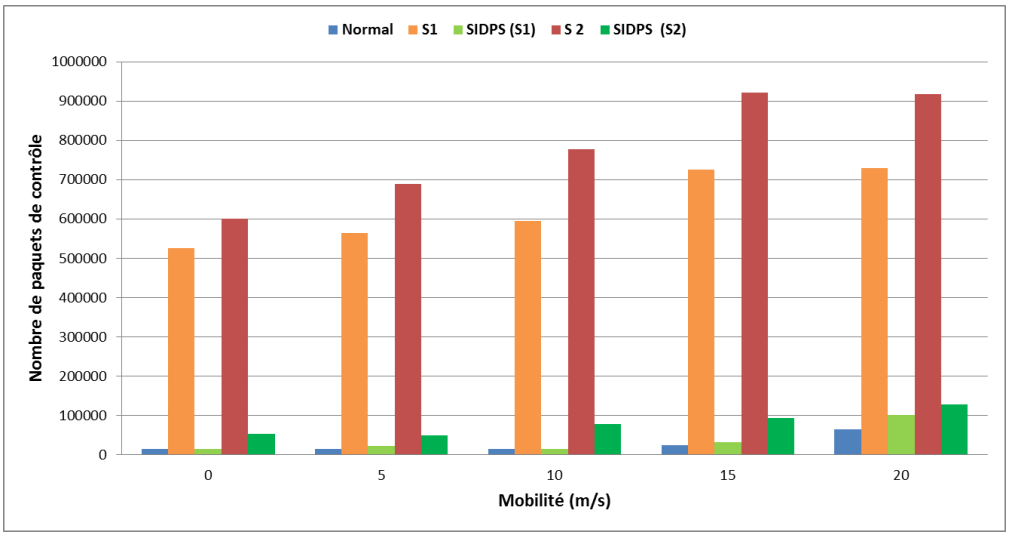


Figure 5.10 Privation du sommeil

Comme le montre la figure 5.11, SIDPS assure un taux de livraison de paquet similaire à celui obtenu lorsqu'il n y a pas d'attaque dans le réseau. SIDPS détecte la tentative d'isolation dans le premier scénario comme une modification du format de la demande de route. Dans les scénarios 2 et 3, l'attaque est détectée comme une modification du nombre des sauts.

5.1.4.4. Perturbation des routes

Le nœud malveillant peut empêcher l'établissement des routes en manipulant les informations de routage portées par les demandes et réponses de route reçues. Pour cette attaque nous considérons les trois scénarios suivants:

- ***Scénario 1*** : le nœud malveillant tente de casser une route existante en fabriquant une demande de route. Il prétend retransmettre une demande de route (en mettant le nombre de sauts au moins à 1) du nœud destination vers le nœud source, avec une adresse IP non existante au niveau de l'entête IP. Ce qui mène le nœud source à mettre à jour la route vers la destination en passant à travers le nœud inexistant.
- ***Scénario 2*** : chaque fois que le nœud malveillant reçoit une demande de route, il fabrique une réponse de route avec un numéro de séquence de destination élevé et un nombre de sauts réduit. Il envoie la réponse de route, en prétendant qu'il la reçu du nœud destination, et il met au niveau de l'entête IP une adresse IP source inexistante.
- ***Scénario 3*** : le nœud malveillant tente de casser une route existante en fabriquant une réponse de route avec un numéro de séquence de destination élevé et un nombre de sauts réduit. Il prétend retransmettre la réponse de route fabriquée du nœud destination vers le nœud source, en mettant au niveau de l'entête IP une adresse IP source inexistante.

La figure 5.12 montre que le taux de livraison des paquets (PDR) est similaire au taux de livraison obtenu lorsqu'il n'y pas de nœuds malveillants dans le réseau. Ce qui prouve l'efficacité de SIDPS à détecter et à prévenir les tentatives de perturbation de routes. Dans le premier scénario SIDPS détecte l'attaque comme une modification du nombre des sauts (transitions incorrectes : TI1 et TD1). Au niveau du nœud récepteur dont l'adresse IP est l'adresse source de la demande de route SIDPS détecte l'attaque comme une modification du format de la demande de route (transitions incorrectes : TO1, TI1, TI4, TI5, TI6, TD1, TD2) parce que le RREQ ID et le numéro de séquence reçus sont supérieurs à ceux du nœud hôte. Dans le scénario 2, SIDPS non seulement détecte l'attaque comme une manipulation du nombre de sauts (transitions incorrectes : TI1 et TD1), mais aussi comme une fabrication de la réponse de route, parce que le nœud malveillant n'existe pas dans la liste des expéditeurs de la demande de route correspondante. En ce qui concerne le scénario 3, l'attaque est détectée comme une fabrication de réponse de route parce que l'adresse de destination n'appartient pas à la liste des destinations cherchées.

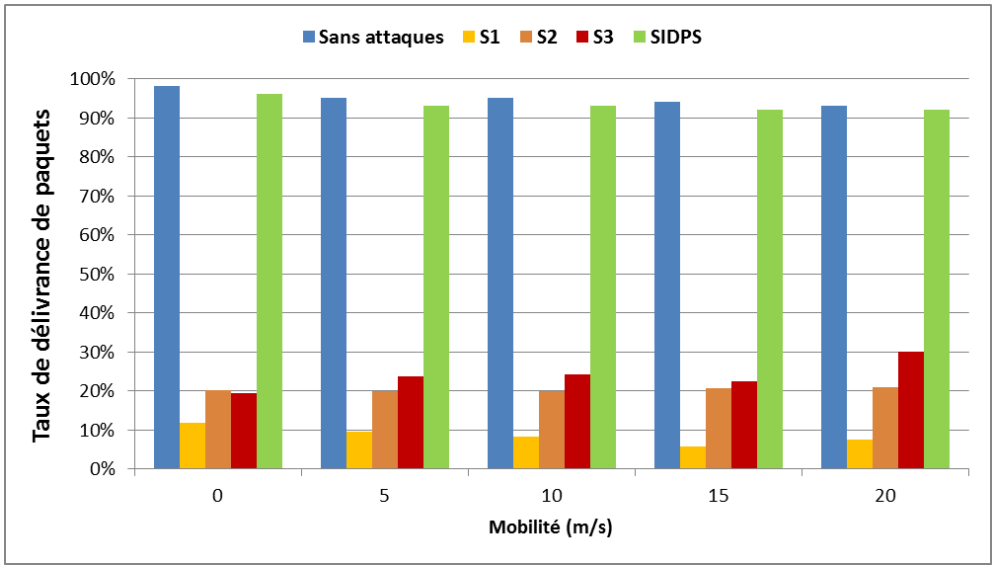


Figure 5.11 Isolation des nœuds

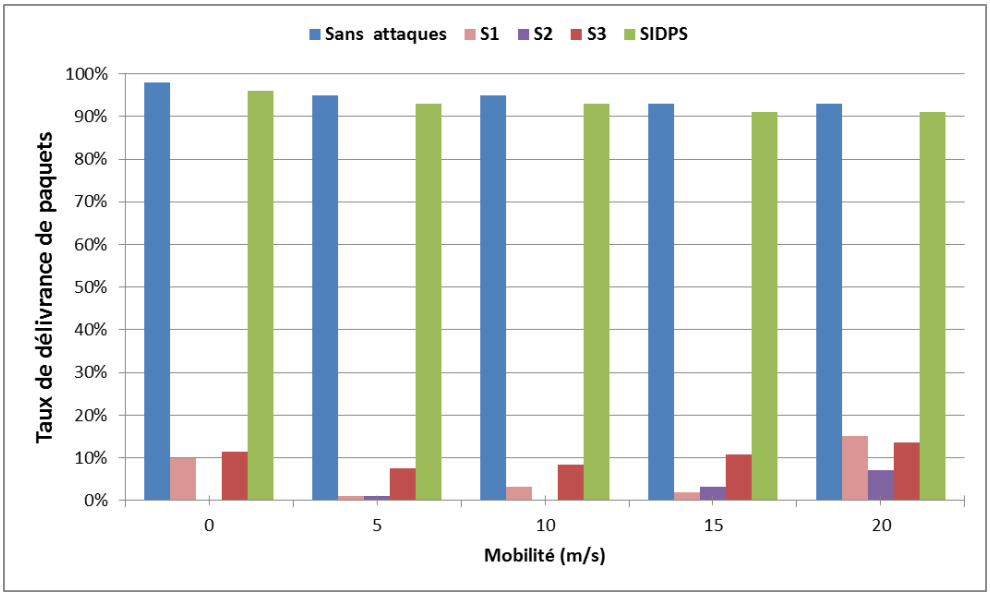


Figure 5.12 Perturbation des routes

5.1.4.5. Trou noir & trou gris

Dans cette attaque, d'abord le nœud malveillant s'insère dans la route en utilisant une des techniques d'invasion de routes décrites précédemment. Une fois sur la route, il supprime tous (ou sélectivement dans le cas du trou gris) les paquets de données qu'il reçoit. La figure 5.13 montre que SIDPS assure un taux de délivrance de paquets similaire au taux normal. SIDPS détecte les attaques du trou noir et du trou gris comme des tentatives d'invasion de routes. Ainsi SIDPS détecte et arrête l'attaque au stade de l'invasion de route, ce qui permet d'empêcher le nœud malveillant d'intercepter et supprimer les paquets de données.

5.1.4.6. Déni de service DDOS (plusieurs nœuds malveillants)

Pour cette attaque nous considérons un scénario, où il y a plusieurs nœuds malveillants, exécutants les attaques suivantes : privation de sommeil ; isolation des nœuds ; perturbation des routes ; trou noir & gris. Nous ne considérons pas l'attaque d'invasion de route parce qu'à elle seule ne constitue pas de risque sur la disponibilité du réseau et du service de routage. La figure 5.14 montre que SIDPS maintient le taux de délivrance similaire au taux normal, sauf dans le cas de privation de sommeil où le taux de délivrance baisse légèrement et progressivement avec l'augmentation des nœuds malveillants. Cette perte est due à l'étroite corrélation entre le nombre des nœuds malveillants et le nombre de paquets de contrôle générés. Plus le nombre des paquets qui circulent dans le réseau est important, plus le réseau est congestionné, et plus les files d'attente des nœuds seront débordées. Par conséquent un nombre considérable de paquets sera supprimé.

Attaques	**Scénarios**	**Type de violation**	**Transitions**
Invasion des routes	Scénario 1	Modification du format de la RREQ	TO1, TI1, TI4, TI5, TI6, TD1, TD2
	Scénario 2	Modification du nombre de sauts	TI1, TD1
	Scénario 3	Fabrication de RREP, modification du nombre de sauts	TI1, TD1
			TI1, TI2, TD1
	Scénario 4		TI1, TI2, TD1
Privation de sommeil	Scénario 1	Timing et taux d'émission	TO3, TI6
	Scénario 2		
Isolation des nœuds	Scénario 1	Modification du format de la RREQ	TO1, TI1, TI4, TI5, TI6, TD1, TD2
	Scénario 2	Modification du nombre de sauts	TI1, TD1
	Scénario 3		
Perturbation des routes	Scénario 1	Modification du nombre de sauts, modification du format de la RREQ	TO1, TI1, TI4, TI5, TI6, TD1, TD2
	Scénario 2	Modification du nombre de sauts, fabrication de RREP	TI1, TI2, TD1
	Scénario 3		TI1, TI2, TD1
Trou noir & trou gris		Invasion de route	TO1, TI1, TI2, TI4, TI5, TI6, TD1, TD2
DoS		Privation de sommeil, isolation des nœuds, perturbation des routes, trou noir	TO1, TO3, TI1, TI2, TI4, TI5, TI6, TD1, TD2

Tableau 5.3 Détection des attaques

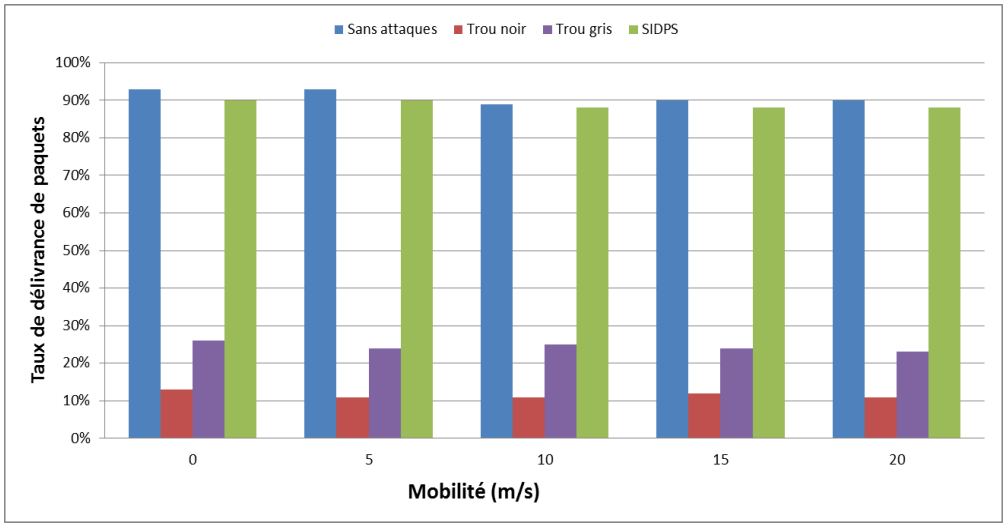


Figure 5.13 Trou noir & trou gris

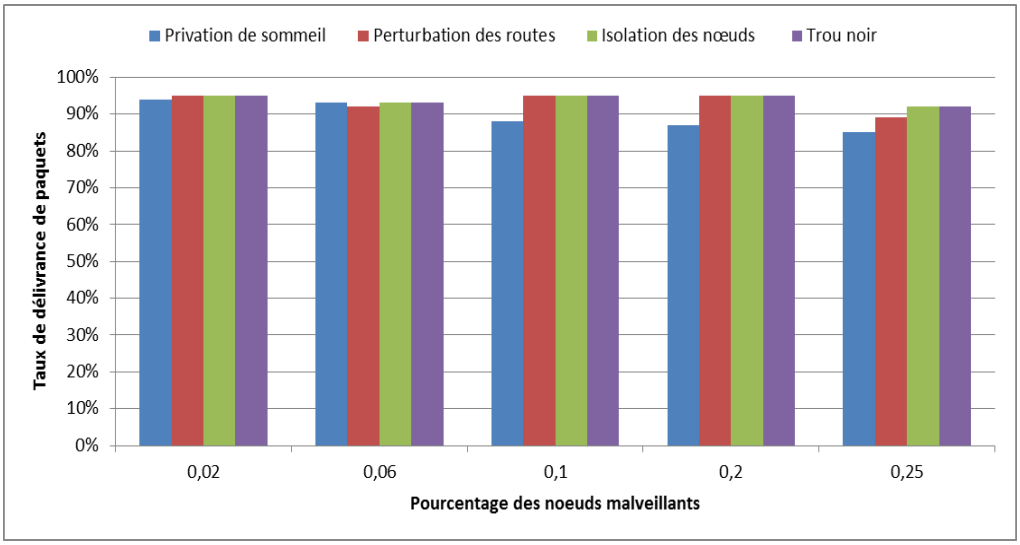


Figure 5.14 Déni de service distribué DDoS

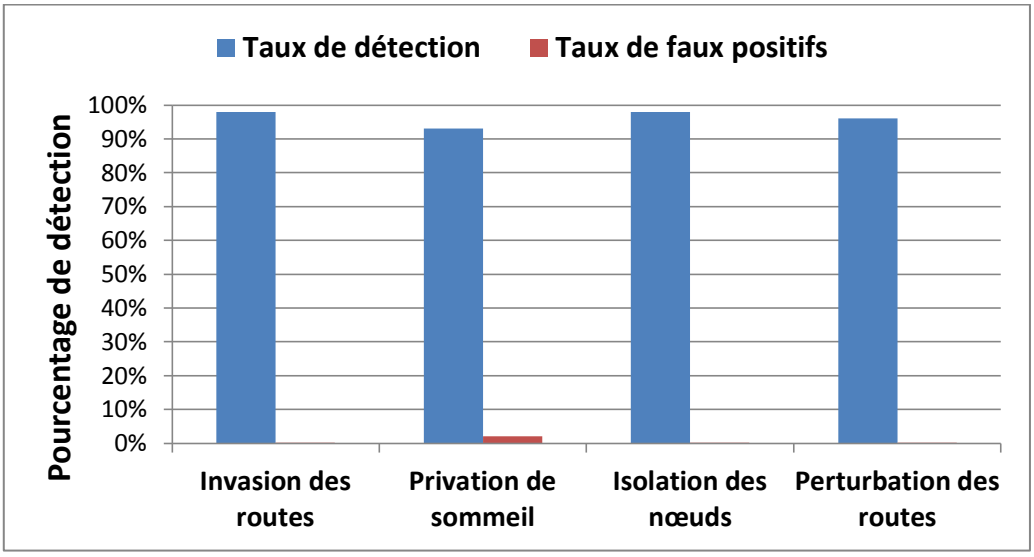


Figure 5.15 Taux de détection et faux positifs

5.1.5. Faux positifs

Nous observons quelques cas de faux positifs, à savoir des nœuds légitimes qui sont détectés a tort comme malveillants. Le taux de faux positifs correspond au nombre de nœuds qui sont détectés comme malveillants au moins une fois par le système de détection d'intrusion. Les résultats de la simulation montrent un faible taux de faux positif qui est égale à 2% (voir figure 5.15), dans le cas de privation de sommeil. Les cas des faux positifs concernent les violations du timing et la fréquence de génération sous forte mobilité, où les nœuds rejoignent et quittent le voisinage fréquemment. Un nœud honnête qui rejoigne le réseau ou le voisinage récemment, n'ayant pas assez de connaissance sur le nœud expéditeur, peut retransmettre les messages du nœud malveillant et par conséquent être détecté comme malveillant. Il y a aussi le cas où un nœud malveillant rejoigne un nouveau voisinage. Certains nœuds du voisinage n'ayants pas suffisamment d'information sur le nouveau nœud peuvent retransmettre à tort ses messages. En ce concerne les autres attaques le taux de faux positif est nul.

5.2. Conclusion

Dans de ce chapitre, nous avons proposé SIDPS un système de détection et prévention d'intrusion pour sécuriser le routage dans les réseaux ad hoc mobiles. Le système proposé utilise la technique de détection à base de spécification pour détecter et prévenir les attaques qui violent la spécification du protocole de routage. Nous avons proposé une nouvelle méthode d'extraction automatique de la spécification qui à partir de l'exécution du protocole de routage génère le modèle de spécification. Le comportement normal du protocole est modélisé par un ensemble d'automates à états finit qui définissent l'ordonnancement et les conditions d'exécution des opérations de routage. Non seulement le système proposé permet la détection et la prévention de la majorité des attaques de routage, mais il considère aussi la réponse aux nœuds malveillants. La réponse aux attaques se fait grâce à un système de réponse adaptatif qui permet de neutraliser les nœuds malveillants et les exclure du réseau. Les résultats de simulation ont montré que le système proposé arrive à détecter la majorité des attaques qui violent la spécification avec un taux de détection élevé et un très faible taux de faux positifs.

Chapitre 6 Un système hybride pour la détection et la prévention d'intrusion dans les réseaux ad hoc

Dans ce chapitre, nous proposons un nouveau système de détection et prévention d'intrusion (IDPS) basé sur l'anomalie pour détecter et prévenir les attaques à retransmission rapide telles que trou de ver et Rushing. Un type d'attaque que le nœud malveillant utilise pour être sélectionner sur les routes découvertes et intercepter le maximum de trafic. Le système que nous proposons emploie une approche statistique qui exploite le concept d'équilibrage de charge. Une approche qui permet d'identifier les nœuds malveillants comme des points de concentration du trafic dû à leurs taux de sélection élevé. Non seulement l'algorithme que nous proposons permet d'identifier et d'écarter les nœuds malveillants du réseau, mais aussi d'éliminer les points de concentration du trafic et assurer une répartition de charge. Sur une architecture hiérarchique nous combinons le système proposé avec le système de détection et prévention d'intrusion basé sur la spécification SIDPS proposé dans le chapitre précédent. Ensuite nous comparons le nouveau système de détection et prévention d'intrusion HSFA avec les mécanismes existants dans la littérature. La comparaison montre que HSFA surpasse les autres mécanismes en termes de détection et taux de faux positifs sans pour autant affecter les performances du réseau.

Ce chapitre est organisé comme suit : Dans la section 6.1 nous proposons un système de détection et prévention d'intrusion basé sur l'anomalie pour détecter et empêcher les attaques à retransmission rapide. La section 6.2 propose un système hybride pour la détection et prévention d'intrusion, et présente une comparaison du système proposé avec les mécanismes qui existent dans la littérature. Finalement, la section 6.3 conclut ce chapitre.

6.1. Détection et prévention des attaques à retransmission rapide

6.1.1 Détection et prévention au niveau hôte

Bien que le nœud malveillant n'opère pas de la même manière pour les attaques trou de ver et Rushing, son objectif est le même ; retransmettre rapidement la demande de route pour atteindre la destination avant les autres nœuds. Nous proposons une approche statistique pour détecter la retransmission rapide des messages de routage. L'idée est de permettre aux nœuds voisins du nœud malveillant de détecter que ce dernier possède une grande capacité à attirer les routes et qu'il constitue un point de concentration du trafic. Nous exploitons le concept d'équilibrage de charge qui

permet l'identification et l'élimination des points de concentration du trafic dans le réseau. L'équilibrage de charge (appelé dans certains ouvrages répartition de charge ou en anglais load balancing) consiste à répartir le trafic réseau sur différents nœuds. Il vise à garantir qu'aucun nœud dans le réseau n'est sous-chargé ou surchargé et à instaurer une charge uniforme pour tous les nœuds. Ce type de mécanisme s'appuie sur un élément, appelé répartiteur de charge (en anglais *load balancer*) dont la fonction est d'assurer la répartition de la charge de routage entre les nœuds afin d'éviter qu'un nœud soit inactif alors que du trafic reste en attente sur d'autres nœuds. Des solutions telles que les algorithmes de routage multi-chemin permettent de distribuer le trafic entre une source et une destination sur plusieurs chemins différents, en considérant le niveau de saturation des nœuds intermédiaires composants les différents chemins. Dans le cadre de cette thèse nous utilisons l'équilibrage de charge pour identifier et éviter les nœuds malveillants qui essaient toujours d'attirer le trafic vers eux. L'équilibrage de charge permet de choisir le meilleur chemin en évitant les nœuds saturés. L'idée est de permettre à tous les nœuds d'avoir les mêmes chances d'être sélectionnés lors du processus de découverte de route, évitant ainsi de passer par les nœuds malveillants. Puisque le nœud malveillant arrive toujours à retransmettre et à faire parvenir la demande de route avant les autres en utilisant un tunnel (trou de ver) ou en ignorant les délais d'attente (Rushing). Nous proposons un algorithme d'équilibrage de charge dynamique basé sur une approche statistique. L'algorithme permet aux nœuds voisins du nœud malveillant de détecter que ce dernier est toujours le premier à retransmettre les demandes de routes, et qu'il possède une grande capacité à attirer le trafic.

6.1.1.1 Découverte de route avec équilibrage de charge

Afin d'identifier les nœuds malveillants qui font de la retransmission rapide des demandes de routes, nous intégrons dans le protocole de routage AODV la fonctionnalité d'équilibrage de charge. Chaque nœud garde le nombre total de demandes de routes reçues (les demandes de route dupliquées ne sont pas comptées) dans un compteur "*RREQ_total*". Chaque nœud doit garder aussi pour chacun de ses voisins un compteur ''*RREQ_count*'', qui permet de compter le nombre de demandes de routes reçues pour la première fois de la part de chaque voisin. Nous ajoutons à l'entrée de la table de routage un nouveau champ ''*RREQ_count*''. Grace aux deux compteurs (*RREQ_total, RREQ_count*) chaque nœud peut calculer le taux de sélection de chacun de ses voisins. Le taux de sélection est calculé comme suit: *(RREQ_count) / (RREQ_total + 1)*. Le taux de sélection représente la probabilité que le nœud qui a retransmis la demande de route en premier soit sur la route découverte. Un taux de sélection élevé signifie que le nœud en question est probablement un nœud malveillant. Nous modifions la procédure de découverte de route de telle façon que l'équilibrage de charge sera considéré lors de la sélection des nœuds.

La nouvelle procédure de réception de la demande de route est décrite par l'algorithme AIDS dans la figure 6.1. Lorsqu'un nœud reçoit une nouvelle demande de route, il incrémente son compteur RREQ_total, et le compteur RREQ_count du nœud expéditeur. Ensuite il calcule le taux de sélection SR de l'expéditeur. Si le taux de sélection de l'expéditeur est inférieur au seuil autorisé, alors la demande de route est ajouté à la table d'historique, et la route inverse (vers le nœud source) est créé ou mise à jour. Si le nœud récepteur est la destination alors il envoie une réponse de route, sinon il retransmet la demande de route vers la destination. Dans le cas où le taux de sélection SR est égal ou supérieur au seuil, le nœud récepteur abandonne la demande de route. Le nœud récepteur n'ajoutera pas de route inverse jusqu'à qu'il reçoit la demande de route de la part d'un autre nœud dont le taux de sélection est inférieur au seuil. Il est possible d'exploiter la retransmission rapide des demandes de route pour atteindre la destination plus rapidement. Dans ce cas, le nœud récepteur retransmet la demande de route même si le taux de sélection du nœud expéditeur dépasse le seuil autorisé. Cependant le nœud récepteur n'ajoutera pas la route inverse jusqu'à qu'il reçoit la demande de route de la part d'un nœud dont le taux de sélection est inférieur au seuil.

6.1.1.2 Détection des nœuds malveillants

Puisque au départ les taux de sélection de tous les voisins sont à zéro, il n'est pas possible pour le nœud récepteur d'observer la capacité d'attraction de route du nœud malveillant. Par conséquent les nœuds malveillants sont inévitables pour une durée de temps qui peut être très courte si le trafic réseau est intense. Puisque le nœud malveillant arrive constamment à retransmettre les demandes de routes avant les autres en exécutant l'attaque du trou de ver ou du Rushing, ses nœuds voisins incrémentent entre temps son compteur RREQ_count. Quand le taux de sélection du nœud malveillant atteint le seuil, ses nœuds voisins rejetteront les demandes de routes reçues de sa part. Si par exemple un nœud X, dont le taux de sélection observé par son nœud voisin Y dépasse le seuil autorisé, alors le nœud Y refuse de retransmettre les demandes de routes reçues de la part du nœud X. Après un certain nombre de découvertes de routes, le taux de sélection du nœud X enregistré par le nœud Y diminuera progressivement. On explique cela par l'accroissement du compteur *RREQ_total*, puisque le taux de sélection est définit par *(RREQ_count)/(RREQ_total + 1)*, donc quand le dénominateur devient grand le taux de sélection diminue.

Nous reprenons les mêmes paramètres de simulation utilisés dans le chapitre 5 pour évaluer l'algorithme proposé AIDS (tableau 6.1). Le trou de ver est mis en œuvre par la création d'un tunnel entre deux nœuds malveillants, comme le montre la figure 6.2. Le premier nœud encapsule la demande de route reçue RREQ en WRREQ, dont le format est connu uniquement par les nœuds

malveillants. Ensuite il transmet la demande de route WRREQ au deuxième nœud malveillant sans incrémenter le nombre de sauts. Dans le cas de l'attaque Rushing, le nœud malveillant ignore les temps d'attentes imposés par le mécanisme du Backoff (standard 802.11) pour transmettre sa demande de route rapidement et ainsi avoir un avantage temporel par rapport aux demandes de route légitimes.

Algorithm 1: Decouverte de route avec equilibrage de charge

```
begin
    if (orig, BID) est nouveau then
        RREQ_total ++
        RTE_ip_src.RREQ_count + +
        if SR(ip_src) < Seuil then
            Ajouter RREQ a la table d'historique
            Creer ou mettre a jour RTE_orig
            if my_addr ≠ RREQ.dst then
                RREQ.HC ++
                Diffuser RREQ
            else
                Envoyer RREP
            end
        else
            Supprimer RREQ de la table d'historique // pour traiter RREQ
            Supprimer RREQ
        end
    else
        Supprimer RREQ
    end
end
```

Figure 6.1 Découverte de route avec équilibrage de charge

Paramètres de simulation	Valeurs
Modèle de propagation	Free Space
Type d'antenne	Omni-directional
Modèle de mobilité	Random waypoint (RWP)
Protocole MAC	802.11
Dimension du terrain	1000 m x 1000 m
Nombre de nœuds	50
Temps de simulation	300 s
Temps de pause	2 s
Type du trafic	CBR (Constant Bit Rate)
Volume du trafic	5 paquets/ seconde
Nombre de connexions	50

Tableau 6.1 Paramètres de simulation

Nous utilisons deux métriques pour évaluer l'algorithme proposé, le taux de paquets de données transférés à travers le nœud malveillant et le taux de délivrance. Un taux de paquets élevé signifie que le nœud est sur plusieurs routes et ainsi participe dans l'acheminement d'important volume de

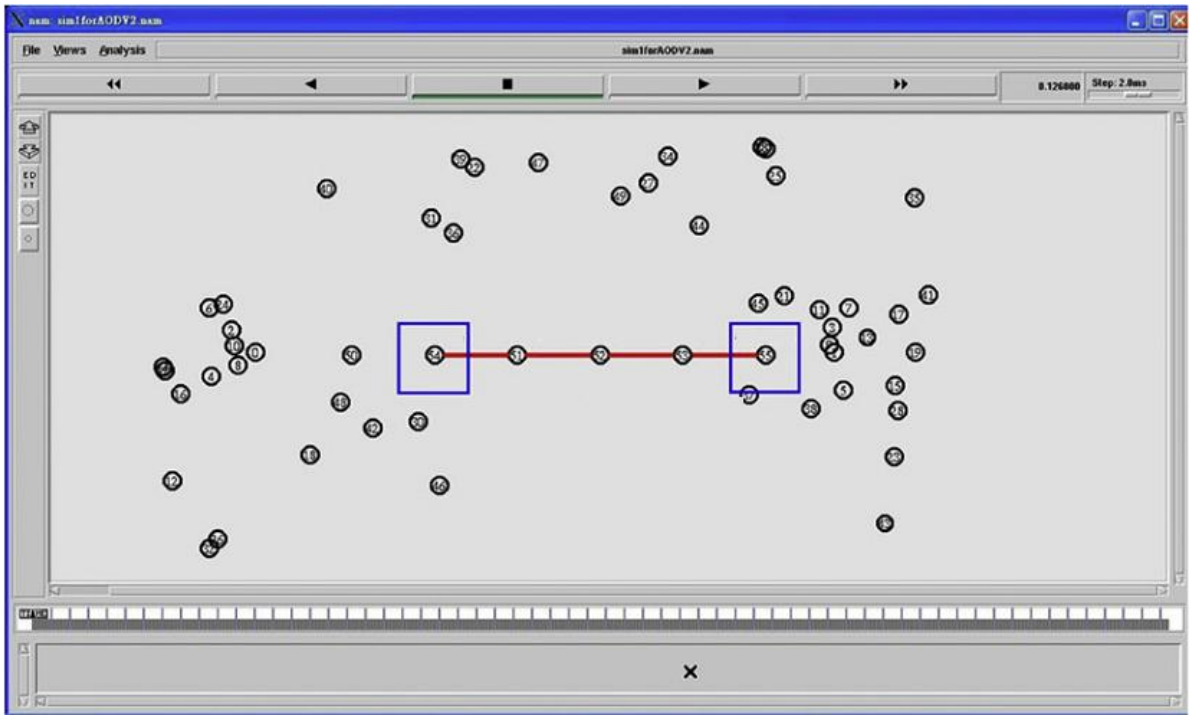


Figure 6.2 Topologie initiale du réseau

données. Le taux de délivrance permet d'évaluer l'impact des éventuelles déconnexions causées par l'écartement des nœuds légitimes placés dans des positions stratégiques de connectivité. Afin d'examiner si le seuil de retransmission affecterait la performance du nouvel algorithme de découverte de route, nous utilisons différents seuils de sélection. Le premier seuil (0,51) signifie que le nœud voisin ne peut être sélectionné consécutivement dans une route qu'une seule fois, le second (0,67) pas plus de deux fois, le troisième (0,76) pas plus de trois fois.

Une fois le seuil de retransmission est atteint, la demande de route reçue de la part du nœud en question sera rejetée. La figure 6.3, montre que le nouvel algorithme réduit le nombre de paquets de données acheminés à travers les nœuds malveillants avec plus de 83% en utilisant le seuil 0,51, et avec environ 72 % en utilisant le seuil 0.67. Cependant le seuil 0,51 baisse le taux de délivrance (PDR) à presque 70 %, ce qui est inférieur au taux de délivrance obtenu en utilisant le seuil 0,67 qui est égale à 84% (voir figure 6.4). Donc 0,67 est le seuil optimal qui offre le meilleur taux de délivrance tout en évitant la sélection des nœuds malveillants.

Bien que l'algorithme de découverte de route proposé arrive à détecter et à éviter les nœuds malveillants du réseau, il induit une baisse du taux délivrance des paquets. La baisse du taux de délivrance est due aux déconnexions réseau causées par l'écartement des nœuds légitimes qui sont placés dans des positions stratégiques du réseau. Afin de réduire le nombre de déconnexions réseau

et ainsi maximiser le taux de délivrance, nous proposons (dans la section suivante) à partir de l'algorithme proposé un système de détection et prévention d'intrusion qui prend en considération la mobilité des nœuds et la densité du réseau.

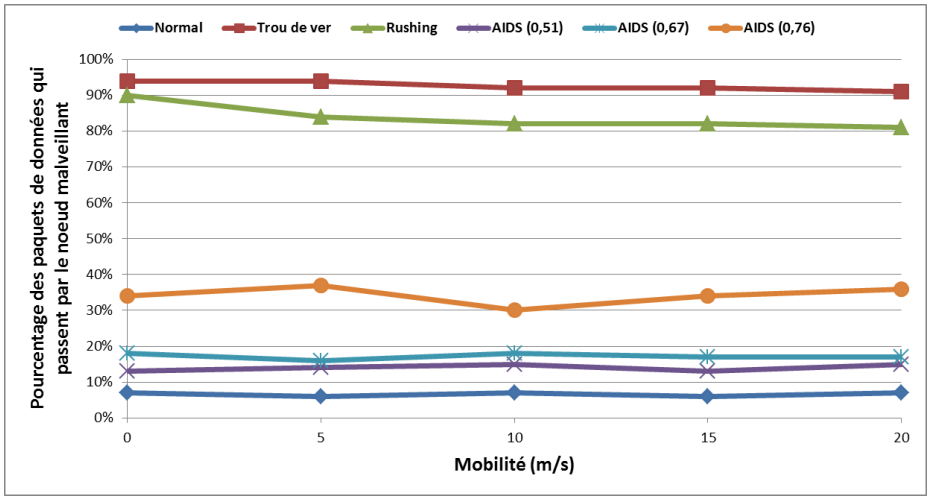


Figure 6.3 Détection du trou de ver & Rushing par AIDS

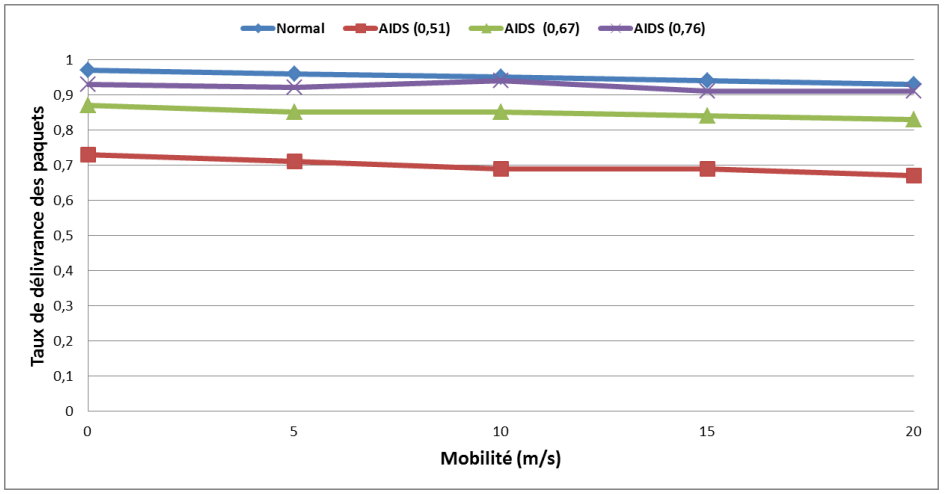


Figure 6.4 Taux de délivrance du AIDS sous différents seuils

6.1.2 Détection et prévention d'intrusion au niveau Cluster

Puisque la topologie du réseau est dynamique et les nœuds sont mobiles, un nœud ne restera pas pour longtemps dans la même position. Donc en estimant la mobilité du réseau on peut déterminer si le taux de sélection élevé est dû à la position stratégique du nœud et à la faible mobilité, ou qu'il s'agit de comportement malveillant. Dans un réseau à forte densité, plusieurs chemins peuvent exister entre deux nœuds, par conséquent les taux de sélection des nœuds ne seront pas disparates. Le système de détection et prévention d'intrusion que nous proposons est supposé s'exécuter dans

une topologie réseau hiérarchique organisée en clusters. Nous choisissons les nœuds les plus capables en termes de capacités de traitement en tant que chefs de clusters (CHs), les autres nœuds sont des membres du cluster (CNs). Nous supposons que la communication entre les CHs et CNs est sécurisée. Le chef du cluster (cluster head) collecte les données à partir des membres (CNs) sous forme de deux matrices : la matrice des taux de sélection SRM et la matrice des conditions réseau NCM. Chaque membre du cluster envoie les taux de sélection de ses nœuds voisins au chef du cluster sous forme de matrice SRM:

$$SRM = \{(n_1, SR_1), (n_2, SR_2), .., (n_i, SR_i)\} \quad (1)$$

Nous considérons avec le taux de sélection la mobilité et la densité, qui sont négativement corrélées avec le taux de sélection. Sous forte mobilité le nœud se déplace fréquemment et change de voisinage. Par conséquent le taux de sélection observé par ses nœuds voisins change (augmente ou diminue). Pareil pour la densité, dans un réseau à forte densité de nœuds, il existe plusieurs routes entre chaque paire de nœuds. Par conséquent les nœuds ont approximativement la même probabilité d'être sélectionnés pendant la découverte de route, ainsi les valeurs des taux de sélection ne seront pas disparates. La mobilité du réseau peut être estimée à base de plusieurs paramètres tels que : le nombre et la fréquence des messages RREQ, RREP et RERR. La mobilité peut être aussi estimée à partir du nombre des mises à jour de la table de routage. La densité globale du cluster est estimée à partir de la densité locale estimée par chaque nœud. La densité locale est estimée en se basant sur le nombre des voisins. La matrice $NCM\ (2)$ est utilisée pour envoyer au chef du cluster les paramètres liés à la mobilité et la densité locales observées par chaque nœud. Elle contient les statistiques sur le protocole de routage, nous considérons les paramètres suivants :

$$NCM = \{Nb\_RREQ, Nb\_RREP, Nb\_RERR, Nb\_neighbors\ , Nb\_RT\_updates\} \quad (2)$$

Le chef du cluster collecte périodiquement les matrices NCM et SRM à partir de chaque nœud du cluster pour calculer la distribution de la charge du trafic sur les nœuds. A partir des taux de sélection observés et collectés, l'IDPS (au niveau du chef de cluster) calcule pour chaque membre du cluster le taux de sélection moyen $\bar{SR}$ (équation 3) à partir des taux de sélection observés SR_ob_i. Les taux de sélection moyens sont sauvegardés dans l'ensemble ASR. Ensuite l'IDPS calcule la distance entre le taux de sélection moyen de chaque nœud et l'ensemble ASR (équation 4) pour repérer les taux de sélection aberrants. Pour chaque nœud du cluster l'IDPS calcule la déviation standard SD entre les taux de sélection observés par les voisins de ce dernier. La déviation standard est utilisée pour mesurer et quantifier la variation ou la dispersion entre les valeurs des taux de sélection observés SR_ob_i (équation 5). Puisque le nœud malveillant propage des informations de routage attrayantes

(des routes plus courtes et fraiches) au niveau de son voisinage, tous ses voisins lui assignent un taux de sélection élevé. Par conséquent la déviation standard entre les taux de sélection observés sera très petite.

$$\bar{SR} = \frac{\sum_{i=1}^{n} SR\_ob_i}{n} \qquad (3)$$

$$d(\ \bar{SR}, ASR) = \inf\{d(\ \bar{SR}, y) |\ y\ \in ASR\} \qquad (4)$$

$$SD = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(SR_i - \ \bar{SR})^2} \qquad (5)$$

Le taux de sélection final est calculé comme suit :

$$SR = \frac{\bar{SR} * d}{SD} \qquad (6)$$

A partir des nombres des RREQ, RREP, RERR et les mises à jour de la table de routage, nous estimons la mobilité comme forte ou faible. Pareil pour la densité, à partir du nombre des voisins de chaque nœud nous estimons si la densité est forte ou faible. La proposition d'algorithme d'estimation de mobilité et densité n'est pas l'objectif de cette thèse, nous utilisons l'algorithme proposé dans [143]. Nous définissons trois seuils : SR_T pour le taux de sélection, M_{high} pour la mobilité, et D_{high} pour la densité. L'algorithme de détection et prévention d'intrusion au niveau cluster C-AIDS est décrit dans la figure 6.5. L'algorithme C-AIDS vérifie pour chaque nœud n_i du cluster si le taux de sélection a atteint le seuil SR_T pendant le dernier intervalle de temps T_i. Si c'est le cas L'IDPS examine la mobilité et la densité, si l'un d'eux a atteint le seuil alors le nœud est ajouté à la liste des nœuds malveillants *malicious_list*. Sinon le nœud sera ajouté à la liste des nœuds suspects *suspect_list*. Si le nœud existe déjà dans la liste des nœuds suspects *suspect_list,* alors il sera déplacé à la liste des nœuds malveillants. Ensuite l'IDPS génère et diffuse le message d'alerte AP pour informer les nœuds du cluster de la présence de nœud malveillant. Quand un nœud reçoit le message d'alerte il vérifie s'il l'a déjà reçu, si c'est ce n'est pas le cas alors il met le(s) nœud(s) en question dans sa liste noir et diffuse le paquet. Les paquets reçus de la part des nœuds appartenant à la liste noire sont ignorés.

***Algorithm*: Détection et prévention au niveau cluster (C-AIDS)**

Do *after each TI*

. Collect SRM and NCM from CNs

. Estimate Mobility and density

for $\forall i$

. Calculate for each CN the average selection rate $\bar{SR}$ *from the observed SR_ob* (*eq* 3)

. Calculate for each CN the distance $d(\bar{SR}, ASR)$ (*eq* 4)

. Calculate for each CN the standard deviation SD between SR_ob (*eq* 5)

. Calculate for each CN its final selection rate value SR (*eq* 6)

End do

For $\forall i$ *perform comparison*

if $(SR > SR_T)$ ***then***

if $(Mobility \geq M_{high})$ *OR* $(Density \geq D_{high})$ ***then***

Add $\boldsymbol{n_i}$ *to Malicious_list*

Else

if $n_i \in$ *Suspect_list* ***then***

Add $\boldsymbol{n_i}$ *to Malicious_list*

Else

Add $\boldsymbol{n_i}$ *to Suspect_list*

End for

Figure 6.5 Pseudocode de l'algorithme de détection et prévention d'intrusion

Nous reprenons les trois seuils utilisés précédemment (pour l'algorithme AIDS) comme seuils moyens $\bar{SR}$ pour calculer les nouveaux seuils (pour l'algorithme C-AIDS), à savoir : (0,51) qui signifie que le nœud voisin ne peut être sélectionné consécutivement dans une route qu'une seule fois, le second (0,67) pas plus de deux fois, le troisième (0,76) pas plus de trois fois. Nous calculons les nouveaux seuils de sélection en utilisant l'équation 6, nous fixons la distance *d* à 1/2 du seuil moyen, et la déviation standard *SD* à 1/3 du seuil moyen. Ainsi nous obtenons les trois seuils suivants : 1.005, 1.14 et 1.215. La figure 6.6 montre que le seuil 1,005 réduit presque à 84 % le nombre de paquets de données acheminés à travers les nœuds malveillants ce qui est similaire au pourcentage obtenu par AIDS en utilisant le seuil 0,51. Le seuil 1,14 réduit à 74 % le nombre de paquets de données transmis à travers le nœud malveillant ce qui est similaire au pourcentage obtenu par AIDS en utilisant le seuil 0,67. D'après les résultats obtenus, nous constatons que C-AIDS et AIDS fournissent la même performance en termes de détection, cependant C-AIDS donne un taux de délivrance meilleur comparé à AIDS (figure 6.7). Le taux de délivrance est passé de 70% à 83% pour le seuil moyen 0,51

et de 84% à 95% pour le seuil moyen 0,67. L'introduction de la mobilité et la densité dans l'algorithme de détection et prévention à permet de minimiser le nombre de déconnexions réseau tout en garantissant un taux de détection élevé. Nous estimons la charge de routage en calculant le rapport entre le nombre de paquets de contrôle et le nombre de paquets de données délivrées à leurs destinations avec succès. La figure 6.8 montre que C-AIDS introduit un faible surcoût de routage supplémentaire par rapport à AODV et AIDS. Le nombre de paquets de contrôle supplémentaire sont générées suite à la collecte des données (SRM et NCM) et la diffusion des paquets d'alertes AP.

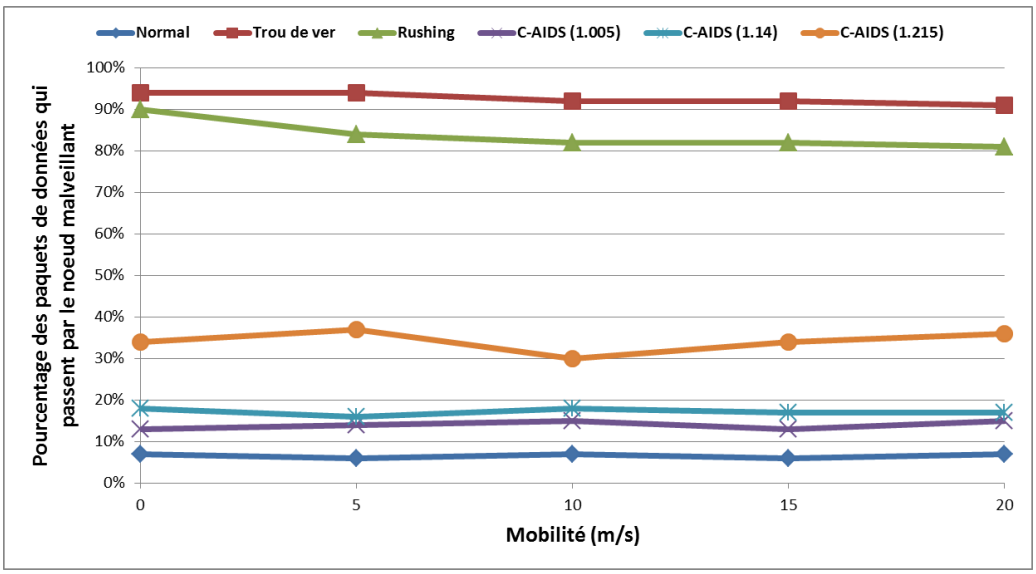


Figure 6.6 Détection du trou de ver & Rushing par C-AIDS

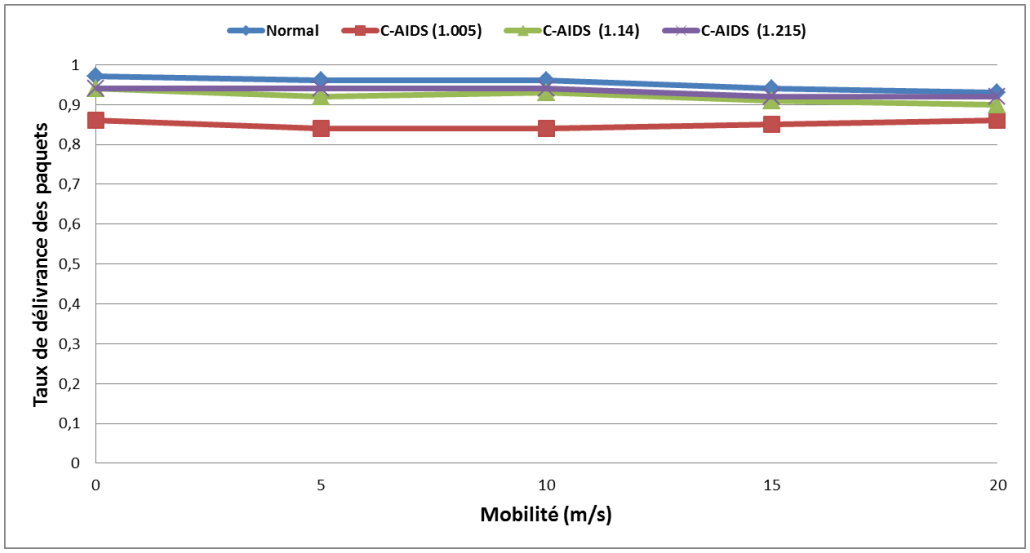


Figure 6.7 Taux de délivrance de C-AIDS sous différents seuils

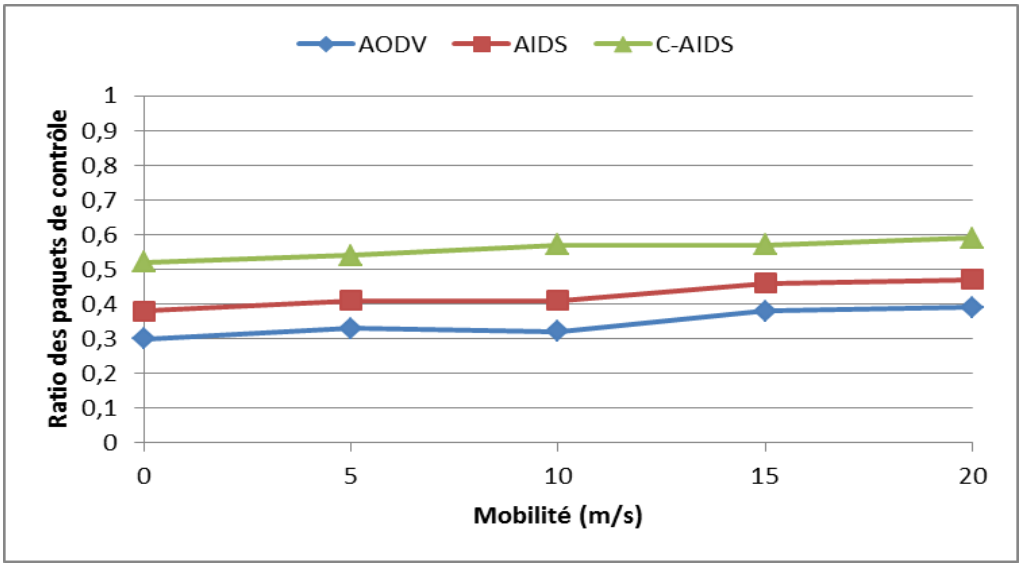


Figure 6.8 Charge de routage

6.2. Un système hybride pour la détection et prévention d'intrusion

La plupart des systèmes de détection et prévention d'intrusion proposés dans la littérature protègent le routage contre un type particulier d'attaques de routage et par conséquent n'assurent pas une protection complète. La détection basée sur l'anomalie est la technique la plus utilisée, puisque elle permet la détection des attaques inconnues. Cependant, elle est inefficace contre les attaques qui violent la spécification et génère un taux élevé de faux positifs si elle n'est pas bien paramétrée. La détection basée sur la spécification génère un faible taux de faux positifs mais ne détecte que les attaques qui violent la spécification. Nous pensons que la combinaison des deux techniques permet de fournir un mécanisme de sécurité robuste et capable à détecter plusieurs types d'attaques avec un faible taux de faux positifs.

Nous combinons le système de détection d'intrusion basé sur l'anomalie C-AIDS (présenté dans la section 6.1) avec le système de détection d'intrusion basé sur la spécification SIDPS proposé dans le chapitre 5. Le nouveau système de détection et prévention d'intrusion que nous appelons HSFA (Hybrid Security Framework for Ad hoc networks) s'exécute sur une topologie réseau hiérarchique organisée en clusters. Nous choisissons les nœuds les plus capables en termes de capacités de traitement en tant que chefs de clusters (CHs), les autres nœuds sont des membres du cluster (CNs). Nous supposons que la communication entre les CHs et CNs est sécurisée. Les chefs des clusters exécutent le système de détection et prévention basé sur l'anomalie C-AIDS. Les membres du cluster exécutent le système de détection et prévention basé sur la spécification SIDPS (voir figure 6.9).

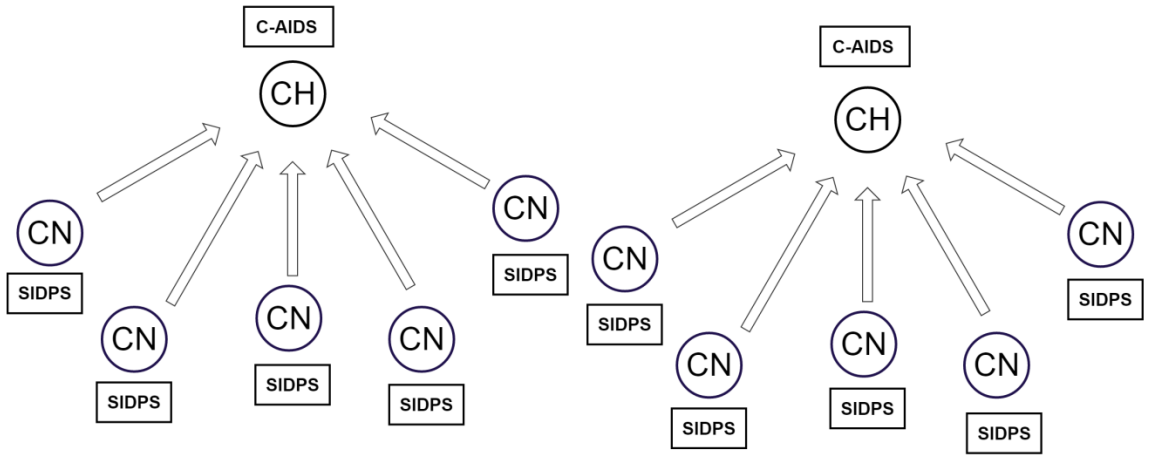


Figure 6.9 Architecture HSFA

SIDPS contrôle les interactions du nœud avec les autres nœuds du réseau et détecte les éventuelles violations de spécification. C-AIDS surveille et analyse le processus de découverte de route pour détecter et empêcher les attaques à retransmission rapide. Le système à réponse adaptative (décrit dans le chapitre 5) est déclenché à chaque fois qu'une intrusion est détectée. Sa fonction est de prendre l'action défensive adéquate contre le nœud malveillant.

6.2.1 Comparaison entre HSFA et les recherches connexes

Puisque HSFA est un mécanisme de sécurité pour la détection et la prévention d'intrusion, nous le comparons uniquement aux mécanismes appartenant à cette catégorie. Nous conduisons une comparaison en se basant sur les critères suivants :

- **Attaques détectées**

Contrairement aux mécanismes proposés dans [126, 129] qui se limitent à la détection de quelques attaques particulières, HSFA peut détecter un grand nombre d'attaques. Non seulement HSFA est capable de détecter les attaques qui violent la spécification a l'instar des mécanismes proposés dans [89, 95, 97], mais aussi les attaques qui ne violent pas la spécification telles que les attaques à retransmission rapide à l'instar des mécanismes proposés dans [127,113].

- **Spécification du protocole**

HSFA présente plusieurs améliorations par rapport aux autres mécanismes de sécurité qui utilise la technique de détection d'intrusion à base de spécification. Un de ses principaux points forts, est sa

capacité à détecter la majorité des attaques de routage, cela est dû au fait que : (i) il est basé sur un modèle de spécification complet, et non pas sur certains aspects de la spécification tel qu'il est le cas dans [89, 90, 91, 95]; (ii) il surveille et contrôle l'échange de tous les messages de routage. Contrairement aux systèmes de détection d'intrusion à base de spécification proposés dans [84, 95, 97] qui surveillent et contrôlent uniquement les actions du nœud hôte (le nœud sur lequel s'exécute l'IDS), HSFA contrôle les actions du nœud hôtes et ses interactions avec les autres nœuds du réseau. HSFA n'est pas basé sur une solution matérielle telle que les blocs mémoires partagés comme il en est le cas dans [84], ou les plates-formes de confiance (TrustZone SoC) comme dans [97]. Contrairement au mécanisme proposé dans [95] qui fournit une protection intermittente des nœuds visités en utilisant des agents mobiles, HSFA assure une protection continue de tous les nœuds du réseau. Laisser les nœuds sans protection pendant l'immigration des agents mobiles, les rendent des cibles faciles pour les nœuds malveillants. Contrairement au système de détection d'intrusion proposé par Tseng et al [89], HSFA n'utilise pas l'écoute en mode promiscuité pour détecter les intrusions. Non seulement le mode promiscuité consomme les ressources du nœud, mais il est aussi sujet aux erreurs qui peuvent résulter suites aux collisions.

- **Surcoût du routage**

HSFA ne génère qu'un faible taux de paquets de contrôle (voir figure 6.10) ce qui n'est pas le cas dans [89, 95, 113, 127, 129], ni un taux élevé de faux positifs tel qu'il est le cas dans [111, 113, 127].

- **Réponse aux intrusions**

A l'instar des mécanismes proposés dans [95, 127], HSFA ne se limite pas à détecter et éliminer les attaques, mais il riposte aussi aux intrusions grâce à un système de réponse actif et adaptatif qui permet d'isoler le nœud malveillant. Ce qui n'est pas le cas pour la majorité des mécanismes proposés dans la littérature, à l'exemple des mécanismes proposés dans [126, 111, 97]. Ces derniers se limitent à la détection et l'élimination sans considérer la réponse aux intrusions. Certains travaux tels que [89, 113, 129] considèrent la réponse aux intrusions, mais ils proposent des systèmes à réponse passive qui se limitent à signaler l'intrusion en envoyant des alarmes.

- **Taux de détection**

Afin de quantifier la comparaison nous avons choisi parmi les mécanismes de sécurité discutés précédemment, SIDE [97] et IDAR [127]. Les autres propositions soit ne fournissent pas des résultats expérimentaux comme [89, 95], où traitent un type particulier d'attaques tel qu'il est le cas dans [129], où utilisent des paramètres et des métriques d'évaluation différents, tels que [113, 111, 127].

Mécanismes	Architecture	Technique de détection	Attaques considérées	Protocoles	Mécanisme de réponse	Contributions	Limites
Tseng et al. [89]	Distribuée & coopérative	A base de spécification	Suppression, modification, fabrication	AODV	Alarmes	Premier IDS à base de spécification	Surcoût de routage ; consommation des ressources
Panos et al. [95]	Agents mobiles	A base de spécification	Empoisonnement de table de routage, trou noir, DoS	AODV	Alarmes, écarter le nœud malveillant	Détection d'intrusion multicouche à base de spécification	Protection intermittente ; pas de validation expérimentale
Panos et al. [97]	Autonome	A base de spécification	Modification, fabrication, privation de sommeil, trou noir & gris	AODV	Non-considéré	Spécification complète ; détection temps réel	Support matériel ; consommation des ressources
Shakshuki et al. [129]	Autonome	A base d'acquittement	Suppression, fausse accusation, transmission à puissance limitée	DSR	Alarmes	Résout les faiblesses du Watchdog	Surcoût de routage ; consommation des ressources
Nadeem et al. [127]	Hiérarchique organisée en clusters	Hybride (à base d'anomalie & signature)	Privation de sommeil, Trou noir& gris, Rushing	AODV	Isoler le nœud malveillant	IDS général ; taux de détection élevé; isolation du nœud malveillant	Surcoût de routage ; consommation des ressources ; fausses accusations
Alattar et al. [126]	Distribuée et coopérative	A base de signature	Suppression, Modification, fabrication	OLSR	Non-considéré	Adaptatif ; investigation coopérative	fausses accusations, consommation des ressources ;
Jabbehdari et al. [111]	Non spécifié	A base d'anomalie	DoS	DSR	Non-considéré	Détection à base de réseaux de neurones ; identification du nœud malveillant	Définition manuelle des classes d'apprentissage ; mises à jour continues
Barani et al [113]	Autonome	A base d'anomalie	Privation de sommeil, Trou noir& gris, Rushing, Trou de ver	AODV	Alarmes	Adaptation rapide aux changements de topologie	Surcoût de routage considérable
HSFA	Autonome	A base de spécification	Toutes les attaques sauf la suppression	AODV	Isoler le nœud malveillant	Détection temps réel ; équilibrage de charge ; taux de détection élevé; isolation du nœud malveillant	Faible surcoût de routage

Tableau 6.2 Comparaison des mécanismes de détection d'intrusion

De plus SIDE et IDAR sont basés sur le protocole de routage AODV. Pour faire une comparaison équitable, nous reprenons les mêmes paramètres de simulation (nombre de nœuds : 50, mobilité : de 0 à 20, nombre de nœuds malveillants entre 0 et 10), et nous considérons uniquement les attaques détectées par les trois mécanismes (privation de sommeil, trou noir, Rushing). Les résultats expérimentaux (voir figure 6.10) montrent que HSFA surpasse IDAR en termes de taux de détection

et de faux positifs, à l'exception de l'attaque Rushing où HSFA présente presque le même taux de faux positif qu'IDAR. Nous remarquons que SIDE et HSFA présentent des taux de détection similaires en ce qui concerne les attaques qui violent la spécification. Cependant HSFA ne nécessite pas un matériel spécial pour fonctionner tel qu'il est le cas pour SIDE qui a besoin de plateforme de confiance (TrustZone SOC). Contrairement à SIDE, HSFA ne demande pas de calcul et d'espace mémoire additionnel (nécessaire aux procédures d'attestation à distance). HSFA génère un faible surcoût de routages comparé à IDAR [127]. De plus HSFA ne dépend pas entièrement d'un ensemble de nœuds particuliers (SIDPS est basé sur l'hôte) tel qu'il est le cas pour IDAR, ce qui constitue un point de défaillance pour l'architecture du mécanisme. IDAR et HSFA fournissement une réponse adaptative contre les nœuds malveillants. IDAR isole le nœud malveillant complètement ou le contourne, en considérant le dommage causé par le nœud malveillant. HSFA isole le nœud malveillant complètement mais pour différentes périodes en considérant la récurrence de l'attaque. Le tableau 4 présente un résumé de la comparaison entre HSFA et les principaux travaux proposés dans la littérature.

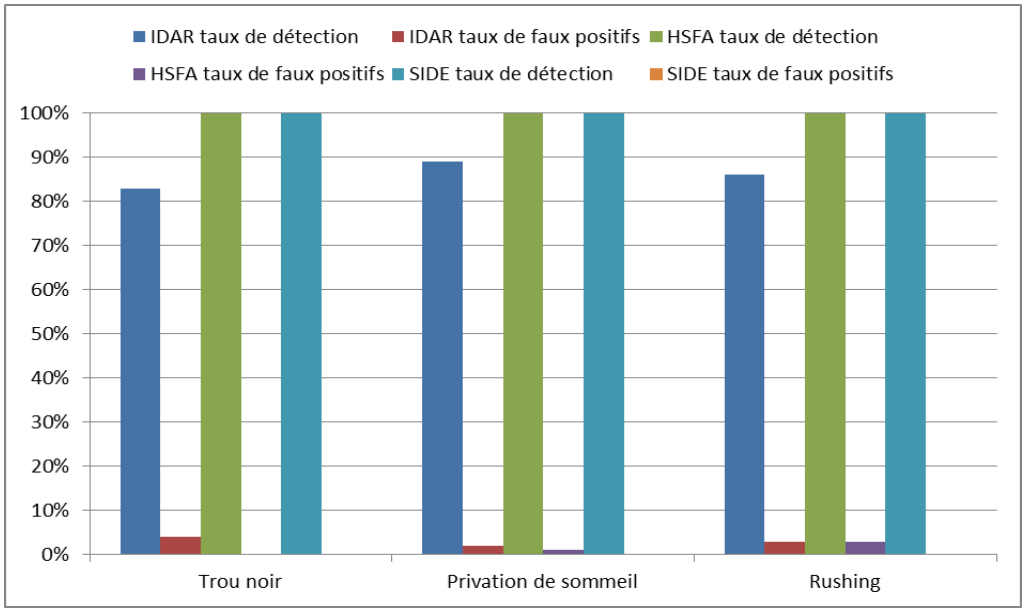


Figure 6.10 Comparaison entre HSFA et d'autres mécanismes

6.3. Conclusion

Dans de ce chapitre, nous avons proposé un système de détection et prévention d'intrusion pour les réseaux ad hoc mobiles. Le système proposé utilise la technique de détection à base d'anomalie pour détecter et prévenir les attaques qui utilisent la retransmission rapide telles que : trou de ver et Rushing. Nous avons proposé une approche statistique qui exploite le concept d'équilibrage de

charge. L'approche proposée permet d'identifier les nœuds malveillants comme des points de concentration du trafic dû à leurs taux de sélection élevé. Ensuite, nous avons proposé un nouveau système de détection et prévention d'intrusion qui combine l'approche statistique pour la détection à base d'anomalie (présentée dans ce chapitre) et la technique de détection à base de spécification (présentée dans le chapitre 5). La comparaison du nouveau système HSFA avec les mécanismes existants dans la littérature a montré que ce dernier surpasse les autres mécanismes en termes de détection et taux de faux positifs sans pour autant affecter les performances du réseau.

Chapitre 7 Conclusion

Le travail de recherche présenté dans cette thèse a atteint l'objectif fixé initialement. Ce chapitre complète la thèse en résumant le travail de recherche et les contributions qui ont été réalisés. Enfin, nous discutons les domaines des futures recherches possibles.

Le service de routage des données dans les réseaux ad hoc est vulnérable à diverses attaques. Bien que des efforts considérables ont été fait pour sécuriser ces réseaux, la majorité de ces travaux se sont focalisés sur la protection du réseau contre un type spécifique d'attaque. Par conséquent, la motivation de cette thèse émerge du besoin de sécuriser les réseaux ad hoc de diverses attaques. Dans cette thèse trois systèmes de détection et prévention d'intrusion sont proposés:

Le premier système de détection et prévention d'intrusion est basé sur la spécification. Il est conçu pour détecter les attaques qui violent la spécification du protocole de routage telles que : modification, fabrication et rejeu. Le système emploie une technique de détection qui grâce à un ensemble de règles définit le comportement normal du nœud du point de vue de l'opération de routage. Les règles de détection sont générées automatiquement sous forme de machine à états finis grâce à une méthode d'extraction automatique de la spécification. La méthode proposée pour l'extraction de la spécification s'inspire de la programmation logique inductive (PLI) et prend avantage des points communs entre les protocoles de routage ad hoc. Une instance du système est exécutée au niveau de chaque hôte du réseau afin de contrôler et protéger les interactions du nœud avec les autres nœuds du réseau. Les résultats des simulations ont montré que le système proposé permet de détecter et empêcher la majorité des attaques qui violent la spécification du protocole de routage avec un nombre de faux positifs avoisinant les 2%.

Le second système de détection et prévention d'intrusion est basé sur l'anomalie, il permet de détecter et prévenir les attaques à retransmission rapide telles que trou de ver et Rushing. Un type d'attaque que le nœud malveillant utilise pour intercepter le maximum de trafic sans violer la spécification du protocole de routage. Le système que nous avons proposé utilise une approche statistique qui exploite le concept d'équilibrage de charge. Une approche qui permet d'identifier les nœuds malveillants comme des points de concentration du trafic dus à leurs taux de sélection élevé. Non seulement le système proposé permet d'identifier et d'écarter les nœuds malveillants du réseau, mais aussi d'éliminer les points de concentration du trafic et assurer une répartition de charge. Les résultats des simulations montrent la capacité du système proposé à détecter et à

empêcher les attaques à retransmission rapide sans affecter les performances du protocole de routage.

Finalement, le troisième système que nous proposons combine les avantages de la technique de détection basée sur la spécification et de la détection basée sur l'anomalie. Un système de détection et prévention d'intrusion hybride, qui combine les deux systèmes décrit précédemment sur une architecture hiérarchique. Une détection à base de spécification au niveau de chaque hôte du cluster, permettant au nœud de surveiller ses interactions avec les autres nœuds et détecter localement les éventuelles violations de la spécification. Une détection basée sur l'anomalie au niveau des chefs des clusters pour détecter les retransmissions rapides et identifier et écarter les nœuds malveillants du réseau. La comparaison avec les mécanismes de sécurité existants a montré que le système proposé surpasse ces derniers en termes de taux de détection, taux de faux positifs et du nombre d'attaques détectées.

Les travaux réalisés dans le cadre de cette thèse ont permet d'atteindre l'objectif fixé initialement (dans la section 1.2); la sécurisation du routage dans les réseaux ad hoc en proposant des systèmes de détection et prévention d'intrusion adaptés à ce type de réseau. Par ailleurs, plusieurs perspectives peuvent être considérées pour améliorer davantage ces travaux. D'abord, il est intéressant d'étendre le champ de détection au niveau des couches transport et liaison de données pour détecter plus d'attaques et assurer une meilleure protection du réseau. De ce fait les systèmes de détection et prévention d'intrusion devraient être multicouches et multi-protocoles.

Les services de sécurité de base (décrits dans la section 2.3.1) telles que l'authentification des nœuds, l'intégrité et la confidentialité des données, et la non répudiation sont importants pour protéger les mécanismes proposés dans cette thèse des utilisations abusives. Il est important de garantir ces services pour le déploiement des mécanismes proposés dans un environnement réel. Cependant assurer ces services de sécurité basiques dans les réseaux sans fil multi-sauts est une problématique de recherche à multiple facettes. Donc nous pensons que cette problématique pourrait être le sujet d'une future recherche pour améliorer les travaux proposés dans cette thèse.

Dans le cadre de cette thèse nous avons montré que les systèmes proposés sont efficaces et qu'ils s'adaptent parfaitement aux caractéristiques des réseaux ad hoc mobiles MANETs. Les systèmes proposés pourraient facilement s'adaptés à d'autres classes des réseaux multi-sauts, tels que les réseaux maillés WMNs et véhiculaires VANETs.

Références

[1] RFC 2501 S. Corson and J. Macker "MANETs: Routing protocols performance issues and evaluation considerations", Informational RFC MANET working group , published January 1999.

[2] F. Dressler "A study of self organization mechanism in ad hoc and sensor networks", Proceeding of ACM International Journal of Computer Communication, Vo1.31, No.l3, pp 3018-3029.2008.

[3] K. Obraczka and G. Tsudik " Multicast routing issues in ad hoc networks", Proceeding of the IEEE International Conference on Universal Personal Communications(ICUPC), October 1998.

[4] X. Hong, K.Xu and M. Gerla "Scalable routing protocols for mobile ad hoc networks", Proceeding of the IEEE Network, Vo1.l6. No.4, pp 11~21, August 2002.

[5] C.C. Chiang and M. Gerla, "Routing and Multicast in Multihop, Mobile Wireless Networks" , Proceedings of The IEEE ICUPC'97, San Diego, CA, 1997.

[6] G. Pei, M. Gerla, X. Hong, and C.C. Chiang, "A Wireless Hierarchical Routing Protocol with Group Mobility" Proceedings of the IEEE WCNC'99, New Orleans, September 1999.

[7] D.L. Gu, G.Pei, H.Ly, M.Gerla and X.Hong "Hierarchical routing for multi -layer ad hoc wireless networks with UAVs", Proceedings of the IEEE MILCOM, 2000.

[8] J. C. Navas and T. lmielinski, "Geographic Addressing and Routing" Proceedings of the Third ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'97), Budapest, Hungary, 26_30 th September 1997.

[9] Y.-B. Ko and N. H. Vaidya, "Location-aided routing(LAR) in mobile ad hoc networks", Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking (Mobicom98), pp 66-75, 1998.

[10] C.E. Perkins and P. Bhagwat. Highly Dynamic Destination-Sequenced Distance-Vector outing (DSDV) for Mobile Computers. In Proc. of the conference on Communications architectures, protocols and applications (SIGCOMM'94), pages 244–254. ACM, August 1994.

[11] T. Clausen and P. Jacquet. Optimized Link State Routing Protocol OLSR. http ://tools.ietf.org/html/rfc3626, October 2003. RFC3626. (Consulté le 11 juin 2014)

[12] Johnson, D. (2007). The Dynamic Source Routing Protocol (DSR). http://tools.ietf.org/html/rfc4728. (Consulté le 11 juin 2014)

[13] Perkins, C., Belding-Royer, E., & Das, S. (2003). Ad hoc On-Demand Distance Vector Routing. http://tools.ietf.org/html/rfc3561. (Consulté le 11 juin 2014)

[14] Zygmunt J. Haas. A new routing protocol for the reconfigurable wireless networks. In Proceedings of 6th IEEE International Conference on Universal Personal Communications, IEEE

ICUPC'97, October 12-16, 1997, San Diego, California, USA, volume 2, pages 562–566. IEEE, IEEE, 1997.

[15] Z.J. Haas, M.R. Pearlman, and P. Samar. The zone routing protocol (zrp) for ad hoc networks. http ://tools.ietf.org/id/draft-ietf-manet-zone-zrp-04.txt, July 2002. Internet-Draft.

[16] Z.J. Haas, M.R. Pearlman, and P. Samar. The intrazone routing protocol (iarp) for ad hoc networks. http ://tools.ietf.org/html/draft-ietf-manet-zone-iarp-02, July 2002. Internet-Draft.

[17] Z.J. Haas, M.R. Pearlman, and P. Samar. The interzone routing protocol (ierp) for ad hoc networks. http ://tools.ietf.org/html/draft-ietf-manet-zone-ierp-02, July 2002. Internet-Draft.

[18] T. Clausen, C. Dearlove, and J. Dean. Mobile ad hoc network (manet) neighborhood discovery protocol (nhdp). http ://tools.ietf.org/html/draft-ietf-manet-nhdp-12, March 2010. Internet-Draft.

[19] Z.J. Haas, M.R. Pearlman, and P. Samar. The bordercast resolution protocol (brp) for ad hoc networks. http ://tools.ietf.org/html/draft-ietf-manet-zone-brp-02, July 2002. Internet-Draft.

[20] Panda, I. (2012). A survey on routing protocols of manets by using qos metrics. International journal of advanced Research in computer science and software engineering, volume2.

[21] F. Anjum and P. Mouchtaris "Security for Wireless Ad hoc networks", Book from senior scientist at Telcordia Technologies, Published by John Wiley & Sons, New Jersey, 2007.

[22] W.Lou ,W.Liu and Y.Fang "SPREAD: enhancing data confidentiality service in mobile ad hoc networks", Proceedings of the IEEE INFOCOM,2004.

[23] K.Seng and W.K.G Seah "Routing security and data confidentiality of mobile ad hoc networks", Proceedings of the IEEE Vehicular Technology Conference vrc, AprIl2003.

[24] E. Cayirci and C. Rong, Security in Wireless Ad Hoc and Sensor Networks. John Wiley & Sons. The Atrium, Southern Gate, Chichester, West Sussex, PO19 8SQ, UK. 2008.

[25] B. Wu, J. Chen, et al J. Wu and M. Cardei.2007. "A Survey of Attacks and Countermeasures in Mobile Ad Hoc Networks," in Wireless Network Security, Y. Xiao, X. Shen and D.-Z. Du, Eds. NY: Springer US, 2007, pp.103-135.

[26] Ieee 802.11 wireless lan media access control (mac) and physical layer (phy) specifications, ansi/ieee std 802.11. 1999.

[27] Abdelaziz, A.K., Nafaa, M., Salim, G., Security Attacks in Mobile Ad Hoc Networks. In Book chapter in Business Intelligence and Mobile Technology Research: An Information Engineering Perspective, edt. by Sean Eom (Cambridge Scholars Publishing).

[28] RACHEDI, A. (2008) Contributions à la sécurité dans les réseaux mobiles ad Hoc. PhD thesis, Université d'Avignon et des Pays de Vaucluse.

[29] Boussad, A. S. (2011) Sécurisation des Réseaux Ad hoc :Systèmes de Confiance et de Détection de Répliques, PhD thesis, Université de Limoges

[30] Ayachi, M. A. (2011). Contributions à la détection des comportements malhonnêtes dans les réseaux ad hoc AODV par analyse de la confiance implicite. PhD thesis , Université de Rennes 1 / Université 7 Novembre à Carthage.

[31] M.O. Pervaiz, M. Cardei and J. Wu, "Routing Security in Ad Hoc Wireless Networks," in Network Security, S. Huang, D. MacCallumand D.Z. Du, Eds. NY : Springer US, 2010, pp. 117-142.

[32] B. Wu, J. Chen, et al J. Wu and M. Cardei.2007. "A Survey of Attacks and Countermeasures in Mobile Ad Hoc Networks," in Wireless Network Security, Y. Xiao, X. Shen and D.-Z. Du, Eds. NY: Springer US, 2007, pp.103-135.

[33] Y.C. Hu, A. Perrig, and DB Johnson. Wormhole attacks in wireless networks. IEEE Journal on Selected Areas in Communications, 24(2) :370–380, 2006.

[34] Y.C. Hu, A. Perrig, and D.B. Johnson. Rushing attacks and defense in wireless ad hoc network routing protocols. In Proc. of the 2nd ACM workshop on Wireless security, page 40. ACM, 2003.

[35] John R. Douceur. The sybil attack. In IPTPS '01 : Revised Papers from the First International Workshop on Peer-to-Peer Systems, pages 251–260. Springer-Verlag, 2002.

[36] C.Piro, C.Shields and B.Levine, "Detecting the Sybil Attack in Mobile Ad hoc Networks", Proc. IEEE International Conference on Security and Privacy in Communication Networks, Aug-Sep. 2006

[37] Jan von Mulert, Ian Welch, Winston K.G. Seah, Security threats and solutions in MANETs: A case study using AODV and SAODV, Journal of Network and Computer Applications, Volume 35, Issue 4, July 2012, Pages 1249-1259, ISSN 1084-8045.

[38] E. Cayirci, C. Rong, Security in Wireless Ad Hoc and Sensor Networks, John Wiley and Sons, 2009.

[39] P-W. Yau, S. Hu and C. J. Mitchell, "Malicious attacks on ad hoc network routing protocols," International Journal of Computer Research, vol. 15, no 1, pp. 73-100, 2007.

[40] P.G. Argyroudis, D. O'Mahony, "Secure routing for mobile ad hoc networks," Communications Surveys & Tutorials, IEEE, vol.7, no.3, pp. 2- 21, 2005.

[41] P. Devi and A. Kannammal, "A hybrid defense mechanism for DDoS attacks using cluster analysis in MANET," in Proceedings of the International Conference on Advances in Computing, Communications and Informatics (ICACCI '12), 2012, pp. 287-291

[42] B. Wu, J. Chen, et al J. Wu and M. Cardei.2007. "A Survey of Attacks and Countermeasures in Mobile Ad Hoc Networks," in Wireless Network Security, Y. Xiao, X. Shen and D.-Z. Du, Eds. NY: Springer US, 2007, pp.103-135.

[43] A.D. Wood and J. A. Stankovic, "Denial of service in sensor Networks," Computer, vol.35, no.10, pp. 54- 62, Oct 2002.

[44] Hu Y-C. Packet leashes: a defense against wormhole attacks in wireless ad hoc networks. In: Proceedings of the 22nd annual joint conference of the IEEE computer and communications (INFOCOM), San Francisco, CA, USA, 2003. p. 1976–86

[45] Hu L, Evans D. Using directional antennas to prevent wormhole attacks. In: Proceedings of the network and distributed system security symposium, San Diego, CA, USA, 2004.

[46] F.Tseng, L. Chou and H.Chao, "A Survey of Black Hole Attacks in Wireless Mobile Ad Hoc Networks", Journal on Human-Centric Computing and Information Sciences, Springer, Vol.1, No.4, pp. 1-16, 2011.

[47] E.Padilla, N.Aschenbruck, P.Martini, M.Jahnke and J.Tolle, "Detecting Black Hole Attack in Tactical MANETs using Topology Graph", Proc. IEEE Conference on Local Computer Networks, 2007.

[48] M. Medadian, M.H. Yektaie and A.M. Rehmani, "Combat with Black Hole Attack in AODV Routing Protocol in MANETs", Proc. IEEE Asian Himalayas International Conference on Internet, Nov. 2009.

[49] X.Y. Zhang, Y. Sekiya and Y. Wakahara, "Proposal of a Method to Detect Black Hole Attack in MANETs", Proc. IEEE International Symposium on Autonomous Decentralized System ISADS, 2009.

[50] P. Papadimitratos and Z.J. Haas, "Secure Message Transmission in Mobile Ad Hoc Networks", Elsevier Journal of Ad Hoc Networks, Vol.1, No.1, pp 193-209, 2003.

[51] P. Papadimitratos, Z.J. Haas and P. Samar, "The Secure Routing Protocol (SRP) for Ad Hoc Networks", IETF Internet Draft , December 2002, available at http://www.ietf.org/proceedings/56/I-D/draftpapadimitratos-secure-routing-protocol-00.txt.

[52] A. Rawat, P.D. Vyavahare and A.K. Ramani, "Evaluation of Rushing Attack on Secure Message Transmission (SMT/SRP) Protocol for Mobile Ad Hoc Networks", Proc. International Conference on Personal Wireless Communications (ICPWC), Jan. 2005.

[53] A. Nadeem, M. Howarth, A survey of manet intrusion detection and prevention approaches for network layer attacks, Communications Surveys Tutorials, IEEE 15 (4) (2013) 2027–2045.

[54] D.Monica, J. Leitao, L. Rodrigues and C.Riberio, "On the use of Radio Resource Test in Wireless Ad Hoc Networks", Proc.Workshop on Recent Advances in Intrusion Tolerant Systems, Portugal, June 2009.

[55] J. Sen, M. Chandra, P. Balamurlidhar, S.G. Hariharaand H.Reddy, "A Distributed Protocol for Detection of Packet Dropping Attack in Mobile Ad hoc Networks", Proc. IEEE Conference on Telecommunication and Malaysian International Conference on Communication (ICT-MICC), 2007

[56] S. Marti, T.J. Giuli, K.Lai and M. Baker, "Mitigating Routing Misbehaviour in Mobile Ad Hoc Networks", Proc. International Conference on Mobile Computing and Networking, pp 255- 265, 2000.

[57] S.B. Lee and Y.H. Choi, "A Resilient Packet Forwarding Scheme against Maliciously Packet Dropping Nodes in Sensor Networks", Proc. ACM workshop on Security of Ad Hoc and Sensor Networks (SANS 2006),pp 59-70, USA, Oct. 2006.

[58] O.F. Gonzalez-Duque, M. Howarthand G. Pavlou, “Detection of Packet Forwarding Misbehaviour in MobileAd hoc Networks”, Proc. International Conference on Wired/Wireless Internet Communications (WWIC 2007), pp 302-314, Portugal, June 2007.

[59] O.F. Gonzalez-Duque, G. Ansa, M.Howarth and G. Pavlou, “Detection and Accusation of Packet Forwarding Misbehaviour in Mobile Ad hoc Networks”, Journal of Internet Engineering, Vol.2, No.8, pp 181-192, June 2008.

[60] O.F. Gonzalez-Duque, A.M. Hadjiantonis, G. Pavlou and M. Howarth, “Adaptive Misbehaviour Detection and Isolation in Wireless Ad Hoc networks Using Policies”, Proc. IFIP/IEEE International Symposium on ntegrated Network Management (IM 2009), pp 242- 250, NY, USA, June 2009.

[61] H. Yang, J. Shu, X. Meng, and S. Lu,“SCAN: Self-Organized Network Layer Security in Mobile Ad Hoc Networks,” IEEE Journal on Selected Areas in Communications, Vol. 24, No. 2, pp. 261-273, 2006.

[62] Bradley, K.A., Cheung, S., Puketza, N., Mukherjee, B., Olsson, R.A.: Detecting disruptive routers: a distributed network monitoring approach. In: Proceedings of the 1998 Symposium on Security and Privacy, May 1998, pp. 115–124 (1998)

[63] P.Yi, Z.Dai, Y. Zhong and S.Zhang, “Resisting Flooding Attack in Ad Hoc Networks”, Proc. IEEE International Conference on Information Technology Coding & Computing ITCC, April 2005.

[64] Y. Guo and S. Perreau, "Detect DDoS Flooding Attacks in Mobile Ad Hoc Networks," International Journal of Security and Networks, Vol. 5, No.4, pp. 259 - 269, 2010.

[65] T. Martin, M. Hsiao, H. Dong and J. Krishnaswami, “Denial-ofService Attacks on Battery Powered Mobile Computers”, Proc. IEEE International Conference on Pervasive Computing and Communications (PerCom), March 2004.

[66] W.Yu and K.Ray, “Defense Against Injecting Traffic Attack in Cooperative Ad Hoc Networks”, Proc. IEEE GLOBECOM, St. Louis, Missouri, USA, Dec. 2005.

[67] Marco Conti, Chiara Boldrini, Salil S. Kanhere, Enzo Mingozzi, Elena Pagani, Pedro M. Ruiz, Mohamed Younis, From MANET to people-centric networking: Milestones and open research challenges, Computer Communications, Volume 71, 1 November 2015, Pages 1-21, ISSN 0140-3664, http://dx.doi.org/10.1016/j.comcom.2015.09.007.

[68] Institute of Electrical and Electronics Engineers. IEEE Std 802.15.1-2005, Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Wireless Personal Area Networks (WPANs), 14 June 2005.

[69] ASTM International, Standard Specification for Telecommunications and Information Exchange Between Roadside and Vehicle Systems - 5 GHz Band Dedicated Short Range Communications (DSRC) Medium Access Control (MAC) and Physical Layer (PHY) Specifications, April 2009.

[70] Nadia haddadou these

[71] J. Anderson. Computer security threat monitoring and surveillance, 1980.

[72] Dorothy E. Denning. An intrusion-detection model. IEEE Transactions on software engeneering, SE-13 :222–232, 1987.

[73] R. Heady, G. Luger, A. Maccabe, and M. Servilla. The architecture of a network level intrusion detection system. Technical report, Computer Science Department, New Mexico, 1990. 39

[74] D. Denning. An intrusion detection model. IEEE Transactions on Software Engineering, 13(2):222–232, 1987. 22, 39

[75] E. Lundin and E. Jonsson. Survey of intrusion detection research. Technical Report 02–04, Department of Computer Engineering, Chalmers University of Technology, 2002. 40, 42

[76] P. Kazienko and P. Dorosz. Intrusion detection systems (IDS), 2004. http://www.windowsecurity.com/articles/IDS-Part2-Classification-methodstechniques.html.

[77] R.S. Puttini, J.-M. Percher, L. Me, O. Camp, R. Jr. Sousa, C.J.B. Abbas, and L.J. Garcia-Villalba. A modular architecture for distributed IDS in MANET. In Proceedings of the Computational Science and Its Applications: LNCS 2669, pages 91–113, 2003.

[78] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. ACM Computing Surveys, 41, 2009.

[79] P. Uppuluri and R. Sekar. Experiences with specification-based intrusion detection. In Proceedings of the Recent Advances in Intrusion Detection (RAID'01), LNCS 2212, pages 172–189. Springer, 2001

[80] C.-Y. Tseng, P. Balasubramayan, C. Ko, R. Limprasittiporn, J. Rowe, and K. Lewitt. A specification-based intrusion detection system for AODV. In Proceedings of the ACM Workshop on Security in Ad Hoc and Sensor Networks (SASN), 2003.

[81] S. Axelsson. Intrusion detection systems: A survey and taxonomy. Technical Report 99–15, Department of Computer Engineering, Chalmers University of Technology, 2000.

[82] E. Lundin and E. Jonsson. Survey of intrusion detection research. Technical Report 02–04, Department of Computer Engineering, Chalmers University of Technology, 2002.

[83] Y. Zhang and W. Lee. Intrusion detection in wireless ad hoc networks. In Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MobiCom'00), pages 275–283, 2000.

[84] Y. Huang, Wei Fan, Wenke Lee, and Philip S. Yu. Cross-feature analysis for detection ad-hoc routing anomalies. In Proceedings of the 23rd International Conference on Distributed Computing Systems (ICDCS), 2004

[85] A.B. Smith. An examination of an intrusion detection architecture for wireless ad hoc networks. In Proceedings of the 5th National Colloquium for Information System Security Education, 2001. 7, 44, 60, 117

[86] T. Anantvalee and J. Wu. A survey on intrusion detection in mobile ad hoc networks. In Wireless Network Security, pages 159–180. Springer, 2007.

[87] O. Kachirski and R. Guha. Effective intrusion detection usign multiple sensors in wireless ad hoc networks. In Proceedings of the 36th IEEE International Conference on System Sciences, 2003.

[88] S. Laniepce, J. Demerjian, and A. Mokhtari. Cooperation monitoring issues in ad hoc networks. In Proceedings of the International Conference on Communications and Mobile Computing, pages 695–700, 2006.

[89] C.-Y. Tseng, P. Balasubramayan, C. Ko, R. Limprasittiporn, J. Rowe, and K. Lewitt. A specification-based intrusion detection system for AODV. In Proceedings of the ACM Workshop on Security in Ad Hoc and Sensor Networks (SASN), 2003.

[90] E. Hansson, J. Gronkvist, K. Persson, and D. Nardquist. Specification-based intrusion detection combined with cryptography methods for mobile ad hoc networks. Technical report, FOI Swedish Defence Research Agency/Command and Control Systems, 2005.

[91] H.M. Hassan, M. Mahmoud, and S. El-Kassas. Securing the AODV protocol using specification-based intrusion detection. In Proceedings of the 2nd ACM International Workshop on Quality of Service and Security for Wireless and Mobile Networks, pages 33–35, 2006.

[92] C.H. Tseng, T. Song, P. Balasubramanyam, C. Ko, and K. Levitt. A specificationbased intrusion detection model for OLSR. In Proceedings of the 8th International Symposium on Recent Advances in Intrusion Detection (RAID'05), LNCS 3858, pages 330–350. Springer, 2005.

[93] J.-M. Orset, B. Alcalde, and A. Cavalli. An efsm-based intrusion detection system for ad hoc networks. In Proceedings of the Automated Technology for Verification and Analysis, pages 400–413, 2005

[94] Stakhanova, N.; Basu, S.; Wensheng Zhang; Xia Wang; Wong, J., "Specification Synthesis for Monitoring and Analysis of MANET Protocols," in Advanced Information Networking and Applications Workshops, 2007, AINAW '07. 21st International Conference on , vol.1, no., pp.183 187, 21-23 May 2007

[95] Panos, C., Xenakis, C., & Stavrakakis, I. (2010). A novel Intrusion Detection System for MANETs. In SECRYPT (Ed), International Conference on Security and Cryptography (SECRYPT), Athens (1-10). USA: IEEE.

[96] Hsiao-Ching Lin; Ming-Kung Sun; Han-Wei Huang; Tseng, C.-Y.H.; Hui-Tang Lin, "A Specification-Based Intrusion Detection Model for Wireless Ad Hoc Networks," in Innovations in Bio-Inspired Computing and Applications (IBICA), 2012 Third International Conference on , vol., no., pp.252-257, 26-28 Sept.

[97] Christoforos Panos, Christos Xenakis, Platon Kotzias, Ioannis Stavrakakis, A specification-based intrusion detection engine for infrastructure-less networks, Computer Communications, Volume 54, 1 December 2014, Pages 67-83.

[98] N.Ye, X.Li, Q.Chen, M.Emran and M.Xu, “Probabilistic Techniques for Intrusion Detection Based on Computer Audit Data”, IEEE Transactions on Systems, Man, and, Cybernetics, Vol. 31, No. 4, July 2001.

[99] N.Ye and Q.Chen, “An Anomaly Detection Technique based on a CHISQUARE Statistics for Detecting Intrusion into Information System”, International Journal of Quality and Reliability Engineering International, Vol.17, No.6, pp 105-112, 2001.

[100] H. Debar, M. Becker and D. Siboni, “A Neural Network Component for an Intrusion Detection System”, Proc. IEEE Computer Society Symposium on Security and Privacy, Oakland, May 1992.

[101] G.F.Cretu, J.Parekh, K.Wang and S.J.Stolfo, “Intrusion and Anomaly Detection Model Exchange for Mobile Ad-Hoc Networks”, Proc. IEEE Consumer Communication and Networking Conference, 2006.

[102] Y. Liu, C. Comaniciu and H. Man, "Modelling Misbehaviour in Ad Hoc Networks: a Game Theoretic Approach for Intrusion Detection”, International Journal of Security and Networks, Vol. 1, Nos.3/4, pp. 243 - 254, 2006.

[103] Panaousis, E.A.; Politis, C., "A game theoretic approach for securing AODV in emergency Mobile Ad Hoc Networks," in Local Computer Networks, 2009. LCN 2009. IEEE 34th Conference on , vol., no., pp.985-992, 20-23 Oct. 2009

[104] H.Jiang and H.Wang, “Markov Chain Based Anomaly Detection for Wireless Ad-Hoc Distribution Power Communication Networks”, Proc. IEEE Power Engineering Conference, 2005.

[105] B. Sun, K. Wu and U.W. Pooch, “Routing Anomaly Detections in Mobile Ad Hoc Networks”, Proc. IEEE International Conference on Computer Communication and Networks ICCCN, 2003.

[106] B.Sun, K.Wu, Y.Xiao and R.Wang, “Integration of Mobility and Intrusion Detection Wireless Ad Hoc Networks”, Journal of Communication Systems, Wiley International, Vol. 20, No. 6, pp. 695-721, 2007.

[107] N.Ye, X.Li, Q.Chen, M.Emran and M.Xu, “Probabilistic Techniques for Intrusion Detection Based on Computer Audit Data”, IEEE Transactions on Systems, Man, and, Cybernetics, Vol. 31, No. 4, July 2001.

[108] N.Ye and Q.Chen, “An Anomaly Detection Technique based on a CHISQUARE Statistics for Detecting Intrusion into Information System”, International Journal of Quality and Reliability Engineering International, Vol.17, No.6, pp 105-112, 2001.

[109] A.Nadeem and M.Howarth, “Adaptive Intrusion Detection & Prevention of Denial of Service Attacks in MANETs”, Proc. ACM International Wireless Communication and Mobile Computing Conference (IWCMC 09), Leipzig Germany, June 2009.

[110] A.Mitrokosta, N.Komninos and C.Douligeris, “Intrusion Detection with Neural Networks and Watermarking Techniques for MANETs”, Proc. IEEE International Conference on Pervasive Services, pp 118-127, July 2007.

[111] S.Jabbehdari, S.H. Talari and N.Modiri, “A Neural Network Scheme for Anomaly Based Intrusion Detection Systems in Mobile Ad Hoc Networks”, Journal of Computing, Vol. 4, No. 2, pp-61-66, 2012.

[112] Sevil Sen and John A. Clark. 2011. Evolutionary computation techniques for intrusion detection in mobile ad hoc networks. Comput. Netw. 55, 15 (October 2011), 3441-3457.

[113] Barani, F., & Abadi, M.I., (2012). BeeID: intrusion detection in AODV-based MANETs using artificial bee colony and negative selection algorithms, The ISC International Journal of Information Security, 1, 4.

[114] Barani, F., "A hybrid approach for dynamic intrusion detection in ad hoc networks using genetic algorithm and artificial immune system," in Intelligent Systems (ICIS), 2014 Iranian Conference on , vol., no., pp.1-6, 4-6 Feb. 2014.

[115] D.S. Bauer, F.R. Eichelman, R.M. Herrera and A.E. Irgon, “Intrusion Detection an Application of Expert Systems to Computer Security”, Proc. IEEE International Conference on Security Technology, 1989.

[116] T. Lunt and R. Jagannathan, “A Prototype Real Time Intrusion Detection Expert System”, Proc. IEEE Symposium on Security and Privacy, Oakland, April 1988.

[117] N. Habra, B. L. Charlier, A. Mounji and I. Mathie, “Asax_ Software Architecture and Rule Based Language for Universal Audit Trail Analysis”, Proc. European Symposium on Research in Computer Security ESORICS, France, November 1992.

[118] K.Ilgun, R.A.Kemmerer, and P.A.Porras, “State Transition Analysis: a Rule Based Intrusion Detection Approach”, IEEE Transactions on Software Engineering, Vol.21, No.3, pp-181-199, March 1995.

[119] G. Vgina, S. Gawalani, K. Srinivasan, M. Belding-Royer and A. Kemmerer, “An Intrusion Detection Tool for AODV Based Ad Hoc Wireless Networks”, Proc. IEEE Annual Computer Security Application Conference ACSAC, 2004.

[120] N.Komninos, D. Vergados and C.Douligeris, “Detecting Unauthorized and Compromised Nodes in Mobile Ad Hoc Networks”, Journal of Ad Hoc Networks, Elsevier, Vol. 5, No. 3, pp 289–298, April 2007.

[121] Porras, P.A. and Neumann, P. G., “EMERALD: Event monitoring enabling responses to anomalous live disturbances,” in Proceedings of the 20th National Information Systems Security Conference, Oct. 1997. 9.1.1

[122] Javitz, H. S. and Valdes, A., “The SRI IDES statistical anomaly detector,” in Proceedings of the IEEE Research in Security and Privacy, (Oakland, CA), pp. 316–376, May 1991.

[123] Anderson, D., Frivold, T., and Valdes, A., “Next-generation intrusion detection expert system (NIDES): A summary,” Tech. Rep. SRI-CSL-95-07, Computer Science Laboratory, SRI International, Menlo Park, CA, May 1995.

[124] F. Anjum, D. Subhadrabandhu and S. Sarkar, “Signature Based Intrusion Detection for Wireless Ad Hoc Networks: a Comparative Study of Various Routing Protocols”, Proc. IEEE Vehicular Technology Conference (VTC), Oct 2003.

[125] A.B.Smith, “An Examination of Intrusion Detection Architecture for Wireless Ad-Hoc Networks”, Proc. National Colloquium for Information System Security Education, May 2001.

[126] Alattar, Mouhannad, “Supervision de la sécurité pour des réseaux ad hoc mobiles : un système léger, robuste, et fiable de détection d'intrusion”, thesis 2013

[127] Nadeem, A.; Howarth, M., "A generalized intrusion detection & prevention mechanism for securing MANETs," in Ultra Modern Telecommunications & Workshops, 2009. ICUMT '09. International Conference on , vol., no., pp.1-6, 12-14 Oct. 2009

[128] Adnan Nadeem, Michael P. Howarth, An intrusion detection & adaptive response mechanism for MANETs, Ad Hoc Networks, Volume 13, Part B, February 2014, Pages 368-380

[129] Shakshuki, E.M.; Nan Kang; Sheltami, T.R., "EAACK—A Secure Intrusion-Detection System for MANETs," in Industrial Electronics, IEEE Transactions on , vol.60, no.3, pp.1089-1098, March 2013

[130] S. Sen, Evolutionary Computation Techniques for Intrusion Detection in Mobile Ad Hoc Networks, Ph.D. Thesis, University of York, UK, 2010.

[131] P. Ning and K. Sun. How to misuse AODV : a case study of insider attacks againstmobile ad-hoc routing protocols. In Proc. IEEE Systems, Man and Cybernetics Society, Information Assurance Workshop (IAW’03) , pages 60–67. IEEE, June 2003.

[132] Schneier, B., (1999). Attack Trees - Modeling Security Threats. Dr. Dobb’s Journal, 24(12), 21–29.

[133] Ebinger, P., & Bucher, T. (2006). Modelling and analysis of attacks on the MANET routing in AODV. In Springer-Verlag (Ed), the 5th international conference on Ad-Hoc, Mobile, and Wireless Networks (ADHOC-NOW'06), Ottawa, Canada (294-307). Berlin, Heidelberg: Springer-Verlag.

[134] Guerrero-Zapata M, Asokan N. Securing ad hoc routing protocols. In: Proceedings of the 2002 ACM workshop on wireless security (WiSe 2002), Atlanta, Georgia, USA, 2002. p. 1–10.

[135] H. Renard. Équilibrage de Charge et Redistribution de Données sur PlatesFormes Hétérogènes. PhD thesis, École Normale Supérieure de Lyon - Laboratoire de l’Informatique du Parallélisme, Décembre 2005

[136] Ns-2: The Network Simulator, 2015. http://isi.edu/nsnam/ns/

[137] Opnet, 2015. http://www.opnet.com/

[138] Omnet++, 2015. http://www.omnetpp.org/

[139] GloMoSim, 2015. http://pcl.cs.ucla.edu/projects/glomosim/lastaccess

[140] Request for Comments (RFC), 2015. https://www.ietf.org/rfc.html

[141] Ko, c. Logic induction of valid behavior specifications for intrusion detection. (2000). In Sp ’00: Proceedings Of The 2000 IEEE Symposium On Security And Privacy.

[142] Muggleton, s. H. And raedt, l. D. (1994). Inductive logic programming: theory and methods. Journal of logic programming, 19 (20), 629–679.

[143] F. Jaddi and B. Paillassa, "Mobility and Density Self-Adaptive Routing Strategies in Ad Hoc Networks," 2006 IEEE International Conference on Mobile Ad Hoc and Sensor Systems, Vancouver, BC, 2006, pp. 685-690.

Annexe A Liste des publications

Revues internationales :

- Abdelaziz Amara Korba, Mehdi Nafaa, Salim Ghanemi, “An efficient intrusion detection and prevention framework for ad hoc networks”, Information and computer security, Emeraled (Accepté).

- Abdelaziz Amara Korba, Mehdi Nafaa, Salim Ghanemi, “Hybrid Intrusion Detection Framework for Ad hoc networks", International Journal of Information Security and Privacy, IGI-global (Accepté).

Conférences internationales :

- Amara Korba, A.; Nafaa, M.; Ghanemi, S.; Ghamri-doudane, Y. "Anomaly-based intrusion detection for Ad hoc Networks", in Global Information Infrastructure and Networking Symposium (GIIS 2016) (soumis).

- Amara Korba, A.; Nafaa, M.; Ghanemi, S., "Specification-based intrusion detection for Ad hoc Networks", in 2016 International Conference on Multimedia Computing and Systems (ICMCS’16) (Accepté).

- Amara Korba, A.; Nafaa, M.; Ghanemi, S., "Analysis of security attacks in AODV" in 2014 International Conference on Multimedia Computing and Systems (ICMCS), vol., no., pp.752-756, 14-16 April 2014.
  doi: 10.1109/ICMCS.2014.6911193

- Amara Korba, A.; Nafaa, M., Salim, G., "Survey of Routing Attacks and Countermeasures in Mobile Ad Hoc Networks," in Computer Modelling and Simulation (UKSim), 2013 UKSim 15th International Conference on , vol., no., pp.693-698, 10-12 April 2013.
  doi: 10.1109/UKSim.2013.48

- Amara Korba, A.; Nafaa, M.; Ghanemi, S, “Security Attacks in Mobile Ad Hoc Networks” in International Conference on Information Systems and Technologies "ICIST'2013" (Tangier, Morocco).

Chapitres de livres :

Amara Korba, A.; Nafaa, M.; Ghanemi, S. (2013). Security Attacks in Mobile Ad Hoc Networks. In Cambridge Scholar Publishing, Business Intelligence and Mobile Technology Research: An Information Systems Engineering Perspective. ISBN (10): 1-4438-5507-3, ISBN (13): 978-1-4438-5507-5.

Annexe B Glossaire

AIDP	Anomaly-based Intrusion Detection Protocol
AODV	Ad Hoc On-demand Distance Vector Routing
ARP	Adresse Resolution Protocol
BRP	Bordercast Resolution Protocol
BSA	Basic Service Area
BSM	Basic Security Module
BSS	Basic Service Set
CA	Certificate Authority
CBR	Constant Bit Rate
CGSR	Cluster gateway switch routing
CGSR	Cluster Gateway Switch Routing
CPN	Colored Petri Networks
CSMA	Carrier Sense Multiple Access
CTS	Clear To Send
DCA	Distributed Clustering Algorithm
DCF	distributed coordination function
DM	Derived Matrix
DMCA	Distributed Mobility Adaptive Clustering
DoS	Denial Of Service
DSDV	Destination Sequenced Distance Vector Routing
DSR	Dynamic Source Routing
DSR	Dynamic Source Routing
DSRC	Dedicated Short Range Communication
EAACK	Enhanced Adaptive ACKnowledgment
EFSM	Extended Finite state machines

ESS	Extended Service Set
EWMA	Exponentially Weighted Moving Average
FSM	Finite state machines
GIDP	Generalized Intrusion Detection & Prevention
GPS	Global Positioning System
GTMR	Game Theoretic MANET Routing
HIDS	Host Based Intrusion Detection
HIDS	Host-based intrusion detection system
HSR	hierarchical state routing
IARP	Intrazone Routing Protocol
IBSS	Independent Basic Service Set
ID	Intrusion Detection
IDAR	Intrusion Detection & Adaptive Response
IDEF	Intrusion Detection Exchange Format
IDP	Intrusion Detection & Prevention
IDS	Intrusion Detection System
IERP	IntErzone Routing Protocol
IETF	Internet Engineering Task Force
IETF	Internet Engineering Task Force
IP	Internet Protocol
IRA	Intrusion Response Action
IRS	Intrusion Response System
ITP	Initial Training Profil
ITS	Intelligent Transport System
KBID	Knowledge Based Intrusion Detection
KDC	Key Distribution Centre
LAN	Local Area Network

MAC	Medium Access Control
MANET	Mobile Ad Hoc Network
MN	Manager Node
MPR	Multipoint relay
NAM	Network animator
NCM	Network Characteristic Matrix
NHDP	Neighborhood Discovery Protocol
NIDS	Network-based intrusion detection system
NPD	Network Performance Degradation
NWS	Neighbour Watch System
OBU	On Board Unit
OLSR	Optimized link State Routing
OTCL	Object oriented TCL
PAN	Personal Area Network
PDA	Personal Digital Assistant
PDR	Packet Delivery Ratio
PIL	Potential Intruder List
PRNET	Packet Radio Network
QoS	Quality of Services
RERR	Route Error Packet
RFC	requests for comments
RREP	Route Reply Packet
RREQ	Route Request Packet
RSU	Road Side Board
RTS	Request To Send
RU	Rushing
RWD	Random Walk Detectors

RWP	Random Way Point Model
SCAN	self-organized network-layer security in mobile ad hoc networks
SIDE	Specification-based Intrusion Detection
SMT	Secure Message Transmission
SNMP	Simple Network Management Protocol
SRP	Secure Routing Protocol
SVM	Support Vector Machine
TTL	Time To Live
UDP	User Datagram Protocol
V-2-I	Vehicle-to-Infrastructure Communication
V-2-V	Vehicle-to-Vehicle Communication
VANET	Vehicular Ad Hoc Network
WEP	Wired Equivalent Privacy
Wi-Fi	Wireless Fidelity
WLAN	Wireless Local Area Network
WMAN	Wireless Metropolitan Area Network
WMN	Wireless Mesh Network
WPA	Wi-Fi Protected Access
WPAN	Wireless Personal Area Network
WRP	Wireless Routing Protocol
WSN	Wireless Sensor Networks
WWAN	Wireless Wide Area Network
ZHLS	Zone-Based Hierarchical Link State
ZRP	Zone Routing Protocol