N°d'ordre : 17/2012-M/IN

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE DES SCIENCES ET DE LA TECHNOLOGIE « HOUARI BOUMEDIENE »

FACULTE D'ELECTRONIQUE ET D'INFORMATIQUE

MÉMOIRE

Présenté pour l'obtention du diplôme MAGISTER

En: INFORMATIQUE

Spécialité : Informatique Mobile

Par: HARBOUCHE OUSSAMA

Sujet:

Détection et Elimination des Nœuds Malicieux dans un VANET (Vehicular Ad-Hoc NETwork)

Soutenu Publiquement le 03/01/2012, devant le jury composé de :

Mme M. BOUKALA	**Professeur / à l'USTHB**	**Présidente**
Mme S. MOUSSAOUI	**Maîtres de Conférences/A, à l'USTHB**	**Directrice de mémoire**
Mme N. NOUALI	**Maîtres de Recherches/A, au CERIST**	**Examinatrice**
Mr M. BENCHAIBA	**Maîtres de Conférences/A, à l'USTHB**	**Examinateur**

Remerciements

Louange à notre Seigneur 'ALLAH' qui m'a doté de la merveilleuse faculté de raisonnement. Louange à notre créateur qui m'a incité à acquérir le savoir. C'est à lui que m'adresse toute ma gratitude en premier lieu.

En second lieu, je tiens à remercier Mme S. MOUSSAOUI ma promotrice pour sa disponibilité, ses idées, ses conseils et ses encouragements qui m'ont permis de mener à bien ce mémoire.

Je remercie particulièrement les honorables membres de jury qui ont pris la peine de lire et d'évaluer ce mémoire.

Je tiens aussi à remercier l'ensemble des enseignants de l'USTHB, sans exception, ainsi que les employés qui ont rendu plus confortable notre formation au sein de l'université

Un grand MERCI à tous les membres de ma famille et spécialement mes parents qui m'ont soutenu tout au long de mes études et qui ont fait en sorte, par leur amour, leur affection et leur soutien financier, que je puisse avoir les meilleures conditions possibles pour continuer mes études et aller de l'avant. Qu'ils trouvent ici ma gratitude et mon amour pour eux.

Résumé

Notre travail consiste en la conception d'une nouvelle technique de détection et d'élimination des nœuds malicieux pour les réseaux véhiculaires qui répond aux défis imposés par la nature des applications véhiculaires. Notre objectif est double: (a) optimiser les paramètres de sécurité de ces réseaux et (b) améliorer les performances des applications de sécurités pour ces réseaux.

Pour se faire, nous avons étudié dans un premier temps les approches proposées dans ce domaine pour les réseaux véhiculaires. Ces solutions ont étés analysées et critiquées. Elles nous ont permis de concevoir, dans un second temps, un nouveau protocole plus adapté.

Notre contribution a été d'apporter une amélioration aux mécanismes de détection des nœuds malicieux et de compléter les solutions existantes par un outil efficace d'élimination de noeuds malicieux.

Les simulations effectuées ont permis de montrer l'efficacité de notre solution par rapport aux solutions existantes par rapport aux paramètres de performances et de sécurité visés.

Mots clés: VANETs, Sécurité, Détection de nœuds malicieux, Algorithmes distribués, modèles de mobilité.

Abstract

Our work proposes the design of a new method of detection and elimination of the malicious nodes on the vehicular networks. This solution is an answer to the challenges imposed by the nature of the vehicular applications. Our objective is double: (a) to optimize security settings of these networks and (b) to improve the application performances of safety measures for these networks.

To be done, we initially studied the approaches suggested in this field for the vehicular networks. These solutions enabled us to design, in the second time, a new protocol.

Our contribution is an improvement of the detection mechanisms of the malicious nodes and proposes an effective tool for their elimination.

The simulations show the effectiveness of our solution compared to the existing solutions by referring to the parameters of performances and security concerned.

Key words: VANETs, Security, Malicious node detection, distributed Algorithms, mobility models.

Table des matières

Table des figures

Introduction

Les réseaux locaux sans fil IEEE 802.11 constituent de nos jours le standard des WLANs le plus largement déployé et utilisé à travers le monde. Les contextes d'utilisation de ces réseaux sont divers et vont principalement du cadre domestique, aux lieux publics (e.g. gares, hôtels, restaurants, etc.) à travers notamment des HotSpots, en passant par le cadre du travail. Poussés précipitamment sur le marché, les WLANs 802.11 n'ont pu intégrer des mécanismes de sécurité robustes qu'après la déferlante des attaques dont ces réseaux ont fait l'objet et la prise de conscience progressive de l'étendue des vulnérabilités dans leur conception initiale. Dans le contexte des réseaux véhiculaires où un consensus technologique semble se dégager autour du standard DSRC/IEEE 802.11p pour les déploiements à venir, il est urgent de tirer les leçons des premiers déploiements des WLANs et donc d'éviter que la sécurité ne soit pensée, une fois de plus, a posteriori.

La sécurité des réseaux véhiculaires est donc aujourd'hui un enjeu majeur dont il faut se saisir pour garantir la plus large adoption possible de ces réseaux aussi bien par les usagers de la route dont on attend l'utilisation que les opérateurs dont on attend le déploiement. Cette sécurité s'inscrit dans un contexte particulier marqué, par une forte dynamique des nœuds avec des pointes de vitesse pouvant atteindre 200 Km/h, une aggravation de l'instabilité de la propagation radio, une connectivité intermittente, une topologie dynamique mais contrainte par celle des routes, un réseau potentiellement très étendu appelant une administration nécessairement hétérogène, un potentiel énergétique important,… etc. En adjonction à cette liste non exhaustive, il faut certainement compter avec la nature des applications ou des services qui y sont opérés et qui sont susceptibles d'induire des exigences de sécurité différenciées à l'image de la dichotomie du modèle de communication induite par les applications. Tout ce contexte, pris dans sa globalité, crée pour la sécurité des réseaux véhiculaires de nombreuses possibilités d'investigation dont on ne peut pourtant pas encore dire, à l' aube des contributions actuelles, qu'elles soient pleinement explorées.

Quand un dispositif commence à envoyer de mauvaises informations, la solution à long terme est l'autorité de certification (par exemple, le Département de Motor Vehicles) pour retirer les qualifications du dispositif offensant. Cependant, ce processus prend du temps, de

la collection des évidences ou accusations, à la résolution des réclamations contestées. Les attaques intérimaires et continues pourraient mettre en danger la sûreté des passagers. Ainsi, il y a un besoin d'isoler rapidement de tels dispositifs errants et de les empêcher de diffuser des données incorrectes. Une solution pour les voitures observant la mauvaise conduite est d'exclure temporairement le mauvais dispositif responsable jusqu'à ce que l'autorité de certification soit informée et qu'elle prenne une mesure appropriée.

Dans ce travail, nous nous intéressons particulièrement à la conception d'une nouvelle technique de détection et d'élimination des nœuds malicieux dans un VANET pour les réseaux véhiculaires mobiles qui prend en considération les caractéristiques spécifiques à ces réseaux et minimise l'impact des attaques menés par les nœuds malicieux sur un VANET. Dans ce travail, on a été emmené à réaliser une solution complète capable de détecter les comportements malicieux et d'éliminer les nœuds qui y sont responsables. Tout en prenant en considération les divers paramètres de sécurité et de performance du réseau. Pour se faire, nous avons étudié dans un premier temps les approches proposées. Ces solutions ont étés analysées et critiquées. Elles nous ont permis de concevoir, dans un second temps, un nouveau protocole prenant en considération les contraintes imposées par notre environnement et réalisant une grande partie des buts fixés. Dans ce protocole, on fait usage d'une stratégie de suicide temporaire qui mène à l' exclusion du véhicule malicieux et à la remise en circulation des véhicules honnêtes après la collecte des évidences.

Ce mémoire est organisé en cinq chapitres de la manière suivante: Le premier chapitre présente des généralités sur les réseaux véhiculaires avec une description de leurs architectures et leurs caractéristiques principales ainsi que leurs domaines d'application. Dans le deuxième chapitre, nous présentons dans un premier temps la sécurité des réseaux véhiculaires et ses principes. Dans le troisième chapitre, nous présentons les comportements malicieux dans les réseaux véhiculaires, et les techniques de détection et d'élimination des nœuds malicieux proposés dans ce contexte, une étude détaillée et des critiques sont apportées à ces méthodes. Le quatrième chapitre est consacré à la description détaillée de nos contributions. Ces dernières sont proposées comme solutions aux problématiques posées par la mobilité des nœuds afin de fournir une efficacité optimale à notre mécanisme. Une évaluation des performances de nos propositions a été réalisée sous le simulateur NS2. Enfin, nous terminons par une conclusion générale qui résume nos observations retenues de l'étude de l'état de l'art et de la discussion des résultats obtenus par notre solution, et présente les perspectives envisageables à notre travail.

Chapitre I

Définition et caractéristiques

1 Introduction

Les réseaux de véhicules, ou VANET (*Vehicular Ad Hoc Networks)*, sont une technologie émergente intégrant les dernières techniques de communication. Chaque nœud du réseau est un véhicule équipé d'une ou plusieurs interfaces radio sans fil. Les véhicules communiquent entre eux grâce à cet équipement. Un réseau de véhicules fournit [1] (1) le long de la route, une connectivité au monde extérieur par l'intermédiaire de passerelles vers d'autres réseaux, et (2) une communication inter-véhiculaire pour les véhicules intelligents ou ITS (*Inteligeant Transportation Systems)*. Sans infrastructure, le réseau est un réseau ad hoc. Un protocole doit donc être utilisé pour assurer les communications inter-véhiculaires. Les réseaux de véhicules sont aussi appelés IVC, pour *Inter-Vehicule Communication* ou IVCS pour *Inter-Vehicular Communication Systems*.

La recherche sur les réseaux de véhicules ou les communications inter-véhiculaires a commencé au Japon au début des années 1980 par la JSK (*Association of Electronic Technology for Automobile Traffic and Driving*). Plus tard, California PATH [2] et Chauffeur [3] ont présenté des techniques permettant de relier deux véhicules ou plus, pour former un convoi. Récemment, le projet Européen CarTalk 2000 [4] tente de résoudre des problèmes liés à la sécurité ou au confort des passagers par l'intermédiaire de communications inter-véhiculaires. Depuis 2002, avec le développement rapide de technologies sans fil, le nombre de publications dans le domaine des réseaux de véhicules a rapidement augmenté. Dans cette dynamique, divers workshops ont été créés, comme ACM *International Workshop on Vehicular Ad Hoc Network* en 2004 ou *International Workshop on Inteligent Transportation* en 2003.

Les véhicules intelligents sont la principale application des réseaux de véhicules, notamment pour augmenter la sécurité ou le confort des passagers. Les fonctionnalités sont [1] la surveillance du trafic, le contrôle du trafic, la visibilité augmentée des carrefours dangereux, la détection de collisions, les services d'information de proximité, le calcul de trajet en temps réel selon le trafic. D'autres applications, en dehors des transports intelligents, permettent de fournir la connectivité à Internet, ou toute communication entre véhicules, comme les jeux ou l'échange de fichiers.

Ce chapitre donne une vue d'ensemble sur les réseaux véhiculaires, montrant leurs architectures potentielles, leurs caractéristiques et leurs scénarios possibles de déploiement.

Les réseaux véhiculaires, les avantages et les applications réelles sont présentés d'une vue d'utilisateur du réseau en s'appuyant sur des exemples potentiels de services. Un certain nombre de défis techniques dans le déploiement de réseau véhiculaire est montré dans la suite du chapitre. En conclusion, quelques travaux de standardisation ainsi qu'un certain nombre de projet en cours d'évolution ont été accentués.

2 Réseau véhiculaire: définition

Les réseaux véhiculaires constituent une nouvelle classe des réseaux sans fil qui ont émergé grâce aux avancés dans les technologies sans fil et de l'industrie automotrice. Les réseaux véhiculaires sont spontanément formés entre les véhicules mobiles équipés d'interfaces sans fil qui pourraient être de technologies homogènes ou hétérogènes. Ces réseaux, également connus sous le nom de VANETs, sont considérés en tant qu'une des applications réelles des réseaux mobiles ad-hoc, permettant la communication entre véhicules voisins aussi bien qu'entre véhicules et équipements fixes voisins, habituellement décrits comme équipements de bord de la route.

Ces réseaux attirent une attention considérable de la communauté de la recherche aussi bien que l'industrie automotrice. De grands intérêts pour ces réseaux sont également montrés de la part des autorités gouvernementaux et des organismes de normalisation. Dans ce contexte, un système de communications à courte portée (DSRC : *Dedicated Short-Range Communications*) a émergé en Amérique du Nord, où 75 mégahertz du spectre ont été approuvés par la FCC (*Federal Communication Commission*) des États-Unis en 2003 pour de tel type de communication qui vise principalement les réseaux véhiculaires [5]. D'autre part, le *Car-to-Car Communication Consortium* (C2C-CC)[6] a été lancé en Europe par des fabricants de voiture et des OEM (*Original Equipment Manufacturers*), avec l'objectif principal d'accroitre la sécurité et l'efficacité de la circulation routière au moyen de la communication inter-véhicule. IEEE avance également dans la famille 1609 de normes pour l'accès sans fil dans les environnements véhiculaires (WAVE : *Wireless Access in Vehicular Environments*)[7].

3 Architectures et caractéristiques des réseaux de véhicules

3.1 Architectures des réseaux véhiculaires

Les réseaux véhiculaires peuvent être déployés par des opérateurs du réseau, des fournisseurs de service ou par coopération entre les opérateurs, les fournisseurs, et une autorité gouvernementale. Les avancées récentes en technologies sans fil et les tendances d'avancement dans les scénarios de réseau ad hoc permettent un certain nombre d'architectures de déploiement pour les réseaux véhiculaires, dans les environnements d'autoroute, rural et urbain. De telles architectures devraient permettre la communication entre les véhicules voisins et entre les véhicules et l'équipement fixe voisin de bord de la route. [8]

Trois alternatives de réseaux véhiculaires

(i) Un réseau ad hoc véhicule-à-véhicule sans fil pur (V2V) permettant la communication véhiculaire autonome sans l'appui d'infrastructure (figure 1(b)),

(ii) Une épine dorsale câblée avec des *hots spots* sans fil qui peuvent être vus comme un WLAN (figure 1(a)),

et (iii) une architecture hybride véhicule-à-route (V2R) qui ne se fondent pas sur une infrastructure fixe d'une façon constante, mais peuvent l'exploiter pour une exécution améliorée et entretenir l'accès quand elle est disponible. En fait l'architecture V2R inclut implicitement la communication V2V (figure 1(c)).

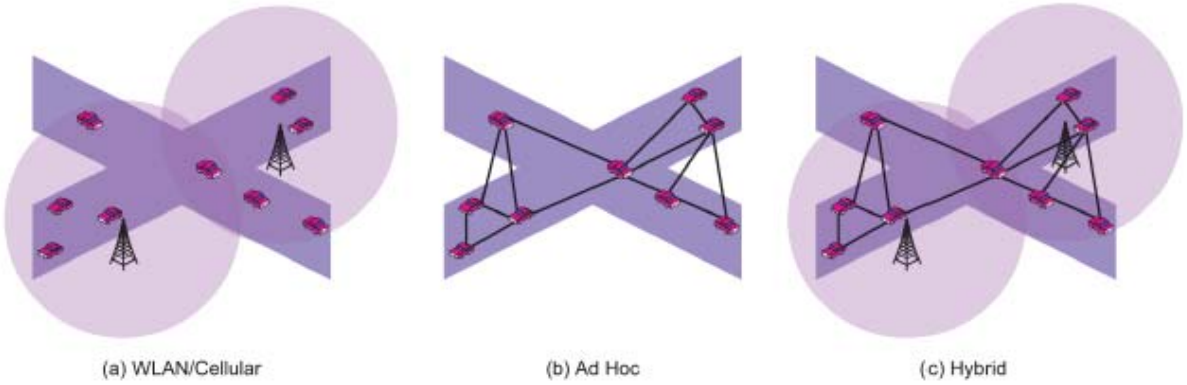


Figure 1. Trois catégories d'architectures pour les réseaux de véhicules [1]

Une architecture de référence pour les réseaux véhiculaires a été proposée dans le C2C-CC, distinguant trois domaines : '*in-vehicle*','*ad hoc*' et '*infrastructure domain*' [6,9].

a) in-vehicle domain: La figure 2 montre cette architecture de référence. Le domaine '*in-véhicule*' se rapporte à un réseau local à l'intérieur de chaque véhicule logiquement composé de deux types d'unités :

(i) une unité de bord (OBU : *on-board unit*)

(ii) une ou plusieurs unités d'application (AUs : *Application Unit(s)*).

Un OBU est un dispositif dans le véhicule ayant des aptitudes de communication (filaire et/ou sans fil), alors qu'un AU est un dispositif exécutant une ou un ensemble d'applications le tout en se servant des aptitudes de la communication de l'OBU. En effet, un AU peut être une pièce intégrée d'un véhicule reliée de manière permanente à un OBU. Il peut également être un dispositif portable tel qu'un ordinateur portable ou un PDA qui peuvent dynamiquement être attaché (et détaché) à un OBU. L'AU et les OBU sont habituellement reliés via un câble, alors que le raccordement sans fil est également possible (en utilisant par exemple, Bluetooth, WUSB, ou UWB). Cette distinction entre l'AU et l'OBU est logique, et ils peuvent également résider dans une unité physique simple.

b) ad hoc domain: Le domaine ad hoc est un réseau composé de véhicules équipés d'OBUs et d'unités de bord de la route (RSUs : *RoadSide Units*) qui sont stationnaires le long de la route. Les OBUs des différents véhicules forment un réseau ad hoc mobile (MANET), où un OBU est équipé d'appareils de communication, incluant au moins un appareil de communication sans fil à courte portée assurant la sécurité routière. Les OBUs et les RSUs peuvent être vus comme des nœuds mobiles et des nœuds statiques d'un réseau ad hoc, respectivement. Un RSU peut être attaché à un réseau d'infrastructure, qui alternativement peut être relié à l'Internet. Les RSUs peuvent également communiquer entre eux directement ou par l'intermédiaire de la connexion multi-hop entre deux nœuds, leur rôle primaire est l'amélioration de la sécurité routière, en exécutant des applications spéciales et en envoyant, recevant ou en expédiant des données dans le domaine ad-hoc.

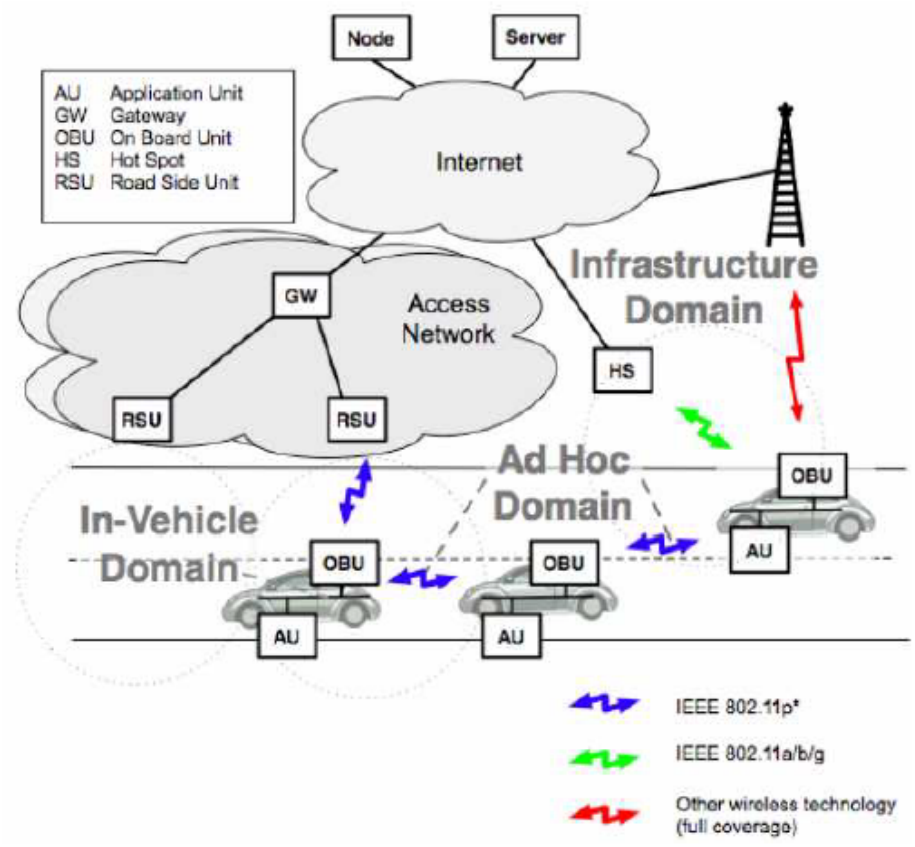


Figure 2.Architecture ad-hoc hybride C2C-CC [6]

Deux types d'infrastructure d'accès au domaine existent : RSU et hot spot. Les RSUs permettent aux OBUs d'accéder à l'infrastructure et, par conséquent, d'être relié à l'Internet. Les OBUs peuvent également accéder à Internet par l'intermédiaire des hot spot (Wifi hot spot). En absence des RSUs et de hot spot, les OBUs peuvent utiliser les possibilités de communication des réseaux de radio cellulaires (GSM, GPRS, UMTS, WiMax, et 4G) si elles sont intégrées dans les OBU.

3.2 Scénarios possibles de déploiement pour les réseaux véhiculaires

En faisant recours à l'architecture de référence de C2C-CC ainsi qu'aux avancés hétérogènes en technologies des communications, les réseaux véhiculaires ont potentiellement deux principaux scénarios de communication: scénario de communication de la Car-to-Car (C2C) et scénario de communication de la Car-to-Infrastructure (C2I). [8]

Ces scénarios de communication permettent un certain nombre d'options de déploiement pour les réseaux véhiculaires. Le déploiement de réseau véhiculaire peut être intégré dans les

hot spot sans fil le long de la route. De tels hot spot peuvent être actionnés individuellement à la maison ou au bureau ou par des fournisseurs de service sans fil d'Internet. D'une part, le déploiement de réseau véhiculaire peut être intégré dans les systèmes cellulaires existants. D'autre part, Les véhicules peuvent eux même communiquer avec d'autres véhicules directement sans infrastructure de communication, où les véhicules peuvent coopérer et expédier l'information au nom de l'un l'autre. Nous notons qu'une combinaison de ces cas de déploiement est également possible.[8]

D'ailleurs, la future architecture pour les systèmes de transport intelligents (ITS) considère les véhicules en tant que nœuds actifs responsables de collecter et d'expédier les informations critiques. En conséquence, la coexistence de réseau véhiculaire et de réseau de capteurs aurait lieu potentiellement, là où les véhicules pourraient rassembler et traiter l'information à l'aide de capteurs intelligents et échanger l'information avec d'autres nœuds (fixes ou mobile) dans un système de télécommunication mondiale.[10]

3.3 Caractéristiques des réseaux véhiculaires

Un réseau ad hoc de véhicules possède des caractéristiques particulières par rapport à un réseau ad hoc classique. En plus du fait que les nœuds sont des routeurs pour les autres nœuds, que le réseau soit auto-organisé, les réseaux ad hoc de véhicules ont les propriétés suivantes [1] :

Environnement varié : Les réseaux ad hoc de véhicules peuvent fonctionner dans trois types d'environnement : autoroutier, urbain, rural. Les caractéristiques de ces environnements seront détaillées par la suite.

Forte dynamique de la topologie : La vitesse élevée du déplacement des véhicules entraine des changements de topologie fréquents du réseau ad hoc. En effet, si on considère deux véhicules roulant en sens opposé à 25 m/s (90 km/h), avec une couverture radio de 250m, alors la durée de la communication directe entre les véhicules est seulement de 10 secondes.

Faible connectivité du réseau : La densité d'un réseau de véhicules est très variable. Une forte densité de véhicules permet au réseau d'être connexe, et donc il existe toujours un chemin entre deux nœuds qui souhaitent communiquer. A l'inverse, une faible densité de véhicules a pour conséquence un taux élevé de ruptures de communications, un délai

d'acheminement plus long si le véhicule conserve le paquet, voire une impossibilité pour deux véhicules de communiquer.

Énergie et stockage suffisant : Un véhicule produit sa propre énergie électrique en roulant et possède une batterie de grande capacité comparativement à l'énergie nécessaire à un système électronique de communication. Cette batterie permet de stocker cette énergie quand le véhicule est à l'arrêt.

Mobilité prévisible : Un véhicule est contraint de suivre la trajectoire de la route. Son déplacement est alors prévisible, d'autant plus si le système possède un plan du réseau routier. Cette mobilité prévisible permet de prévoir la position d'un véhicule dans le futur.

Information de localisation disponible : Actuellement, la majorité des véhicules neufs possède un système de géo-localisation. Pour un coût quasiment nul, le système peut obtenir la position géographique du véhicule, ainsi que sa direction, s'il est en mouvement. Cette information est utilisée par les protocoles ad hoc géographiques mais également pour tous les services disponibles aux passagers où la localisation est indispensable.

3.4 Environnements routiers

Comme on l'a mentionné précédemment, nous recensons trois types d'environnement où fonctionnent les réseaux ad hoc de véhicules : autoroutier, urbain, rural.

Le contexte urbain a pour particularité une densité de véhicules importante. Mais la présence de nombreux bâtiments ou d'arbres perturbent la transmission radio et donc il n'existe pas toujours une communication en ligne directe entre deux véhicules séparés par une distance inférieure à la couverture radio. Dans les villes américaines, on remarque que le réseau routier est bien souvent très proche d'une grille.

Dans un contexte rural, le réseau routier est peu dense, et n'a pas de caractéristique géométrique particulière. Les obstacles sont moins présents qu'en environnement urbain. En France, la majorité du réseau routier est en contexte rural. Malheureusement, c'est le contexte le plus difficile : faible densité de véhicules, donc peu de connectivité, et couverture totale du réseau routier non envisageable.

Le contexte autoroutier est particulier. Le réseau autoroutier est par nature adapté à la circulation de véhicules à grande vitesse. Les conséquences sont une route proche d'une ligne, au moins deux voies de circulation par sens de circulation, terre-plein central, infrastructure liée à la sécurité renforcée. Les véhicules se déplacent à grande vitesse dans deux directions opposées. Le réseau de véhicules est alors un réseau fortement dynamique mais est considéré à une seule dimension, il s'apparente alors à une ligne.

4 Applications des réseaux véhiculaires

L'intégration d'une interface réseau, un récepteur de GPS, des capteurs et un ordinateur embarqué donne une bonne occasion d'établir un puissant système de *car-safety*, capable de recueillir, de traiter et de distribuer les informations. De nombreuses applications peuvent être déployées dans un réseau établi avec un tel véhicule équipé et une infrastructure appropriée. Généralement, du point de vue de la connectivité elles pourraient être divisées en quatre groupes principaux : *car-to-car traffic, car-to-infrastructure, car-to-home* et *routing based* applications. Ces applications sont soit *safety-related* soit *comfort-related* (commercial)[11].

4.1 Applications de sécurité (safety-related)

Elles peuvent être groupées en trois classes principales: assistance (navigation, évitement coopératif de collisions et changement de route), information (information de limitation de vitesse ou de domaine d'activités) et avertissement (obstacle ou avertissements d'état des routes). Ils exigent habituellement la communication directe due à leur nature critique. Une telle application serait dédiée aux avis d'urgence, par exemple alarmes de freinage de secours. En cas d'accident (l'événement de déclenchement des airbags), un avis est envoyé aux voitures suivantes. Cette information pourrait également être propagée par l'entraînement de voitures dans la direction opposée et être de ce fait donnée aux véhicules qui devraient se diriger à l'endroit de l'accident (police, pompiers…).

4.2 Applications de confort (comfort-related)

Le but général de ces applications est d'améliorer le confort des passagers et l'efficacité du trafic. Cela pourrait inclure la localisation du POI(*Points Of Interest*) la plus proche, l'information courante du trafic ou du climat. Toutes sortes d'applications, qui peuvent fonctionner sur la pile TCP/IP pourraient être appliquées ici, par exemple, des jeux sur Internet ou la transmission de messages instantanée. Une autre application est la réception des

données des véhicules commerciales et de l'infrastructure de bord de la route au sujet de leurs entreprises (*'wireless advertising '*). Les entreprises (centres commerciaux, restauration rapide, stations de service, hôtels) peuvent installer des passages stationnaires pour communiquer des données de vente à des clients potentiels passant près de ces passages. En outre, ces services pourraient être intégrés avec des paiements électroniques.

La caractéristique importante de ces applications est qu'ils ne devraient jamais interférer avec des applications de sécurité. Dans ce contexte, donner la priorité au trafic de sécurité et à l'utilisation de canaux physiques séparés est une solution fiable.

5 Technologies d'accès dans les VANETs

Malgré les efforts de recherches académiques et industriels continus sur les VANETs, beaucoup de défis de recherches sont apparus. Du point de vue réseau, la sécurité est l'un des défis les plus importants. Les applications de sûreté véhiculaire représentent le challenge principal pour les VANETs. Là où la sécurité des personnes est en jeu, il est naturellement essentiel de protéger les VANETs contre les abus. D'autre part, comme tous les autres réseaux sans fil, le protocole MAC devrait jouer un rôle essentiel dans les transmissions de paquet de manière assez efficace dans les VANETs, selon les conditions (QoS) de qualité du service des applications.

Le MAC contrôle l'accès des clients aux voies de communication dans les réseaux câblés, par exemple, l'accès multiple avec détection de porteuses et de collision est utilisé dans l'Ethernet d'IEEE 802.3. Le MAC des réseaux câblés peut à peine être directement appliqué aux réseaux sans fil, par exemple, IEEE 802.3 ne peut pas être utilisé dans les LAN sans fil (WLAN) à cause de la conception des émetteurs récepteurs de RF (radio fréquence) qui ne peuvent pas transmettre et recevoir simultanément sur le même canal [11]. D'ailleurs, en raison du manque de contrôle centralisé et des ressources sans fil limitées en largeur de bande, les nœuds des réseaux ad-hoc sans fil doivent se coordonner afin de partager le canal sans fil d'une manière équitable, efficace et distribuée. En plus, la collision et l'interférence dégradent sévèrement l'exécution du système de communication et l'alimentation électrique précieuse sur chaque mode mobile est ainsi gaspillée. Par conséquent cette coordination distribuée exige plus de fonctionnalités des protocoles de la couche MAC dans les réseaux mobiles ad-hoc.

5.1 Caractéristiques du MAC VANET

L'environnement d'application spécial des VANETs et les caractéristiques des nœuds mobiles (véhicules), font le scoop du MAC des VANETs en guise des spécificités suivantes :

1) Distribué et fortement adaptatif : Les véhicules bougent à des vitesses variables sous différentes situations de trafic et peuvent aussi atteindre de très grande vitesse. L'environnement de communication change constamment, en raison des constructions de routes. Le MAC du VANET devrait pouvoir ainsi adapter rapidement et collectivement son comportement, le tout en permettant la modification dynamique de l'environnement de communication. Par exemple, l'exécution du MAC devrait être adaptée à la modification fréquente et rapide de l'environnement. Et les nœuds voisins exécutant le même MAC devraient avoir une corrélation avec leur point de vue sur le statut et la modification de l'environnement.

2) Retard délimité: Un aspect important des applications VANET est la sûreté du transport. Les messages concernant la sûreté du transport doivent être fournis avant que les contre-mesures de sauvetage ne soient choisies et exercés. Autrement l'information devient inutile et hasardeuse même si elle vient juste de se produire. Les messages de sûreté dans les VANET ont ainsi une période de vie très courte, ce qui signifie qu'ils doivent être transmis avec succès pendant leurs temps de vie. Une condition de retard de 100 millisecondes est généralement déterminée pour la plupart des applications de sûreté [12].

3) Taux élevé de réception de paquet : Une délivrance réussie signifie que l'émetteur confirme que le paquet a été reçu par les récepteurs. Dans les VANETs, la livraison de données souffre de la collision et de l'interférence d'accès qui sont communs aux MANET traditionnel. En plus, Elle est affectée par la dégradation des canaux en raison de l'effet *Doppler* et du *multipath delay spread* [13,14,15]. Les contre-mesures pour remédier au bas taux de réception de paquet provoquent l'*overhead* qui a comme effet d'entraîner un plus long retard [16] et une circulation plus dense [17].

4) Trafic de *Broadcast* fréquent : Les paquets de *Broadcast* sont fortement utilisés dans les VANET [12]. Les modèles ad-hoc populaires de MAC, tels qu'IEEE 802.11, ne supportent pas un service de *Broadcast* fiable. Le MAC du VANET devrait traiter ce problème.

5.2 Technologies d'accès véhiculaires

A- Dedicated Short Range Communications (DSRC)

DSRC est un service de communication à moyenne portée qui supporte les applications de sécurité publiques et privées dans une bande licenciée d'ITS de 5.9 gigahertz (5.85-5.925 gigahertz) dans des environnements de communication véhicule-to-infrastructure (V2I) et véhicule-to-véhicule (V2V). DSRC est censé être un complément aux transmissions cellulaires en fournissant des taux de transfert de données très élevés dans des circonstances où la minimisation du temps de latence dans les liaisons et l'isolement des zones de communication relativement petites sont importants. [18]

Fonctionnement du DSRC

Les RSUs effectuent des *broadcast* de messages vers les OBUs, à une fréquence de 10 fois par seconde, contenant le type d'applications qu'il supporte et le canal. L'OBU écoute sur le canal 178, authentifie la signature numérique du RSU. Il exécute les applications de sûreté d'abord, s'il y en a, et commute ensuite le canal pour exécuter les applications de non-sûreté. Quand il n'y a plus d'applications à exécuter, l'OBU revient au canal de contrôle et écoute les émissions. [19]

B- Wireless Access in Vehicular Environments (WAVE)

WAVE est un mode de fonctionnement employé par les dispositifs IEEE 802.11 pour fonctionner sur la bande DSRC. Il est basé sur la série de standards IEEE P1609, qui définissent l'architecture, les modèles de communication, la structure de gestion, la sécurité et les dispositifs d'accès physique aux communications véhiculaires. Les composants architecturaux primaires sont les *Road-Side Units* (RSUs), les *OnBoard Units* (OBUs), et l'interface de WAVE. La série de standard IEEE P1609 comportent les normes suivantes, chacune d'elles est concernée par une zone spécifique.

→ IEEE P1609.1 [20] traite le gestionnaire de ressources WAVE. Il décrit les éléments clé de l'architecture du système WAVE et définit les flux de données et les ressources aussi bien que les formats de message de commande et les formats de stockage des données. Il spécifie également les types de dispositif qui peuvent être supportés par les OBUs.

→ IEEE P1609.2 [21] traite les Services de Sécurités pour les Applications et les Messages de contrôle. Il définit les formats de message et de traitement sécurisés ainsi que les circonstances d'utilisation des échanges de messages sécurisés.

→ IEEE P1609.3 [22] traite les Services réseau. Il définit les services de la couche réseau et transport, y compris l'adressage et le routage, à l'appui du *secure WAVE data exchange*. Il décrit également les *WAVE Short Messages* (WSM), fournissant une alternative efficace à l'IP *(WAVE-spécifique)* qui peuvent être directement supportés par les applications VANET. Il traite également le *Management Information Base* (MIB) pour la pile de protocole WAVE.

→ IEEE P1609.4 [23] décrit les améliorations du MAC 802.11 pour supporter WAVE.

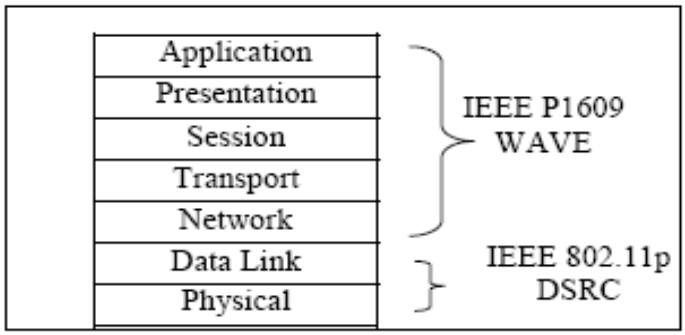


Figure 3.La pile de protocole WAVE

C- IEEE 802.11p

C- IEEE 802.11p

IEEE 802.11p [24] est un projet d'amendement aux standards IEEE 802.11 **[25]** pour ajouter du soutien au WAVE. Il définit des perfectionnements à 802.11 requis pour supporter les *Intelligent Transport Systems* (ITS). Il traite essentiellement les couches *data-link* et physiques du modèle OSI ; son but est de fournir des communications sans fil à courte distance entre :

- Bord de la route et unités radio mobiles

- Unités mobiles

- Unités portatives et mobiles

5.3 Histoire de standardisation du WAVE

Aux États-Unis, l'effort initial pour normaliser la technologie radio de DSRC a eu lieu dans le groupe de travail d'ASTM 2313 [26]. En 2004, cet effort migre au groupe de standardisation IEEE 802.11 comme une technologie radio de DSRC, il s'agissait essentiellement du standard IEEE 802.11a ajusté pour de basses opérations d'*overhead* dans le spectre du DSRC **[27]**. IEEE 802.11p n'est pas une norme autonome. Elle est destinée à modifier la norme globale d'IEEE 802.11 [28].

Une implication particulière de migrer de la norme DSRC vers l'IEEE 802.11 est que WAVE est entièrement destiné à servir comme norme internationale applicable à d'autres parties du monde aussi bien qu'aux États-Unis. La norme d'IEEE 802.11p est sensée :

Décrire les fonctions et les services exigés par les stations *WAVE-conformant* (compatible) pour fonctionner dans un environnement dynamiquement variable et échanger des messages sans devoir joindre un *BASIC Service Set* (BSS), comme c'est le cas dans les standards IEEE 802.11 traditionnels.

Définir les techniques de signalisation WAVE et les fonctions d'interfaces qui sont contrôlées par le MAC IEEE 802.11.

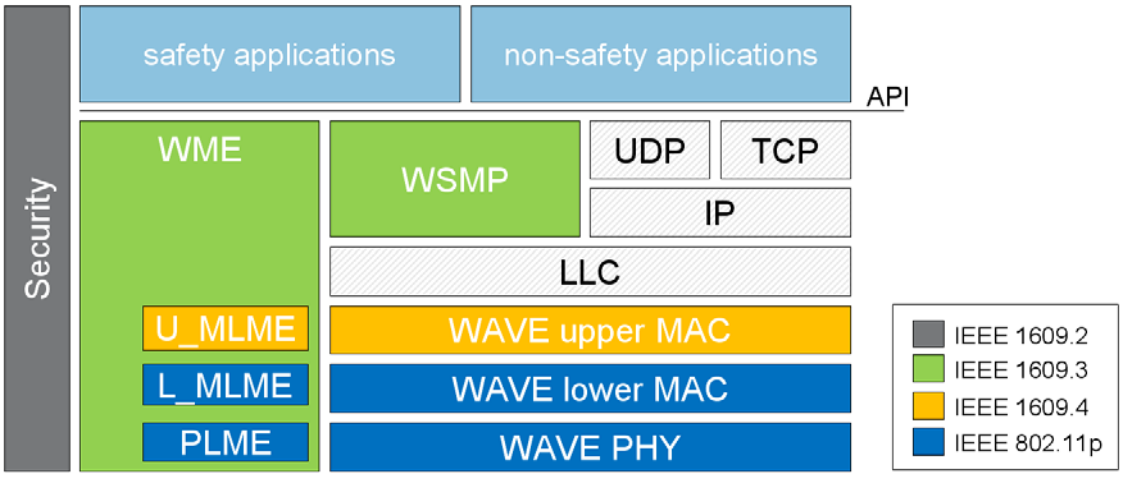


Figure 4.Standards de communication DSRC

Selon la Figure 4, IEEE 802.11p WAVE est seulement une partie d'un groupe de standards liés aux couches de protocoles pour des exécutions basées sur DSRC. La norme d'IEEE 802.11p est limitée par la vision IEEE 802.11, qui est strictement une norme d'un

niveau MAC et PHY censée fonctionner dans un canal logique simple. Toutes les connaissances et les complexités liées aux plans et aux concepts opérationnels du canal DSRC sont prises à soin par les couches supérieures du standard IEEE 1609**[15]**. La norme d'IEEE 1609.4 se repose bien sur l'IEEE 802.11p et permet l'exécution des couches supérieures à travers des canaux multiples, sans exiger la connaissance des paramètres de la couche PHY **[23]**.

5.4 Fonctionnement de WAVE et du protocole MAC

La norme WAVE utilise un concept multi canal qui peut être utilisé pour les messages sécuritaires et informationnels. Le spectre est rangé aux dessus de 5 gigahertz et compte sept canaux de largeur de bande de 10 mégahertz **[29]**. La bande est gratuite mais licenciée (restrictions d'utilisation et de technologie). Elle utilise un canal de contrôle (CCH : *Control CHannel*) - ch 178 - réservé aux applications de sûreté et au contrôle système. Les six autres canaux sont utilisés en tant que canaux de service (SCHs : *Service CHannels*), supportant principalement des applications appropriées de non-sûreté.

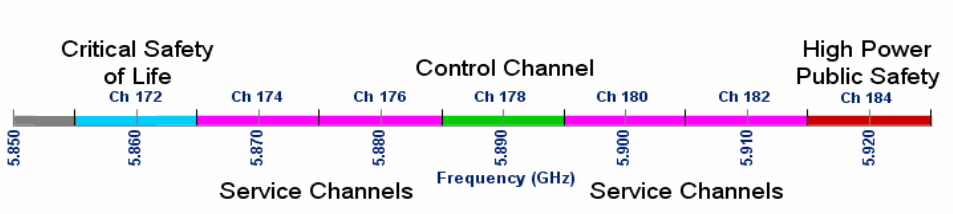


Figure 5.Les canaux de transmission WAVE

Sachant que l'intervalle de temps de communication des *WAVE STAtions* (STAs) est très limité, l'*overhead* devra être le plus bas possible. Ainsi, un *WAVE Basic Service Set* (WBSS) n'a pas besoin d'une authentification MAC et d'une association antérieure pour être autorisé à transmettre des données. Dans un WBSS, un utilisateur de WBSS doit seulement recevoir la trame d'annonce de service WBSS (WSA : *WBSS Service Announcement*) d'un fournisseur de WBSS avant de commencer les transmissions. Contrairement à l'IEEE 802.11 ordinaire, les trames *beacon* ne sont pas utilisées.

Par conséquent, pour réaliser la synchronisation une référence externe de temps comme le GPS(*Global Positioning System*) doit être employée. Pour les dispositifs radio simples, les canaux ne peuvent pas être utilisés simultanément. Par conséquent la coordination de l'accès

aux canaux doit être faite efficacement. Un schéma de synchronisation global de la coordination des canaux est en cours de développement par IEEE P1609.4. Le temps du canal est divisé en intervalles de synchronisation avec une longueur fixe de 100ms, se composant d'intervalle CCH ou d'intervalle SCH. Tous les dispositifs doivent basculer en mode CCH pendant les intervalles de CCH, où des trames prioritaires sont transmises. Pendant les intervalles de SCH, les dispositifs peuvent sur option commuter au mode SCH, qui sont utilisés pour les applications de non-sûreté **[30]**.

5.5 MAC P1609.4/IEEE 802.11p

Le protocole MAC de WAVE compte une couche MAC de base et une autre d'extension. Le MAC de base est le IEEE 802.11 DCF basés sur le CSMA/CA et utilise le *Request-To-Send/Clear-To-Send* (RTS/CTS) et le *Network Allocation Vector* (NAV). La couche de MAC d'extension utilise le mécanisme *Enhanced Distributed Channel Access* (EDCA) Initialement fourni par IEEE 802.11e.

Ensemble, le procédé d'accès de canal pour CCH et SCHs, comme nous pouvons voir dans la Figure 7, inclut le *Listen Before Talk* (LBT) et un *back-off* aléatoire, et permet l'accès prioritaire au canal. Le *back-off* se compose d'un délai d'attente aléatoire et d'un autre fixe. Le délai d'attente fixe est un certain nombre "slots" donné par le paramètre *Arbitration Inter-Frame Space* (AIFS). Le délai d'attente aléatoire est également un certain nombre de slots, mais ce facteur est tiré de la *Contention Window* (CW). La taille initiale de la CW est indiquée par le facteur CWmin. Chaque fois qu'une tentative de transmission échoue la taille de la CW est augmentée jusqu'à atteindre la taille indiquée par le paramètre CWmax **[31]**. La priorité des messages est fournie en utilisant différents paramètres d'accès du canal (AIFS et CW), avec quatre catégories d'accès (ACs) : *BacKground traffic (BK), Best Effort traffic (BE), VOice traffic (VO) and VIdeo traffic (VI).* La figure 6 affiche les configurations des paramètres par défaut utilisées dans IEEE 802.11p pour différents types de trafic d'application.

AC	CWmin	CWmax	AIFSN
VI	**3**	**7**	**2**
VO	**3**	**7**	**3**
BE	**7**	**225**	**6**
BK	**15**	**1023**	**9**

Figure 6.Configurations des paramètres pour les différentes catégories d'application selon IEEE 802.11p

Par conséquent, dans la couche de MAC il y a deux degrés de contention. D'abord le paquet rivalise intérieurement (basé sur ACs) pour choisir quel paquet sera transmis, et deuxièmement le paquet choisi rivalisera extérieurement en utilisant les paramètres d'accès au canal. Puisque c'est un mécanisme basé sur le conflit, l'exécution des applications sensibles au débit dans les scénarios fortement peuplés peut être améliorée.

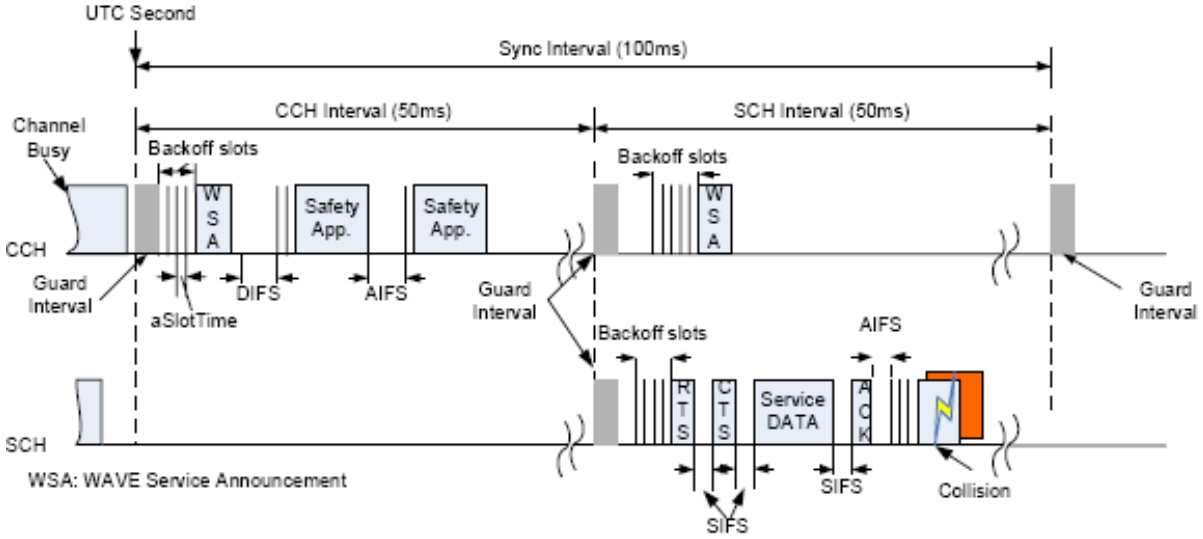


Figure 7.Processus d'accès au canal IEEE P1609.4/IEEE 802.11p MAC

De la figure 6, on peut voir clairement que les trafics de voix et de vidéo peuvent être servis avec une haute priorité en sélectionnant des tailles de la fenêtre basses du *backoff* et un temps d'espace inter-frame plus court. En conséquence, le débit de ces types de trafic peut être augmenté en choisissant la petite fenêtre du *backoff* qui réduit le délai d'attente avant d'être serv1. Cependant, parfois le nombre de véhicules en transmission simultanés est grand dans l'environnement du réseau véhiculaire, et par conséquent les nœuds deviennent fortement agressifs ce qui mènera à un bas débit et de forte chance de collision. En d'autres termes, un véhicule devrait augmenter la longueur des intervalles de temps de *backoff* plutôt qu'utiliser CWmin = 3 et CWmax = 7 quand il existe d'autres nœuds contractuels **[30]**.

6 Projets existants

Plusieurs projets de recherches au niveau national ou international ont été lancés: aux USA avec VII, en Europe avec CVIS, au Japon avec SmartWay.

6.1 USA: Vehicle-Infrastructure Integration (VII)

Le but du projet *Vehicle-Infrastructure Integration* (VII) du département *États-Unien* des Transports est de fournir des équipements de communication interopérables entre les véhicules sur une route, équipés d'un équipement embarqué spécifique ; et une infrastructure le long de la route pour le développement de systèmes permettant d'améliorer la sécurité, l'efficacité et le confort des réseaux de transport. Il est basé sur l'utilisation de la norme IEEE 802.11p, développée conjointement avec le projet. VII relie différents acteurs concernés : industries, autorité des transports et organisations professionnelles. Le projet a trois priorités : évaluation du modèle économique, validation des technologies utilisées et développement de structures légales et politiques permettant un succès sur le long terme. Le projet prévoit d'équiper 15 millions de nouveaux véhicules par an et de couvrir 70 % de toutes les intersections à feux tricolores dans 454 zones urbaines dans un rayon de 2 minutes de temps de trajet du centre ville. Plusieurs déploiements ont déjà été réalisés en Floride, Michigan et Californie, permettant d'évaluer les technologies utilisées.

6.2 Europe: European Commission's Cooperative Vehicle-Infrastructure System (CVIS)

Les objectifs de CVIS sont :

– d'unifier les solutions techniques permettant aux véhicules et aux infrastructures de communiquer entre-eux de manière continue et transparente en utilisant différents média de communications disponibles,

– définir et valider une architecture ouverte, un concept pour des systèmes coopératifs et développer des composants communs supportant le modèle de coopération dans les applications de la vie courante,

– d'étudier l'intégration avec les utilisateurs, la sécurité des données, la protection de la vie privée, l'ouverture et l'interopérabilité des systèmes, les risques et responsabilités, les politiques publiques nécessaires, le modèle économique et le plan de déploiement.

Le projet Calm fait partie intégrante du projet CVIS. *SafeSpot* est également un projet européen orienté sur la conception de système coopératif pour la sécurité sur route. Plusieurs projets Allemands existent également : Now, Car2Car et FleetNet. Lara (La Route Automatisée) est un projet Français lancé récemment, plutôt orienté pour l'assistance à la conduite, voire le pilotage totalement automatisé des véhicules.

6.3 Japon: SmartWay

Le principal apport de SmartWay, basé au *Tokyo Metropolitan Expressway* est la réalisation de démonstrations utilisant un réseau de véhicules comme l'assistance à la conduite en temps réel, la délivrance de messages à l'intérieur du véhicule ou la communication à deux sens pour le paiement électronique. Au Japon, tous les nouveaux véhicules sont équipés d'un système de navigation intégré et d'un composant appelé *Vehicle Information and Communication System* VICS. Les informations agrégées par le *Japan Road Traffic Information Center* sont transmises à cet équipement par liaison optique (IR) ou par radio sur la fréquence des 2.4 Ghz. *SmartWay* a également mis en avant que les 75% de véhicules au Japon utilisant les autoroutes, sont équipés d'un *Electronic Toll Collection* (ETC) dont le fonctionnement est basé sur la bande de fréquence dsrc 5.8 Ghz. Cet équipement sert au télépéage.

7 Conclusion

Bien que les réseaux véhiculaires diffèrent quelque peu des réseaux sans fil traditionnels, notamment par leur potentiel énergétique, il est intéressant de constater que ces réseaux n'ont pas encore pu voir le jour sur le marché que dans des architectures à infrastructure. Cette tendance confirme la nécessité de s'atteler au développement des réseaux véhiculaires ad-hoc hybrides si l'on souhaite voir se matérialiser l'important potentiel applicatif des VANETs. En effet, les réseaux ad-hoc hybrides apparaissent comme une option crédible permettant de tirer parti concomitamment des avantages applicatifs, des caractéristiques des réseaux à infrastructure et des réseaux ad-hoc mais aussi d'engranger des gains de performance (e.g. débit, délai, optimisation de l'exploitation des ressources radio,…etc.) et des avantages de la rationalisation du déploiement des points d'accès ou des stations de base. Cette perspective que présentent les réseaux ad-hoc hybrides est de nature à assurer des débouchés commerciaux au concept des communications ad-hoc et ainsi garantir des déploiements substantiels. Les projets actuellement en cours sur les réseaux véhiculaires ne s'y trompent pas puisqu'ils s'inscrivent en grande majorité dans un contexte architectural ad-hoc hybride même si les applications commerciales les plus porteuses (1.e. applications non liées à la sécurité routière) sont encore très peu prises en compte. Les réseaux ad-hoc hybrides et en particulier leur déclinaison dans les réseaux véhiculaires restent donc un champ d'investigation prometteur appelant encore de nombreux investissements.

C'est dans ce contexte, que nous inscrivons les travaux de cette thèse dans une architecture des réseaux véhiculaires ad-hoc hybrides; architecture que nous instancions sur le plan technologique dans les réseaux véhiculaires DSRC (*Dedicated Short Range Communications*) [19] dont le standard en cours de développement IEEE 802.11p [32] spécifie les couches physique et liaison de données pour les communications véhiculaires. En nous situant ainsi dans des réseaux s'appuyant sur une technologie de type WLAN, nous anticipons une généralisation de ces derniers, portée par le coût relativement faible que promet l'opération des services sur ces réseaux par rapport aux réseaux de type WWAN (*e.g.* 3G). Il est par ailleurs attendu que la généralisation de ces réseaux soit aussi portée par des incitations publiques dans l'optique du renforcement de la sécurité routière. De plus, le concept ad-hoc hybride appliqué aux réseaux DSRC permet d'étendre la zone de couverture des services, de favoriser l'accès ubiquitaire à ces derniers et de réduire les coûts de déploiement de l'infrastructure fixe.

Dans le chapitre qui va suivre nous nous intéresserons à la sécurité dans les WLANs dans le contexte spécifique des réseaux véhiculaires.

Chapitre II

Sécurité des VANETs

1 Introduction

Les réseaux locaux sans fil IEEE 802.11 constituent de nos jours le standard des WLANs le plus largement déployé et utilisé à travers le monde. Les contextes d'utilisation de ces réseaux sont divers et vont principalement du cadre domestique, aux lieux publics (e.g. gares, hôtels, restaurants, etc.) à travers notamment des HotSpots, en passant par le cadre du travail. Poussés précipitamment sur le marché, les WLANs 802.11 n'ont pu intégrer des mécanismes de sécurité robustes qu'après la déferlante des attaques dont ces réseaux ont fait l'objet et la prise de conscience progressive de l'étendue des vulnérabilités dans leur conception initiale. Dans le contexte des réseaux véhiculaires où un consensus technologique semble se dégager autour du standard DSRC/IEEE 802.11p pour les déploiements à venir, il est urgent de tirer les leçons des premiers déploiements des WLANs et donc d'éviter que la sécurité ne soit pensée, une fois de plus, a posteriori.

La sécurité des réseaux véhiculaires est donc aujourd'hui un enjeu majeur dont il faut se saisir pour garantir la plus large adoption possible de ces réseaux aussi bien par les usagers de la route dont on attend l'utilisation que les opérateurs dont on attend le déploiement. Cette sécurité s'inscrit dans un contexte particulier marqué, comme souligné dans le chapitre précédent, par une forte dynamique des nœuds avec des pointes de vitesse pouvant atteindre 200 Km/h, une aggravation de l'instabilité de la propagation radio, une connectivité intermittente, une topologie dynamique mais contrainte par celle des routes, un réseau potentiellement très étendu appelant une administration nécessairement hétérogène, un potentiel énergétique important,… etc. En adjonction à cette liste non exhaustive, il faut certainement compter avec la nature des applications ou des services qui y sont opérés et qui sont susceptibles d'induire des exigences de sécurité différenciées à l'image de la dichotomie du modèle de communication induite par les applications. Tout ce contexte, pris dans sa globalité, crée pour la sécurité des réseaux véhiculaires de nombreuses possibilités d'investigation dont on ne peut pourtant pas encore dire, à l'aube des contributions actuelles, qu'elles soient pleinement explorées.

Dans ce chapitre nous nous investirons spécifiquement dans la sécurité des réseaux véhiculaires. Nous ouvrirons une parenthèse particulière sur la nature des services mis en œuvre dans ces réseaux avant de présenter des exemples d'attaques dans ces réseaux ainsi qu'une taxonomie correspondante. Il sera ensuite possible de mettre en évidence les

principales exigences et les défis de sécurité dans ces réseaux puis de présenter quelques contributions dans ce domaine.

2 Caractéristiques applicatives

La sécurité des réseaux véhiculaires est encore aujourd'hui un champ d'investigation assez peu exploré. Nous présentons dans cette section un panorama des attaques, des enjeux et des contributions dans ce domaine. Mais avant, il est utile de revenir, au-delà des caractéristiques générales des réseaux véhiculaires (en termes de mobilité, de connectivité et de topologie) dont nous avons fait état dans le Chapitre 1, sur les caractéristiques applicatives de ces réseaux.

Il est attendu que les réseaux véhiculaires soient l'instrument de la fourniture d'une très grande variété d'applications au rang desquelles on peut citer les alertes accident, ralentissement, déviation, travaux, intempéries, la conduite coopérative, la surveillance de l'état des véhicules, la localisation des véhicules, la gestion de flotte de véhicules, la messagerie instantanée, les jeux en réseau, l'accès Internet, les paiements automatiques,… etc. Nous répertorions ces applications suivant 2 grandes classes à savoir:

Les applications ITS (Intelligent Transportation System): Ce sont des applications liées à la sécurité routières (*i.e.* des applications impactant directement la sécurité des personnes et des biens) et visant à bâtir un système de transport intelligent. En d'autres termes, l'objectif ultime de ces applications est de réduire l'accidentologie routière et d'améliorer les conditions de circulation. Ces applications ont constitué dans les différents travaux de recherche et projets gouvernementaux menés à travers le monde, le fondement premier du concept de réseau véhiculaire. Du point de vue du modèle de communication, on relève dans les applications ITS, une prééminence très prononcée du *Broadcast* ou du *Multicast* sur les autres formes de communication. Cette prééminence est bien sûr liée à la nature même de ces applications où les transmissions se font presque toujours à l'intention de tous ou d'un groupe. Parmi les exemples d'applications cités précédemment, les applications dites ITS pourront être: la conduite coopérative, l'aide aux dépassements de véhicules, les alertes accident, ralentissement, déviation, travaux, intempéries, etc.

Les applications non-ITS: Ce sont des applications commerciales, de confort, de divertissement ou plus généralement toutes les autres applications ne faisant pas partie de

la catégorie des applications ITS. Si ces applications ont émergées conceptuellement à la suite des applications ITS, leur mise en œuvre concrète a en revanche pris le pas sur les premières. Cette avance est principalement due à la préexistence d'un certain nombre d'infrastructures sur lesquelles ces applications sont déployées et à leur potentiel commercial beaucoup plus important. S'agissant du modèle de communication de ces applications, il n'est pas surprenant, compte tenu de leur accès plus discriminant, de constater la prééminence de l'Unicast. Parmi les exemples d'applications donnés plus haut, les applications dites non-ITS sont: la surveillance de l'état des véhicules, la localisation des véhicules, la gestion de flotte de véhicules, la messagerie instantanée, les jeux en réseau, l'accès Internet, les paiements automatiques,… etc.

3 Attaques dans les réseaux véhiculaires

Nous donnons dans cette section, une classification générique des attaques recensées ou à venir dans les réseaux véhiculaires. Nous illustrons ensuite cette classification par quelques exemples concrets.

3.1 Taxonomie des attaques

La sécurisation des réseaux véhiculaires passe par la détermination d'une typologie des attaques dans ces réseaux. Compte tenu de la diversité des applications que l'on peut y opérer et de celle des environnements d'opération, il est aisé d'imaginer que ces réseaux feront l'objet de nombreuses attaques dont certaines pourront même relever du terrorisme. A partir de la taxonomie introduite dans [33], nous définissons quatre grandes déclinaisons pour toute attaque dans ces réseaux:

Interne ou Externe: Une attaque est dite interne si elle est instiguée par une entité identifiée comme légitime par les autres nœuds du réseau. De manière courante, une entité sera déclarée légitime si elle est authentifiée dans le réseau. Les attaques internes font partie des attaques les plus dangereuses puisque l'attaquant est injustement considéré comme étant de confiance et a généralement accès aux services du réseau. Une attaque externe est, quant à elle, menée par une entité a priori considérée et reconnue comme illégitime. L'attaquant dans ce cas n'est généralement pas authentifié dans le réseau et n'a pas accès aux services de ce réseau. Il est donc de ce fait limité dans la diversité des attaques qu'il peut entreprendre.

Intentionnelle ou Non intentionnelle: Une attaque est dite intentionnelle si elle est instiguée par une entité malveillante visant délibérément à remettre en cause le bon fonctionnement du réseau. Ce type d'attaque est à distinguer d'une attaque non intentionnelle ou involontaire qui peut par exemple être le fait d'une erreur de transmission radio ou d'une erreur protocolaire dans le réseau.

Active ou Passive: Une attaque est dite active lorsque l'attaquant injecte, modifie ou supprime du trafic dans le réseau. A contrario, dans une attaque passive, l'attaquant ne fait qu'écouter et collecter le trafic pour une éventuelle utilisation malveillante ultérieure.

Indépendante ou Coordonnée: Une attaque est dite indépendante lorsqu'elle est menée de manière isolée par un seul attaquant. Elle est en revanche dite coordonnée lorsque plusieurs attaquants partageant la même intention se concertent pour la mener.

3.2 Exemples d'attaques

En raison de l'impossibilité d'envisager toutes les attaques possibles dans les réseaux véhiculaires, nous nous limitons ici à la présentation et à la déclinaison dans la taxonomie introduite plus haut, de quelques exemples parmi les plus significatifs:

Attaque sur l'intimité numérique: Dans cette attaque, l'entité malveillante essaie d'obtenir l'identité ou des informations personnelles d'un utilisateur du réseau. Il peut également s'agir pour l'attaquant de tracer l'activité et les déplacements de cet utilisateur. Pour identifier et tracer une victime, l'attaquant peut utiliser toute chaîne de caractères identificatrice dont la récurrence est constatée dans les échanges de la victime. Cette chaîne de caractères peut être une adresse IP, une adresse MAC, des informations d'identification d'un certificat,… etc. Au-delà des chaînes de caractères, l'empreinte radio de la victime peut également être utilisée: on parle alors d'attaque de la couche physique. La Figure 8 illustre une attaque sur l'intimité numérique et en particulier une identification non-autorisée. D'après la taxonomie des attaques qui a été définie, cette attaque peut être Interne ou Externe, Intentionnelle, Passive et Indépendante.

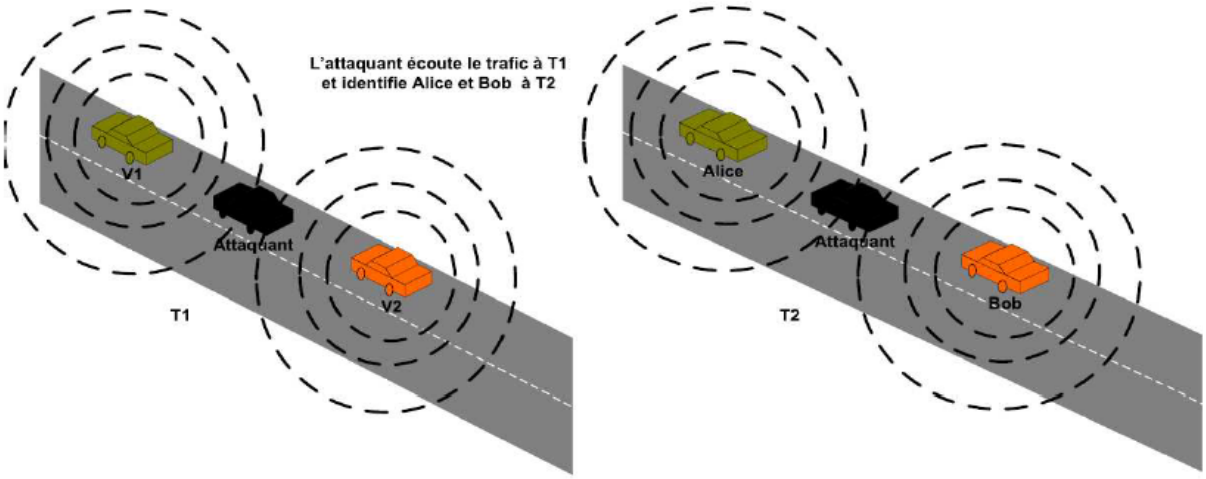


Figure 8. Identification non autorisée

Attaque sur la cohérence de l'information: Dans cette attaque, l'entité malveillante porte atteinte à la cohérence des informations acheminées dans le réseau en les modifiant ou en injectant des informations erronées. L'intention de l'attaquant peut être d'altérer la perception qu'ont ses victimes, de sa position, de sa vitesse, de sa direction, et plus généralement des conditions de circulation. Ce faisant, l'attaquant peut par exemple provoquer un changement d'itinéraire de ses victimes. Les Figures 11 et 12 illustrent ce cas de figure. Sur la Figure 9 un attaquant diffuse des informations de trafic erronées et sur la Figure 12 des attaquants indiquent de fausses données de localisation amenant les victimes à admettre l'existence d'un bouchon qui en réalité n'existe pas. L'attaque de la Figure 9 est Interne ou Externe, Intentionnelle, Active et Indépendante alors que celle de la Figure 12 est Interne ou Externe, Intentionnelle, Active et Coordonnée.

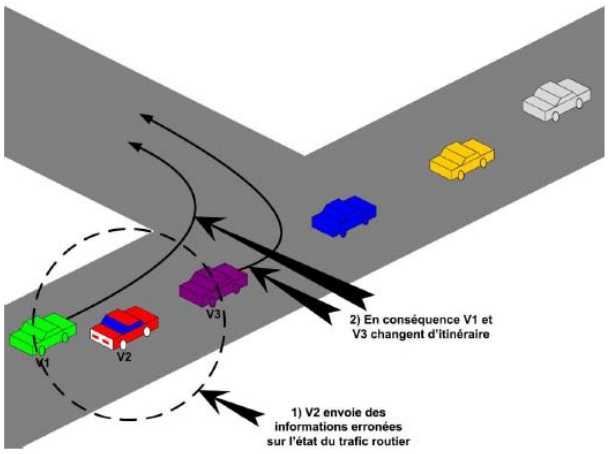


Figure 9. Injection d'informations de trafic erronées

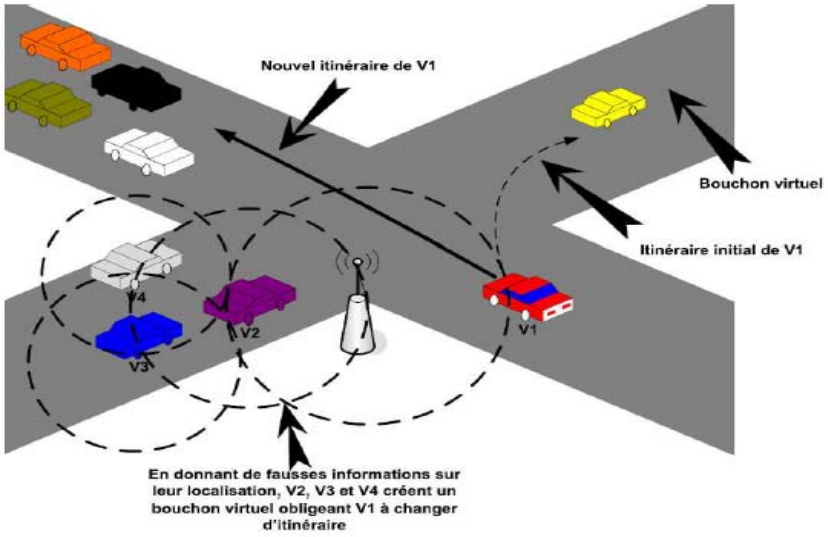


Figure 10. Fausses déclarations de localisation

Usurpation d'identité ou de rôle: Dans cette attaque, l'entité malveillante utilise une fausse identité ou de fausses lettres de créance pour se faire passer pour une entité légitime ou pour jouir des privilèges de cette dernière. La Figure 11 illustre un cas d'usurpation d'identité. L'attaque illustrée peut être Interne ou Externe, Intentionnelle, Active et Indépendante.

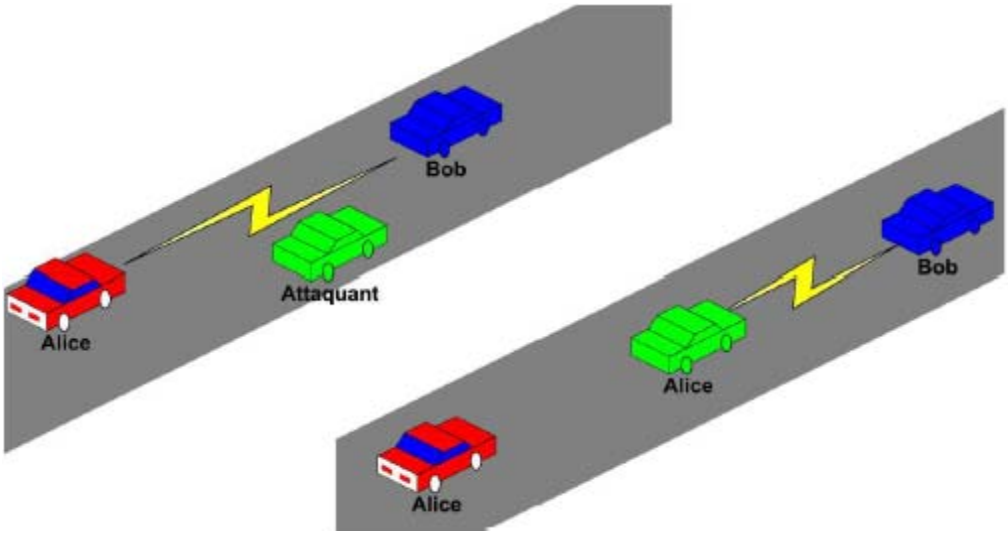


Figure 11. Usurpation d'identité

Déni de service: Dans ce type d'attaque, l'entité malveillante empêche l'accès normal aux services du réseau. Ce type d'attaque peut être monté en brouillant le canal radio, en surchargeant et en épuisant les ressources du réseau par des requêtes abondantes, en

exploitant la vulnérabilité des protocoles, en ayant une attitude non coopérative (*e.g.* refus de relayer des paquets),… etc. La Figure 12 illustre une attaque par déni de service aboutissant à une collision, où l'attaquant empêche l'échange de messages critiques entre des véhicules s'apprêtant à prendre une intersection. Cette attaque peut être Interne ou Externe, Intentionnelle, Active et Indépendante.

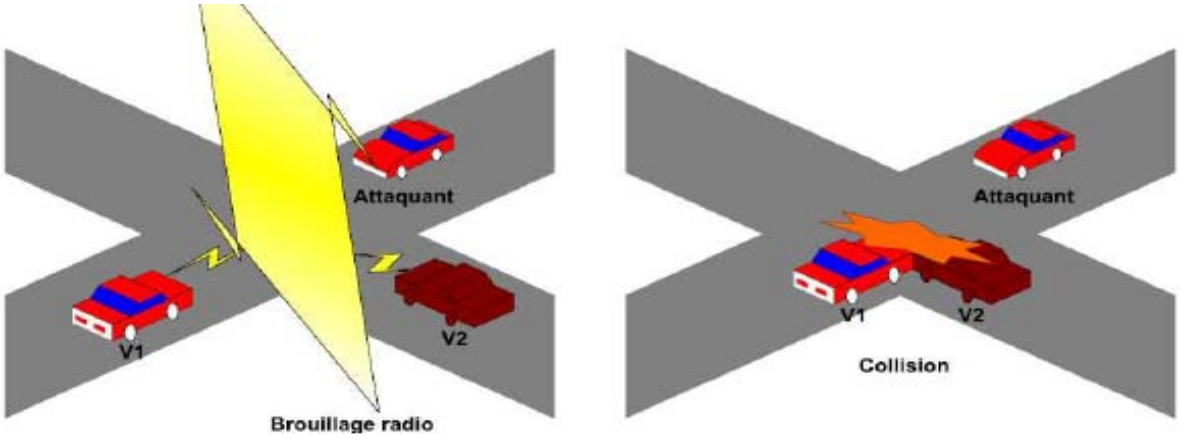


Figure 12. Déni de service par brouillage du canal radio

Ecoute du réseau: Dans cette attaque, l'entité malveillante collecte les données transmises dans le réseau afin d'en extraire une information dont elle pourrait tirer profit. La Figure 13 illustre une telle attaque dans laquelle l'attaquant espionne une transaction commerciale, typiquement un paiement électronique, en vue d'en extraire un mot de passe. Cette attaque peut être Interne ou Externe, Intentionnelle, Passive et Indépendante.

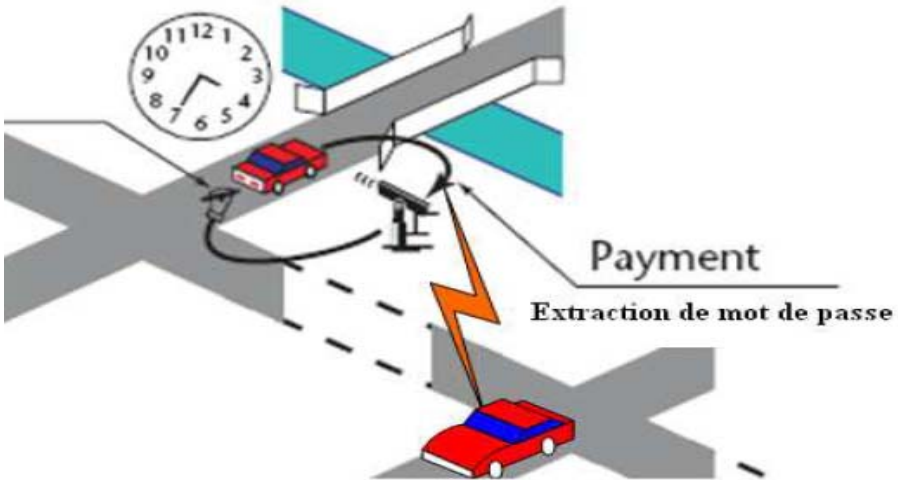


Figure 13. Extraction du mot de passe d'une transaction commerciale

3.3 Exigences et défis de sécurité

Nous présentons dans cette section les principales exigences de sécurité ainsi que les défis qui se posent à la sécurité dans un contexte d'opération des services ITS et non-ITS dans les réseaux véhiculaires. Ces exigences et ces défis sont définis pour être pris en compte aussi bien dans la conception architecturale de ces réseaux que dans la conception des protocoles de sécurité, des algorithmes cryptographiques et des implémentations matérielles et logicielles mis en œuvre dans ces réseaux. Au rang de ces exigences et défis de sécurité, on peut citer:

La confidentialité: Les applications ITS ne peuvent atteindre leur objectif de prévention et de réduction des accidents de la route que si le maximum - si ce n'est la totalité - des véhicules coopèrent étroitement à leur mise en œuvre. Il n'est donc, dans ce contexte, pas question de discriminer l'accès aux informations diffusées dans le réseau suivant que le véhicule est authentifié ou non. L'application du principe de confidentialité aux services ITS serait en effet contre-productive dans la mesure où les véhicules non authentifiés et donc ne pouvant déchiffrer ces services, feraient courir, du fait de leur non-information, un risque important d'accident aux véhicules authentifiés qui eux, peuvent déchiffrer ces services. A la différence donc des services non-ITS qui ont une vraie vocation commerciale et peuvent de ce fait appliquer la confidentialité pour assurer l'accès discriminé aux services, les services ITS doivent impérativement être accessibles à tous les véhicules du réseau qu'ils soient authentifiés ou non; et ce, dans l'intérêt de la sécurité de tous.

L'authentification de la source: Une des principales restrictions, dans un contexte d'opération des services ITS, réside dans l'obligation pour toute entité générant et diffusant des messages ITS, d'y adjoindre une preuve d'authenticité (*e.g.* une signature). Cette restriction est faite pour éviter que des entités malveillantes ou non authentifiées puissent générer et diffuser des messages ITS sans qu'il soit possible d'en vérifier l'authenticité. Cette exigence de sécurité ne vaut que pour les services ITS dans la mesure où ils sont les seuls qui soient quasi-exclusivement régis par le modèle de transmission en Broadcast ou en Multicast.

La non-répudiation: En raison de l'impact que peuvent avoir les applications ITS sur la sécurité des biens et des personnes, il est indispensable que toute entité générant ou modifiant des messages ITS soit toujours identifiable avec certitude. En d'autres termes, cette entité, après avoir émis un message, ne doit pas pouvoir ensuite nier cette action. Assurer la non-répudiation pour les services ITS, va donc éliminer toute possibilité pour une entité malveillante d'injecter des informations erronées et de causer éventuellement des accidents sans être confondue. S'agissant de la mise en œuvre de la non-répudiation, la signature numérique qui est majoritairement utilisée pour réaliser l'authentification entre des parties étrangères - l'une à l'autre – sans qu'il soit besoin de recourir à une entité de confiance en ligne, peut aussi la garantir. Pour ce faire, une signature doit être systématiquement ajoutée aux messages générés ou modifiés. A la différence des services ITS qui se distinguent par une exigence forte de non-répudiation, les services non-ITS peuvent s'en passer dans la plupart des cas; à l'exception notable de certains services non-ITS sensibles comme par exemple ceux impliquant des paiements.

L'authentification mutuelle, l'autorisation et le contrôle d'accès: La nature commerciale ou transactionnelle des services non-ITS fait qu'il est nécessaire, plus qu'ailleurs, d'y appliquer le principe de l'authentification mutuelle, que ce soit entre les véhicules et l'opérateur réseau ou entre les véhicules uniquement. C'est en effet de cette authentification mutuelle que découle la mise en œuvre de l'autorisation et du contrôle d'accès. Par ailleurs, l'authentification mutuelle dont l'objet est d'assurer que les entités en communication sont bien celles qu'elles prétendent être ou qu'elles ont bien les privilèges qu'elles prétendent détenir, va permettre de mettre en échec les attaques impliquant des usurpations de rôle ou d'identité. Une approche simple de mise en oeuvre de l'authentification peut consister à utiliser des clés de groupe symétriques (en anglais, *Symmetric group keys).*

Cette approche bien que facile à mettre en œuvre ne peut concerner malheureusement qu'un très petit nombre de véhicules placés sous la même autorité. Pour des déploiements à grande échelle, cette approche présente 2 inconvénients majeurs: (i) il suffit de compromettre un nœud pour compromettre la sécurité de tout le réseau et (ii) les nœuds ayant la clé peuvent se faire passer les uns pour les autres; ce qui empêche toute confidentialité et non-répudiation. Une autre approche d'authentification peut consister à utiliser des clés symétriques individuelles (en anglais, *Symmetric pairwise keys*) au lieu des

clés de groupe. Seulement cette approche souffre d'une non-scalabilité intrinsèque puisque le nombre de clés à gérer augmente de manière linéaire avec le nombre de noeuds du réseau.

Reste donc la cryptographie à clé publique qui, dans le contexte des réseaux véhiculaires, est seule à pouvoir permettre la réalisation de l'authentification tout en satisfaisant les exigences de scalabilité, de non-répudiation et de confidentialité. S'agissant du problème de la performance de la cryptographie à clé publique, il se pose moins dans le contexte des réseaux véhiculaires où des capacités matérielles importantes peuvent être attendues. De plus, avec les avancées réalisées ces dernières années dans le domaine de la cryptographie à clé publique, des choix avisés d'algorithmes peuvent permettre de réaliser des niveaux de performance élevés.

L'intimité numérique: La sensibilité des individus quant à la préservation de leur intimité allant grandissante, il faudra leur assurer dans le contexte des réseaux véhiculaires, aussi bien pour les services ITS et que les services non-ITS, une forme d'anonymat et de non-traçabilité. Si l'anonymat est un concept simple garantissant la non-identification, la non-traçabilité est quant à elle un concept plus étendu recoupant diverses notions. Par exemple, violer la non-traçabilité d'un utilisateur U peut consister à répondre aux questions suivantes: (i) U communique avec qui ? (ii) U envoie quoi ? (iii) U utilise quelle application ? (iv) U se trouve où ? Et où va-t-il ? etc. Dans tous les cas, il est évident que l'anonymat et la non-traçabilité ne peuvent qu'être partiels; et ce, dans la mesure où l'exigence de non-répudiation doit également être garantie. De plus, les obligations légales de traçabilité et d'interception qu'ont les opérateurs vis-à-vis de l'autorité judiciaire doivent également être honorées. En définitive, la préservation de l'intimité numérique (*i.e.* anonymat et non-traçabilité) des utilisateurs du réseau véhiculaire ne peut être applicable que vis-à-vis des autres utilisateurs du réseau ou plus généralement vis-à-vis de toute autre entité autre que les opérateurs de ces réseaux et l'autorité judiciaire. La préservation de l'intimité numérique dans des environnements ouverts et dynamiques comme ceux des réseaux véhiculaires est d'autant plus complexe, que des éventuelles solutions, qui n'existent pas encore à ce jour, doivent aborder le problème de manière holistique en intégrant cette préservation dans toutes les phases de conception de ces réseaux et à tous les niveaux de la pile protocolaire.

Les contraintes temps réel: Une des caractéristiques importantes des applications ITS est leur caractère temps réel et leur sensibilité aux délais. Il est par exemple montré dans [34] que le délai de transmission critique d'un message ITS est de l'ordre de *100* ms. Il importe donc que les mécanismes de sécurité mis en oeuvre dans les réseaux véhiculaires ne soient pas de nature à contrevenir à ces contraintes. Puisqu'il est attendu que les véhicules exécutant les services ITS, aient à faire plus de vérification de signatures que de génération de signatures, on peut par exemple choisir en priorité un crypto-système à clé publique rapide en vérification et n'ayant pas de très mauvaises performances en génération. Si on néglige les temps de calcul - ce que l'on peut légitimement faire puisque les véhicules sont supposées embarquer d'importantes ressources de calcul - alors on devra choisir en priorité le crypto-système à clé publique le plus compact et donc, induisant le moins de délai possible à la transmission. Dans tous les cas, le choix des mécanismes de sécurité à mettre en oeuvre doit être optimisé pour tenir compte du défi posé par les contraintes temps réel des services ITS.

La cohérence des données: Le caractère sensible des applications ITS impose que la cohérence des informations transmises dans le réseau soit garantie. Il est en effet souhaitable que des informations erronées, même lorsqu'elles proviennent d'entités régulièrement authentifiées dans le réseau, puissent être détectées. Il est tout à fait possible qu'une entité légitime puisse devenir malveillante en tentant d'influer sur le trafic routier ou en essayant de causer des accidents. Divers mécanismes permettant d'avoir une certaine assurance de la cohérence des informations ont été proposés. C'est par exemple le cas de ceux qui consistent à corréler l'information initiale avec les informations reçues d'autres sources en s'appuyant sur un système de réputation ou de recommandation. D'autres approches consistent à rechercher une explication plausible à l'information reçue à partir d'un modèle de connaissance du réseau [35].

L'intégrité: Cette exigence de sécurité apporte l'assurance que les données transmises ne sont pas altérées. Elle s'applique aussi bien aux services ITS qu'aux services non-ITS. Elle est en pratique mise en œuvre de manière concomitante avec l'authenticité.

La disponibilité: Les services du réseau véhiculaire, qu'il s'agisse des services ITS ou non-ITS, doivent être disponibles en toute circonstance pour les véhicules légitimes les sollicitant. Pour assurer cette continuité du service, le réseau véhiculaire doit pouvoir

résister le plus possible aux attaques de déni de service (*e.g.* brouillage du canal radio, saturation des ressources du réseau, comportements non coopératifs, etc.). La très grande diversité des attaques de déni de service, fait de la disponibilité un des enjeux de sécurité les plus complexes. Cela étant, l'exigence de disponibilité peut être mise en œuvre en implémentant par exemple des solutions contre les comportements non coopératifs (*e.g.* surveillance du réseau, système de réputation, etc.), contre les brouillages ou surcharges des canaux radio (*e.g.* basculement entre canaux, radio cognitive, etc.), et plus généralement contre toutes les attaques de type DoS les plus importantes.

La forte mobilité: La forte dynamique des noeuds des réseaux véhiculaires constitue un défi majeur à relever dans la conception des mécanismes de sécurité à mettre en oeuvre dans ces réseaux. Si les plateformes embarquées dans les véhicules peuvent être assimilées, du point de vue du potentiel énergétique et des capacités de calcul, aux stations fixes du réseau filaire, il reste qu'à la différence de ces stations fixes, elles sont plus contraintes dans leur connectivité et leur débit. C'est une des raisons pour lesquelles la plupart des protocoles de sécurité des réseaux filaires s'avèrent inadaptés dans les réseaux véhiculaires où il est davantage besoin de protocoles compacts dont l'exécution est rapide. Ainsi par exemple dans des protocoles comme SSL/TLS, DTLS, WTLS on préférera à la place du légendaire crypto-système RSA [36], l'utilisation du crypto-système NTRU [37] pour sa rapidité d'exécution et plus encore du crypto-système ECC (*Elliptic Curve Cryptography*) [38] pour son caractère compact. Dans les systèmes cryptographiques symétriques, on pourra également préférer aux protocoles DES et 3DES, le protocole AES pour sa relative rapidité et son niveau de sécurité supérieur. Des optimisations visant à accroitre la rapidité d'exécution peuvent également être faites par un choix avisé des implémentations matérielles ou logicielles des algorithmes cryptographiques. Il peut être aussi utile dans cette quête de rapidité, de définir des politiques de chiffrement qui soient fonction de la nature des données à chiffrer. Ainsi par exemple, il ne sera pas nécessaire de chiffrer tous les paquets d'un flux vidéo mais uniquement certains paquets essentiels au visionnage du flux. Adapter des protocoles de sécurité aux réseaux véhiculaires peut aussi se traduire par un choix avisé de la couche transport sur laquelle ces protocoles sont implémentés. Ainsi par exemple, au protocole TLS (*Transport Layer Security*) [39] dont l'implémentation est faite au-dessus de TCP (*Transmission Control Protocol*) - un protocole de transport fiable inadapté dans un contexte de transmission erratique -, il sera préféré le protocole DTLS

(*Datagram TLS*) [40] qui lui, est implémenté sur UDP (User Datagram Protocol), un protocole de transport plus tolérant aux pertes.

Alors que le défi de la forte mobilité s'impose à tous les services des réseaux véhiculaires, on relève toutefois que ces services appellent des exigences de sécurité plus ou moins différentes. La Figure 14 illustre cet état de fait en soulignant les similitudes et les différences entre les principales exigences et les principaux défis de sécurité dans les réseaux véhiculaires.

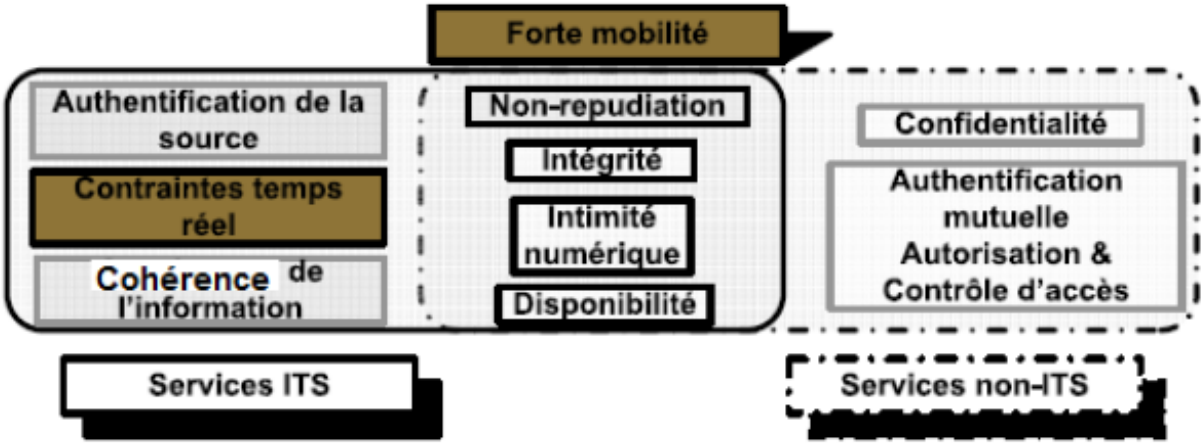


Figure 14. Principaux défis et exigences de sécurité des réseaux véhiculaires

4 Solutions et contributions

La sécurité spécifique des réseaux véhiculaires est un champ d'investigation relativement récent n'ayant pas encore fait l'objet de développement ou de standardisation de solutions complètes. Les principaux travaux dans ce domaine sont pour l'essentiel axés sur la sécurité des services ITS. Ce faisant, ils ne donnent pas de perspective globale intégrant à la fois la sécurité des services ITS et la sécurité des services non-ITS.

Dans [41], il est proposé une architecture de sécurité pour les services ITS dans les réseaux véhiculaires. Cette architecture suppose une architecture réseau dépourvue de toute infrastructure de routage et ne s'appuyant que sur des communications en Broadcast à sens unique des stations de base vers les véhicules. La sécurité des échanges est assurée par des signatures dont les clés sont fournies par une PKI (*Public Key Infrastructure*). De plus, les contraintes temps réels des services ITS sont prises en compte dans les mécanismes de sécurité mis en œuvre. Cette proposition souffre cependant de nombreuses faiblesses dont

son modèle de communication restrictif (*i.e.* le sens de la communication se fait uniquement des stations de base vers les véhicules), l'absence de prise en compte des problématiques d'intimité et des problématiques d'opérateurs (*e.g.* authentification avec l'opérateur ou le fournisseur de service, sécurité des services non-ITS, etc.). Toutes ces faiblesses ne permettent pas d'envisager la solution proposée dans une perspective de déploiement.

L'intimité numérique dans un contexte d'opération des services ITS est plus spécifiquement traitée dans [42]. Il y est introduit une métrique pour l'intimité appelée entropie d'anonymisation. Des recommandations pour l'utilisation de la cryptographie à clé publique ainsi que des recommandations de mécanismes de vérification de la localisation sont également faites. Le principal intérêt de ces travaux est d'introduire des mécanismes notamment des mécanismes d'anonymisation utilisables par extension dans la plupart des services des réseaux véhiculaires.

Dans [43] une architecture de sécurité comprenant une PKI, un système de détection d'intrusion (en anglais, *Intrusion Detection System "IDS"*) et une infrastructure de routage sécurisée, est proposé pour les VANETs. Bien que ces travaux s'efforcent de prendre en compte les exigences de sécurité des services ITS et des services non-ITS, les mécanismes proposés restent assez éloignés des problématiques opérateurs puisque n'envisageant prioritairement que des réseaux ad-hoc purs.

Les travaux réalisés dans [33] et [44] proposent un modèle d'analyse des attaques dans les réseaux véhiculaires ainsi qu'une architecture de sécurité pour les services ITS. Une PKI, des clés publiques anonymes et divers protocoles de révocation des certificats sont spécifiés. Ces contributions souffrent toutefois de ne pas s'inscrire dans un contexte de réseau opéré - ce qui occulte toute perspective de déploiement par des opérateurs - et plus encore, de ne pas prendre en compte toute la diversité des exigences de sécurité des réseaux véhiculaires dont notamment celles relatives aux services non-ITS.

Dans [45], une architecture de sécurité opérateur s'appuyant sur une PKI et le protocole EAP-Kerberos est étudiée. Cette architecture dont l'objet premier est de permettre la fourniture sécurisée de services non-ITS dans un réseau véhiculaire, ambitionne également de sécuriser les applications ITS. Une des insuffisances de cette contribution est l'absence de prise en compte de la mobilité dans la définition des

associations de sécurité avec les points d'accès (*i.e.* les services sont constamment interrompus puisque de nouvelles associations de sécurité doivent être définies chaque fois qu'un véhicule change de point d'accès).

Dans le cadre du projet NOW (Network On Wheels) [46], les études publiées dans [47], introduisent une architecture de sécurité pour les communications véhiculaires et en particulier les services ITS. Cette architecture intègre diverses composantes dont une composante pour l'enregistrement des véhicules, une composante pour la certification, une composante pour la révocation des certificats, une composante pour l'authentification, une composante pour la gestion de l'intimité et en particulier des pseudonymes, une composante pour la gestion de l'intégrité des données, une composante pour l'évaluation de la cohérence des informations, etc. Au-delà de cette contribution architecturale d'intégration, il est proposé dans [48] une infrastructure de routage géographique sécurisée s'appuyant sur une combinaison des schémas de signature saut par saut et de bout-en-bout pour assurer l'authentification, l'intégrité et la non-répudiation. Des mécanismes de corrélation visant à réduire l'impact de l'injection de fausses données dans le protocole de routage sont également proposés. De manière générale, les solutions développées dans le cadre de ces travaux et plus largement dans le cadre du projet NOW intéressent davantage des réseaux véhiculaires spontanés que des réseaux véhiculaires réalisés.

A l'image du projet NOW, des travaux sur la sécurité des réseaux véhiculaires sont également menés dans le consortium C2C-CC (*Car2Car Communication Consortium*) [49], le projet SEVECOM (*Secure VEhicular COMmunications*) [50] et le groupe de travail IEEE P1609.2 [51]. Tous les efforts menés dans ces différentes structures fondent la sécurité des réseaux véhiculaires sur l'utilisation des PKIs et des signatures numériques. Cependant, ils ne s'intéressent bien souvent qu'à la sécurité des applications ITS; le rôle de l'opérateur en tant que fournisseur de services et pilier de la confiance et de la sécurité, étant la plupart du temps ignoré.

5 Discussion

Arrivé au terme de la présentation de la sécurité dans les réseaux véhiculaires, il est intéressant de constater la diversité des exigences de sécurité qui s'y déclinent. On note, par exemple, la possibilité pour tous les véhicules du réseau d'accéder aux services ITS, alors que les services non-ITS ne seront généralement accessibles que des utilisateurs ayant

souscrit un abonnement. On constate malheureusement à l'aube des efforts actuels qu'une vision architecturale globale intégrant la sécurité des services ITS et celle des services non-ITS fait défaut. En effet, la plupart des contributions dans le domaine de la sécurité des réseaux véhiculaires ne traitent que de la sécurité des services ITS. Nous pensons pourtant que les services non-ITS, sans être à l'origine du concept des réseaux véhiculaires, constituent du fait de leur potentiel commercial, une des incitations les plus fortes au déploiement de ces réseaux.

En résumé, nous pensons que la sécurité des réseaux véhiculaires, au-delà de l'adaptation aux caractéristiques de mobilité, de connectivité, de topologie et d'échelle de ces réseaux, doit pouvoir se décliner en toute transparence aussi bien pour les services ITS que les services non-ITS. Dans cette déclinaison nous entendons donner une position centrale de mise en œuvre à l'opérateur afin de hâter l'avènement de ces réseaux. Pour se faire, il est, par exemple, nécessaire de définir de nouveaux mécanismes de protection contre les nœud malicieux et leurs impacts néfaste pour le fonctionnement du réseau tout en étant compatibles avec des réseaux hautement dynamiques s'appuyant sur une technologie de type WLAN comme la technologie 802.11p [32] - à partir desquels va être assurée la sécurité de l'ensemble des services du réseau et des nœud y participant. De tels mécanismes ne sont aujourd'hui répertoriés de façon sérieuse dans aucune des contributions que nous avons exploré.

6 Conclusion

Nous nous sommes intéressés dans ce chapitre à la sécurité des réseaux sans fil dans le contexte des réseaux véhiculaires. Nous avons montré à travers le prisme de leurs caractéristiques spécifiques - notamment leurs caractéristiques applicatives - et de quelques exemples d'attaques, que les exigences et les défis de sécurité ne s'y posaient pas nécessairement de la même manière selon que l'on traite les services ITS ou les services non-ITS. Ces différents défis et exigences de sécurité ont été passés en revue en mettant chaque fois en lumière quelques pistes susceptibles de concourir à leur mise en œuvre. S'agissant des contributions dans ce domaine, nous avons noté de manière générale l'absence d'un modèle de sécurité global intégrant les problématiques de sécurité de l'ensemble des services des réseaux véhiculaires (*i.e.* services ITS et services non-ITS). De plus, ces contributions ne s'inscrivent généralement pas dans un schéma architectural

d'opérateur de réseau ou de service et par conséquent n'adressent pas les problématiques associées. C'est ainsi, par exemple, que la problématique essentielle du traitement des comportements malicieux s'annonce inévitable pour combler les lacunes des solutions proposées afin de pouvoir mettre en œuvre le potentiel énorme des réseaux véhiculaires. Nous pensons pourtant qu'investir les opérateurs dans les prochaines générations des réseaux véhiculaires (*e.g.* DRSC/802.11p) en leur donnant la possibilité d'y opérer aussi bien des services purement commerciaux que des services ITS, tout en étant confiant d'être protégé contre les personnes ou les organismes malveillants visant à perturber le bon fonctionnement de leurs services et à espionner ou voler leurs informations personnelles est un élément incitatif susceptible d'accélérer la concrétisation et le déploiement de ces réseaux.

C'est donc dans ce contexte que nous allons définir dans le chapitre qui va suivre, l'ensemble des méthodes et solutions qui ont été proposées en guise d'outil de détection et de minimisation de l'ampleur de ces types d'attaques ainsi que l'élimination des nœuds ayant été considéré comme source de telles comportements atteignant le bon fonctionnement du réseau et la sécurité des utilisateurs.

Chapitre III

Détection Et Elimination Des Nœuds Malicieux Dans Un VANET

1 Introduction

Les initiatives récentes de recherches supportées par les gouvernements et les fabricants de voitures recherchent à améliorer la sûreté et l'efficacité des systèmes de transport. Les réseaux véhiculaires se trouvent au noyau de ces efforts. Les nœuds des réseaux véhiculaires, c.-à-d., véhicules et des unités d'infrastructure de bord de la route (RSUs) seront équipés de modules de détection, de traitement, et de communication sans fil. Les transmissions Véhicule-à-Véhicule (V2V) et Véhicule-à-infrastructure (V2I) permettront des applications de sûreté qui fournissent des avertissements au sujet des accidents, des états de trafic (par exemple, encombrement, urgence…etc) et d'autres événements.

L'intégration de mécanismes de sécurité dans le VANETs est cruciale pour leur déploiement: leurs fonctionnalités et services nombreuses peuvent être détournés ou corrompus, compromettant la sécurité des véhicules, les gestionnaires et les passagers, aussi bien que l'efficacité du système de transport. Un certain nombre de contributions de la recherche analysent les vulnérabilités [43,52], tracent les grandes lignes des composants, des conditions et des conceptions architecturales de bases [53], et proposent les mécanismes spécifiques [51,54,55,41,57].

La présence d'une autorité, que nous dénotons comme autorité de certification (CA), est exigée dans pratiquement tous les efforts de recherches concernés par la sécurité dans les VANETs. Les processus rigides de gestion d'identité pour les véhicules ont longtemps été mis en place. L'attribution de responsabilité continuera à être cruciale; et les mécanismes de contrôle d'accès seront nécessaires. Sans certificats et clés cryptographiques appropriés, les nœuds ne peuvent pas participer activement à vie du réseau. Néanmoins, la possession d'un certificat ne garantit pas que son support fournit des informations correctes: un nœud peut simplement injecter des données défectueuses (par exemple, fausses alertes, avertissements, coordonnées, …) tout en se conformant aux protocoles mis en application. La sauvegarde du système contre de tels nœuds défectueux ou compromis est déterminante pour sa robustesse. Par conséquent, le besoin d'expulsion des nœuds de conduite malicieuse s'impose de jour en jour. Une approche typique pour

réaliser ceci est la révocation des certificats du nœud. Une fois que cette tâche est exécutée, les messages de ces nœuds sont ignorés.

L'accès opportun à l'information de révocation est un problème particulièrement dur dans VANETs. L'infrastructure de bord de la route peut agir en tant que Gateway du CA au réseau, distribuant les listes de révocation de certificat (CRLs) [33]. Le manque d'une infrastructure omniprésente de bord de la route, particulièrement aux étapes préliminaires de déploiement, et les dimensions importantes d'un VANETs sont des obstacles à l'application des conventions traditionnelles de révocation de certificat. D'ailleurs, à moins qu'un nœud soit retiré pour des raisons administratives (par exemple, le propriétaire de véhicules n'a pas renouvelé son inscription), comment l'autorité peut-elle obtenir et valider les preuves suffisantes qu'un nœud est défectueux ou compromis? Ainsi, un défi supplémentaire est comment les nœuds honnêtes peuvent être protégés jusqu'à ce qu'ils obtiennent l'information de révocation concernant les nœuds malicieux.

Dans ce chapitre, nous allons essayer de voir toutes les contributions qui visent cet objectif de sécurité. Dans la première partie du chapitre, nous montrerons les travaux qui s'intéressent à la détection des fausses informations introduites dans un VANET, et à la détermination des nœuds sources de ces informations. Tandis que dans la deuxième partie du chapitre, nous nous intéresserons aux méthodes employées pour établir la culpabilité de ces nœuds et aux mécanismes déployés pour les exclure temporairement du réseau le temps que le CA les révoque. A la fin de chaque partie, nous discuterons les différents aspects des travaux proposés pour chaque processus. En conclusion, nous présentons les grandes lignes des potentielles améliorations de ces solutions.

2 Détection Des Nœuds Malicieux

Dans cette partie du chapitre, nous allons essayer de couvrir les différents travaux réalisés dans le domaine de la détection des nœuds malicieux jusqu'à aujourd'hui. Pour chaque travail présenté, l'environnement de travail ainsi que le fonctionnement de la solution proposée seront discutés.

2.1 Solution de Philippe Golle et al(2004)

Afin d'atteindre des buts d'exécution, il est largement convenu que les réseaux ad hoc véhiculaires (VANETs) doivent se fonder fortement sur la transmission de nœud-à-nœud,

et de ce fait tenir compte du trafic des données malveillantes. En même temps, l'accès facile à l'information fournie par les VANETs rend potentiellement difficile le maintient de sécurité et de la validation des données. Ce travail propose une approche générale pour évaluer la validité des données des VANETs. Dans cette approche, un nœud recherche les justificatifs possibles pour les données rassemblées en considérant que des nœuds malveillants peuvent être présents. Les explications qui sont compatibles au modèle VANET du nœud sont marquées (signées). Le nœud accepte alors les données marquées avec les niveaux de marquage les plus élevées. Les techniques utilisées pour produire et marquer les explications se fondent sur deux suppositions: (1) les nœuds peuvent détecter séparément la présence des nœuds dans leurs zones de couverture respectives, et (2) un argument de parcimonie reflète exactement le comportement d'adversaire dans un VANET.

2.1.1 Modèle d'environnement

Ce travail propose le modèle suivant d'un VANET: étant donné P un espace euclidien et ||P1, P2|| la distance euclidienne du point P1 à P2. Les événements et les nœuds sont définis comme suit:

Un événement E est une paire E = (D,f), où D est la donnée liée à l'événement et f la fonction de localisation, est une fonction continue f: T→P qui indique l'emplacement de l'événement pendant la durée $T \subseteq \mathrm{R}$ de la vie du nœud. La vie d'un événement peut être un point unique dans le temps T = {t} ou un intervalle de temps T = [t0,t1]. Les données liées à un événement peuvent être, par exemple, l'identité ou la vitesse du nœud à l'emplacement donné par f (T).

Un nœud est un triplet (*N*, f, p), où:

- $N \in \mathbb{N}$ est un nombre entier qui identifie de manière unique le nœud,
- f, la fonction de localisation du nœud, est une fonction continue f: T→P qui indique l'emplacement de l'événement pendant la durée $T \subseteq \mathrm{R}$ de la vie du nœud.
- $\rho \in \mathbb{R}^{+}$est le rayon d'observation du nœud,

Assertions (événements observés). Les nœuds peuvent observer les événements qui sont dans leurs rayons d'observation, et partager leurs observations entre eux. Nous appelons un évènement observé une assertion. L'assertion $\langle (D, f) \rangle o_i$ déclare que le nœud Oi (l'observateur) était témoin de l'évènement (D,f). La règle suivante explique les conditions dans lesquelles un nœud peut enregistrer une assertion. Etant donnée (N_i, f_i, ρ_i) un nœud, E=(D,f) un évènement, Ti la durée de vie du nœud Ni et T la vie de évènement E. Si $T \subseteq Ti$ alors pour tout $t \in T$, nous avons $||f(t) - f_i(t)|| \leq \rho_i$, alors le nœud Ni peut enregistrer l'assertion $\langle (D, f) \rangle_{N_i}$.

Les nœuds peuvent également partager les assertions entre eux. Tandis que dans la pratique le partage des assertions peut être limité par des considérations de la latence et de la largeur de la bande, ce travail assume une propagation idéale dans ce modèle. En d'autres termes, une assertion enregistrée par un nœud est immédiatement disponible à tous les autres nœuds. Cette supposition est justifiée par le fait qu'il considère seulement la propagation locale des assertions dans un voisinage géographique relativement petit (une assertion a moins de valeur pour les nœuds lointains de l'événement). Il dénote la base de données globale de toutes les assertions contribuées par tous les nœuds, par K.

Modèle du VANET. Un modèle du VANET spécifie quels événements ou ensembles d'événements sont possibles. Le modèle peut être basé sur les règles ou basé sur les propriétés statistiques des évènements. Formellement, prenant E comme ensemble de tous les ensembles d'évènements. Le modèle du VANET est une fonction M: → E {valide, incorrect}.A un ensemble des évènements {E1,…, En}∈ E est compatible au modèle du VANET si M (E1,…, En) = valide et contradictoire si M (E1,…, En) = invalide.

Le modèle adversarial est défini comme suit: il suppose que les nœuds malveillants peuvent enregistrer des évènements imprécis ou inexistants, c.-à-d. ils peuvent écrire des assertions fausses dans la base de données K.

Explication d'un ensemble d'événements. Etant donné H ⊆ N un ensemble d'hypothèses possibles. On suppose que l'ensemble H est divisé en un sous-ensemble H^+ d'hypothèses de validité (« correctes ») et d'un sous-ensemble H^- d'hypothèses d'invalidité (« malveillant » ou « défectueux »).

Etant donné K = {<E1> $_{O1}$,…, <En> $_{On}$ } un ensemble d'assertions. Une explication pour K au nœud N est une association pour chaque assertion appartenant à K avec une hypothèse:

$$\mathsf{Exp}_N(K) = \{\langle E_1\rangle_{O_1}^{h_1}, \ldots, \langle E_n\rangle_{O_n}^{h_n}\}$$

$h_i \in H$, tel que le sous-ensemble d'assertions étiquetées avec des hypothèses de validité est compatible au modèle du VANET. Formellement, on a:

$$\mathsf{Exp}_N^{\mathcal{H}^+}(K) = \{\langle E_i\rangle_{O_i}^{h_i} \in \mathsf{Exp}_N(K) \mid h_i \in \mathcal{H}^+\}$$

On a $\mathcal{M}(\mathsf{Exp}_N^{\mathcal{H}^+}(K)) = \mathsf{valid}$. [68]

Notez que l'explication $\mathsf{Exp}_N(K)$ est défini respectivement par rapport au nœud N, puisque différents nœuds peuvent assigner différentes hypothèses à de diverses assertions (considérez par exemple qu'un nœud est susceptible de considérer toujours ses propres assertions comme véridiques).

Ordonnancement des explications. Le modèle de l'adversaire spécifie également un ordre des explications. C'est habituellement un ordre total basé sur une notation des explications qui varieront selon les méthodes statistiques employées. Par exemple, l' *Occam's razor* utilisé pour cette approche note les explications sur la base de leurs simplicités.

Adressage des incohérences. Etant donné une collecte des données K incorrecte sous un modèle du VANET M, et une collection ordonnée d'explications de K, alors les données sont déclarées incorrect (une erreur est détectée) ou les erreurs en K sont corrigés en utilisant les assertions étiquetées par H^+ de la meilleure explication. S'il y a de meilleures explications multiples, leurs assertions étiquetées par H^+ peuvent être assemblées et un sous-ensemble de K peut être corrigé.

2.1.2 Exemple

Pour cet exemple, on suppose que les nœuds peuvent détecter l'emplacement précis de tous les voisins avec lesquels ils peuvent communiquer, et que la détection d'emplacement

est liée a la porté de transmission, de sorte que l'emplacement detecté d'un nœud puisse être associé à sa clé publique. La base de données K se compose des tuples:

$$K = \{\langle N_1, \vec{x}_1\rangle_{O_1}, \langle N_2, \vec{x}_2\rangle_{O_2}, \langle N_3, \vec{x}_3\rangle_{O_3}, \ldots\}, \qquad (1)$$

Là où l'assertion $A_i = \langle N_i, \vec{x}_i\rangle_{O_i}$ peut être interprétée comme suit «le nœud Oi prétend avoir observé le nœud Ni à l'emplacement $\vec{x}_i$ ». Sous un fonctionnement normal du nœud Oi il ne pourra pas observer des nœuds au delà d'un rayon ρ fixe, dans ce cas $\vec{x}_i$ aura la valeur « inaperçue » (Avec la supposition de multiple de voies de transmission qu'on a fait plus tôt, on peut éliminer ces tuples avec $\vec{x}_i$ « inaperçu » de la base des données et leurs valeurs est déduite de leur absence).

Les nœuds peuvent faire des assertions au sujet d'eux mêmes, dans ce cas Oi = Ni, et nous introduisons la notation L correspondant à ces assertions réflexives: $\langle N_i, \vec{x}_i\rangle_{N_i} \implies L(N_i) = \vec{x}_i$.

Le modèle du VANET M(K) pour cet exemple renvoie valide si les deux conditions géométriques suivantes toutes les deux sont valides (et invalides autrement):

1. K contient une assertion réflexive pour chaque nœud.

2. Chaque assertion non-réflexive $\langle N_i, \vec{x}_i\rangle_{O_i}$ en K est conforme à l'assertion réflexive pour le nœud Ni, c.-à-d., nous avons $\vec{x}_i = L(N_i)$ si $\|\vec{x}_i - L(O_i)\| \leq \rho$ et $\vec{x}_i = \text{unobserved}$ autrement.

S'il y a des nœuds malveillants K ne sera pas nécessairement compatible au model M. Dans cet exemple, une explication au nœud N, dénoté ExpN(K), consiste à étiqueter chaque assertion de K avec une des trois désignations, « truthful » si $t \in H^+$. Chaque tuple étiqueté,

$$\langle N_i, \vec{x}_i\rangle_{O_i}^{h_i} \qquad h_i \in \mathcal{H} = \{t, m, s\}$$

Dans ExpN (K) doit répondre aux critères suivants:

1. Si Oi = N et hi = t. en d'autres termes, les observations du nœud construisant l'explication sont considérés véridique.

2. Quand un observateur Oi a été étiqueté un spoof (hi = s), aucun des autres tuples ne doit faire des assertions au sujet d'Oi, tel que h:

$$\langle N_k = O_i, \vec{x}_k \rangle_{O_k}^{h_k}$$

Et ne devrait être étiqueté t à moins que le $\vec{x}_k = \text{unobserved}$.

On permet également à une explication d'inclure de nouveaux tuples étiquetés "*added*," $a \in H+$, avec un nouveau tuple permis pour chaque tuple réflexive qui a été étiqueté m. Le tuple ajouté fournira un $\vec{x}_i^*$ correct d'emplacement qui est compatible à toutes les autres observations véridiques de Ni en K:

$$\langle N_i, \vec{x}_i \rangle_{N_i}^{m} \implies \langle N_i, \vec{x}_i^* \rangle_{N_i}^{a}$$

Les assertions véridiques et ajoutées dans ExpN (k), pris ensemble, devraient être compatibles au modèle du VANET:

$$\mathcal{M}(\mathsf{Exp}_N^{\mathcal{H}^+}(K)) = \mathsf{valid}.$$

Pour terminer le modèle d'adversaire pour cet exemple, on marque des explications Exp_N selon le nombre d'observateurs distincts Oi qui reçoivent l'étiquette malveillante *m* sur un ou plusieurs de leurs tuples. L'explication Exp^*_N avec le nombre minimale de nœuds malveillants est considérée l'explication la plus simple et donc la plus plausible des données. S'il y a assez d'observations en K, alors les données dans Exp^*_N (K) identifiera les nœuds malveillants aussi bien que de fournir les emplacements corrects pour tous les nœuds, véridiques et malveillants (Parfois il peut y avoir plusieurs explications qui sont également probables, dans ce cas il peut encore être possible d'extraire quelques emplacements corrects à partir de l'intersection de ces explications).

Noter que le modèle de l'adversaire pour cet exemple ne fait aucune distinction basée sur le nombre d'assertions malveillantes par un observateur; une fois qu'une des assertions d'un observateur a été étiquetée malveillante, toutes ses assertions seront aussi étiquetées

malveillantes. Tandis qu'il est possible de construire des modèles plus raffinés qui assignent une certaine mesure à la complexité de la déception créée par un observateur malveillant (ou les modèles qui tiennent compte de quelques erreurs bénignes).

Nous notons également que le classement des explications ignore des étiquettes de spoof dans les explications, ainsi les explications meilleures auront plus de nœuds spoof et peu de nœuds malveillants. Cependant, les observations directes des nœuds véridiques (par le deuxième critère de l'écriture de labels décrit ci-dessus) limiteront le nombre de nœuds qui peuvent être étiquetés comme spoofs.

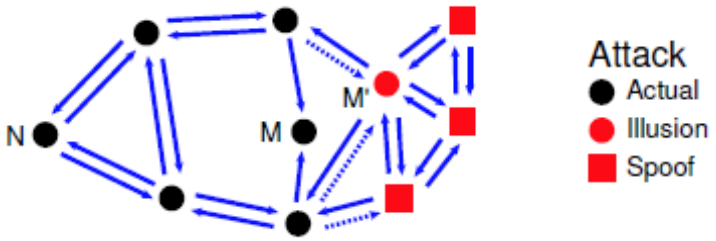


Figure 15. Un nœud malveillant simple M crée des spoofs pour supporter un emplacement faux M'

S'il y a seulement quelques nœuds malveillants alors la meilleure explication (ou les explications si plusieurs sont possibles) peut être calculée exhaustivement comme suit: les explications sont calculées en postulant un nombre restreint de nœuds malveillants, étiquetant toutes les assertions des nœuds malveillants postulés comme malveillantes, le traitement des assertions restantes comme arcs dans un graphique, la traversée courbe de l'observateur Oi au nœud Ni tant que le Ni n'est pas déjà étiqueté m, et marquer les nœuds atteint de cette façon comme véridiques. Tous les nœuds qui n'ont pas pu être atteint sont étiquetés comme spoof. Pas tous ces marquages seront compatibles au modèle M, mais en recherchant le minimum de nœuds malveillants d'abord, l'algorithme peut se terminer quand il a trouvé une ou plusieurs explications de même taille qui passent le test de consistance.

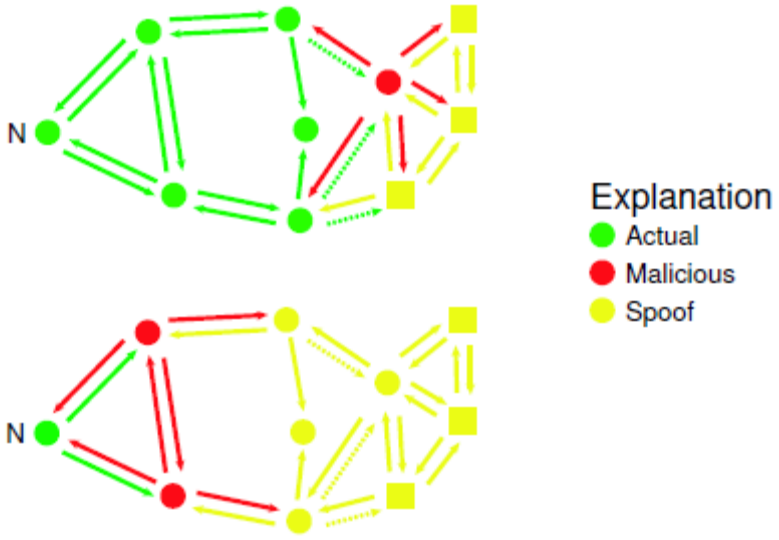


Figure 16. Deux (parmi beaucoup) explications possibles pour les observations contradictoires

La figure 15 affiche à la tentative d'un nœud malveillant simple de créer l'illusion qu'il est à un emplacement voisin. Le nœud malveillant a essayé d'augmenter l'évidence pour son emplacement illusoire en produisant de plusieurs nœuds spoof, affichée comme carrés, pour supporter son illusion. Les flèches bleues affichent les observations apparaissant dans la base de données partagée, et les flèches à tirer affichent les observations manquantes qui créeront des conflits de géométrie dans le modèle, et exposent de ce fait l'attaque. La figure 16 affiche deux explications pour les conflits. Noter que tandis que le nœud malveillant essayait de polariser les explications en ajoutant des nœuds spoof, ce modèle d'attaque particulier, peut favoriser la première explication (correcte) à la seconde parce qu'elle a peu de nœuds étiquetés malveillants. Cependant, noter que la capacité de trouver l'explication correcte est à la charge de la densité du graphique.

2.2 Solution de Bin Xiao et al(2006)

Les attaques Sybil ont été considérées comme une menace sérieuse de sécurité pour les réseaux ad-hoc et les réseaux de capteur. Elles peuvent également causer la détérioration des applications possibles des VANETs en créant une illusion de congestion du trafic. Cet auteur a présenté un régime de sécurité léger pour détecter et localiser les nœuds Sybil dans les VANETs, basé sur l'analyse de statistique de la distribution de la puissance du signal. Cette solution est une approche distribuée et localisée, dans laquelle chaque véhicule sur une route peut exécuter la détection des véhicules Sybil potentiels en vérifiant leurs positions réclamées. Il introduit d'abord un schéma *signal-strength-based* de

base pour la vérification de la position. Cependant, le schéma de base s'avère être inapproprié et vulnérable pour les attaques Sybil. Afin de compenser les faiblesses du schéma de base, l'auteur a proposé une technique pour empêcher des nœuds Sybils de se dissimuler l'un pour l'autre. Dans cette technique, les profils du trafic et le support des stations de base de bord de la route sont utilisés à son avantage. Il a alors, proposé deux algorithmes statistiques pour améliorer l'exactitude de la vérification de la position. Les algorithmes peuvent détecter des attaques Sybils potentielles en observant la distribution de la puissance du signal d'un nœud suspect pendant le temps.

2.2.1 Modèle d'environnement

Dans cette section, nous définissons le modèle d'attaque des attaques de Sybil et puis nous présentons les suppositions de système qui seraient appropriées pour des applications VANETs.

Modèle d'attaque

L'attaque de Sybil se rapporte à un nœud malveillant prenant d'une manière illégitime des identités multiples [61]. Dans les réseaux sans fil, les nœuds mobiles découvrent habituellement de nouveaux voisins en annonçant périodiquement les paquets de *beacon*, dans lesquels ils réclament leurs identités. Cependant, étant donné la nature de la communication sans fil, un nœud malveillant peut facilement réclamer des identités multiples sans être détecté. L'authentification d'identité n'aide pas à empêcher les attaques de Sybil dans les VANETs, puisqu'un nœud malveillant peut encore obtenir l'information supplémentaire d'identité par des moyens non techniques tels que le vol, ou simplement en empruntant à ses amis. Le but de détecter les attaques Sybil est de s'assurer que chaque nœud physique est lié avec seulement une identité légale.

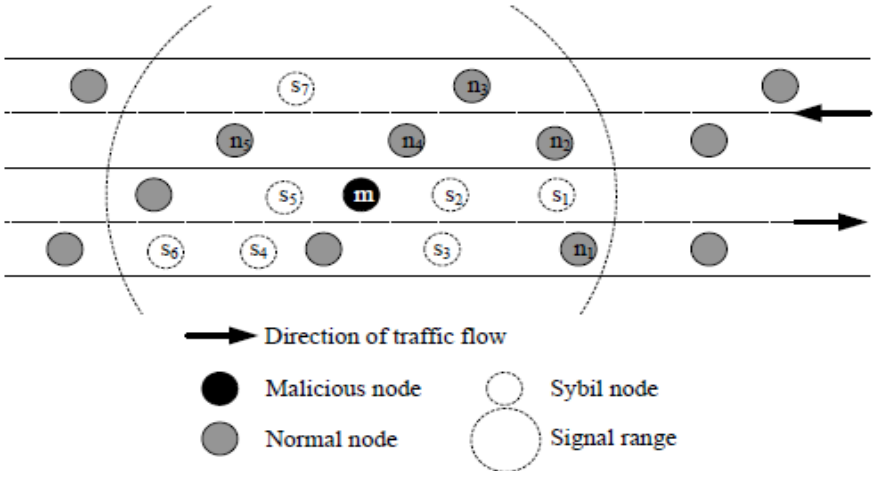


Figure 17. Exemple d'un VANET subissant une attaque sybil

Les attaques de Sybil peuvent encourir de grandes menaces de sécurité à VANETs. D'abord, les nœuds de Sybil peuvent entraîner une illusion de congestion du trafic. Un nœud malicieux peut convaincre les véhicules voisins qu'il y a un encombrement considérable en avant, de sorte qu'ils choisissent des routes alternatives et permettent au nœud un chemin incogestionné à sa destination [33]. En second lieu, les nœuds Sybil peuvent directement ou indirectement injecter des données fausses dans les réseaux, affectant considérablement l'uniformité des données du système. Par exemple, un VANET peut se fonder sur des véhicules votant pour produire un rapport sur l'état du trafic. Cependant, si certains des électeurs sont des véhicules Sybil, l'état peut être dévié selon les avantages du nœud malveillant. En conclusion, les nœuds Sybil peuvent lancer d'autres attaques DoS telles que les attaques de *jamming* et les attaques de suppression de message [52]. Les protocoles de diffusion de données pour les VANETs tel que [62] [63] [64] peut être facilement fendu par des attaques Sybil.

Suppositions

Les suppositions suivantes seraient appropriées pour une application VANET. D'abord, nous supposons qu'il y a une certaine quantité de véhicules voyageant indépendamment sur les routes et à la plupart d'entre eux sont dignes de confiance. Seulement quelques nœuds malicieux peuvent exécuter des attaques Sybil afin d'atteindre leurs buts malveillants. En second lieu, nous supposons que tous les véhicules, y compris les véhicules malveillants, sont équipés des mêmes modules radio, un pour chacun. Le module radio peut être basé sur n'importe quelle technique de transmission de RF (radio

fréquence) fournissant RSSI (*Received Signal Strength Indicator*), comme DSRC [18]. Troisièmement, nous supposons que chaque véhicule est équipé de dispositifs GPS, et les positions GPS sont censées être précises. En conclusion, nous supposons que des stations de base de bord de la route sont abondamment déployées le long des routes, et l'infrastructure d'authentification d'identité telle qu'ELP (*Electronic License Plate*) [42] a été déployé pour le réseau entier. L'authentification d'identité empêche un véhicule malveillant de fabriquer de façon illimitée de fausses identités. Naturellement, comme mentionnés ci-dessus, seule l'authentification d'identité ne peut pas empêcher les attaques Sybil. Veuillez noter également que les mécanismes principaux de détection de cette approche ne sont pas mis en application dans les stations de bord de la route, mais nous avons besoin de l'aide indirecte des stations de base.

2.2.2 Solution

Cette approche de vérification de position se fonde sur la surveillance de la puissance du signal des *beacons* périodiques. Pour la clarté de la description, on a définit trois catégories des rôles des nœuds: demandeur, témoin, et vérificateur. Chaque nœud jouera périodiquement tous ces rôles, mais à divers moments et pour différents buts.

1. Demandeur (Claimer)

Chaque nœud annonce périodiquement un message *beacon* à des intervalles t_b, afin d'effectuer la découverte du visionnage. Dans le message *beacon*, il réclame son identité et sa position telle que la position de GPS. À ce moment, nous nommons le nœud en tant que demandeur. Le but de cette approche est de vérifier sa position réclamée.

2. Témoin (Witness)

Tous les nœuds voisins, dans la portée du signal du demandeur, recevraient le message *beacon* précédent. Ils mesurent la puissance du signal et sauvegardent l'information voisine correspondante dans leurs mémoires. La prochaine fois qu'ils annoncent un message *beacon*, il joindra leur liste voisine, y compris les mesures de puissance du signal pour chaque *beacon* reçue, au message *beacon*. Nous nommons ces nœuds exécutant la mesure et enregistrant les mesures comme témoins.

3. Vérificateur (Verifier)

Après réception d'un message *beacon*, le nœud attend un intervalle de vérification t_v, pendant laquelle il rassemble assez de mesures de puissance de signal au sujet des messages précédents de *beacon* des témoins voisins. La période t_v peut être un peu plus longue que la période t_b, puisqu'après un autre intervalle de t_b, chaque témoin voisin devrait avoir annoncé un *beacon* contenant ses mesures collectées. Avec les mesures rassemblées, le nœud peut localement calculer une position estimée du demandeur, par exemple, par exécution de MMSE (*Minimum Mean-Square Error*) sur la puissance du signal rassemblée et un modèle radio prédéfini. Nous appelons un nœud exécutant la vérification un vérificateur.

Pour obtenir la position estimée, nous calculons d'abord la méthode des moyens carrés:

$$MSE(p) = \frac{\sum_{i=1}^{k}(S_r(w_i) - S_m(w_i, p))^2}{k}$$

p est la position potentielle du demandeur, k est le nombre de témoins, Sr est la puissance du signal reçue aux Wi témoin, Sm est la puissance du signal calculée aux Wi obtenus à partir du modèle de propagation radioélectrique. En variant p, nous pouvons réduire au minimum MSE et finalement obtenir le p estimée optimal de position.

Si la position estimée d'un demandeur est lointaine de sa position réclamée, nous considérons ce nœud comme un nœud suspect. Noter qu'en raison de la nature instable et irrégulière des RF (fréquences radio), nous ne pouvons pas encore affirmer, basé sur les résultats de ce calcul simple, qu'une attaque de Sybil se produit.

Nous prenons la figure 17 comme exemple. Le nœud s1, un demandeur, *broadcast* un *beacon*, réclamant son identité et sa position. Le nœud n1, un vérificateur, rassemble toutes les mesures de puissance du signal des témoins voisins qui ont reçu le *beacon*. Évidemment, la position estimée finale du s1 serait près de la position du nœud m, au lieu de la position s1 réclamée, car le nœud s1 et m sont physiquement le même véhicule.

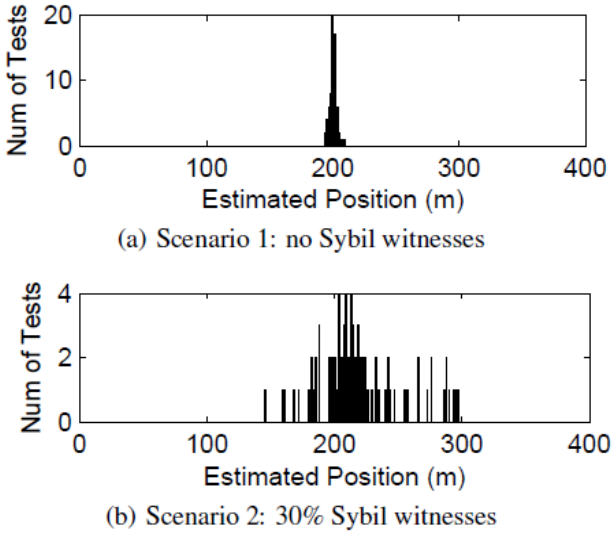


Figure 18. Distribution estimée de la position

D'après les simulations, on peut conclure que: D'abord, l'exactitude de la vérification de position s*ignal-strength-based* est plutôt limitée, avec une erreur d'environ 10m. C'est-à-dire, si deux véhicules physiques sont très proches entre eux, moins que 10m, nous ne pouvons pas nous assurer s'ils sont deux véhicules voisins ou deux véhicules Sybil. En second lieu, les témoins (*Witness*) Sybil pourraient en grande partie affecter les résultats de la vérification de position. Si tous les témoins indiquent la vraie puissance du signal mesuré, la position prévue est très proche de la position physique du demandeur.

2.2.3 Technique d'élimination des témoins Sybil

Avant qu'un vérificateur exécute le calcul de la position, il devrait retirer les mesures de puissance de signal provenant des témoins potentiellement Sybil autant que possible. L'auteur présente une technique pour s'assurer que chaque témoin soit un véhicule physique et non pas un véhicule Sybil. Dans cette technique, il profite pleinement des profils de trafic et du support de station de base dans l'environnement véhiculaire. Comme il a supposé précédemment, les stations de bord de la route, manipulées par les gouvernements, sont abondamment distribuées le long des routes. Sur la base de cette hypothèse, il établit les deux règles suivantes:

Règle 1. Une station de bord de la route émettrait une certification de position pour chaque véhicule passant près d'elle. La certification de position contient un *timestamp* et

des informations sur l'emplacement de la station de base et peuvent donc prouver la présence du véhicule près de la station de base à un certain moment.

Règle 2. Tous les témoins pour un demandeur se composent de véhicules dans le flux de trafic opposé du demandeur.

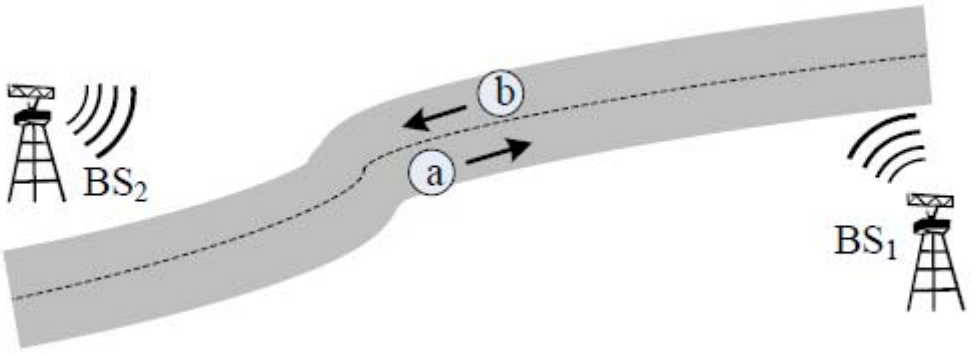


Figure 19. Un scénario avec des infrastructures de bord de la route

Avec la règle 1, nous pouvons nous assurer d'où vient un certain véhicule. En prenons la figure 19 comme exemple. Le nœud a peut obtenir un certificat de position de la station de base BS2, en passant par BS2, et le nœud b obtient également un autre certificat de BS1. Quand a et b se rencontrent, il est facile pour eux de montrer qu'ils viennent de sens inverses en permutant les certificats.

Avec la règle 2, nous pouvons nous assurer que chaque témoin dans le flux de trafic opposé est un véhicule physique au lieu d'un Sybil. L'exemple sur la figure 19 peut illustrer comment cette règle fonctionne. Le nœud malveillant m fabrique 7 nœuds Sybil, dans lesquels, s7 voyage dans le sens inverse et le reste dans le même sens que m. En essayant de vérifier les positions de s1,…, s6, on choisis seulement des témoins dans le flux de trafic opposé(de droit a gauche) tel que les nœuds n2,…, n5. Cependant, avec la règle 1, nous ignorerions le nœud s7, parce qu'il ne peut pas montrer qu'il vient de l'autre flux de trafic de la route, et réellement il ne le fait pas. De cette façon, on peut nous assurer que chaque témoin est un véhicule physique venant du sens inverse.

2.3 Solution de Jonathan Van Eenwyk (2007)

Une recherche significative a été conduite pour adapter les techniques cryptographiques standard aux réseaux véhiculaires [65,33,66]. Cependant, de telles méthodes empêchent seulement l'écoute illicite et le *tampering* des messages, mais n'empêchent pas l'opérateur d'un véhicule d'envoyer délibérément des données malveillantes. La révocation du certificat peut exclure des utilisateurs réseau, mais seulement après un retard significatif, après que de tels messages ont atteint leurs objectifs malicieux. Ainsi, il faut un certain mécanisme pour que les véhicules honnêtes et les stations de base identifient les données malveillantes en temps réel et prennent des mesures appropriées. Cette solution est une tentative de développer une technique implantable pour résoudre ce problème basé sur le travail existant de Philippe Golle et al décrit dans [35].

2.3.1 Modèle D'environnement

Le scénario cible de cette solution se compose de nœuds mobiles sous forme de véhicules communiquant en mode ad-hoc avec leurs voisins, d'autres nœuds et les stations de bord de la route. Les nœuds annoncent périodiquement leurs propres positions, vitesses et *heading*, ainsi que la position des autres nœuds récemment observés. En collectant ces informations, les nœuds honnêtes essayent d'identifier les tentatives des nœuds malveillants de compromettre les décisions des autres nœuds.

Comme dans [35], cette solution assume que les nœuds peuvent 1) associer une communication reçue à une localisation physique, et 2) distinguent les nœuds voisins individuellement. Ces conditions sont raisonnables, relativement à la maturité de la technologie de découverte d'emplacement [67]. En outre, ceci limite les effets des attaques Sybil, parce qu'un nœud simple peut essayer de communiquer en utilisant des identités multiples, mais les autres nœuds l'identifieront comme un nœud simple en raison de son emplacement inchangé. Cependant, parce que les nœuds déménagent, ce problème n'est pas supprimé et le système doit être robuste contre de telles attaques.

Au terme de ce travail, les nœuds produisent leur propre ensemble de paires de clés publiques/privées pour les utiliser en communiquant avec les nœuds voisins. Pour assurer l'intimité, les nœuds créent périodiquement de nouvelles clés. Bien qu'on permette à des nœuds de changer constamment les clés, nous supposons que la plupart des nœuds

honnêtes seront disposés à garder la même clé pour une certaine période; des secondes ou des minutes. Les nœuds malveillants peuvent changer les clés plus fréquemment, mais ceci a une utilité limité en raison des suppositions faites ci-dessus. En outre, les nœuds peuvent s'entendre avec d'autres nœuds en partageant des clés privées, mais de nouveau une telle attaque a une utilité minimal, parce que le partage d'une identité signifie simplement qu'elles pourraient tous être classifiées comme malveillantes.

Cette solution n'est pas limitée aux clés *self-generated*. Cependant, une infrastructure de clé publique comme celle décrite dans [35] pourrait également être utilisée. Dans ce scénario, chaque nœud a un grand nombre de clés assignées par une *Certificate Authority*. Bien qu'on assume que normalement ces clés soient protégées par un dispositif inaltérable, cette solution fonctionnerait même si ces clés ont été compromises. Dans ce cas, le nœud malveillant pourrait employer ces clés pour créer des nœuds Sybil, mais cette solution est conçue spécifiquement pour traiter de tels cas.

2.3.2 Solution

Les sections suivantes décrivent les divers aspects de cette solution et comment elles s'adaptent ensemble.

2.3.2.1 Base de données d'événement

En essayant de solidifier la technique modèle-basée de détection, la première tâche était de donner à chaque nœud sa propre base de données privées au lieu d'assumer une base de données globale d'accès instantané. Les événements qui sont enregistrés dans la base de données ont les zones affichées dans la figure 20.

Champs	Unités
Nœud detecté	**ID du nœud**
Nœud observateur	**ID du nœud**
Position	**X,Y**
Vitesse	**Mettre/second**
Heading	**Radians**
Dernière perception	**seconde**

Figure 20. format des événements enregistrés dans la BDD

Les unités spécifiques décrites sont seulement pour cette simulation. Dans un scénario déployé, les nœuds observateur et détectée sont susceptibles d'être identifiés par leurs clés publiques, la position pourrait être spécifiée en utilisant la longitude et la latitude, et ainsi de suite.

Le dernier champ est un *timestamp* pour chaque évènement. Ceci permet de comparer les observations d'un nœud particulier qui pourrait avoir été enregistré par n'importe qui à tout moment.

Périodiquement, les nœuds *broadcast* le contenu de leurs bases de données. Juste avant cela, les nœuds enregistrent leur position actuelle, vitesse et *heading* dans la base de données. En tant qu'élément de la construction du modèle (décrit plus tard), d'autres nœuds vérifieront cette information contre leurs propres observations (et celles des autres).

En recevant une émission d'un autre nœud, le nœud fusionne le contenu dans sa propre base de données. En outre, le nœud enregistre un nouvel événement pour l'émetteur. La position est basée sur la technologie de détection d'emplacement choisie, mais la vitesse et le *heading* sont laissés à zéro (c.-à-d., ne déménageant pas). C'est pour éviter la complexité ajoutée (et le coût) de sentir les technologies qui peuvent déterminer la vitesse et l'orientation des autres nœuds.

2.3.2.2 Paramètres de synchronisation

L'aspect critique de ce système établit une valeur pour un certain nombre de paramètres synchronisation suivant les indications de la figure 21.

Variable	Description	Valeur
T_B	**Intervalle de *broadcast***	**0-1 sec**
T_L	**Temps de vie d'un évènement**	**15 sec**
T_E	**Temps d'explication**	**5 sec**

Figure 21. Paramètres de synchronisation

Ces paramètres sont cruciaux pour assurer un équilibre entre l'exactitude et le temps système. Actuellement, les événements expirent après 15 secondes. Ceci signifie que, supposant que des nœuds voyagent en moyenne à mètres 10/sec, les nœuds pourront généralement voyager jusqu'à 150 mètres avant qu'une observation expire. Car nous

verrons plus tard, on ne devrait pas permettre à des nœuds de voyager beaucoup plus loin que ceci, autrement il sera beaucoup plus difficile de comparer deux observations séparées. En outre, parce que les émissions se produisent approximativement une fois chaque demi-seconde, un événement simple pourrait se propager jusqu'à 30 sauts plus loin. Nous voulons que les événements se propagent aussi loin que possible afin de fournir plus des données au procédé de construction du modèle, mais de ne pas exister très longtemps jusqu'à ce que l'événement soit dépourvu de toute signification.

2.3.2.3 Construction modèle

Les nœuds honnêtes essayent périodiquement d'expliquer les observations retenues dans leur base de données. Ce processus commence par incrémenter i à partir de 0 jusqu'a N nœuds malveillants, où N est le nombre total des nœuds dans la base de données. Pour chaque itération, nous essayons systématiquement toutes les permutations des *labellings* avec des nœuds malveillants de i. Pour chaque *labellings*, nous marquons au commencement tous les nœuds non-malveillants comme *spoof*, excepté le nœud actuel qui est marqué comme *truthful* (c.-à-d., un nœud se considère honnête).Puis, commençant au nœud actuel, nous suivons périodiquement toutes les observations, en sautant les nœuds malveillants et en marquant les nœuds visités comme *truthful*. Tous les nœuds non visités demeureront comme *spoof*. En conclusion, nous exécutons un contrôle de cohérence sur ce *labeling*. S'il est cohérant, nous choisissons cette explication et terminons l'algorithme; autrement, nous continuons jusqu'à ce que nous trouvions une explication cohérente ou manquons de possibilités.

Contrôle de cohérence

L'élément clé de l'algorithme de construction du model exécute un contrôle de cohérence. Ceci signifie donner un label a tous les nœuds (*truthful, malicious* ou *spoof*), nous vérifions que les observations enregistrées sont conformes basées sur ces deux règles:

1. Tous les nœuds *truthful* doivent déclarer leur position, vitesse, et *heading*. Cette déclaration doit être compatible à toutes les observations par des nœuds *truthful*.

2. Pour les nœuds *malicious* ou *spoof*, ignorer les déclarations, mais vérifier que toutes les observations par des nœuds *truthful* conviennent.

Le contrôle d'uniformité de deux événements est exécuté suivant la *rectangular path prediction* décrite dans la prochaine section. Puisque nous recherchons progressivement de plus en plus des nœuds malveillants, la meilleure explication qui est conforme sera une avec le nombre minimum de nœuds malveillants et ainsi le plus grand nombre d'événements conformes. Dans le pire des cas, aucun des événements n'est conforme, et tous les nœuds (excepté le nœud actuel) sont étiquetés *malicious*.

Comparaison des événements

Puisque l'instant après qu'un événement soit enregistré un nœud est susceptible d'avoir déménagé, des événements ne peuvent pas être comparés littéralement.

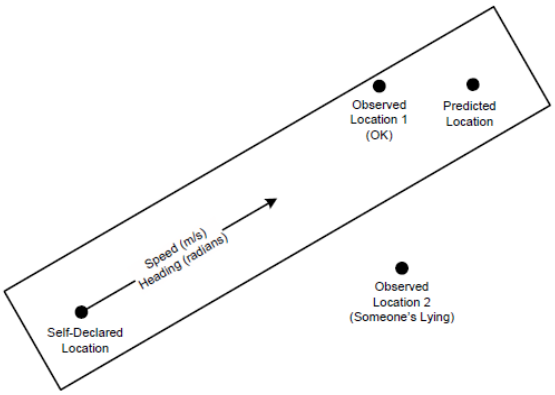


Figure 22. rectangular path prediction

Le point dans le coin bas-gauche est la position, la vitesse, et le *heading* d'un nœud donné (figure 22). À une autre heure (passée ou futur), un autre nœud fait une observation de ce nœud comme indiqué par les deux emplacements observés (1 et 2). La question est: ces événements sont-ils compatibles à l'emplacement avoué par le nœud? Basé sur la vitesse et le *heading* du nœud, nous pouvons prévoir son emplacement à tout moment tant que la vitesse et le *heading* demeurent constants. Cependant, dans les scénarios réels de trafic, il y aura une certaine incertitude dans cette prévision, y compris des facteurs tels qu'une vitesse de nœud/ *heading* changeant ou la dérive d'horloge entre les nœuds. Ainsi, nous laissons une marge fixe d'erreur le long du chemin prévu. Actuellement, ceci est défini à 10 mètres dans chaque direction afin d'égaler raisonnablement l'état des routes typique. Tant que les observations font partie de ce rectangle, nous les considérons comme

étant des événements conformes. Dans le scénario déployé, cette marge devrait être compatible avec les erreurs de la technologie de détection d'emplacement disponibles.

2.4 Solution de Soyoung Park et al.(2009)

En ce document, l'auteur propose une approche de série de *timestamps* pour se défendre contre les attaques Sybil dans un réseau ad-hoc véhiculaire basé sur le support d'unité de bord de la route. L'approche proposée vise l'étape initiale de déploiement des VANETs quand l'infrastructure de base de support de l'unité de bord de la route (RSU) est disponible et une petite fraction des véhicules ont la capacité de communication réseau. À la différence des solutions précédemment proposés qui exigent une infrastructure à clé publique véhiculaire dédiée pour certifier les différents véhicules, dans notre approche les RSUs sont les seuls composants délivrant les certificats. En raison des différences de la dynamique mobile parmi des véhicules, il est rare d'avoir deux véhicules passant par multiple RSUs exactement au même temps. En exploitant cette corrélation spatiale et temporelle entre les véhicules et les RSUs, deux messages seront traités comme attaque Sybil s'ils font émettre des séries similaires de *timestamp*s par RSUs. Cette approche n'a besoin ni d'infrastructure a clé publique véhiculaire ni d'RSUs accédant a internet, ce que lui rendent une solution économique appropriée à l'étape initiale de déploiement des VANETs.

2.4.1 Modèle D'environnement

Cette approche considère l'étape initiale de déploiement de VANET où: (1) seulement une petite fraction des véhicules sur les routes sont équipées de dispositifs de communication sans fil (nous les appelons les véhicules intelligents); (2) il n'existe aucune infrastructure à clé publique véhiculaire dédiée (VPKI) pour différents véhicules intelligents; (3) un nombre limité d'unités de bord de la route sont installés et ils ne peuvent pas avoir l'accès à Internet. Dans un environnement réseau si basique, le véhicule n'a pas une paire de clés privée/publique permanente (ou de long terme) et un certificat. Les suppositions de base sur les véhicules et RSUs sont comme suit:

• Véhicule: Il a une unité de bord (OBU) pour la gestion de réseau, le traitement des messages, un GPS pour la détection d'emplacement et une carte digitale comprenant l'information géographique de route.

• Unité de bord de la route (RSU): Il a un émetteur pour envoyer et recevoir des messages a un saut, mais il n'exige pas d'avoir un accès Internet. En outre, il a un dispositif inaltérable pour stocker les informations de sécurités et produire des paires de clés aléatoires certifiées ou des *timestamps* certifiés. Chaque RSU a ses propres clés privées/publiques et son certificat délivrés par le *Certificate Authority* (CA). Les paires de clés et le certificat sont enregistrés dans son dispositif inaltérable.

• *Certificate Authority* (CA): elle contrôle les RSUs et délivre des certificats pour la clé publique des différents RSUs. Chaque véhicule intelligent est préinstallé avec la clé publique du CA.

Maintenant, on définit un véhicule malveillant pour ce modèle comme suit: un véhicule malicieux peut: (i) collecter n'importe quelles informations a répartir du réseau et (ii) se servir de son propre dispositif de communication manufacturé pour créer de fausses informations GPS, de fausses informations de trafic et toute autres informations d'authentification relative telle que la signature numérique.

2.4.2 Solution

Sur les routes structurées simples qui ont des ruelles multiples et n'ont aucune congestion du trafic, les véhicules se déplacent dynamiquement à différentes vitesses et de manière indépendante. Basé sur ce phénomène, on découvre qu'il serait rare pour que deux véhicules arbitraires passent par des RSUs différents (lointain et indépendant l'un de l'autre) toujours en même temps. Par conséquent, si un message de trafic envoyé à n'importe quel véhicule contient plusieurs *timestamps* par les mêmes RSUs précédemment passé, une attaque Sybil peut être détectée si des messages multiples de trafic contiennent des séries de *timestamps* très similaires. On peut fortement suspecter ces messages comme messages Sybil créés par un véhicule simple.

Cette approche exige que seulement les RSUs puissent émettre les *timestamps* et qu'un véhicule ne peut pas utiliser un *timestamp* obtenu par d'autres véhicules. Par conséquent, dans cette conception, (1) le *timestamp* est digitalement signé par le RSU émetteur et (2) un *timestamp* obtenu par un véhicule contient la clé publique *self-generated* du véhicule, qui ne peut pas être utilisée par d'autres qui ne connaissent pas la clé privée correspondante.

Un véhicule peut créer des demandes multiples afin d'obtenir des *timestamps* multiples d'un même RSU. Cependant, les *timestamps* obtenus par un véhicule simple dans un intervalle de transmission du même RSU seront très étroits dans leurs *timestamps*. Comme mentionné ci-dessus, les messages de trafic avec ces *timestamps* peuvent être facilement détectés comme messages Sybil.

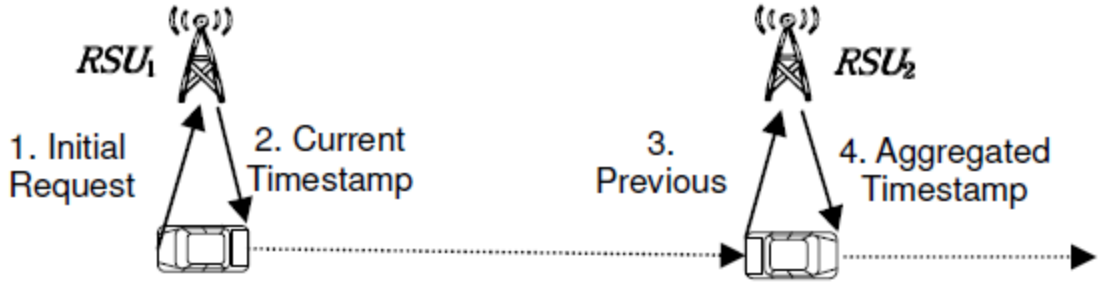


Figure 23. Illustration de l'approche des timestamps

Afin d'employer la trajectoire dynamique des véhicules pour détecter l'attaque Sybil, chaque message de trafic envoyé par un véhicule devrait inclure au moins deux *timestamps* émis des deux derniers RSUs que le véhicule a passés. Une voie directe est d'attacher les deux *timestamps* récemment obtenus au message de trafic.

Cependant, ajouter deux *timestamps* certifiés peut considérablement augmenter la taille des messages de trafic. Puisque chaque *timestamp* est digitalement signé par le RSU émetteur, le message de trafic doit aussi bien inclure les certificats supplémentaires des RSUs émetteurs. En outre, dans une congestion du trafic, plus de deux *timestamps* peuvent être exigés pour être mis dans un message de trafic afin de différencier la trajectoire des différents véhicules; c'est parce que beaucoup de véhicules se déplacent lentement côte à côte et pourraient recevoir des *timestamps* similaires des RSUs situés autour de la zone congestionnée.

Pour résoudre le problème ci-dessus, un *timestamp* agrégé a été employé afin de réduire au minimum l'*overhead* de sécurité. Pour réaliser ceci, un véhicule doit montrer son *timestamp* précédent avant qu'il n'obtienne un nouveau. Un RSU doit vérifier d'abord le *timestamp* fournit, et puis, pour un *timestamp* valide, crée un nouvel *timestamp* agrégé qui contient le *timestamp* actuel et les *timestamps* précédents. Par conséquent, les besoins

de chaque message de trafic se limitent seulement au plus récent *timestamp* agrégé et au certificat simple du RSU émetteur pour la détection d'une attaque Sybil.

2.5 Discussion

Dans cette première partie du chapitre, nous avons pu voir les différentes approches proposés en guise de solution pour la détection du nœud malicieux. On a exposé pour chaque solution le model d'environnement ainsi que les mécanismes sur lesquels est fondé chaque solution proposée. Tout ce travail a été réalisé dans la perspective de pouvoir faire un constat générale sur ces approches, leurs fondements, point fort et faiblesses. Pour ce faire nous allons discuter ces approches une a une dans ce qui suit;

Le premier travail, celui de Philippe Golle et al(2004), est un travail purement mathématique dont l'application sur les VANETs réels semble encore avoir du chemin à faire pour définir les bonnes règles à exploiter dans le modèle mathématique définit, les critères de choix de la solution cohérente accepté et encore plus loin la définition de la solution cohérente elle-même est à revoir. Tout cela nous amène à dire que cette solution est dotée de forts mécanismes mathématiques d'où elle retient sa force. Néanmoins, elle ne peut être déployé pour le moment à cause des insuffisances aperçus dans la définition de ces paramètres ainsi que des décisions déterministes qu'elle prend (malicieux et honnête; négligence des probabilités d'erreur ou de manipulation).

Le travail suivant est celui de Bin Xiao et.al (2006) est une solution statistique dans laquelle l'auteur propose un schéma signal-strength-based basique renforcé par des techniques utilisant les RSUs et les profils du trafic dans les VANETs. Cette approche est destinée exclusivement à la détection des attaques Sybils qui constituent une sous classe des comportements malicieux, ainsi que le problème d'overhead de communication généré à cause du nombre important de message utilisé lors de l'exécution de la solution. Il faut ajouter à cela la négligence total des messages des nœuds provenant du sens inverse et la perte des informations précieuses qui pourraient alerter des dangers potentiellement présents dans leurs trajectoires.

Quant au travail de Jonathan Van Eenwyk (2007), il s'agit d'une pure amélioration du travail de Philippe Golle et al(2004) en éliminant l'hypothèse de la propagation idéale et celle de la base de données communes à tous les nœuds du VANETs pour lui donner un

aspect plus réaliste. Malgré ces améliorations, ce travail souffre du manque de précision de la méthode de détection de position. Ce qui compromet de manière significative les résultats de cette approche ainsi que les diffusions périodiques du contenu des bases de données individuelles. Ceci génère un overhead de communication très important.

Le dernier travail est celui de Soyoung Park et al(2009), qui se base sur une agrégation de timestamps et sur la supposition que deux nœuds ne peuvent pas passer par les même RSUs en mémé temps, alors que c'est très probable dans un environnement urbain où les routes sont très proches presque comme des grilles. Des véhicules circulants sur deux ruelles parallèles peuvent très souvent avoir la même série de timestamps lors d'une certaine période de temps.Dans cette première partie du chapitre, nous avons pu voir les différentes approches proposés en guise de solution pour la détection du nœud malicieux. On a exposé pour chaque solution le model d'environnement ainsi que les mécanismes sur lesquels est fondé chaque solution proposée. Tous ce travail a été réalisé dans la perspective de pouvoir faire un constat générale sur ces approches, leurs fondements, point fort et faiblesses.

3 Elimination des nœuds malicieux

Quand un dispositif commence à envoyer de mauvaises informations, la solution à long terme est l'autorité de certification (par exemple, le Département de *Motor Vehicles*) pour retirer les qualifications du dispositif offensant. Cependant, ce processus prend du temps, de la collection d'évidences à la résolution des réclamations contestées. En ce temps, les attaques intérimaires et continues pourraient mettre en danger la sûreté des passagers. Ainsi, il y a un besoin d'isoler rapidement de tels dispositifs errants et de les empêcher de diffuser des données incorrectes. Une solution pour les voitures observant la mauvaise conduite est d'exclure temporairement le mauvais dispositif responsable jusqu'à ce que l'autorité de certification soit informée et prenne une mesure appropriée. Dans cette partie du chapitre, nous considérons des voies de prendre une décision locale tout en maximisant l'efficacité et la sécurité.

Nous décrivons d'abord un mécanisme de décision local déjà proposé, appelé LEAVE (*Local Eviction of Attackers by Voting Evaluators*), où les nœuds votent pour exclure les dispositifs errants en permutant des réclamations d'inexactitude signée [68]. Nous présentons aussi un nouveau protocole, appelé *Stinger*,[39] dans lequel un nœud peut

unilatéralement retirer un voisin de conduite perçu malicieuse en limitant sa propre participation. Ce dernier est une adaptation du protocole de suicide proposé pour les réseaux ad-hoc dans [69]. Et en fin nous discuterons les imperfections des deux approches proposées pour pouvoir par la suite tirer profit des résultats obtenues par ces deux solutions.

3.1 Environnement de travail

Nous décrivons maintenant les caractéristiques opérationnelles des réseaux véhiculaires sur lesquelles vas être basé notre étude.

3.1.1 Model du système

Les autorités véhiculaires du système sont susceptibles de devenir des autorités de certification (CAs). Chacune serait responsable de la gestion des identités de tous les véhicules enregistrés dans sa région géographique respective. Les véhicules s'inscrivent exactement à un CA. Chaque nœud a une seule identité, une paire de clés cryptographique privée et publique, et un certificat délivré par le CA.

Les messages sont transmis périodiquement, par exemple, chaque 0.3 s pour des messages de sûreté, ou déclenchés par des événements *in-vehicle* ou de réseau. La plupart du trafic sont des émissions à des régions limitées du réseau. Tous les messages sécuritaires incluent la période et les coordonnées géographiques de l'émetteur, en plus de l'autre information spécifique à l'application. Chaque message est également signé et accompagné du certificat de l'émetteur. Il est largement accepté que la cryptographie asymétrique est faisable pour les réseaux véhiculaires [33].

Les messages de sûreté peuvent se propager à travers des sauts multiples. Dans ce cas-ci, ils sont signés et incluent les coordonnées et le *timestamp* du dernier nœud relais, avec la signature du créateur, ses coordonnées et son *timestamp*. Cette chaîne de signatures assure la fraîcheur d'information tout en limitant la propagation d'information illégitime. Un message de sûreté reçu est rejeté si la différence entre son *timestamp* et le *timestamp* du récepteur est plus grande qu'une constante spécifique aux horloges du système, à la propagation et aux retards de traitement. D'ailleurs, un message est rejeté (par un récepteur) si les coordonnées de son émetteur/relais indiquent que le récepteur est en

dehors de l'intervalle nominal maximum de la communication sans fil de l'émetteur. Ces validations sont appliquées à chaque saut.

À la couche liaison de données, nous supposons que 802.11p est utilisé. Ce protocole fournit des intervalles de transmission de 300 à 1000 m, des débits dans l'intervalle de 6-27 Mbps.

Un sous-ensemble de nœuds du réseau forment l'infrastructure, constituée par les stations de base à courte portée et des unités mobiles. Ces dernières incluent les véhicules de sécurité publique (par exemple, aide d'omnibus et véhicules de lutte contre l'incendie), les véhicules de police, et les véhicules de transport en commun. Les nœuds d'infrastructure se servent des *Gateways* du CA à et du réseau véhiculaire; la connexion du CA aux nœuds statiques de l'infrastructure est effectif au-moyen de liens filaires fixent. Cependant, nous ne supposons pas que le CA doit être accessible au réseau véhiculaire à tout moment. Les mécanismes d'exclusion LEAVE et *Stinger* sont exécutés par des véhicules ordinaires, pas par les nœuds de l'infrastructure.

3.1.2 Modèle d'attaque

Un adversaire, ou attaquant, peut contrôler un certain nombre de nœuds qui dévient des protocoles réseau véhiculaires légitimes. Les nœuds peuvent également être défectueux dus aux échecs de matériel. Un examen détaillé des modèles d'adversaire et de défaut est donné dans [53]. Pendant que nos mécanismes proposés s'appliquent aux nœuds malicieux et aux dispositifs errants, nous utilisons les deux termes l'un pour l'autre. Nous soulignons que nous sommes concernés par des nœuds malicieux équipés des qualifications valides.

Nous considérons deux types de stratégie d'attaquant. La « diffusion de fausse information » peut être une attaque très pertinente, une fois comparée aux « écarts des protocoles réseau ». La motivation pour les attaques de diffusion de fausses informations peut être malveillante (par exemple, en envoyant une information de freinage truqué pour provoquer un accident) ou égoïste (par exemple, la revendication qu'un accident s'est produit pour se dégager de l'encombrement). L'adversaire pourrait manipuler les capacités sensorielles ou compromettre le *protocol-stack* et la plateforme informatique [53]. Un attaquant peut également contrôler la communication entrante, par exemple, en effaçant sélectivement des messages reçus par sa plate-forme *on-board*.

La deuxième stratégie d'attaque est l'« abus du mécanisme d'exclusion ». Dans la prochaine section, nous discutons deux propositions pour exclure de mauvais dispositifs de la participation sur le réseau. Ces stratégies, et n'importe quel autre mécanisme qui essaye d'exclure de mauvais dispositifs, peuvent être maltraités par un adversaire essayant de retirer de bons dispositifs au lieu des mauvais.

3.2 LEAVE

LEAVE [68] est illustré sur la figure 24. Les Véhicules détectant un dispositif errant diffuse des messages d'avertissement (*warning*) à tous les véhicules dans leurs zones de transmission. N'importe quel véhicule recevant un message d'avertissement ajoute le dispositif averti à une liste d'accusation. Une fois assez de voix d'avertissement contre un nœud sont rassemblées, son identificateur est ajouté à une liste noire locale.

Après que des nœuds soient ajoutés aux listes noires, des messages supplémentaires de négligence (*disregard*) sont à plusieurs reprises émis au voisinage local du nœud leurs demandant d'ignorer les messages de l'attaquant.

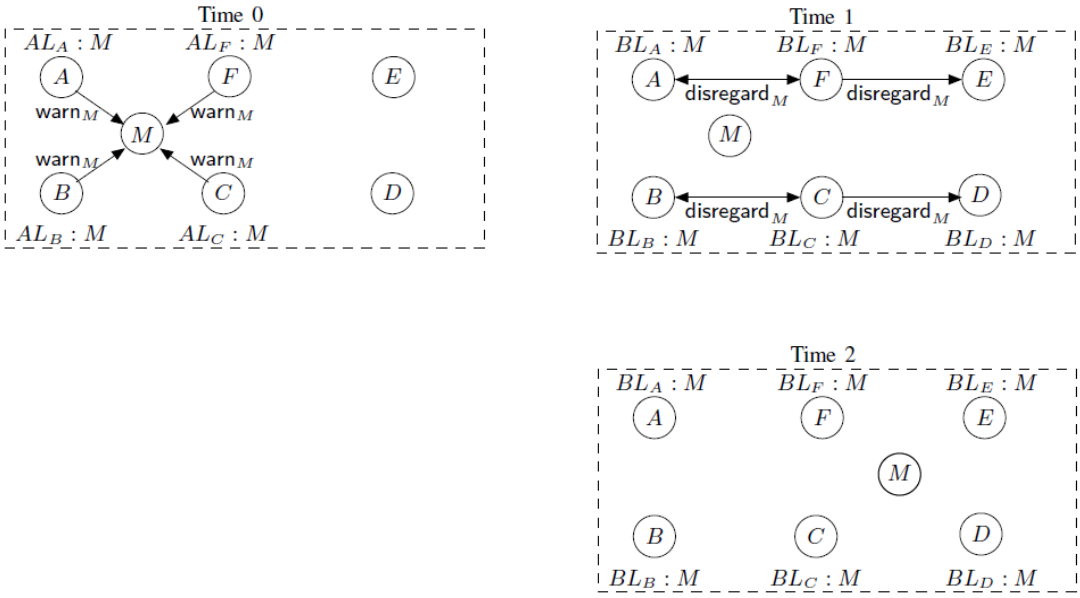


Figure 24. LEAVE (Local Eviction of Attackers by Voting Evaluators)

Par conséquent, les véhicules utilisant LEAVE peuvent être mis au courant des mauvais véhicules avant l'interaction avec ces derniers.

Finalement, les nœuds expulsés sont enregistrés au CA une fois dans la portée d'un nœud d'une infrastructure.

La décision d'avertir qu'un nœud utilisant LEAVE est malveillant est en réalité plus subtile que surpasser un seuil numérique simple de voix d'avertissement. En revanche, elle est basée sur le dépassement d'un quotient d'exclusion, une somme d'accusations pesées relativement à la taille du voisinage d'un véhicule (le papier de LEAVE a utilisé un quotient d'exclusion de 0.5). Le quotient d'exclusion ignore les accusations des utilisateurs qui eux-mêmes ont été accusés par d'autres nœuds, comme proposé par Crepeau et Davis dans [70]. Son calcul est également basé sur des poids affecté aux informations reçues selon l'identité de l'émetteur parce qu'il favorise les informations émises par les véhicules de confiance telle que celle des véhicules de police en leurs affectant des poids supérieurs par rapport au reste des autres véhicules.Pour les messages de négligence, un seuil simple est utilisé (le papier de LEAVE a utilisé un seuil de 4 voix). Pour expliquer leur légitimité, les messages de négligence incluent les signatures des nœuds supportant ce seuil.

Pour être sécurisé, LEAVE exige une majorité honnête: chaque bon nœud doit toujours avoir plus de bons voisins que de mauvais. Si l'attaquant contrôle plus de dispositifs que le seuil exigé pour envoyer des messages de négligence, alors les mauvais dispositifs peuvent éjecter n'importe quel bon dispositif à volonté.

3.3 Stinger

Dans [69], Moore et al proposent plusieurs stratégies qui permettent à des nœuds de retirer les dispositifs compromis d'un réseau ad-hoc. Nous discutons maintenant comment une stratégie présentée -des attaques de suicide- qui peut être adapté pour l'usage dans les réseaux véhiculaires.

Les procédures pour retirer un mauvais dispositif sont beaucoup plus simples une fois prises par un simple nœud. Si un nœud croit qu'un autre nœud se conduit malicieusement, il peut unilatéralement le retirer. Naturellement, un nœud malveillant peut faussement accuser des nœuds légitimes. Par conséquent, l'acte de la punition doit être rendu coûteux pour le nœud qui prend cette décision. Les stratégies de suicide retirent l'accusé et l'accusateur du réseau. Lors de la détection qu'un nœud M s'engage dans une certaine activité illégale, le nœud A envoie une note de suicide $suicide_{A,M}$ avec les identités de A et

de M. Les autres nœuds négligent maintenant les nœuds A et M. Le sacrifice des futures participations est si coûteux qu'il explique sans doute la loyauté de la réclamation du nœud.

Les suppositions environnementales considérées par Moore et al[69] ne correspondent pas directement aux nœuds membres d'un réseaux véhiculaires. Le mécanisme modifié adapté aux VANETs s'appelle *Stinger*, et des notes de suicide s'appellent les *stings*. *Stinger* dévie du suicide aux considérations suivantes:

1) *Stinger* interdit temporairement les dispositifs de transmettre des messages, mais leur permet de continuer à les recevoir et a les expédier;

2) *Stinger* permet à de multiples bons nœuds d'être ignorés par un plus petit nombre de dispositifs afin d'exclure un mauvais nœud simple;

3) *Stinger* permet aux bons dispositifs de continuer d'accuser le mauvais, et cela même après qu'ils ont publié un *sting*.

Nous décrivons maintenant chacune de ces modifications, et leur motivation, plus en détail.

D'abord, le mécanisme de suicide initial a proposé l'exclusion permanente de la participation sur le réseau. Une telle punition est inadéquate pour les communications véhiculaires qui transmettent des informations de sûreté. En revanche, l'exclusion provisoire pourrait être employée pour ignorer rapidement un émetteur errant. Puisque la plupart des interactions sont de courte durée, l'exclusion provisoire est également pertinente en abordant la mauvaise conduite car il se produit sans interruption de la communication se produisant beaucoup plus tard.

Tandis que les directives de *sting* empêchent le mauvais et bon dispositif d'envoyer des avertissements supplémentaires, toutes les deux reçoivent toujours des directives de sûreté des autres voitures. Ceci réduit au minimum l'incidence apparente sur le gestionnaire de sacrifice tout en pénalisant toujours un dispositif malveillant.

En second lieu, le mécanisme initial de suicide a assumé un réseau complètement connecté. Les notes de suicide étaient émises dans tout le réseau de sorte que juste un bon dispositif soit retiré pour chaque mauvais dispositif. Les réseaux véhiculaires seront

composés d'îles déconnectés. Les zones Haut-débit dans les villes demeurent séparées les unes des autres et des *highways*. En outre, les connexions sont éphémères: les voitures sur une autoroute peuvent seulement être dans l'intervalle de transmission pendant quelques secondes. Ainsi, il est impraticable de transmettre des messages de *Stinger* à travers un pays en peu de temps. Au lieu de cela, les messages de *Stinger* doivent demeurer localisés, rediffusés tout au plus quelques fois. Ceci maintient la réponse rapide et réduit au minimum *l'overhead* de communication. Il signifie également qu'il y aura des périodes où plus d'un bon nœud s'est sacrifié pour le même mauvais nœud. Pourtant l'incidence est encore limitée: plutôt qu'avoir un nœud simple retiré pour un mauvais nœud, plusieurs nœuds peuvent être indépendamment retirés pour un même mauvais nœud. Crucialement, aucun dispositif n'ignorera deux nœuds honnêtes pour le même mauvais nœud. C'est parce que les bons nœuds mettent à jour une liste noire locale, et ils ignorent seulement le premier transmetteur de *Stinger* pour chaque dispositif accusé.

La figure 25 montre comment le protocole de *Stinger* fonctionne pendant que les voitures bougent. Le mauvais nœud M est détecté par A, qui annonce le $sting_{A,M}$ pour demander aux véhicules près de A d'ignorer M. Cependant, les nœuds B et C ajoutent A et M à leurs listes noires locales, alors que D et E ne le font pas parce qu'ils n'ont pas reçu le $sting_{A,M}$. Pendant que M entre dans l'intervalle de D et E, E publie un nouveau message d'exclusion pour M, $sting_{E,M}$. D ajoute E et M à sa liste noire locale, mais C ne le fait pas parce qu'il a déjà ignoré M du *sting* de A.

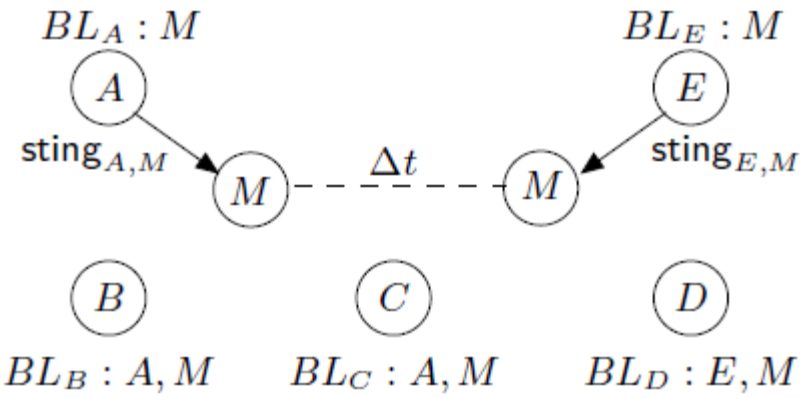


Figure 25. Multiple suicides pour un seul nœud malicieux (stinger)

Cette discussion motive la troisième différence entre le suicide et *Stinger*: les bons dispositifs continuent à accuser le mauvais ceux même après qu'ils ont publié un *sting*. C'est nécessaire pour empêcher un soi-disant attaquant d'autoroute qui annonce largement la mauvaise conduite et déménage autour rapidement pour attirer beaucoup de *sting* et pour empêcher de bons nœuds d'exclure les attaquants ultérieurs.

Les messages *sting* sont localement transmis, et ils peuvent également être rediffusés pour avertir les autres dispositifs au cas où le mauvais dispositif déménagerait plus tard. L'effet de la retransmission de *sting* est montré sur la figure 26.

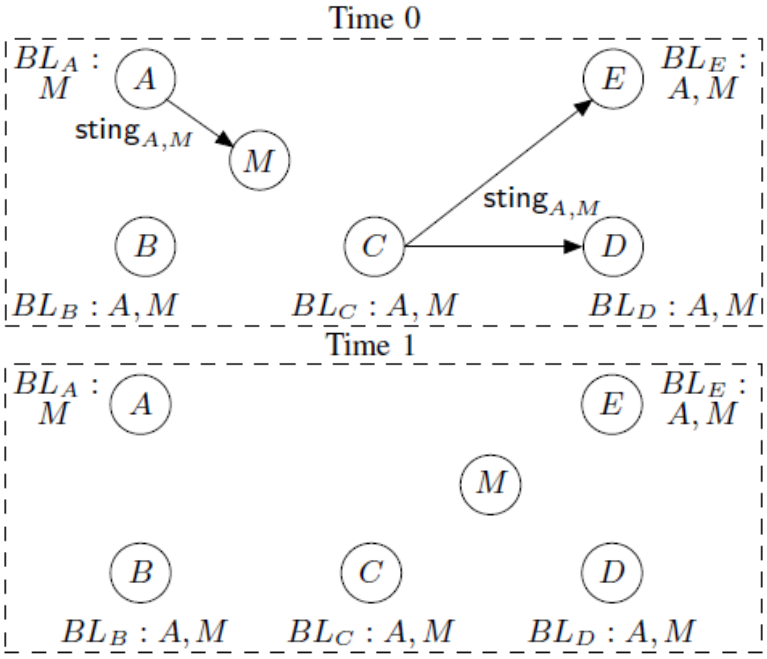
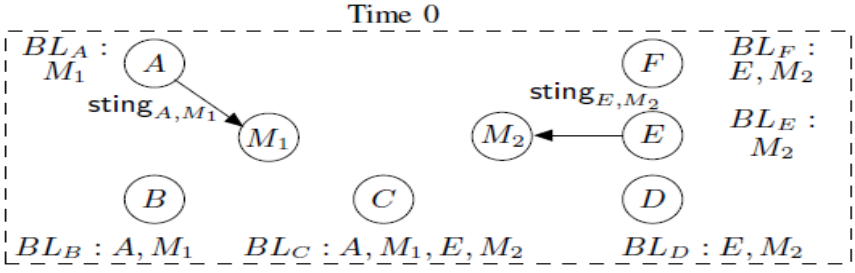


Figure 26. rebroadcast de stings

Au temps 0, le mauvais nœud M est détecté par A qui transmet le $\text{sting}_{A,M}$. les nœuds B et C alors retransmettent le message, informant D et E. Quand M déménage près de D et E au temps 1, M est déjà ignoré par eux.



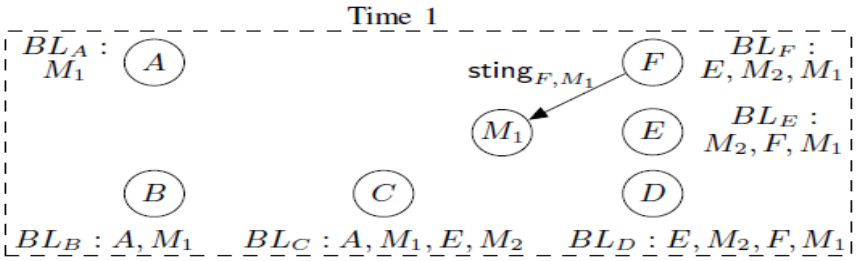


Figure 27. impact adverse sur stinger

Ainsi quel est le coût de *Stinger* en omettant les messages et l'*overhead* de transmission? Les bons dispositifs qui ont publié des *stings* ne peuvent plus avertir leurs voisins s'ils détectent un autre dispositif malicieux. Sur la figure 27, les mauvais nœuds M1 et M2 sont présents dans différentes zones. Les nœuds A et E génèrent des *stings* pour les retirer localement. Cependant, quand M1 entre dans la zone précédemment occupée par M2, E est incapable d'avertir ses voisins. E peut essayer de retirer M1, mais il n'a aucun effet puisque ses voisins ignorent déjà ses messages. Ainsi F est le seul capable de générer le $sting_{F, M1}$. Dans l'analyse suivante, nous mesurons l'incidence défavorable d'exclure les dispositifs honnêtes en mesurant tous les retards introduits en retirant de mauvais dispositifs.

//lors de la detection d'un nœud malicieux m par un nœud a

```
If MDSa(m)=true then
  Add(m,BLa);
  Send «sting(a,m,Signa)»;
End;
```

//Lors de la réception d'un message 'Stinga,m' signé par a par un vehicule x

```
RCV «sting(a,m,Signr)»;
If exists(a,BLx)=true or exists(m,BLx)=true then
  Discard «sting(a,m,Signa)»;
Else
  Add(m,BLx);
  Add(a,BLx);
End;
```

3.4 Discussion

Dans cette section on a identifié un certain nombre de différences entre LEAVE et *Stinger* pour les différents scenarios vus en [71]. Pour récapituler:

- LEAVE est plus résilient aux taux de faux positif différents de zéro.

- Moins de bons nœuds sont ignorés en utilisant LEAVE que *Stinger* quand un attaquant maltraite le mécanisme d'exclusion.

- *Stinger* exclut plus rapidement les mauvais dispositifs, laissant une fenêtre de vulnérabilité plus courte.

- *Stinger* résiste (*scale*) mieux à mesure que la densité des véhicules augmente.

Malheureusement, il n'y a aucun clair vainqueur entre LEAVE et *Stinger*. LEAVE réalise un meilleur travail de traitement des faux positifs que *Stinger*, alors que *Stinger* est sensiblement plus rapide que LEAVE en retirant les mauvais dispositifs dans chaque condition que nous avons testée. La vitesse du déplacement est critique pour limiter la transmission de la fausse information.

Nous concluons qu'une stratégie hybride qui s'adapte aux paramètres de système et à l'environnement pourrait tirer meilleur parti des deux mécanismes, si appliqué correctement. Avec plus de précision, trois facteurs principaux affectent le choix du mécanisme d'exclusion: fenêtre tolérable de vulnérabilité, taux de faux positif, et le pourcentage d'attaquants détecté. Ces facteurs sont influencés par l'application de sûreté en question (par exemple, la fenêtre de vulnérabilité d'une application de risque-avertissement dépend de la distance à l'accident), la mauvaise conduite étant détectée (par exemple, la falsification des informations d'emplacement n'est pas facilement détecté) et la densité de trafic. Par conséquent,

Les auteurs de [71] ont proposé une stratégie hybride qui a essayé de prendre en compte ses critères.Cette solution stipule que:(1) Tant que la fenêtre de vulnérabilité réalisée par LEAVE est tolérable, utiliser LEAVE. Puisque les mécanismes de détection sont peu susceptibles d'avoir des taux de faux positif nuls, *Stinger* ignorera un certain pourcentage de bons voisins.(2) quand la fenêtre tolérable de vulnérabilité est dépassée

(par exemple, en raison du trafic croissant conditionné par la vue locale de la voiture), commuter à *Stinger*, mais permettre toujours aux dispositifs de voter au choix.

Cette solution elle-même augmente posent de nouveaux obstacles en terme d'efficacité et d'applicabilité pour les deux protocoles en même temps et cela est du aux facteurs citées ci-dessus:

- le protocole proposé est basé sur la *vulnerabity-windows* autorisée selon l'emplacement et le type de service, alors que ces deux paramètres eux-mêmes peuvent être facilement manipulés par l'attaquant.

- il suppose que chaque nœud choisi le protocole d'exclusion qu'il veut exécuter ce qui réduit de manière sérieuse l'efficacité du protocole LEAVE en réduisant le nombre de nœuds participant aux vote en empêchant ainsi le seuil d'exclusion d'être atteint, et en même temps pose un handicape au nœuds exécutant *stinger* en réduisant leurs nombre et en les interdisant a participer ultérieurement a LEAVE si elles avaient été déjà suicidé en exécutant stinger.

- il génère un taux de communication plus élevé en utilisant les messages des deux protocoles.

Cela nous amène à éviter l'utilisation d'un tel protocole même si ses fondements théoriques étaient aussi élaborés, mais sa mise en œuvre a mis en évidence ses énormes anomalies.

4 Conclusion

Dans ce chapitre nous avons montrés les différents travaux existant dans la littérature pour ce qui est de la détection des nœuds malicieux et leurs éliminations. Chaque travail dispose de son propre environnement d'exécution et a ses propres points forts ainsi que quelques manques qu'on a pu apercevoir.

Dans l'optique de présenter une approche complète qui prenne en considération les deux phases du traitement (détection et élimination) on va essayer de vous présenter le chapitre suivant qui portera sur la solution que nous avons proposé et qui essayera de palier aux points cités dans ce chapitre.

Chapitre IV

Une Nouvelle Solution Pour La Détection Et L’élimination des Nœuds Malicieux Dans Un VANET

1 Introduction

Assurer l'intégrité des communications véhiculaires de sûreté est primordial. Les émetteurs compromis pourraient envoyer une information factice pour des raisons égoïstes (par exemple, la prétention qu'il y a eu un accident automobile pour détourner le trafic à partir du chemin choisi et pour apprécier un chemin non congestionné) ou malveillantes (par exemple, falsifier les informations d'emplacement pour encourager les collisions).

Quand un dispositif commence à envoyer de mauvaises informations, la solution à long terme est l'autorité de certification (par exemple, *ce que propose* le Département de *Motor Vehicles [71]*) pour retirer les qualifications du dispositif offensant. Cependant, ce processus prend du temps, de la collection de certitudes à la résolution des réclamations contestées. Entre temps, les attaques intérimaires et continues pourraient mettre en danger la sûreté des passagers. Ainsi, il y a un besoin d'isoler rapidement de tels dispositifs corrompus et de les empêcher de diffuser des données incorrectes. Une solution pour les voitures observant une mauvaise conduite est d'exclure temporairement le mauvais dispositif responsable jusqu'à ce que l'autorité de certification soit informée et prenne une mesure appropriée. Dans ce chapitre, nous considérons les solutions possibles pour détecter localement de fausses informations, identifier leurs sources pour enfin procéder a leurs éliminations.

Dans le chapitre précédent nous avons discuté les divers travaux qui ont été accompli dans le domaine de la détection des nœuds malicieux dans les VANETs. Nous avons fait le point sur l'environnement de travail de chacune des solutions proposées ainsi que les mécanismes de fonctionnement de ses solutions. Par la suite, nous avons discuté les différents aspects techniques de ses solutions pour pouvoir faire un constat sur le travail qui reste à accomplir dans cet axe de recherche. Dans la deuxième partie de ce chapitre, nous avons procédé de la même façon, cette fois pour présenter les deux protocoles réalisés dans le domaine de l'élimination des nœuds malicieux et pour analyser leurs différents aspects. Ce travail a été accomplie dans la perspective de proposer une nouvelle solution combinant les deux phases; détection et élimination des nœuds malicieux. Nous espérons ainsi avoir réussi à améliorer certaines des insuffisances remarquées dans les solutions étudiées au chapitre précédent plus particulièrement pour les comportements malicieux visant à troubler les flux de la circulation et les choix des utilisateurs du réseau routier.

2 Modèle d'environnement

Nous décrivons maintenant les caractéristiques opérationnelles des réseaux véhiculaires considérées pour notre étude.

2.1 Modèle du système

Les autorités véhiculaires du système sont susceptibles de devenir des autorités de certification (CAs). Chacune serait responsable de la gestion des identités de tous les véhicules enregistrés dans sa région géographique respective. Les véhicules s'inscrivent à exactement un CA. Chaque nœud a une seule identité, une paire de clés cryptographiques privées et publiques, et un certificat délivré par le CA.

Les messages sont transmis périodiquement, par exemple, chaque 0.3 s pour des messages de sûreté, ou déclenchés par des événements *in-vehicle* ou de réseau. La plupart des messages du trafic sont des émissions vers des régions limitées du réseau. Tous les messages sécuritaires incluent la période et les coordonnées géographiques de l'émetteur, en plus d'autre information spécifique à l'application. Chaque message est également signé et accompagné du certificat de l'émetteur. Il est largement accepté que la cryptographie asymétrique est faisable pour les réseaux véhiculaires [19].

Les messages de sûreté peuvent se propager à travers des sauts multiples. Dans ce cas-ci, ils sont signés et incluent les coordonnées et le *timestamp* du dernier nœud relais, avec la signature du créateur, ses coordonnées et son *timestamp*. Cette chaîne de signatures assure la fraîcheur d'information tout en limitant la propagation d'information illégitime. Un message de sûreté reçu est rejeté si la différence entre son *timestamp* et le *timestamp* du récepteur est plus grande qu'une constante spécifique aux horloges du système, à la propagation et aux retards du traitement. D'ailleurs, un message est rejeté (par un récepteur) si les coordonnées de son émetteur/relais indiquent que le récepteur est en dehors de l'intervalle nominal maximum de la communication sans fil de l'émetteur. Ces validations sont appliquées à chaque saut.

À la couche liaison de données, nous considérons le standard 802.11p [24]. Ce protocole fournit des intervalles de transmission de 300 à 1000 m, des débits dans l'intervalle de 6-27 Mbps.

Un sous-ensemble de nœuds du réseau forme l'infrastructure, constituée par les stations de base à courte portée et des unités mobiles. Ces dernières incluent les véhicules de sécurité publique (par exemple, aide d'omnibus et véhicules de lutte contre l'incendie), les véhicules de police, et les véhicules de transport en commun. Les nœuds d'infrastructure se servent des *Gateways* du CA à et du réseau véhiculaire; la connexion du CA aux nœuds statiques de l'infrastructure est effective au-moyen de liens filaires fixes. Cependant, nous ne supposons pas que le CA doit être accessible au réseau véhiculaire à tout moment. Les taches exécutées par la solution proposée sont exécutées par des véhicules ordinaires, pas par les nœuds de l'infrastructure.

Nous adoptons ainsi les hypothèses suivantes pour notre solution:

- Une majorité de nœuds honnêtes c.-à-d. que la plupart des nœuds du réseau sont censés avoir un comportement honnête (taux de nœuds malicieux<10%).
- Environnement dense (densité des véhicules à cause de l'environnement urbain supposé). Cette hypothèse assure une meilleure exécution du protocole qui se basse sur le voisinage pour la détection.
- Vitesse limitée (la densité et les règles de trafic dans les environnements urbains imposent cette contrainte).
- Chaque nœud est doté d'un modèle de VANET (ensemble d'hypothèses et de règles lui permettant de vérifier la validité des données détectées ou reçues) lui permettant de valider ses propres données (découvertes par lui-même ou reçues par d'autres nœuds).
- Un nœud connaît son voisinage (utilisation de *beacon*) et il est capable d'estimer la position d'un voisin (technologie de localisation basée sur le signale reçu par exemple RSSI).
- Chaque nœud considère que les informations qu'il collecte sont 'valides' avec une grande probabilité. Nous considérons l'information reçue correcte jusqu'à ce que certaines conditions soient découvertes (contradiction dans la base de données ou avec le modèle du VANET).

2.2 Modèle d'attaque

Un adversaire, ou attaquant, peut contrôler un certain nombre de nœuds qui dévient des nœuds réseau véhiculaires légitimes. Les nœuds peuvent également être défectueux à cause d'échecs matériels. Un examen détaillé des modèles d'adversaire et de dégradation est donné dans [45]. Pendant que les mécanismes proposés s'appliquent aux nœuds malicieux et aux dispositifs corrompus (qui présentent des comportements suspects mais involontaires suite à une contamination par un nœud malicieux). Nous traitons les deux cas (nœuds malicieux et nœuds corrompus) de la même manière**.** Nous soulignons que nous sommes concernés par des nœuds malicieux disposant d'identifiants sécurisés reconnus par les autorités de certification (clés, signature, ….).

Nous considérons donc deux types d'attaquants: le nœud malicieux qui génère les fausses informations et le nœud dit corrompus qui va contribuer sans le savoir et de manière involontaire à faire propager cette information. Ces deux attaques agissent à des degrés différents sur la validité des données du système. L'impact de l'attaquant par « diffusion de fausses informations» peut être plus néfaste comparé aux comportements des nœuds corrompus car ces derniers peuvent compromettre la validité des informations du réseau sans avoir de stratégie propre vu qu'ils diffusent des informations alors qu'ils ignorent même qu'elles sont fausses.

La motivation pour les attaques de diffusion de fausses informations peut être malveillante (par exemple, envoie d'une information de freinage faussé pour provoquer un accident) ou égoïste (par exemple, la revendication qu'un accident s'est produit pour se débarrasser de l'encombrement). Un nœud malicieux pourrait aussi manipuler les capacités sensorielles ou compromettre le *protocol-stack* (la pile de protocoles de communications réseaux avec toutes ses couches) et la plateforme informatique [45] déployée au niveau des OBU (*On Board Unit*). Un attaquant peut également contrôler la communication entrante, par exemple, en effaçant sélectivement des messages reçus par sa plate-forme *on-board*.

La deuxième stratégie d'attaque est l'« abus du mécanisme d'exclusion ». Dans la suite du chapitre, nous discutons deux propositions pour exclure les mauvais dispositifs de la participation sur le réseau. Ces stratégies, et n'importe quel autre mécanisme qui essaye

d'exclure de mauvais dispositifs, peuvent être maltraités par un adversaire essayant de retirer de bons dispositifs au lieu des mauvais.

3 Détection de nœuds malicieux

Les sections suivantes décrivent les divers aspects de notre solution pour la détection de nœuds malicieux. Elle est, en effet, basée sur trois concepts fondamentaux:

- Base de données individuelle pour chaque nœud du réseau.
- Chaque RSU émet une certification de position pour chaque véhicule passant près d'elle. La certification de position contient un *timestamp* et des informations sur l'emplacement de la station de base et peuvent donc prouver la présence du véhicule près de la station de base à un certain moment.
- Un degré de confiance supérieur est accordé aux véhicules du flux de trafic opposé au nœud pour éliminer l'influence des nœuds Sybil sur la consistance de la base de données.

Comme la confiance en un nœud est estimée par un sondage des nœuds du voisinage, les nœuds malicieux créent souvent des nœuds virtuels qui vont confirmer ces informations et renforcer le degré de confiance en les données générées. Les nœuds virtuels peuvent s'attribuer une identité certifiée appartenant à un autre nœud et qui a été captée sur le réseau mais il ne peut pas fournir de certification de positionnement délivrée par une RSU.

Les trois concepts définis ci-dessus peuvent être justifiés comme suit:

- nous avons choisi de donner à chaque nœud sa propre base de données privée au lieu d'assumer une base de données globale d'accès instantané, ce qui est plus proche des applications réalistes des VANETs.

- Le voisinage d'un nœud comportant les véhicules, dans la même direction, change peu comparé aux véhicules de son voisinage mais qui se déplacent dans le sens opposé. En effet, l'effet de la vitesse et des orientations opposées fait que le changement de voisinage devienne plus fréquent. D'autre part, souvent, un nœud malicieux partage un même intérêt avec les nœuds de la même direction (exemple, simulation d'embouteillage).

- Les deux autres concepts peuvent donc être justifiés par le fait que l'utilisation des informations provenant des véhicules du flux opposé et ayant un certificat délivré

par un RSU dont la position est connue et qui participent activement à diminuer l'influence des nœuds virtuels. Les véhicules virtuels ne peuvent pas fournir une certification de positionnement. Par conséquent, ils devront tous se servir de la certification du nœud malicieux qui les a créés ce qui permet de démasquer le nœud malicieux dans ce cas.

Périodiquement, les nœuds qui diffusent des informations vérifient le contenu de leur base de données. Juste avant cela, les nœuds enregistrent leur position actuelle, vitesse et heading (angle de déplacement ou direction de déplacement) dans la base de données. En tant qu'élément de la construction du modèle (décrit plus tard), d'autres nœuds vérifieront cette information par rapport à leurs propres observations (et celles des autres).

En recevant une information d'un autre nœud, le nœud intègre le contenu dans sa propre base de données. En outre, le nœud enregistre un nouvel événement pour l'émetteur. La position est basée sur la technologie de détection d'emplacement choisi, mais la vitesse et le *heading* sont ignorés pendant les calculs. C'est pour éviter la complexité ajoutée et le coût des technologies qui peuvent déterminer la vitesse et l'orientation des autres nœuds en considérant la vitesse.

3.1 Paramètres de synchronisation

Ces paramètres sont cruciaux pour assurer un équilibre entre l'exactitude et le temps système. L'aspect critique de ce système établit une valeur pour un certain nombre de paramètres de synchronisation suivant les indications de la figure 28.

Variable	Description	Valeur
T_B	Intervalle de *broadcast*	0-1 sec
T_L	Temps de vie d'un évènement	15 sec
T_E	Temps d'explication	5 sec

Figure 28. Paramètres de synchronisation

Dans la majorité des travaux accomplie sur les VANETs) **[72],** les événements expirent après 15 secondes. Ceci signifie que, supposant que des nœuds voyagent en moyenne à 10 mètres/sec, Ces nœuds pourront généralement voyager jusqu'à 150 mètres avant qu'une observation expire. Car avec des vitesses plus importantes, il serait beaucoup

plus difficile de comparer deux observations séparées. En outre, parce que les émissions se produisent approximativement une fois chaque demi-seconde, un événement simple pourrait se propager jusqu'à 30 sauts plus loin. Nous voulons que les événements se propagent aussi loin que possible afin de fournir plus de données au procédé de construction du modèle, mais en évitant de le maintenir jusqu'à ce que l'événement soit dépourvu de toute signification.

3.2 Construction modèle

Les nœuds honnêtes essayent périodiquement d'expliquer les observations retenues dans leur base de données. Ce processus commence par incrémenter i à partir de 0 jusqu'à N nœuds malveillants, où N est le nombre total des nœuds dans la base de données. Nous faisons appel à un mécanisme d'étiquetage (labelling) ou classification qui attribue à chaque nœud un état reflétant le degré de confiance qui lui est accordée. Pour chaque itération, nous déterminons systématiquement toutes les évaluations du *labelling* d'un nœud ***x*** par un nœud ***i*** en considérants que les informations fournies par les nœuds malveillants sont fausses. Pour chaque labelling, nous marquons, au commencement du protocole, tous les nœuds non-malveillants comme *spoof*, excepté le nœud actuel qui est marqué comme *truthful* (c.-à-d. qu'un nœud se considère honnête). Puis, commençant par le nœud actuel, nous suivons périodiquement toutes les observations, en omettant les nœuds malveillants et en marquant les nœuds visités comme *truthful* du moment qu'ils respectent les conditions du modèle de données valides qui a été défini. Tous les nœuds non visités demeureront comme spoof. En conclusion, nous exécutons un contrôle de cohérence sur ce labeling. Ce contrôle est basé sur des propriétés géométriques et d'autres spécifiques aux protocoles de communication utilisé tel que le fait qu'un nœud ne peut recevoir des messages d'un autre nœud en dehors de sa portée de communication ou que deux nœuds ne peuvent se trouver physiquement au même point géographique. Si les labellings sont cohérents, nous choisissons cette évaluation du *labelling* et nous terminons l'algorithme; autrement, nous continuons jusqu'à ce que nous trouvions une évaluation cohérente.

Contrôle de cohérence

L'élément clé de l'algorithme de construction du model exécute un contrôle de cohérence. Ceci signifie donner un label à tous les nœuds (*truthful, malicious* ou *spoof*). Nous vérifions que les observations enregistrées sont conformes en se basant sur deux règles:

1. Tous les nœuds *truthful* doivent déclarer leur position, vitesse, et *heading*. Cette déclaration doit être compatible à toutes les observations des nœuds *truthful*.

2. Pour les nœuds *malicious* ou *spoof*, ignorer les déclarations, mais vérifier que toutes les observations des nœuds *truthful* conviennent (c.à.d, qu'ils ont été physiquement détectés par d'autres nœuds *truthful*).

Le contrôle d'uniformité de deux événements est exécuté suivant la méthode «*rectangular path prediction»[72]* décrite dans la prochaine section. Puisque nous recherchons progressivement les nœuds malveillants, l'évaluation qui considéré est celle qui est conforme avec un minimum de nœuds malveillants et ainsi le plus grand nombre d'événements conformes. Dans le pire des cas, aucun des événements n'est conforme, et tous les nœuds (excepté le nœud actuel) sont étiquetés *malicious*.

Comparaison des événements

Puisque l'instant après l'enregistrement d'un événement par un nœud, ce dernier est susceptible d'avoir changé de position puisqu'il est mobile; les événements ne peuvent pas être comparés exactement. Il faudra donc, prendre en considération l'évolution dynamique de l'environnement lors des vérifications.

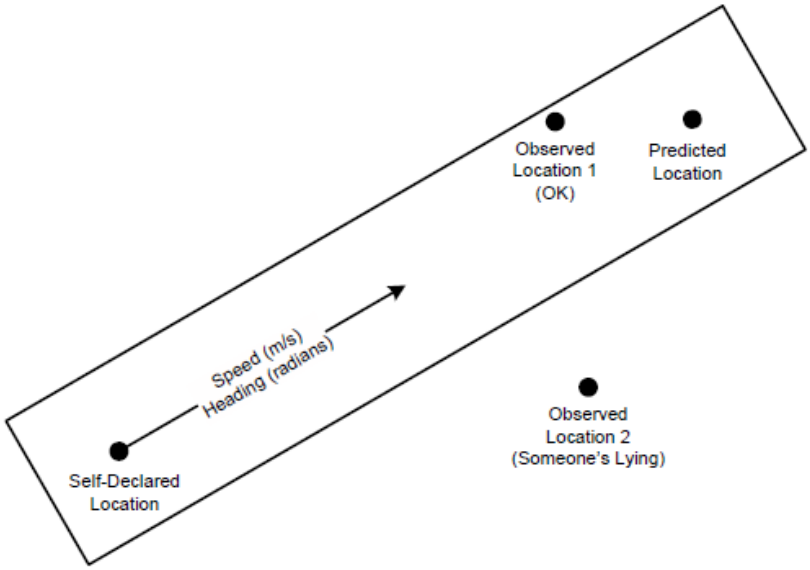


Figure 29. Rectangular path prediction

Le point dans le coin bas-gauche représente la position, la vitesse, et le *heading* d'un nœud donné (figure 29). À une autre heure (passée ou futur), un autre nœud fait une observation de ce nœud comme indiqué par les deux emplacements observés (1 et 2). La question est: ces événements sont-ils compatibles à l'emplacement avoué par le nœud? En se basant sur la vitesse et le *heading* du nœud, nous pouvons prévoir son emplacement à tout moment tant que la vitesse et le *heading* demeurent constants. Cependant, dans les scénarios réels de trafic, il y aura une certaine incertitude dans cette prévision, y compris des facteurs tels que la vitesse des nœuds (*heading* changeant) ou la dérive des horloges des nœuds. Nous prévoyons alors une marge fixe d'erreur le long du chemin prévu. Nous l'avons défini à 10 mètres dans chaque direction afin d'égaler raisonnablement l'état des routes typiques [72]. Tant que les observations font partie de ce rectangle, nous les considérons comme étant des événements conformes. Dans le scénario déployé, cette marge devrait être compatible avec les erreurs de la technologie de détection d'emplacement disponibles.

Plusieurs paramètres ont été considérés dans notre protocole de détection, parmi lesquelles on peut citer; l'affectation d'un facteur de réputation élevé a un voisin duquel on a jamais reçu une information considéré malicieuse d'après le modèle du nœud, mais maitre à jour ce facteur au fur et à mesure de l'évolution du système. Cela est en fait dû au fait que les nœuds peuvent être corrompus, manipulés ou il se peut même que leurs

identifiants soient volés par des nœuds s malhonnêtes. Ce traitement est appliqué sur le nœud lui-même mais à un degré moins sévère pour prendre en compte la manipulation des capacités sensorielles ce qui pourra l'induire à l'erreur. Un autre paramètre est le facteur de confiance qui permet au nœud de décider si une information même dite non malicieuse, pourrai contribuer à la correction de son modèle du VANET, servir à enrichir sa base locale ou être transférée a son voisinage.

```
//A la détection d'un événement par un nœud observateur O

Collect(E);//fonct reflétant la collecte d'une info par le nœud lui-même.
Attrib(<E>Oi,Pi);//fonct attribue probabilité a l'évènement collecté ou reçu
Eval(<E>Oi,Pj,M);//fonct. Evalue la validité d'un évènement/modelé.
Update(<E>Oi,REPi);//fonct mise a jour de la réputation du nœud i
Add(<E>Oi,Ki,REPi);//fonct ajout événement E à base connaissance
Send (<E>Oi,Ni,Ni,*);//fonct diffusion évènement

//A la réception d'une information de O par un nœud N

Rcv(<E>Oj,Nj);//fonct de réception de message par un nœud
Attrib(<E>Oj,Pj);
Eval(<E>Oj,Pj,M);
Update(<E>Oj,REPi,j);
Add (<E>Oj,Ki,REPi,j);
Send (<E>Oj,Nj,Ni,*);
```

4 S-LEAVE (Stinged-LEAVE)

Notre protocole est appelé ainsi (*S-leave*) parce qu'il vise à tirer profit des bienfaits des deux protocoles cité dans le chapitre précédent en même temps. Cela pourrait être expliqué par le fait que notre protocole vise dans un premier temps à exclure les entités malveillantes immédiatement à leur détection dans le système. Cela est en fait effectué même en pénalisant plusieurs entités honnêtes du système. Ces entités sont alors sacrifiées pour limiter l'effet de l'attaque et éviter l'infection du réseau par de fausses informations qui pourraient nuire à la sécurité des nœuds ou aux résultats des protocoles réseau exécutés. Le suicide des nœuds honnêtes du système est une démarche temporaire au cours du temps nécessaire à la collecte de données suffisantes pour cerner le ou les nœuds malveillants. Cette détection se base sur un seuil de nombre d'accusations du nœud dit malveillant par les autres nœuds. La détection d'un nœud se base sur le nombre des accusations. Dès que ce nombre atteint un certain seuil ***S1***, le nœud est déclaré malveillant. Le seuil S1 est un paramètre du système.

Une fois le seuil de détection atteint, tous les nœuds accusant le nœud malicieux sont rétablis sur le réseau. De plus, de nouveaux privilèges leurs sont accordés en termes de compensation et de reconnaissance pour leurs intégrité et envers le reste des nœuds du réseau. Cet avantage leur donne un privilège qui leur permettra de rester actifs même après avoir accuser d'autres nœuds malicieux dans le futur. Mais, ce privilège est limité pour prendre en compte la possibilité de manipulations ou de défaillances qui pourraient compromettre le bon fonctionnement d'un nœud même s'il est considéré comme étant honnête et même s'il a contribué a l'élimination de plusieurs nœuds malhonnêtes auparavant. Le fonctionnement détaillé du protocole proposé sera expliqué dans les points suivants:

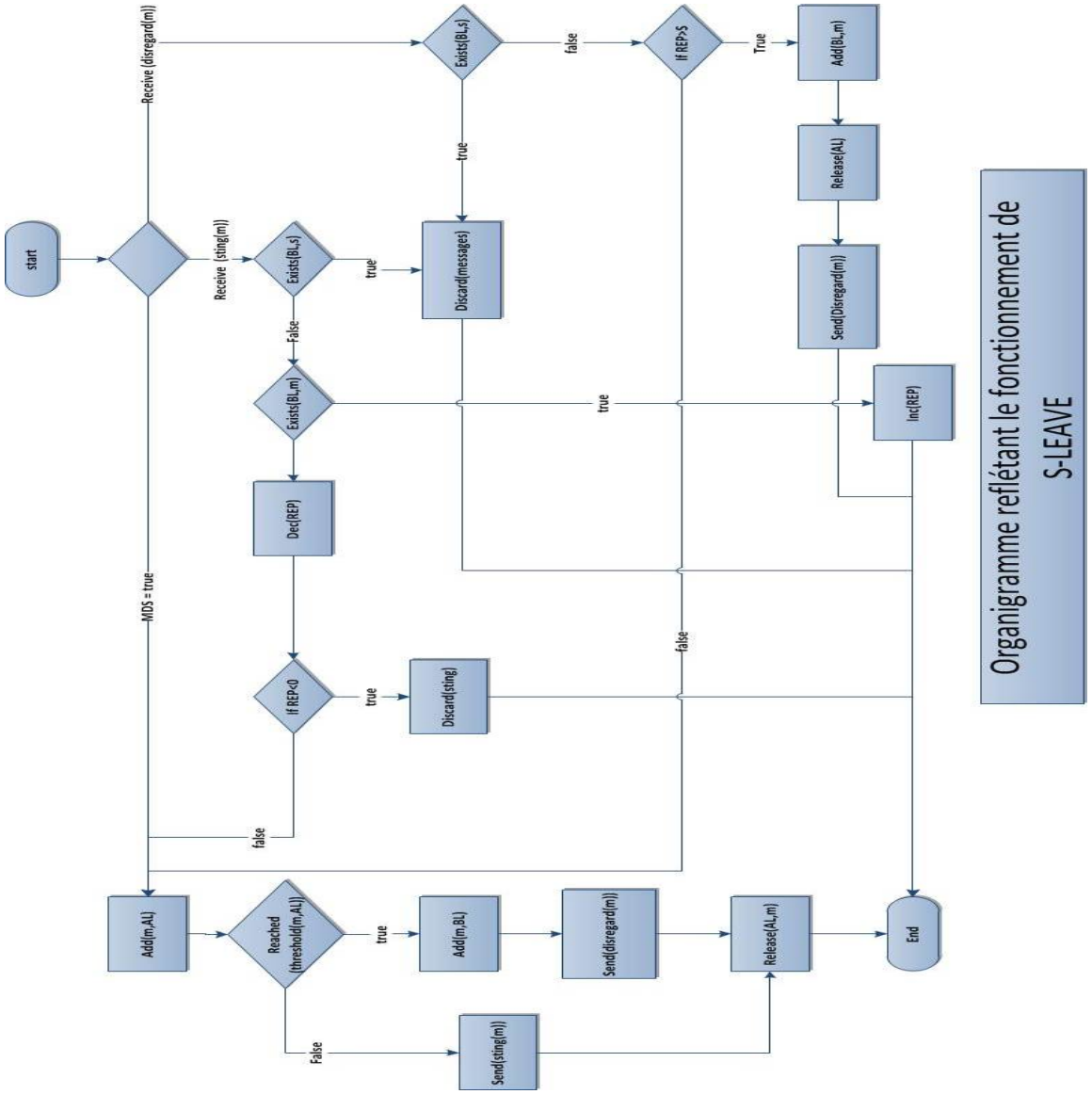
Lorsqu'un nœud détecte qu'un autre nœud m est malicieux, il génère un $sting_{a,m}$ et l'envoie à son voisinage direct s'il considère qu'il n'a pas encore collecté assez de votes contre ce nœud. Autrement, il envoie un message $disregard_{a,\mathrm{m}}$. Et, il libère tous les nœuds ayant accusé ce nœud et qui figurent dans son *Accusationlist*. De plus, il leur affecte un facteur de confiance supérieur à celui qui leurs était affecté auparavant. Ce facteur leur permettra de continuer à participer à la détection dans le future sans être éliminé de façon directe du réseau.

Ces facteurs permettront à chaque nœud d'avoir un certain historique comportemental de ses voisins, ce que lui permettra par la suite de prendre des décisions futures plus pertinentes.

Chaque nœud recevant ce *sting* calculera un facteur lui permettant de savoir s'il doit ou non ignorer les prochaines participations de la source du *sting* (le nœud accusateur) au réseau. Cela n'est effectivement effectué qu'après avoir vérifié que la source n'apparait pas dans sa *Blacklist* bien sûr. Puis, il exécute le même processus qu'un nœud qui a lui-même détecté un autre nœud malicieux pour savoir s'il doit générer un *disregard* ou non contre le nœud accusé.

Chaque nœud recevant un *disregard* de la part d'un autre nœud du voisinage vérifiera s'il attribue un facteur de confiance assez important à ce nœud pour exécuter la procédure

correspondante à la réception d'un tel message. Autrement, ce message est traité comme un simple *sting*.



4.1 Les structures utilisées

Nous utilisons, pour l'implantation de ce protocole, un ensemble de structures de données dont:

ALx : (*Accusation List*) est une liste bidirectionnelle locale au nœud x, dont la tête de chaque sous liste est le nœud suspecté malicieux pour x et les autres éléments de la liste constitue le voisinage accusant la tête.

`BLx:` (*Black List*) est une liste unidirectionnelle locale au nœud x, dont chaque élément est un nœud dont la culpabilité a été établie pour x.

`REPx:`(*reputation list*) est une liste d'enregistrement permettant de maintenir à jours la vue de x sur tous ses voisins (leurs réputations).

4.2 Les messages utilisés

Un certain nombre de messages sont utilisés pour le contrôle des nœuds malicieux dont:

`Sting:` Sting(s, m, $Sign_s$, r)

est un message indiquant que le nœud ***s*** a détecté que le nœud ***m*** est un nœud malicieux et, qu'il a inclus sa signature dans le message et que ***r*** est le dernier nœud relai sur lequel a transiter le message.

`Disregard:` disregard (s, m, $Sign_a$, r)

est un message indiquant que le nœud ***s*** a établie après avoir atteint le seuil de culpabilité requis que le nœud ***m*** est un nœud malicieux, qu'il a inclus sa signature dans le message et, que ***r*** est le dernier nœud relai sur lequel le message a transiter.

4.3 Les fonctions utilisées

Pour plus de clarté, nous définissons les principales fonctions utilisées:

MDSa (m): prend la valeur « true » pour signifier que le ***MDS*** (*Misbahavior Detection System:* protocole de détection décrit dans la première phase du chapitre**)** du nœud a détecté que le nœud m est un nœud malicieux.

Add (m, s, AL_a): est une fonction permettant au nœud a d'ajouter le nœud m a son *accusationlist* en stipulant que c'est le nœud ***s*** qui vient de l'accuser.

Add (m, BLa) fonction permettant au nœud a d'ajouter le nœud m a sa *blacklist*.

Reached(threshold(m, ALa)) est une fonction dont le résultat est vrai si et seulement si le seuil requis pour l'établissement de la culpabilité de ***m*** pour ***a*** est atteint.

Discard(Sting(s, m, $Sign_s$, r)): fonction permettant de supprimer les informations relatives à un message.

Exists(s, Bl_a) fonction permettant de rechercher si un élément ***s*** apparait dans une liste ***Bl_a***.

Release(Al_a, x)*:* fonction permettant de supprimer un élément ***x*** de la liste ***AL_a***.

Procédure Détection:

```
// Détection d'un nœud malicieux m par un nœud a
If MDSa(m)=true then
     Add(m,a,ALa);
     If reached(threshold(m,ALa)) then
          Add(m,BLa);
          Send «disregard(a,m,Signa,a)»;
          foreach x Liste(m,ALa) do
               release(Ala,x);
               inc(REPa,x,2);
          end;
     Else
          Send «Sting(a,m,Signa,a)»
     End;
End;
```

Procédure Réception Sting:

```
//A la réception d'un message sting(s,m) par un nœud a

RCV «Sting(s,m,Signs,r)»;
If exists(s,Bla)=true or exists(r,BLa)=true then
     discard«Sting(s,m,Signs,r)»
Else
     if exists(m,BLa)=true then
          inc(REPa,s,2);
          inc(REPa,r,2);
     else
          dec(REPa,s,1);
          dec(REPa,r,1);
          if val(REPa,s)<0 or val(REPa,r)<0 then
               discard«Sting(s,m,Signs,r)»;
          else
               add(ALa,s,m)
               If reached(threshold(m,ALa)) then
                    Add(m,BLa);
                    foreach x Liste(m,ALa) do
                         release (Ala,x);
                         inc(REPa,x,2);
                    end;
                    Send «disregard(a,m,Signa,a)»;
               Else
                    Send «Sting(a,m,Signa,a)»
               End;
          End;
     End;
End;
```

Procédure Réception Disregard:

```
 //A la réception d'un mess. disregard(s,m) par un nœud a

RCV «disregard(s,m,Signs,r)»;
If exists(s,BLa)=true or exists(r,BLa)=true then
     dicard«disregard(s,m,Signs,r)»;
Else
     if val(Repa,s)>=V1 and val(Repa,r)>=1 then
          add(BLa,m);
          foreach x Liste(m,ALa) do
               release (Ala,x);
               inc(REPa,x,2);
          end;
          send «disregard(s,m,Signs,a)»
     else
          dicard«disregard(s,m,Signs,r)»;
          call>>>RCV «Sting(s,m,Signs,r)»;
     end;
end;
```

V1 est un paramètre système utilisé par le nœud pour refléter le de degrés de méfiance dans ses attitudes envers ses voisins. Plus V1 est grand plus le nœud est méfiant. Pour notre cas d'étude nous l'avons fixé à 3

5 CONCLUSION

Les protocoles introduits dans ce chapitre constituent une des toutes premières contributions adressant la problématique de l'exclusion des nœuds malicieux dans les réseaux véhiculaires. Après avoir minutieusement étudié le problème d'exclusion des nœuds malicieux et après avoir étudié les différents aspects proposés dans les solutions existantes, nous avons essayé de proposer une contribution prenant en compte les faiblesses des solutions existantes et essayant d'améliorer la prise en compte des nœuds malicieux sur un réseau de type VANET.

Notre protocole ayant été conceptuellement défini et détaillé, il ne reste à évaluer ses performances du point de vue réseau. Nous avons, à cet effet, choisi de mener une évaluation par le biais d'un simulateur de réseaux. Cette simulation et les résultats de l'évaluation vont l'objet des discussions du chapitre suivant.

Chapitre V

Évaluation des Performances de S-LEAVE

1 Introduction

L'évaluation d'un protocole sur un système doit passer par une mise en œuvre pratique dans des conditions réelles. Cette approche quoique directe et très efficace, s'avère très coûteuse. En effet, en l'absence d'une plateforme d'expérimentation réelle, la simulation reste le moyen le plus accessible.

Plusieurs simulateurs sont disponibles tels que NS2, Opnet, GLOMOSIM, etc... Notre choix est porté sur NS2 car c'est le simulateur de réseaux le plus utilisé par la communauté ad hoc, en plus du fait que la plutpart des protocoles traités dans ce mémoire, ont été simulé avec NS2.

L'environnement des VANETs possède des caractéristiques particulières, principalement liées au réseau routier qui définit une mobilité dirigée, des vitesses très grandes, un espace géographique illimité et un grand nombre de véhicules dont la connectivité dépend des facteurs de la route outre les caractéristiques des technologies sans fil. Il est clair que les modèles de mobilités définis sous NS comme le RDM, RWP, RW, ne sont plus adaptables à notre environnement, d'où la nécessité d'utiliser de nouveau générateur de modèles de mobilité adapté aux réseaux véhiculaires. Il existe plusieurs générateurs de mobilité tels que SUMO, VISSIM, CARISMA, etc…. Nous avons choisi d'utiliser le SUMO à cause de sa disponibilité (il est gratuit) en plus du fait qu'il permet de générer facilement des modèles de mobilité très réalistes.

Ce chapitre est divisé en deux grandes parties. Dans la première partie nous présentons les différentes techniques utilisées dans l'évaluation des performances des protocoles dans un environnement mobile ad hoc véhiculaire. Nous présentons ensuite, les différents modèles de mobilité ainsi que les générateurs de mobilité. Tout au long de ces présentations, nous faisons le point sur les différents choix auxquels nous avons opté.

Dans la deuxième partie, nous présentons les différentes simulations effectuées pour les trois protocoles (LEAVE, Stinger et S-LEAVE) ainsi que l'interprétation des résultats obtenus.

2 Les techniques d'évaluation des performances:

Il existe différentes techniques pour l'évaluation des performances d'un système [73]:

2.1 La mesure (émulation):

Il s'agit de faire des mesures et de les analyser directement sur un système réel. Cette technique permet de comprendre le vrai comportement du système. Cependant elle n'est pas toujours réalisable car le fonctionnement du système réel pourrait être perturbé, en plus, les résultats issus de la mesure, ne reflètent qu'une seule trajectoire du système.

2.2 La modélisation:

Il s'agit de réduire le système en un modèle mathématique (Les automates, Les réseaux de pétri, Les approches probabilistes, Les approches déterministes, etc...) et de l'analyser numériquement. Généralement, des hypothèses sont posées pour simplifier l'étape de modélisation du système et rendent l'évaluation numérique faisable. Ces hypothèses simplificatrices peuvent toucher à la fidélité de la représentation du système.

2.3 La simulation:

Il s'agit d'implanter un modèle simplifié du système à l'aide d'un programme de simulation adéquat. Cette méthode traduit le comportement du système à évaluer d'une manière réaliste. La simulation permet en plus de visualiser les résultats sous forme de graphes faciles à analyser et à interpréter. La simulation n'est pas une méthode exacte, et nécessite de prêter une attention particulière aux interprétations des résultats. La simulation a pour avantage de permettre l'étude du système en variant ses divers paramètres.

3 Environnement de simulation:

Il existe plusieurs outils de simulation, citons: NS2 (network simulator 2), GloMoSim (Global Mobile Simulator), Opnet (Open Network), …etc. Afin de faciliter le choix définitif d'un simulateur, des aspects peuvent être considérés tels que: la précision des modèles, le passage à l'échelle, la possibilité d'analyser les résultats, etc...

Notre choix s'est porté sur le Network Simulator NS2 [74,75] et ceci pour différentes raisons qui sont:

- Son modèle libre permet l'ajout très rapide de modèles correspondants à des technologies émergentes. Ceci le rend meilleur logiciel de simulation par évènements discrets d'après les spécialistes des télécommunications,

- ❖ Le logiciel NS2 est fourni avec une interface graphique (NAM) permettant de démontrer le fonctionnement des réseaux,

- ❖ Il permet d'étendre le simulateur et de le personnaliser, et il donne une liberté de programmation vu qu'il est fourni avec son code source qu'on peut modifier et recompiler autant qu'on le souhaite.

- ❖ La plupart des protocoles dédies aux VANET sont simulés avec NS.

La distribution standard de NS2 est exécutée sous Linux. Cependant, elle peut être utilisée sous Cygwin (émulateur Linux pour Windows). Notre choix s'est porté sur la plateforme de Linux pour la simple raison que NS2 est nettement plus facile à installer et à configurer sous linux que sous Windows.

3.1 Le Network Simulator NS2:

Le simulateur de réseaux NS2 est un outil (logiciel) de simulation de réseaux informatiques. Il repose sur deux langages de programmation C++ et OTCL (Object Tools Command Langage) dérivé de TCL (Tools Command Langage). Il offre plusieurs outils tels que le Nam (Network Animator), qui est un outil d'animation basé sur Tcl/TK, utilisé dans NS afin de visualiser le tracé de simulation des réseaux, ainsi que les tracés de données. Le ns2 est aussi composé de l'outil Xgraph qui permet de tracer des courbes à partir de fichier de données.

Un modèle de réseau sous NS est constitué de:

- Nœuds: Un nœud est une collection de 'classifiers' et d'agents. Il existe deux types de nœuds dans NS, les nœuds unicast et les nœuds multicast.

- Liens de communication entre les nœuds: Servent à raccorder les nœuds entre eux.

- Agents de communication: Ils modélisent les constructeurs et les consommateurs des paquets. Ces agents sont attachés aux nœuds et connectés les uns aux autres, afin d'échanger des données entre les nœuds.

- Application: Génère le trafic de données selon certaines applications (CBR, FTP), et se sert des agents de transport.

3.1.1 Les modèles de mobilité sous NS2 [03]:

NS2 définit trois modèles de mobilité implémentés qui sont:

- Le modèle Random WayPoint (RWP):

Dans ce modèle, la mobilité des nœuds est aléatoire et leur distribution est uniforme dans l'espace de simulation. En effet il consiste:

- Au placement d'un certain nombre de mobiles dans une zone carrée dans laquelle ils ne peuvent pas sortir,
- Á l'affectation d'une position, d'une vitesse et d'une destination initiale à chaque mobile,
- Au déroulement proprement dit de la simulation, où à chaque fois que les mobiles atteignent leurs destinations dans le carré, ils repartent vers d'autres destinations choisies aléatoirement après un éventuel temps de pause.

- Le modèle Random Walk (RW):

Un nœud mobile dans ce modèle se déplace de son endroit courant à un nouvel endroit en choisissant aléatoirement une vitesse et une direction dans des gammes prédéfinies ([speedmin, speedmax] et [0, 2π] respectivement), suivant lesquelles il se déplace. Lorsqu'un nœud mobile atteint la frontière de simulation, il rebondit avec l'angle déterminé par la direction entrante puis il continue le long du nouveau chemin.

- Le modèle Random Direction Model (RDM):

Ce modèle essaye d'alléger le modèle RWP, en fournissant un nombre de voisins constant. Les nœuds mobiles se déplacent vers une direction sélectionnée aléatoirement en tant que modèle de mobilité de Random Walk, où ils se déplacent vers la frontière de la simulation dans cette direction. Une fois que la frontière est atteinte, et pendant un certain temps, le nœud mobile fait une pause, puis choisit une autre direction angulaire entre (0°et 180°) afin de poursuivre son chemin.

3.1.2 Le langage TCL/OTCL:

Le TCL (Tools Command Language) est un langage interprété, traité par un interpréteur TCL (NS par exemple). Les programmes écrits en TCL sont en fait des fichiers texte constitués de commandes. Ces fichiers sont nommés scripts. Le langage OTCL est une extension orientée objet de TCL.

Jusqu'ici, nous avons présenté les différents outils permettant la simulation complète d'un réseau. Le schéma suivant (Figure 30) représente les démarches à suivre pour utiliser ces outils:

Á travers OTCL, l'utilisateur décrit l'environnement de la simulation: La topologie du réseau, les caractéristiques des liens physiques, les protocoles utilisés…etc. Cette description n'est rien qu'un ensemble de commandes écrites dans un fichier texte, appelé « script ». NS2 interprète le script OTCL et exécute la simulation. Les résultats obtenues (Fichier trace et fichier NAM) peuvent être visualisés avec l'outil NAM (Network Animator), et analysés à partir des courbes tracées par l'outil Xgraph.

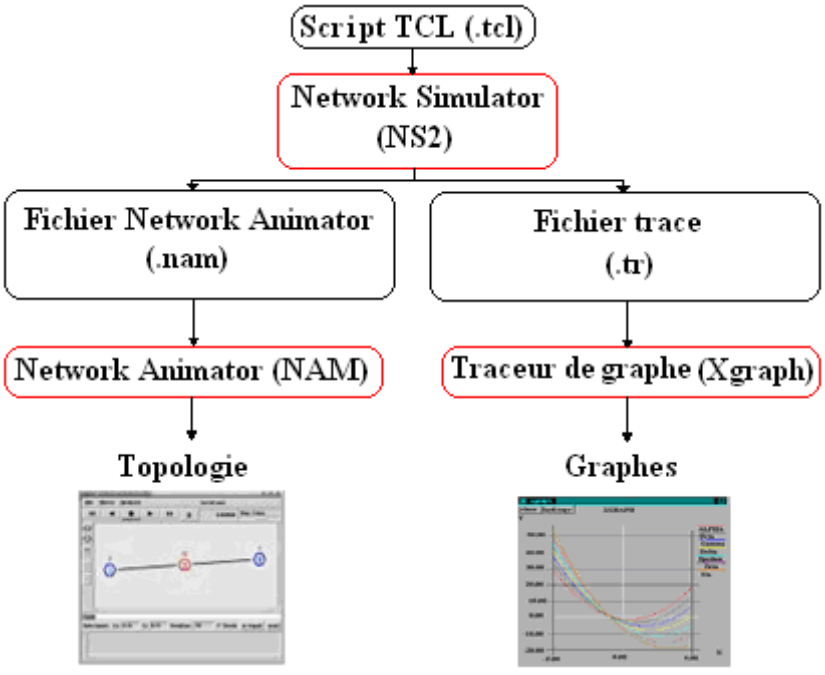


Figure 30. Flot de simulation avec NS2.

3.2 Générateurs de mobilité:

Beaucoup d'études ont montré qu'un modèle de mobilité réaliste avec un niveau suffisant de détails est essentiel pour la précision des résultats de simulation du réseau [77].

Aucun des modèles de mobilité offerts par NS ne convient aux VANETs, à cause des caractéristiques propres aux réseaux véhiculaires qui ne sont pas pris en charge par ces modèles comme:

- la mobilité qui est dirigée et influencée par le code de la route et le trafic routier,
- et, les véhicules qui sont dotés d'une grande vitesse,

ce qui cause un changement rapide de la topologie, des partitionnements très fréquents et de courtes durées de liaisons de communication. Tout ça nécessite l'utilisation d'outil de génération de scénarios de mobilité adéquat. Il existe plusieurs outils de simulation:

- Le simulateur SUMO:

Le SUMO (Simulation of Urban MObility) est un simulateur de mobilité open source programmé en C++. Il permet de générer des scenarios de mouvement au niveau microscopique, c'est-à-dire qu'il peut générer des routes, des véhicules, ainsi que des feux de signalisation. Il est accompagné d'un outil appelée MOVE (*MObility model generator for VEhicular networks*) [78], qui permet de convertir les mouvements crée par le SUMO en langage utilisé par des simulateurs tels que NS2, OPNET. Les réseaux routiers simulés peuvent être soit générés manuellement ou importés depuis des bases de données. Ce simulateur comporte une interface graphique permettant de contrôler le début de la simulation. Le principal inconvénient du SUMO est que l'itinéraire des véhicules est calculé avant le début de la simulation, c'est-à-dire que les véhicules ne pourront pas changer d'itinéraire suite à des informations reçues par le réseau au cours de la simulation.

- Le simulateur VISSIM:

C'est un logiciel de simulation microscopique très détaillé. Il permet une représentation de la réalité, aussi bien en urbain qu'en interurbain. Tous les modes motorisés sont pris en compte, ainsi que les cyclistes et les piétons. Il fournit des interfaces programmables, ce qui lui donne la possibilité d'être intégré à d'autres programmes.

L'inconvénient majeur du VISSIM est que l'utilisation des cartes routières nécessite l'utilisation d'un autre outil qui est le convertisseur de carte VISUM. Un autre inconvénient commun au SUMO est que l'itinéraire des véhicules est calculé avant le début de la simulation.

- Le simulateur CARISMA:

Le simulateur de trafic CARISMA est un simulateur réaliste contenant des dispositifs microscopiques et macroscopiques. Le principal avantage du CARISMA est que le calcul de l'itinéraire des véhicules se fait durant la simulation. Les obstacles de la route (bâtiments etc.…) sont pris en compte durant les communications inter-véhiculaire d'où la nécessité de calculer périodiquement la connectivité des véhicules. Le principal inconvénient est que cet outil existe seulement en version payante.

Notre choix s'est porté sur le MOVE, qui est utilisé avec le SUMO, pour différentes raisons qui sont:

- Il est gratuit,
- Il permet de générer facilement des modèles de mobilité réalistes pour la simulation des réseaux VANET,
- Il est fourni avec son code source qu'on peut modifier et recompiler au besoin,
- le MOVE fournit un ensemble d'interfaces graphiques qui permettent à l'utilisateur de générer rapidement des scénarios de simulation réaliste sans se soucier de la rédaction des scripts de simulation,
- Il génère un fichier trace qui peut être directement utilisé par des outils de simulation NS2 ou OPNET.
- Il permet de visualiser le mouvement des véhicules, et cela grâce au SUMO.

3.3 Le simulateur MOVE:

C'est un logiciel programmé en java pour générer des simulations du trafic routier SUMO d'une manière rapide et en utilisant l'interface graphique. Il permet de générer un fichier trace qui contient des informations concernant le mouvement des véhicules, et qui peut être directement utilisé par des outils de simulation comme NS-2.

Le MOVE comporte deux composants principaux: Map Editor et Vehicle Movement Editor, comme le montre la Figure 31.

- Map Editor:

C'est un éditeur de carte, employé pour créer la topologie de la route. La création de la carte routière peut se faire de trois manières différentes: elle peut être créée manuellement par l'utilisateur, générée aléatoirement ou bien tirée d'une base de données TIGER (Topologically Integrated Geographic Encoding and Referencing, du bureau de recensement des Etats-Unis) sur la base d'une carte électronique réelle.

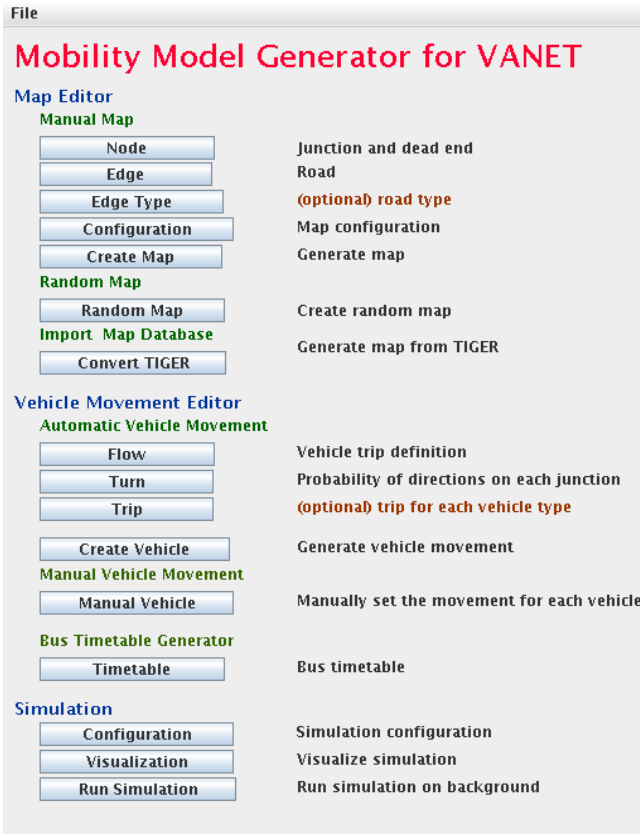


Figure 31. l'interface graphique du MOVE.

- Vehicle Movement Editor:

C'est un éditeur de mouvement qui permet de générer automatiquement ou manuellement le mouvement des véhicules. Grâce à cet éditeur, l'utilisateur peut spécifier plusieurs propriétés concernant les itinéraires des véhicules incluant le nombre de véhicules dans une route particulière, l'heure de départ des véhicules, l'origine et la destination des véhicules, la durée du voyage, la vitesse des véhicules (accélération, décélération, vitesse maximale), etc... En outre, l'utilisateur peut définir la probabilité

qu'un véhicule prenne une direction donnée lorsqu'il est dans une jonction (par exemple 0.5 pour tourner à gauche, 0.3 pour tourner à droite et 0.2 pour aller directement).

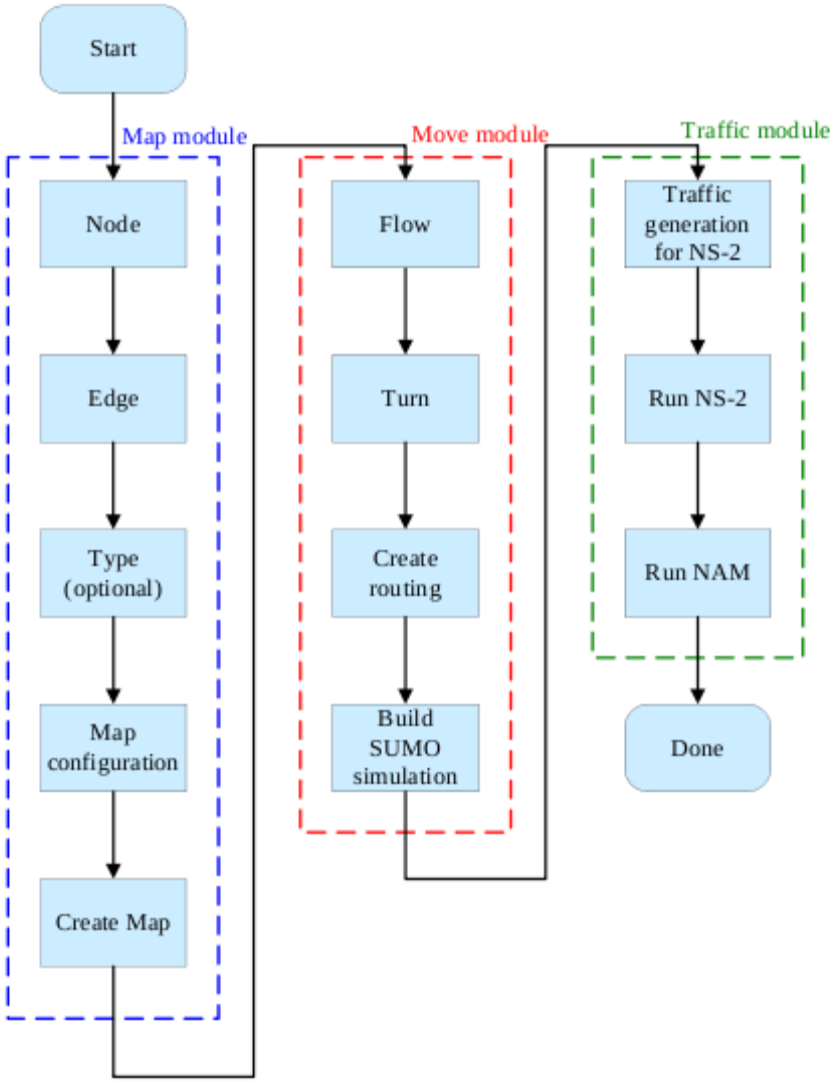


Figure 32. Génération du fichier de mobilité avec MOVE.

Le schéma précédent (Figure 32) résume les étapes à suivre pour la génération du fichier de mobilité avec MOVE. En effet la création du fichier de mobilité se fait en trois étapes:

- **Map module:** cette partie permet de crée la topologie de la carte routière (nombre de route, nombre de voie par route, feux de signalisation, longueur des routes, vitesse maximale autorisé sur chaque route,...etc).

• **Move module:** elle permet de créer les véhicules qui vont se déplacer sur la carte crée précédemment. Elle permet de les initialiser, c'est-à-dire spécifier leur point de départ, leur point d'arrivée, la vitesse maximale qu'ils peuvent atteindre, ainsi que d'autres propriétés

3.4 Mise en œuvre de la simulation:

La mise en oeuvre de la simulation doit passer par l'installation des outils de simulation. Pour NS-2 nous avons utilisé le package NS-ALLINONE-2.34 (contenant NS-2 en version 2.34 ainsi que les principaux programmes requis). L'installation du SUMO et du MOVE, qui permet de convertir les mouvements crées par le SUMO en langage TCL, requière de nombreuses librairies, des compilateurs à jour et des drivers adaptés. De plus la compilation de certains codes est très longue, une bonne connaissance du système d'exploitation est indispensable pour ne pas perdre un temps précieux (En annexe, les étapes d'installation du NS-2, du SUMO et du MOVE).

3.4.1 génération des scénarios de mobilité:

Pour réaliser un scénario de mobilité, nous avons besoin de deux éléments: une carte routière et un flux de véhicules en mouvement. Ces deux éléments peuvent être générés manuellement ou aléatoirement. Nous avons choisi de créer nous même les carte routière et de générer aléatoirement le mouvement des véhicules sur ces cartes afin d'avoir des simulations réalistes.

- Création des cartes routières

La création d'une carte routière est effectuée à l'aide du Map Editor. Cela nécessite l'introduction de deux types d'informations ' Node' et ' Edge'.

• **Node**

C'est un point particulier de la carte qui peut être soit une jonction ou une extrémité d'une route. Cependant, si le 'Node' est une jonction, alors il peut être soit une jonction normale ou avec feu de signalisation (traffic lights) (voir Figure 33). Chaque 'Node' est caractérisé par son identificateur (ID), ses coordonnes (X,Y), et sa nature (feu de signalisation ou extrémité de la route). Ces informations sont enregistrées dans un fichier <nom du fichier>.nod.xml.

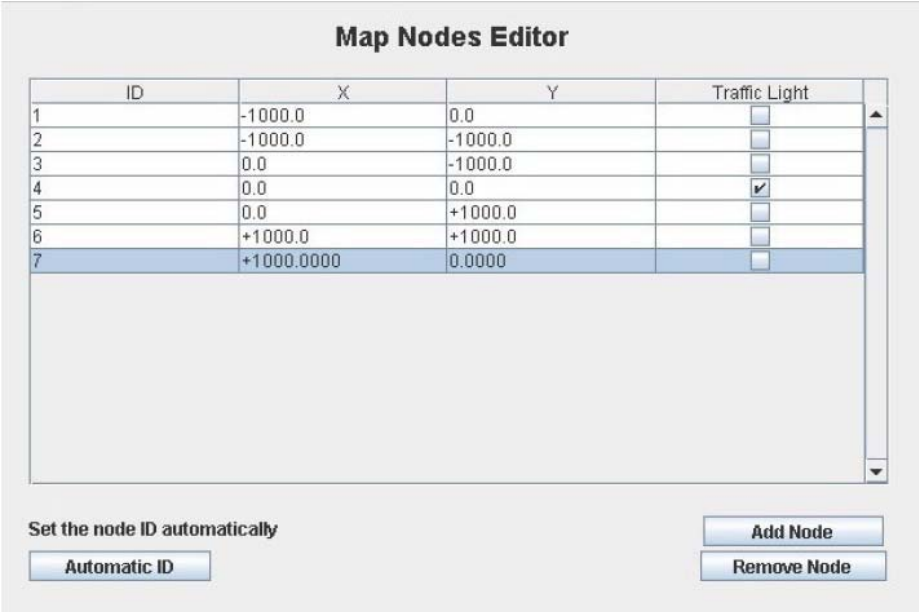


Figure 33. Editeur de 'Node'.

- **Edge**

C'est la route qui relie deux 'Nodes' dans la carte. Chaque 'Edge' est caractérisé par un identificateur (ID), une longueur (Length), une priorité (priority), ses extrémités ('Node' de départ et 'Node' d'arrivée), ainsi qu'une vitesse (Speed) qui représente la vitesse maximale qu'un véhicule peut atteindre sur cet 'Edge' (voir Figure 34). Il est possible de spécifier le nombre de voies que peut contenir un 'Edge'. Ces informations sont enregistrées dans un fichier <nom du fichier>.edg.xml.

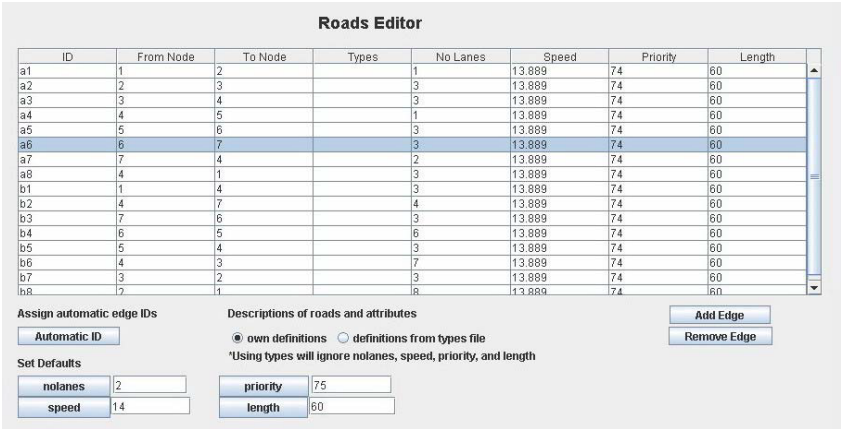


Figure 34. Editeur de 'Edge'.

Il faut préciser que le mouvement d'un véhicule sur un 'Edge ' eij se fait du 'Node' i vers le 'Node' j, c'est-à-dire qu'un Edge ne peut avoir qu'un seul sens de circulation. Si on veut que le véhicule se déplace dans le sens inverse, il faudra créer un autre 'Edge' 'eji'.

Afin d'avoir une carte routière bien structurée, nous avons choisi d'utiliser les notations suivantes durant toute la simulation:

- L'identificateur de chaque 'Node', est sous la forme 'node i' (i=1,….).

Exemples:

ID: "node1" X: "0.0" Y: "0.0" Traffic_light: true.
ID: "node2" X: "100.0" Y: "0.0" Traffic_light: false.
ID: "node3" X: "0.0" Y: "100.0" Traffic_light: true.

- L'identificateur de chaque 'Edge', est sous la forme 'e jk '. Les 'j,k' font référence au 'i' constituant les extrémités (node) du Edge.

Exemples:

- ID: "e12" from_node: "node1" to_node: "node2" type: "" (lesser vide) no_lanes: "1" speed: "40.0" priority: "50".

- ID: "e13" from_node: "node1" to_node: "node3" type: "" (lesser vide) no_lanes: "2" speed: "45.0" priority: "60".

- Lors de la création des cartes routières, il faut faire en sorte à ce que le point (Xmin,Ymin) se situant à l'extrémité inferieure gauche des bornes de la carte ait les coordonnées (0,0), autrement il y'aura une incompatibilité entre les coordonnées du MOVE et ceux du NS-2, c'est-à-dire que les différents points constituant la carte routière, dans NS-2, aurons des coordonnées différents de ceux introduit dans le MOVE.

Exemple:

Etant donnée les coordonnées d'un Node (200,300):

- Si (Xmin,Ymin)=(0,0), alors le Node aura les mêmes coordonnées dans NS-2.

- Si (Xmin,Ymin)= (100,100), alors le Node aura les coordonnées (100,200),

(X=200-100=100, Y=300-100=200) dans NS-2.

- Génération du mouvement des véhicules

La génération du mouvement des véhicules se fait à l'aide du 'Vehicle Movement Editor' en spécifiant l'identificateur de chaque flux de véhicules, son point de départ et son point d'arrivée (Edge), le début et la fin du mouvement, ainsi que le nombre de véhicules dans chaque flux (voir Figure 35). Ces informations sont enregistrées dans un fichier: <nom du fichier>.flow.xml.

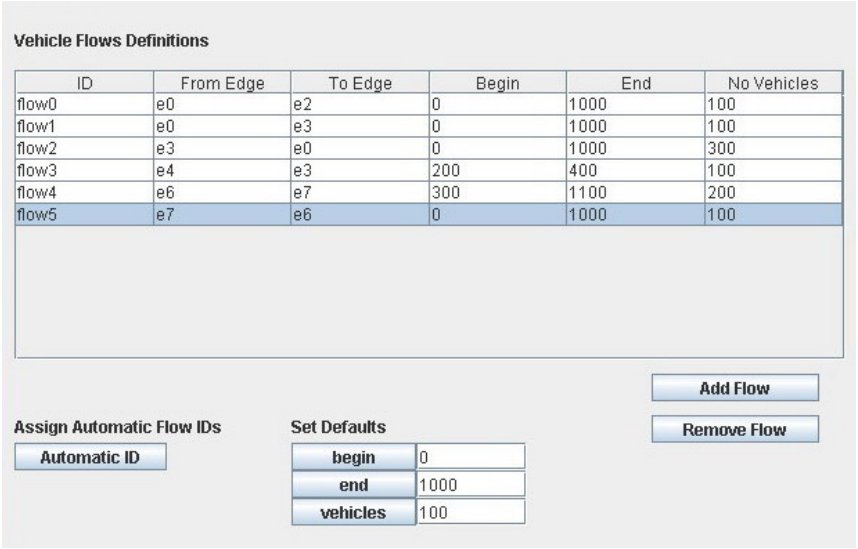


Figure 35. Éditeur de mouvement de véhicules

Afin de pouvoir déterminer le mouvement de chaque véhicule durant la simulation, nous avons affecté à chaque flux un nombre de véhicule égal à un.

Exemple:

-ID: "flow1" From Edge: "e12" To Edge: "e21" Begin: "0" End: "1000" No Vehicles: "1".

3.4.2 Codification des cartes routières:

Le codage des cartes est effectué par une procédure de codage, en utilisant les fichiers précédents (nod.xml et edg.xml). Ce codage permet de créer les structures de données qui seront enregistrées dans un fichier prêt à être utilisé dans le script TCL. La procédure de codage permet de parcourir le fichier nod.xml, les coordonnées ainsi que l'identificateur de

chaque point de la carte seront enregistrés en faisant appel à « coord_ » qui est un objet de la classe « carte_rout », et qui a pour attributs: id_nod, X_nod, Y_nod.

La Figure 36 représente un extrait d'un fichier nod.xml.

```
- <nodes>
    <node id="node1" x="200.0" y="0.0" type="priority"/>
    <node id="node2" x="1000.0" y="0.0" type="priority"/>
    <node id="node3" x="1400.0" y="0.0" type="priority"/>
    <node id="node4" x="2200.0" y="0.0" type="priority"/>
    <node id="node5" x="2600.0" y="0.0" type="priority"/>
```

Figure 36. Extrait d'un fichier nod.xml.

En parcourant le fichier edg.xml (Figure 37), les routes ainsi que leur longueur seront enregistrées respectivement dans les matrices MAT (i,j) et L(i,j), où les 'i' représenteront la première extrémité de la route, et les 'j' la deuxième extrémité.

Exemple:

Si on prend le fichier de la Figure 37 alors:

MAT (node1, node10)=e110. L (node1, node10)= 200.
MAT (node2, node11)=e211. L (node2, node11)= 200.
MAT (node3, node12)=e312. L (node3, node12)= 200.
MAT (node4, node13)=e413. L (node4, node13)= 200.
MAT (node5, node14)=e514. L (node5, node14)= 200.

```
- <edges>
    <edge id="e110" fromnode="node1" tonode="node10" priority="75" nolanes="2" speed="40" length="200"/>
    <edge id="e211" fromnode="node2" tonode="node11" priority="75" nolanes="2" speed="40" length="200"/>
    <edge id="e312" fromnode="node3" tonode="node12" priority="75" nolanes="2" speed="40" length="200"/>
    <edge id="e413" fromnode="node4" tonode="node13" priority="75" nolanes="2" speed="40" length="200"/>
    <edge id="e514" fromnode="node5" tonode="node14" priority="75" nolanes="2" speed="40" length="200"/>
```

Figure 37. Extrait d'un fichier edg.xml.

La détection des intersections se fait à l'aide de la matrice MAT. Si un Nœud donné relie plus de trois routes alors c'est une intersection. En parcourant tous les éléments de la matrice, les intersections seront sauvegardées dans une liste «listinters », et les informations concernant chaque intersection seront sauvegardées en faisant appel à « inters_ » qui est un objet de la classe « carte_rout ».

3.4.3 Direction de mouvement

La direction du mouvement indique la trajectoire de circulation des véhicules. Cette direction est représentée par l'identificateur de la route sur laquelle se trouve le véhicule: si un véhicule est sur la route 'e110' alors sa direction est 'e110', s'il est sur la route 'e25' alors sa direction est 'e25' etc...

Afin de déterminer la direction de chaque véhicule durant notre simulation, nous avons eu recours au fichier <nom fichier>.move.trace (Figure 38), crée par le SUMO lors de la génération du mouvement des véhicules.

Figure 38. Extrait d'un fichier.move.trace.

3.4.4 Les coordonnées et vitesses des véhicules

Le fichier de mobilité généré par le MOVE nous permet d'avoir les coordonnées et la vitesse de chaque véhicule durant la simulation. Cependant, un des inconvénients du MOVE est que ces informations sont générées chaque unité de temps (Figure 39), c'est-à-dire qu'on ne pourra pas avoir les informations d'un véhicule donnée à un temps T€]0,1[ou T€]1,2[à titre d'exemples. Mais plutôt a des unités de temps: T=0, T=1, T=2, T=3 etc...

```
# Now produce node movements
$ns_ at 1.0 "$node_(0) setdest 6.0 300.0 1.67"
$ns_ at 1.0 "$node_(1) setdest 6.000001 600.000001 1.92"
$ns_ at 2.0 "$node_(2) setdest 300.000002 7.000002 2.59"
$ns_ at 2.0 "$node_(0) setdest 10.000003 300.000003 3.38"
$ns_ at 2.0 "$node_(1) setdest 10.000004 600.000004 3.88"
$ns_ at 3.0 "$node_(2) setdest 300.000005 12.000005 4.55"
$ns_ at 3.0 "$node_(0) setdest 15.000006 300.000006 5.18"
$ns_ at 3.0 "$node_(1) setdest 16.000007 600.000007 6.02"
$ns_ at 4.0 "$node_(2) setdest 300.000008 19.000008 6.91"
$ns_ at 4.0 "$node_(0) setdest 21.000009 300.000009 6.58"
```

Figure 39. Extrait du fichier.tcl générer par le MOVE.

3.4.5 Les feux de signalisation

La gestion des feux de signalisation se fait automatiquement par le MOVE. Cependant les fichiers générés par ce dernier ne précisent pas la durée de séjour des véhicules dans une intersection signalisée (Afin de palier à ce problème, il faut évaluer la durée moyenne des véhicules dans chaque intersection, et cela en créant plusieurs scenarios de mobilité. Ces informations sont sauvegardées dans le fichier « temp_inter.tcl » qui sera intégré au script TCL).

4 Mise en œuvre comparative des protocoles

Nous avons réalisé plusieurs simulations pour évaluer les performances de nos propositions en termes de paramètres de sécurité et de performance. Puis, nous avons comparé ces résultats à ceux de l'algorithme LEAVE et Stinger proposé respectivement par [35] et [71]. Pour cela, nous avons exploité le simulateur NS-2 [75] pour implémenter l'algorithme proposé. En plus, puisque la mobilité est prise en considération dans le modèle d'exécution, pour générer des modèles de mobilité des nœuds nous avons utilisé le modèle « Manhattan » [72] pour évaluer notre contribution car, il se rapproche plus des déplacements qui peuvent être effectués par nos entités. Dans ce modèle, chaque nœud choisit sa direction et sa vitesse de déplacement après chaque intervalle de temps et il peut y avoir de même des périodes de pause. Les déplacements des nœuds sont limités à la topologie fournit par le générateur (ils ne peuvent pas se déplacer dans une zone non définit dans la topologie c-a-d ils ne peuvent pas sortir des routes) dans une zone géographique de 2,4 km par 2,4 km. La vitesse est fixée à 60km/h, et nous pouvons la varier selon les besoins de la simulation. Pour comparer les performances de notre

contribution aux autres algorithmes nous allons suivre le plan d'évaluation de performances suivant:

4.1 Les paramètres de simulation

L'objectif de cette étude est de proposer une méthode de détection et d'élimination des nœuds malicieux dans un VANET qui soit efficace en termes de paramètres de sécurité et de performances sur ces réseaux. Cette évaluation est faite en analysant quelques méthodes existantes et en utilisant notre modèle de mobilité. Le système étudié consiste en un réseau véhiculaire équipé de RSU déployé de manière uniforme sur une partie du réseau et respectant le modèle de mobilité choisit.

Les paramètres qui affectent notre système sont:

Les paramètres liés au réseau

- La nature de la surface: la zone d'intérêt ou la zone de surveillance peut être sous forme rectangulaire ou carrée. Nous nous plaçons dans le cas des surfaces 2D.
- Le nombre de nœuds déployés .

Les paramètres liés aux modèles de mobilité

Notre étude se limite à un type de modèle qui est le modèle Manhattan [72]. La génération d'un modèle de mobilité nécessite de fixer les paramètres liés à chaque modèle comme la vitesse, l'accélération, la trajectoire….

Les paramètres liés aux nœuds

Pour toutes les simulations, les nœuds sont considérés mobiles et homogènes. Ils ont le même rayon de couverture et de transmission. Les RSU sont fixes et loin de la zone d'intérêt.

Pour que l'évaluation des performances de nous propositions par simulation soit efficace, il fallait que nous prennions en considération les spécificités des réseaux véhiculaires pour que la simulation soit réalisable dans des conditions qui se rapprochent de la réalité. Certains paramètres sont sélectionnés en se basant sur des travaux antérieurs pour des applications similaires ainsi que sur la capacité du simulateur. Le tableau ci-dessous résume les valeurs des paramètres utilisés:

Paramètre	Valeur
Surface du réseau	2.4km X 2.4 km
Position de la RSU	Distribution uniforme avec une Couverture non totale du réseau
densité des nœuds	Variable [2..8] véhicule/vois/km
Taille du paquet	2000 bits
Période de broadcast de détection	0.5s
Nombre d'envois de données	3 fois par période
Bande passante	2.7MB/s
Modèle de mobilité	Manhattan

Figure 40. Paramètres de simulation

Les nœuds utilisés dans la simulation sont considérés comme homogènes: possédant les mêmes capacités de calcul, de mémoire et, de portée de transmission. Ils sont aussi équipés par les mêmes interfaces de communication IEEE 802.11p. Les nœuds sont déployés à l'intérieur d'un carré de taille 2.4km par 2.4 km créant ainsi des voies pour le déplacement des nœuds sur la topologie crée. La position des nœuds dans les topologies générées suit le modèle de mobilité choisit.

Les nœuds suivent le modèle de mobilité Manhattan, ils se déplacent de façon aléatoire en respectant les limites de la topologie et la vitesse. L'accélération permise est limitée.

4.2 Les métriques d'évaluation de performances

Pour quantifier les performances des 3 protocoles: LEAVE, Stinger et S-LEAVE deux types de métriques sont sélectionnés: des métriques de sécurités et d'autres de performances. Ces métriques sont affectées par des facteurs comme le nombre de nœuds utilisés dans les scénarios de simulation, le type du protocole...etc.

Les valeurs moyennes des paramètres sont obtenues par l'exécution de 20 simulations indépendantes pour chaque scénario.

Nous calculons trois métriques de sécurité et de performance:

1) Le temps moyen de vulnérabilités des bons dispositifs: La vulnérabilité d'un nœud est définie par sa présence dans la portée de transmission d'un dispositif malveillant non encore bloqué.

2) Pourcentage moyen des voisins honnêtes ignorés,

3) Nombre de messages moyen reçus par nœud.

Les deux premières métriques décrivent des propriétés de sécurité des protocoles, alors que la troisième est utilisée pour calculer le coût en communication du protocole.

La plupart des attaques envisagées sur les réseaux véhiculaires sont *timecritical*. Les attaquants diffusent de fausses informations qui poussent les véhicules vers une décision incorrecte. Par conséquent, les mauvais dispositifs doivent être détectés et retirés le plutôt possible. Nous avons mesuré le temps moyen de vulnérabilités des bons dispositifs par un mauvais nœud. Le temps de vulnérabilité d'un nœud est défini par le temps de sa présence dans la portée de transmission d'un dispositif malveillant non encore bloqué.

La deuxième métrique de sécurité que nous mesurons est le pourcentage moyen des voisins honnêtes ignorés suite à l'exécution du protocole d'exclusion des nœuds malicieux. Chaque voisin ignoré réduit le nombre de dispositifs qui peuvent transmettre une information de sûreté, aussi bien que celui de ceux qui participent aux mécanismes d'exclusion adoptés par Stinger ou LEAVE. Cette métrique est habituellement plus importante pour Stinger, puisque les bons dispositifs renoncent à la participation afin de retirer de mauvais dispositifs. Cependant, il est encore possible que de bons nœuds soient exclus dans LEAVE (en raison du vote accidentel ou malveillant contre de bons dispositifs).

La dernière métrique offre une comparaison utile des coûts des tâches effectuées par chaque stratégie.

5 Résultats

Dans ce qui suit, nous allons présenter et analyser les résultats de simulation obtenus suivant les métriques de performances discutées précédemment.

5.1 Impact de la variation des capacités de détection des nœuds

Dans cette série de tests, nous avons fait varier la distance maximale de détection de 100 à 300m (portée de communication du nœud), Nous avons aussi fait varier le taux de faux positifs (un nœud honnête détecté autant qu'un malicieux) et le taux de faux négatifs (un nœud malicieux détecté autant qu'un honnête.) d'un nœud. Nous avons étudié l'impact de ses variations sur le temps moyen de vulnérabilité et sur le pourcentage de voisins honnêtes ignorés par un nœud.

Après simulation, nous découvrons que le temps moyen de vulnérabilité dans le cas de notre protocole est très intéressant par rapport à celui des deux autres protocoles. Ceci lorsque la capacité de détection d'un nœud est inférieur à son rayon de communication et que l'écart se réduit proportionnellement à l'augmentation des capacités de détection mais il reste toujours moins efficace par rapport à S-LEAVE (figure 41). Cela (l'efficacité de notre protocole) pourrait, en effet, être expliqué par le fait qu'à la différence de LEAVE et Stinger notre protocole permet d'éliminer les nœuds malicieux dès qu'ils sont détectés par un nœud simple. De plus, notre proposition permet de remettre en marche le nœud qui se sacrifie dès que la culpabilité du nœud malicieux est découverte. Ce nœud est rétabli en tant que nœud honnête en lui permettant de continuer à participer au protocole de détection de nœuds malicieux.

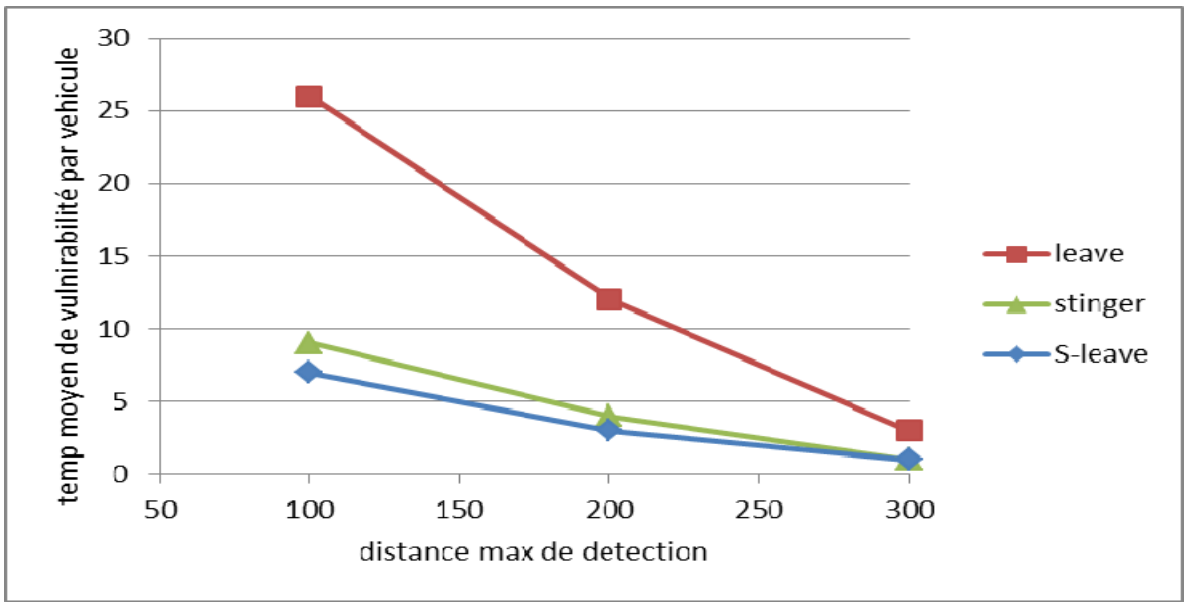


Figure 41. temps moyen de vulnérabilité rapport à la Variation de la distance maximale de détection

En plus de présenter un large avantage sur LEAVE du point de vue temps moyen de vulnérabilité, S-LEAVE affiche une nette supériorité sur Stinger par rapport au pourcentage des voisins honnêtes ignorés (figure 42). Cela peut être expliqué par le fait qu'un nœud se suicidant dans le protocole Stinger ne peut se mettre en activité à nouveau. Par contre, dans S-LEAVE ces nœuds peuvent être remis en service une fois que suffisamment d'accusations sont collectées.

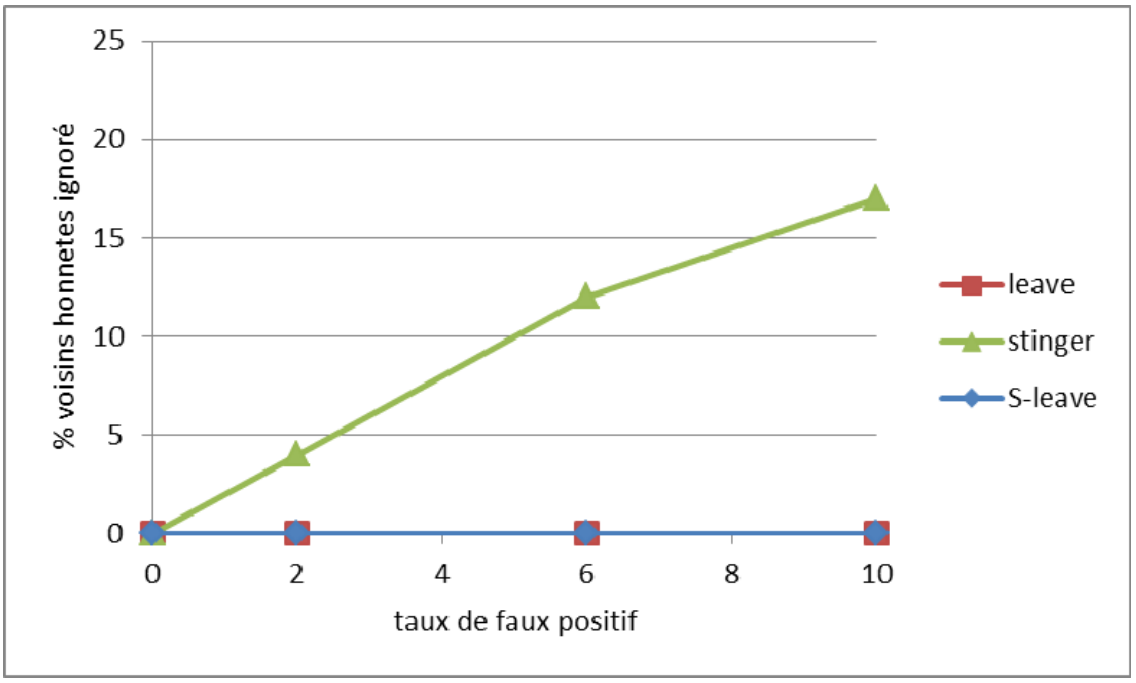


Figure 42. Variation du taux de faux positifs par rapport % de voisins honnêtes ignorés

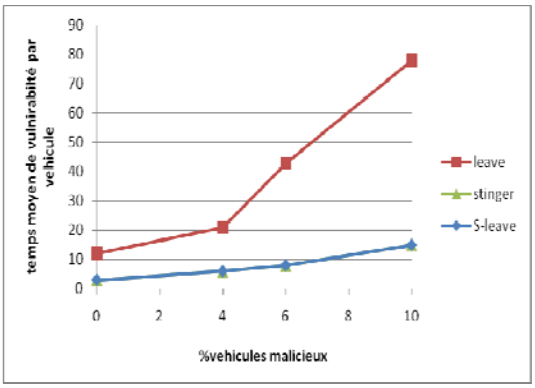
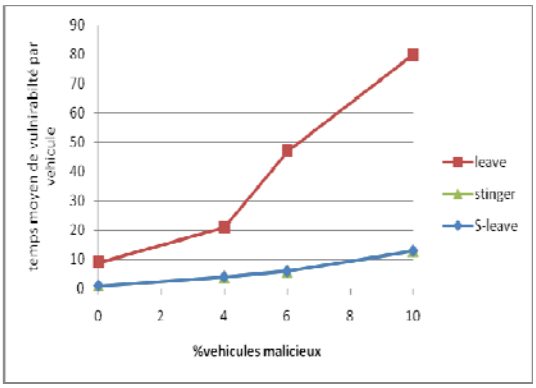
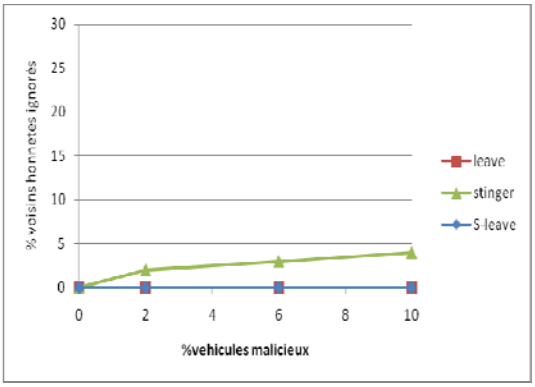
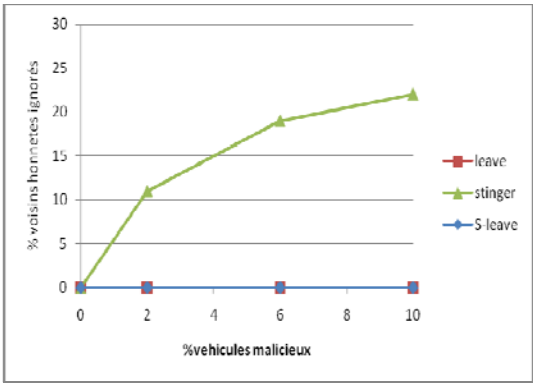
En conclusion de cette série de simulation nous avons pu voir que S-LEAVE offre en même temps l'avantage d'un temps de vulnérabilité minimale et d'un taux de participation maximal par rapport aux deux autres protocoles. Ceci permet une participation plus importante des nœuds aux activités du réseau et en conséquence l'amélioration de la qualité des services offerts.

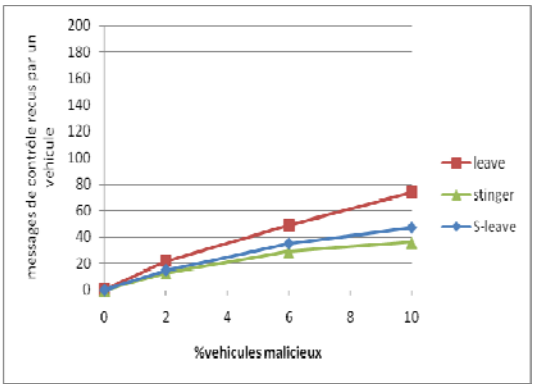
5.2 Impact de la variation des capacités et stratégies adverses

Dans cette série de simulation nous avons étudié l'impact de la variation des capacités des stratégies adverses sur le temps moyen de vulnérabilité, le pourcentage de voisin honnêtes ignorés et le nombre de messages reçus par chaque véhicule (figure 43). Sur cette figure, la colonne de droite affiche les résultats obtenus après diffusion de fausses informations alors que la deuxième montre l'effet de l'abus du mécanisme d'exclusion(ou

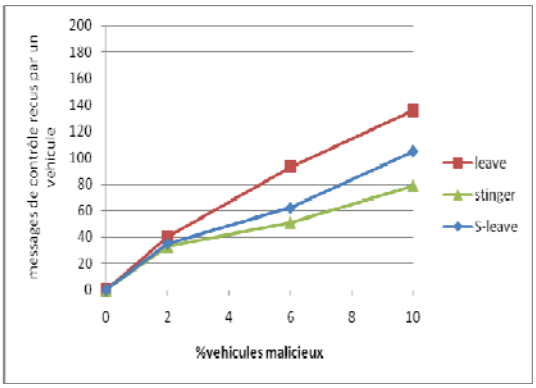
le but est de causer le plus de dommages en exploitant le mécanisme d'exclusion lui-même).

Pour cela nous allons a chaque fois augmenter le pourcentage des nœuds malicieux dans le système de 2 a 10%.

Diffusion de fausses informations	Abus du mécanisme d'exclusion
 (a)	 (b)
 (c)	 (d)



(e)



(f)

Figure 43. Variation des stratégies adverses

En considérant les graphes de la figure 43 deux par deux (a&c, b&d et e&f), nous remarquons qu'à chaque fois que S-LEAVE presente un résultat optimal relativement aux deux autres protocoles, que se soit pour la diffusion de fausses informations (a&c) ou l'abus du mecanisme d'exclusion (b&d), nous remarquons un coût en nombre de messages recus (e&f). Cependant, ce coût reste tout de même acceptable en vu des apports du protocole par rapport aux deux autres.

5.3 Impact de la variation des conditions de trafic

Dans cette serie de simulation nous avons etudié l'impact des differentes conditions de trafic (densité du trafic et vitesse moyenne) sur le temps moyen de vulnérabilité, sur le pourcentage de voisins honnêtes ignorés et le nombre de messages recus par chaque vehicule (figure 44). La colonne de droite de la figure ci-dessous affiche les resultat obtenus après diffusion de fausses informations alors que la deuxième montre l'effet de l'abus du mécanisme d'exclusion.

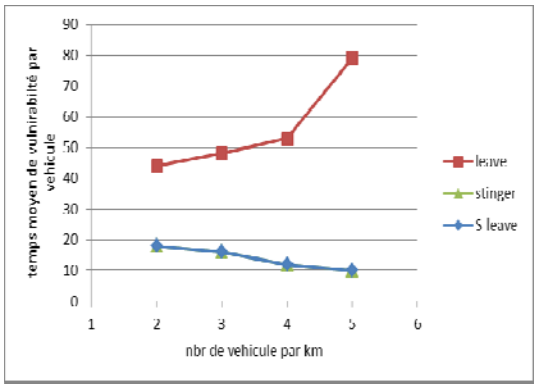
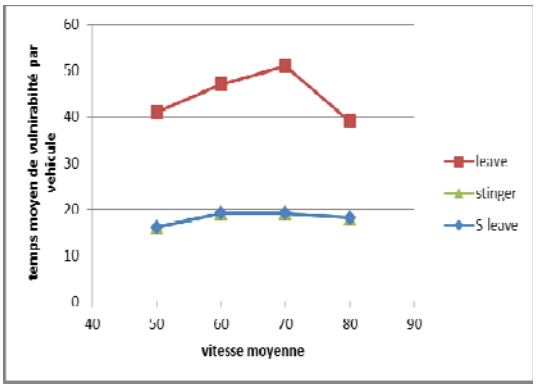
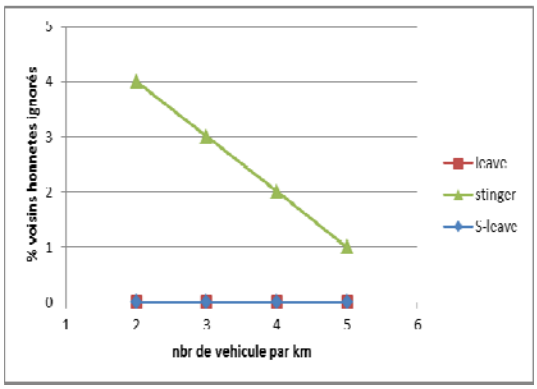
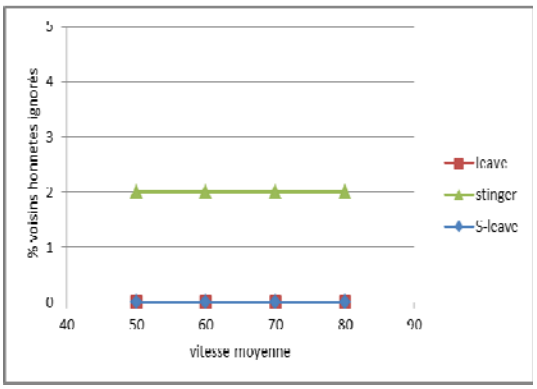
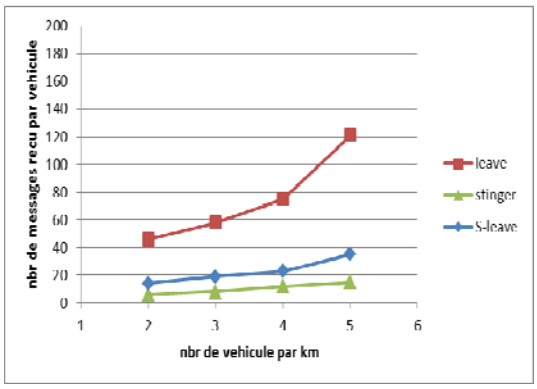
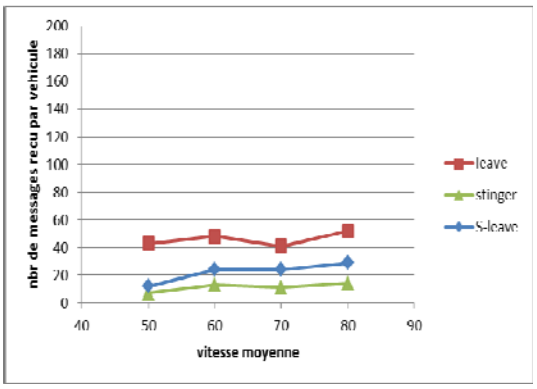
Densité	vitesse
	
(a)	(b)
	
(c)	(d)
	
(e)	(f)

Figure 44. Variation des conditions de trafic

De la même façon que la série précédante, si nous considérons les graphes de la figure 44 deux par deux (a&c, b&d et e&f), nous remarquons qu'à chaque fois que S-LEAVE présente un résultat optimal par rapport aux deux autres protocoles de référence, que se soit pour la densité(a&c) ou la vitesse(b&d), un coût en nombre de messages recus (e&f) qui reste tout de même acceptable en vu des apports du protocole par rapports aux autres.

6 Conclusion

Dans ce chapitre, nous avons comparé deux protocoles LEAVE et Stinger ainsi que notre protocole proposé pour exclusion de dispositifs malicieux ou corrompus. Nous l'avons appliqué dans le cadre des réseaux véhiculaires où l'exclusion rapide est critique et difficile à réaliser étant donnés les propriétés éphémères de l'environnement.

Suite à une analyse basée sur la simulation, nous avons constaté que les deux protocoles ont chacun des avantages et des inconvénients complémentaires. Notamment, Stinger est plus rapide que LEAVE et il est plus adapté à l'augmentation de la densité de trafic. Mais, LEAVE est plus résilient aux positifs faux et à l'accroissement du pourcentage des attaquants.

Ces constations argumente le choix de concevoir un algorithme hybride, permettant à des dispositifs de participer à la détection même au prix d'une élimination temporairement le temps qu'assez d'accusations soient collectées pour établir la culpabilité d'un nœud malicieux.

L'évaluation de ces algorithmes a montré que notre stratégie présente un intérêt certains en efficacité du protocole dans l'environnement fortement dynamique caractéristique des réseaux véhiculaires.

Conclusion générale

L'intérêt pour les réseaux véhiculaires est de plus en plus grand mais les travaux dans ce domaine et en particulier dans le champ de la sécurité restent relativement modestes. Les travaux réalisés dans ce mémoire sont parmi les premiers visant une solution complète à la problématique de traitement des comportements malicieux dans ces réseaux. Cette problématique est en fait primordiale pour la sécurité de ces réseaux et le bon fonctionnement des services offerts par ces derniers. Pour dénouer les substances de cette problématique on est contraint de passer par la détection des informations malicieuses et par la suite à l'élimination des nœuds qui y sont responsables.

Dans cette optique, plusieurs travaux ont été réalisés pour apporter une solution qui tienne compte des caractéristiques spécifiques des VANETs et des buts de sécurité à atteindre. Néanmoins, la majorité de ces travaux restent très loin de l'objectif fixé; soit à causes des hypothèses proposées qui s'éloignent totalement des fondements des réseaux véhiculaires et même ad-hoc, soit par la négligence des premières étapes de mise en œuvre de ces réseaux et même parfois le traitement d'une partie du problème et non son ensemble.

Afin de concevoir une solution complète capable de traiter toutes les phases de traitements des comportements malicieux et prenant en considération toutes les contraintes imposées par cet environnement, nous avons choisie d'aborder cette problématique afin d'apporter de nouvelles solutions qui peuvent combler le vide de sécurité causé et afin de prendre en considération les paramètres de performances et de scalability des réseaux véhiculaires pour que la solution proposée puisse être appliqué sur les VANETs concrets. Cette solution fait usage d'une stratégie de suicide temporaire qui mène à l' exclusion du véhicule malicieux et à la remise en circulation des véhicules honnêtes après la collecte des évidences.

L’évaluation des performances du protocole proposé a été réalisée en utilisant le simulateur de réseaux NS2. Les simulations effectuées ont permis de montrer l’efficacité de notre solution concernant les buts fixés, contribuant ainsi à l’amélioration de la sécurité des réseaux véhiculaires.

Enfin, des expérimentations sur un réseau réel de véhicules nous permettraient de mieux appréhender la robustesse de notre protocole dans un environnement où les phénomènes physiques (effet Doppler, multi-chemin, interférences, etc...) entraînent souvent des pertes de paquets. Il serait également intéressant d’évaluer d’autres modèles qui se rapprochent aux différents cas d’un réseau de véhicules réel.

Bibliographie

[1] F. Li and Y. Wang, Routing in vehicular ad hoc networks : A survey, Vehicular Technology Magazine, IEEE 2 (2007), no. 2, 12–22.

[2] J. K. Hedrick, M. Tomizuka, and P. Varaiya, Control issues in automated highway systems, IEEE Control Systems Magazine 14 (1994), no. 6, 21–32.

[3] O. Gehring and H. Fritz, Practical results of a longitudinal control concept for truckplatooning with vehicle to vehicle communication, Intelligent Transportation System, 1997. ITSC'97., IEEE Conference on (Boston, MA, USA), November 1997, pp. 117–122.

[4] D. Reichardt, M. Miglietta, L. Moretti, P. Morsink, and W. Schulz, CarTALK 2000 : safe and comfortable driving based upon inter-vehicle-communication, Intelligent Vehicle Symposium IEEE, vol. 2, June 2002, pp. 545–550.

[5] DSRC (Dedicated Short Range Communication) Technical Committee, "Dedicated Short Range Communications (DSRC) Message Set Dicticonary," SAE Standard J2735, work in progress, March 2006.

[6] C2C-CC « CAR 2 CAR Communication Consortium Manifesto, Overview of the C2C-CC System», http://www.car-2-car.org/fileadmin/downloads/C2C-CC_manifesto_v1.1.pdf, August 2007.

[7] IEEE P802.11p: Wireless Access in Vehicular Environments (WAVE), draft standard edition.

[8] Hassnaa Moustafa ,Yan Zhang 'Vehicular Networks Techniques, Standards, and Applications', AUERBACH PUBLICATIONS, © 2009 by Taylor & Francis Group, LLC

[9] Panagiotis Papadimitratos, Levente Buttyan ,Tamás Holczer, Elmar Schoch, Julien Freudiger , Maxim Raya, Zhendong Ma ,Frank Kargl, Antonio Kung, Jean-Pierre Hubaux ,'Secure Vehicular Communication Systems: Design and Architecture', IEEE Communications Magazine , November 2008

[10] Florian Doetzer, Timo Kosch, and Markus Strassberger, "Classification for traffic related intervehicle messaging," in Proceedings of the 5th IEEE International Conference on ITS Telecommunications, Brest, France, June 2005. [Online]. Available: http://www13.in.tum.de/personen/doetzer/publications/Doetzer-05ClassificationTrafficRelatedMessaging.pdf

[11] Jakub Jakubiak, Yevgeni Koucheryavy "State of the art and research challenges for VANETs", 1-4244-1457-january 2008 © IEEE

[12] Xianbo Chen, Hazem H. Refai, Xiaomin Ma ,' SDMA: On The Suitability for VANET', ICTTA 2008. 3rd International Conference on Information and Communication Technologies ,IEEE,2008

[13] The CAMP Vehicle Safety Communications Consortium, "Vehicle Safety Communications Project Task 3 Final Report - Identify Intelligent Vehicle Safety Applications Enabled by DSRC," DOT HS 809 859 NHTSA, USDOT, 2005.

[13] S. Sibecas, C. Corral, S. Emami and G. Stratis,"On the Suitability of 802.11a/RA for High-Mobility DSRC," Proc. IEEE Vehicular Technology Conference, Oct. 2002.

[14] Jijun Yin, Tamer ElBatt, Gavin Yeung, Bo Ryu, Stephen Habermas, Hariharan Krishnan, Timothy Talty "Performance evaluation of safety applications over DSRC vehicular ad hoc networks,"VANET '04: Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks .October 2004.

[15] B. Gallagher, H. Akatsuka, and H. Suzuki, "Wireless communications for Vehicle safety: Radio Link Performance & Wireless Connectivity Methods," IEEE VT Magazine, 2006.

[16] K. Tang and M. Gerla, MAC reliable broadcast in ad hoc networks, IEEE MILCOM, 2001.

[17] Qing Xu, Tony Mak, Jeff Ko, and Raja Sengupta, "Vehicle-to-Vehicle Safety Messaging in DSRC", Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks (VANET'04), October 1, 2004, Philadelphia, PA.

[18] Armstrong Consulting, http://www.leearmstrong.com/DSRC/DSRCHomeset.htm

[19] DSRC - Linking the Vehicle and the Road, Bill Jones, ITS Joint. Program Office, U.S. Department of Transportation, 2005

[20] IEEE Standard 1609.1, Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE) - Resource Manager, 2006

[21] IEEE Standard 1609.2, IEEE Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE) - Security Services for Applications and Management Messages

[22] IEEE Standard 1609.3, IEEE Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE)- Networking Services, 2007

[23] IEEE Standard 1609.4, IEEE Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE)- Multi-Channel Operation, 2007

[24] IEEE P802.11p: Wireless Access in Vehicular Environments (WAVE), draft standard edition.

[25] IEEE Standard 802.11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, 1999 Edition

[26] ASTM DSRC STD E2313-02 "Standard Specification for Telecommunications and Information Exchange Between Roadside and VehicleSystems - 5 GHz Band Dedicated Short Range Communications (DSRC) Medium Access Control (MAC) and Physical Layer (PHY) Specifications", 2002

[27] "IEEE P802.11p/D3.0, Draft Amendment for Wireless Access in Vehicular Environments (WAVE)," July 2007.

[28] "IEEE Std. 802.11-2007, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications," IEEE Std. 802.11, 2007.

[29] Eichler, S., "Performance Evaluation of the IEEE 802.11p WAVE Communication Standard", IEEE 66th Vehicular Technology Conference, Sept. 30 – Oct. 3 2007, pp. 2199-2203

[30] Yunpeng Zang; Stibor, L.; Walke, B.; Reumerman, H.-J.; Barroso, A., "Towards Broadband Vehicular Ad-Hoc Networks – The Vehicular Mesh Network (VMESH) MAC Protocol", IEEE Wireless Communications and Networking Conference, 11-15 March 2007, pp. 417-422

[31] B. C. Seet, G. Liu, B. S. Lee, C. H. Foh, K. J. Wong, and K. K. Lee, "A-STAR: A mobile ad hoc routing strategy for metropolis vehicular communications," In Proc. of 3rd International Networking Conference IFIP-TC6 (IFIP '04), Athens, Greece, Dec 2004. Lecture Notes in Computer Science 3042:989–999.

[32] Status of Project IEEE 802.11p, IEEE Task Group TGp, http://grouper.ieee.org/groups/802/11/Reports/tgp_update.htm

[33] M. Raya and J. Hubaux, "The Security of Vehicular Ad Hoc Networks", ACM Workshop on Security of Ad Hoc and Sensor Networks (ACM SASN), 2005

[34] X. Yang, J. Liu, F. Zhao and N. Vaidya, "A vehicle-to-vehicle communication protocol for cooperative collision warning" International Conference on Mobile and Ubiquitous Systems (MobiQuitous), 2004

[35] P. Golle, D. Greene, and J. Staddon, "Detecting and Correcting Malicious Data in VANETs", ACM International Workshop on Vehicular Ad Hoc Networks (VANET), 2004

[36] R. L. Rivest, A. Shamir, and L. M. Adleman, "A method for obtaining digital signatures and publickey cryptosystems", Communications of the ACM, 1978

[37] NTRU Communications and Content Security, http://www.ntru.com

[38] N. Koblitz, "A Course in Number Theory and Cryptography", Springer-Verlag, 1987 Communication Simulation Systems Based on Properties of Urban Areas", International Journal of Computer Science and Network Security (IJCSNS), VOL.6 No.10, 2006

[39] IETF, "The Transport Layer Security (TLS) Protocol Version 1.1", RFC 4346, April 2006

[40] IETF, "Datagram Transport Layer Security", RFC 4347, April 2006

[41] Magda El Zarki, Sharad Mehrotra, Gene Tsudik and Nalini Venkatasubramanian, "Security Issues in a Future Vehicular Network", EuropeanWireless, 2002

[42] Jean-Pierre Hubeaux, Srdjan, Capkun and Jun Luo, "The Security and Privacy of Smart Vehicles", IEEE Computer Society, 2004

[43] Jeremy Blum, Azim Eskandarian, "The Threat of Intelligent Collisions", IEEE Computer Society,2004

[44] Maxim Raya and Jean-Pierre Hubaux, "Securing Vehicular Ad Hoc Networks", Journal of Computer Security (JCS) - special issue on Security on Ad Hoc and Sensor Networks, January 2007

[45] H. Moustafa, G. Bourdon and Y. Gourhant, "Providing authentication and access control in vehicular network environment", IFIP/SEC, May 2006

[46] Network-on-Wheels (NOW) project, www.network-on-wheels.de

[47] M. Gerlach, A. Festag, T. Leinmüller, G. Goldacker and C. Harsch, "Security Architecture for Vehicular Communication", International Workshop on Intelligent Transportation (WIT), 2007

[48] C. Harsch, A. Festag and P. Papadimitratos, "Secure Position-Based Routing for VANETs", IEEE VTC-Fall, 2008

[49] CAR 2 CAR Communication Consortium (C2C-CC), http://www.car-to-car.org/

[50] SEcure VEhicular COMmunication (SEVECOM) project, http://www.sevecom.org/

[51] IEEE P1609.2, "Standard for Wireless Access in Vehicular Environments: Security Services for Applications and Management Messages", 2006

[52] B. Parno and A. Perrig, “Challenges in securing vehicular networks,” in Proceedings of HotNets-IV, 2005.

[53] P. Papadimitratos, V. Gligor, and J.-P. Hubaux, “Securing vehicular communications - assumptions, requirements, and principles,” in Workshop on Embedded Security in Cars (escar’06), 2006.

[54] M. Gerlach, “VaneSe - An approach to VANET security,” in V2VCOM,2005.

[55] M. Jakobsson and S. Wetzel, “Efficient attribute authentication with applications to ad hoc networks,” in Proceedings of VANET’04, 2004.

[56] M. Raya, P. Papadimitratos, and J.-P. Hubaux, “Securing vehicular communications,” IEEE Wireless Communications Magazine, vol. 13, no. 5, pp. 8–15, 2006.

[57] R. Housley, W. Polk, W. Ford, and D. Solo, “Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile,” RFC 3280, 2002.

[58] A. Mishra, K. Nadkarni, and A. Patcha, “Intrusion detection in wireless ad hoc networks,” IEEE Wireless Communications, vol. 11, no. 1, pp. 48–60, Feb. 2004.

[59] T. Leinm¨uller, C. Maih¨ofer, E. Schoch, and F. Kargl, “Improved security in geographic ad hoc routing through autonomous position verification,” in Proceedings of VANET’06, 2006.

[60] S. Kullback and R. Leibler, “On information and sufficiency,” The Annals of Mathematical Statistics, vol. 22, no. 1, pp. 79–86, Mar. 1951.

[61] J. Newsome, E. Shi, D. Song, and A. Perrig. The Sybil Attack in Sensor Networks: Analysis & Defenses. In Proc. of the Third International Symposium on Information Processing in Sensor Networks (IPSN 2004), pp. 259-268, 2004.

[62] R.M. Yadumurthy, A. Chimalakonda, M. Sadashivaiah, and R.Makanaboyina. Reliable MAC Broadcast Protocol in Directional and Omni-directional Transmissions for Vehicular Ad hoc Networks. In Proc. of the 2nd ACM international workshop on Vehicular ad hoc networks (VANET 2005), pp. 10-19 , 2005.

[63] H. Wu, R. Fujimoto, R. Guensler, and M. Hunter. MDDV: A Mobility-Centric Data Dissemination Algorithm for Vehicular Networks. In Proc. of the 1st ACM international workshop on Vehicular ad hoc networks (VANET 2004), pp. 47-56, 2004.

[64] G. Korkmaz and E. Ekici. Urban Multi-hop Broadcast Protocol for Inter-Vehicle Communication Systems. In Proc. of the 1st ACM international workshop on Vehicular ad hoc networks (VANET 2004), pp. 76-85, 2004.

[65] J. Y. Choi, M. Jakobsson, and S. Wetzel. Balancingauditability and privacy in vehicular networks Proceedings of the 1st ACM international workshop on Quality of service and security in wireless and mobile networks, pages 79–87, 2005.

[66] M. Raya, D. Jungels, P. Papadimitratos, I. Aad, and J.-P. Hubaux. Certificate revocation in vehicular
networks. Technical report, 2006 2006.

[67] J. Hightower and G. Borriello. A survey and taxonomy of location systems for ubiquitous computing. university of washington, 2001.

[68] M. Raya, P. Papadimitratos, I. Aad, D. Jungels, J.-P. Hubaux, “Eviction of misbehaving and faulty nodes in vehicular networks,” IEEE Journal on Selected Areas in Communication, vol. 25, no. 8, pp. 1557–68, 2007.

[69] T. Moore, J. Clulow, S. Nagaraja, and R. Anderson, “New strategies for revocation in ad-hoc networks,” in 4th European Workshop on Security and Privacy in Ad-hoc and Sensor Networks (ESAS), Springer Lecture Notes in Computer Science (LNCS), vol. 4572, pp. 232–246, 2007.

[70] C. Cr´epeau and C. Davis, “A certificate revocation scheme for wireless ad hoc networks,” in 1st ACM Workshop on Security of Ad-hoc and Sensor Networks (SASN), 2003, pp. 54–61.

[71] T. Moore, M. Raya, J. Clulow, P. Papadimitratos, R. Anderson, and J.-P. Hubaux. Fast exclusion of errant devices from vehicular networks. In Proceedings of SECON’08.

[72] Jonathan Van Eenwyk ,University of Kansas,” Detecting Malicious Data in Vehicular Networks”, VANET ’07 Montreal, Canada Copyright 2007 ACM

[73] Amrouni.R,Djoudi.F:Dissémination d'information dans un réseau Ad-Hoc véhiculaire.Mémoire de fin d'études, 2009.

[74] http://www-sop.inria.fr/rodeo/personnel/Pierre.Ansel/Manuel_NS1.3.pdf.

[75] http://nile.wpi.edu/NS/

[76] http://www.phdgroup.org/LebaneseUniversityArchive/CSTI/2005-2006/5.pdf

[77] http://lens1.csie.ncku.edu.tw/MOVE/index.htm

[78] http://www.csie.ncku.edu.tw/~klan/move/manual.pdf