Université Abou Bekr Belkaid
Tlemcen Algérie
جامعة أبي بكر بلقايد
تلمسان الجزائر
République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Laboratoire STIC

PFE

Présenté

A L'UNIVERSITE DE TLEMCEN

FACULTE DE TECHNOLOGIE

DEPARTEMENT DE TELECOMMUNICATIONS

Pour l'obtention du diplôme de

Master 2

Spécialité : " **R**éseaux **M**obiles et **S**ervices de **T**élécommunication"

Présenté Par :

MEZIANI Badr et OUAHIANI Mohammed

Sujet :

Sécuriser la communication de groupe dans les réseaux VANETs « protocole Diffie-Hellman »

Soutenu le **14 Juin 2016**, devant le jury composé de :

HADJILA Mourad	**M.C.B**	**Président**
BOUABDELAH Reda	**M.C.A**	**Examinateur**
MOUSSAOUI Djillali	**M.C.A**	**Encadreur**

REMERCIEMENTS

Nous remercions d'abord la grâce de **l'Allah,** pour nous avoir guidés et éclairer sur la bonne voie du savoir pour continuer ce travail et atteindre les objectifs traces.

Nous exprimons nos remerciements particulièrement et les plus sincères à notre encadreur ***Mousaoui Djilali*** Maitre de conférences A de nous avoir encadrés pour réaliser ce travail par ses précieux conseils et de nous avoir donné le meilleur de son savoir et aide.

Nous remercions vivement ***Hadjila Mourad*** Maitre de conférences (B), et ***Bouabdelah Reda*** Maitre de conférences (A) a l'université de Tlemcen, d'avoir acceptons de faire partie de nos jury de thème.

Résumé

L'évolution progressive des technologies sans fil a donné naissance à une nouvelle génération des réseaux utilisée dans les communications véhicule à véhicule ou véhicule à infrastructure afin d'améliorer la sécurité routière via l'échange des messages d'alerte entre les véhicules ou encore afin d'offrir de nouveaux services de confort aux usagers des routes. Ces types de réseaux sont très dynamiques avec des architectures fortement décentralisées et dont les services sont organisés de manière autonome.

Le problème dans ces réseaux consiste à déterminer le protocole de routage le plus adapté à cet environnement, et ensuite à le sécuriser afin de fournir un acheminement optimal et sécurisé pour les données.

Dans cette thèse, nous avons proposé quelques solutions de sécurité pour les réseaux ad hoc véhiculaire à savoir :

La sécurisation du protocole de routage DSR "Dynamic Source Routing" avec Diffie-Hellman.

Mots clés : Réseaux ad hoc véhiculaires (VANETs), protocole de routage, Diffie-Hellman, sécurisation de DSR avec Diffie Hellman.

Abstract

The technological advancement in Wireless network has give the birth to a new generation of networks used in vehicle-to-vehicle or vehicle to road side unit communications to improve road safety by exchanging warning messages between vehicles or to offer new comfort services to road users.

These types of networks are highly dynamic with highly decentralized architectures and whose services are organized independently.

The problem in these networks is to determine the best routing protocol suited to this environment characterized by rapid topology changes, and then secure it to provide a safe and optimal route to data.

In this work, we propose some security mechanisms for vehicular ad hoc networks, first, we talking about Vanet, Secondly two cryptographic security mechanisms have been proposed to secure the DSR"Dynamic Source Routing".

Keywords: Vehicular Ad hoc networks (VANET), routing protocol, Diffie-Hellman, secure DSR with Diffie-Hellman.

الملخص:

التطور التدريجي لتكنولوجيا اللاسلكية قد أدى إلى ظهور جيل جديد من الشبكات المستخدمة في الاتصالات الجديدة من بينها شبكات المركبات .وتهدف هذه الشبكات إلى دمج تكنولوجيا المعلومات والاتصالات في صناعة السيارات لتحسين السلامة والراحة عبر شبكات الطرق.

التوجيه هو عنصر مهم في شبكات المركبات VANETs لتحديد كيفية نقل مختلف الرسائل والبيانات .

الهدف من موضوعنا هو دراسة مختلف بروتوكولات التوجيه وأدائها وفي هذه الفرضية اقترحنا بعض الحلول في VANTEsألا وهي تأمين بروتوكولDSR.

كلمات البحث :شبكات VANETs، بروتوكولات التوجيه، بروتوكولات التأمين، Diffie–Hellman ، تأمين بروتوكولDSR .

Table des matières

Chapitre II : La sécurité dans les réseaux VANETs

Chapitre III : Le protocole DH-DSR

Table de matière

Liste des figures

Liste des tableaux

Introduction générale

Un réseau ad hoc véhiculaire est un ensemble de nœuds mobiles (véhicules) autonomes et coopératifs qui se déplacent et communiquent par une transmission sans fil et ne suppose pas d'infrastructure de gestion préexistante. Le réseau ad hoc se forme de manière spontanée et provisoire dès que plusieurs nœuds se trouvent à portée radio les uns des autres ou un véhicule peut communiquer directement avec un autre ou en comptant sur la coopération des voisins pour router les paquets vers la destination.

Les protocoles de routage ad hoc ont été conçus sans aucun contrôle de sécurité et font l'hypothèse d'un comportement honnête entre les entités qui collaborent. La réalité peut toutefois être très différente en présence d'entités malveillantes capables de détourner le bon déroulement des opérations de routage pour servir leur intérêt. Parmi ces attaques, motivées par l'égoïsme ou par la malveillance figurent : la modification des paquets, l'injection des données et la génération de faux messages, la rupture de l'acheminement ou la suppression de paquets.

Le problème dans ces réseaux consiste à déterminer le protocole de routage le plus adapté à cet environnement caractérisé par des changements rapides de la topologie, ensuite à le sécuriser afin de fournir un acheminement optimal et sûr pour les données.

Pour remédier à ces vulnérabilités, plusieurs protocoles de routages sécurisés pour les réseaux mobiles ont été proposés dans lesquels des primitives cryptographiques interviennent, comme les signatures numériques, le MACs (Message Authentication Code) ou le chiffrement asymétrique .On peut citer par exemple : SRP (Secure Routing Protocol) , SAODV (Secure Ad-Hoc On demand Distance Vector), ARIADNE et SOLSR (Secure Optimized Link State Routing). Ces protocoles sont hautement spécialisés pour une attaque précise et ils n'offrent pas la possibilité de détecter de nouvelles attaques, ni même de défendre le réseau contre des nœuds internes compromis. Cependant, ces protocoles sont toujours vulnérables à certaines attaques, telles que Rushing Attack et Route cache poisoning.

D'autre part, les solutions proposées nécessitent un temps de synchronisations supplémentaire des horloges des nœuds, utilisé pour la détermination des intervalles de temps de validité des clés cryptographique et certificats. Ce qui rend ces solutions inadaptées pour les VANETs.

Dans le cadre de cette étude, nous nous sommes intéressés aux problématiques de sécurité des communications véhiculaires. L'objectif principal consiste à proposer des mécanismes de sécurité adaptés aux caractéristiques des réseaux VANETs et à leurs applications.

Dans un premier temps, nous avons effectué un état de l'art sur les principaux concepts, caractéristiques et challenges liés aux réseaux VANETs. Le système de communication entre véhicules a été détaillé en présentant les différentes architectures de réseaux VANETs : V2V (Véhicule à Véhicule), V2I (Véhicule à Infrastructure) et hybrides (mélange des deux précédentes). Nous avons pu apporter une vue sur un ensemble d'algorithmes de routage.

Deuxièmement nous présentons un récapitulatif sur les mécanismes de base de la sécurité en générale, nous passons en revue la sécurité dans les réseaux sans-fil, ensuite nous présentons les problèmes et les mécanismes de base de sécurité dans les VANETs, enfin nous étudions les techniques et solutions de sécurité existantes qui peuvent être mises en œuvre afin de sécuriser les informations échangées à travers ces réseaux.

La troisième étape de notre étude, consiste à sécuriser le protocole DSR. En effet, le routage a un rôle primordial vu sa fonctionnalité dans l'acheminement des données entre les nœuds du réseau. Il est donc nécessaire de bloquer toutes les tentatives qui visent à modifier ces fonctionnalités par un nœud malveillant.

Cependant les contraintes liées à l'absence d'infrastructure de gestion centralisée, à l'utilisation de canaux de communication sans fil, ainsi qu'à l'absence de coopération entre les nœuds, rendent cette tâche difficile pour les réseaux ad hoc.

Dans cette partie de notre travail sur la sécurisation du protocole de routage DSR, nous avons proposé deux extensions à ce protocole :

- Etablir des clés secrètes de Diffie-Hellman entre deux véhicules au moment de découverte de route. L'idée consiste à ajouter les paramètres de Diffie-Hellman aux messages RREQ et RREP qui seront utilisées pour calculé la clé secrète au sein de chaque nœuds.

- Ajouter au paquet DSR, une signature numérique basée sur la cryptographie symétrique générée à l'aide de l'algorithme AES et la fonction de hachage MD5 ce qui minimise le temps de calcul de la signature numérique.

I.1. Introduction :

Les réseaux sans fil ont connu ces dernières années un essor spectaculaire et s'imposent aujourd'hui de façon indéniable. Parmi les technologies récentes de communication sans fil sont les réseaux véhiculaires (VANET), très inspiré des MANET (réseau mobile ad hoc).

VANET permet aux véhicules de communiquer via des messages d'alertes de sécurité envoyés entre eux. Leur élaboration s'appuie sur l'émergence des systèmes de transports intelligents (Intelligent transportation Systems- ITS), qui ont comme objectif principale d'améliorer la sécurité routière.

En effet, grâce à des capteurs installés au sein des véhicules qui incluent des protocoles de routages sécurisé, les communications véhiculaires permettent aux conducteurs d'être avertis suffisamment tôt de dangers éventuels.

Dans ce chapitre nous commençons dans un premier temps la mise en réseau des MANET, leurs caractéristiques, et nous abordons les réseaux Ad Hoc véhiculaires (VANET) en décrivant les entités communicantes, les modes de communication et les caractéristiques de ces réseaux et leurs objectives. Puis en citant quelques applications pour VANET. Ainsi, nous parlons sur les différents défis qui ont un impact sur le futur déploiement des réseaux véhiculaires et au final, on classifier les protocoles de routages dans les réseaux VANET.

I.2.Les réseaux sans fils « Wireless network » : [1]

C'est un réseau qui ne nécessite pas une connexion filaire entre les machines, la communication se fait par voie hertziennes. Les réseaux sans fils peuvent être organisés en deux classes :

- **Réseaux avec infrastructure:**

Qui est constitué en minimum d'un seul point d'accès. Dans cette classe l'étendu du réseau est définit par le point d'accès (comme les réseaux cellulaires, GSM, et le WIFI).

- **Réseaux sans infrastructure (ad hoc):**

Qui représente un ensemble de station qui communique directement entre elles et forme un réseau point à point (P2P pour Peer to Peer). Dans ce cas toutes les stations constituent en elles-mêmes un point d'accès.

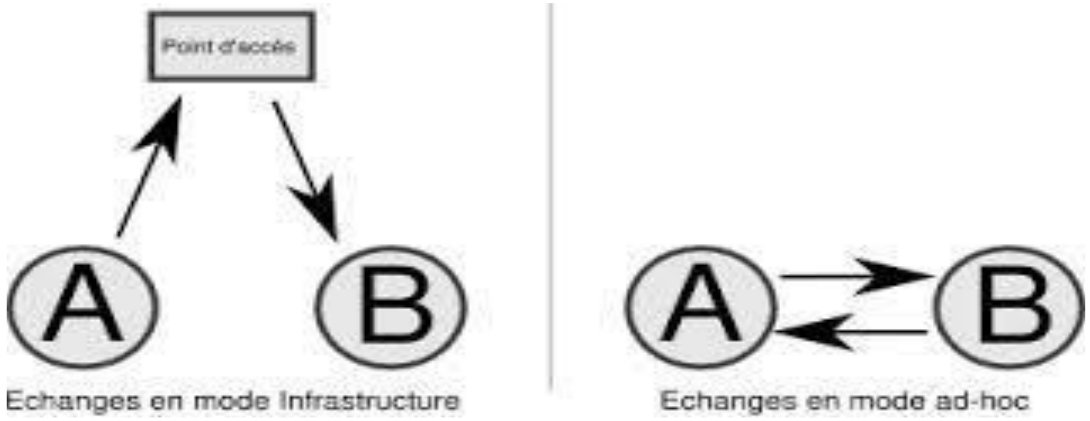


Figure I.1 : mode infrastructure et mode ad-hoc

I.2.1.Réseaux sans fils Ad hoc (WANET) :

D'après le point précédent, les réseaux ad hoc sont des réseaux P2P sans fils capables de s'organiser sans infrastructure spontanément et d'une manier autonome. Le mode Ad hoc permet de crée rapidement un réseau sans fil avec des entités communicante appelé des nœuds là où il n'existe pas d'infrastructure filaire **[2]**.

I.2.2.Réseaux mobile ad hoc (MANET) :

C'est un système autonome composé des nœuds mobiles dynamiques interconnectés par des liens sans fil sans l'utilisation de l'infrastructure fixe.

Les nœuds sont libres de se déplacer de façon aléatoire, par conséquence, la structure du réseau change fréquemment et d'une manière imprévisible.

Dans ce qui suit on va s'intéresser particulièrement aux réseaux VANET, ses caractéristiques, objectifs, avantage et inconvénients. Ainsi qu'à leurs protocoles de routage.

I.3. Les réseaux Ad Hoc Véhiculaire (VANET) :

I.3.1.Pourquoi VANET ?

De notre jour, de plus en plus de foyers possèdent au moins un véhicule. Cette situation a conduit à une grande augmentation du trafic routier causant de multiples problèmes de confort et de sécurité.

Le véhicule, cet engin que nous a offert la technologie moderne, et qui nous a tant facilité la tâche des déplacements à longue distance est entrain de montrer ses méfaits. En effet, la

circulation en voiture est devenue dans certaines villes telles que Moscou, Pékin ou Mexico, une épreuve quotidienne à cause des embouteillages. Un problème encore plus important est celui de la sécurité, les statistiques de l'OMS (*Organisation Mondiale de la Santé*) montres qu'il y'a en moyenne 1,2 millions de décès et entre 20 et 50 millions de blessures grave causés par les accidents de la route.

Pour pallier aux problèmes de sécurité et de circulation routière, de nombreuses initiatives ont été prises par les gouvernements, les associations et les constructeurs automobiles .Parmi ces initiatives les compagnes de sensibilisations, l'instauration d'un code de la route stricte et l'amélioration et l'incitation au transport en commun. Une autre solution, d'œuvre technologique est apparue au début des années 1990, nommée ITS (*Intelligent Transportation Systems*), elle consiste à intégrer les nouvelles technologies de l'information et de la communication afin de rendre plus efficace le system routier, la sécurité et le confort des usagers.

Après l'investissement de plusieurs pays ces vingt dernières années, une nouvelle technique de communication, dédiée spécialement a l'ITS à émerger en 2002.Cette technologie a été nommée DSRC (*Dedicated Short Range Communication*) et qui support les communications a courte et à moyenne portée, Tel que : V2I (*Vehicle To Infrastructure)* et V2V ou VANET (*Vehicle to Vehicle, Vehicle Ad hoc Network*)**.**

I.3.2. Définition d'un réseau VANET : [1]

Un réseau VANET est un réseau de communication entre véhicules intelligents équipés de calculateurs, de périphériques réseau et de différents types de capteurs.

Les VANET fonts parti de la famille des réseaux mobiles MANET qui fonctionnent dans des réseaux à liaison point à point sans infrastructure, c'est-à-dire que tout nœud constituant le réseau est un point d'accès. Dans un réseau VANET les nœuds sont les véhicules intelligents appartenant au réseau.

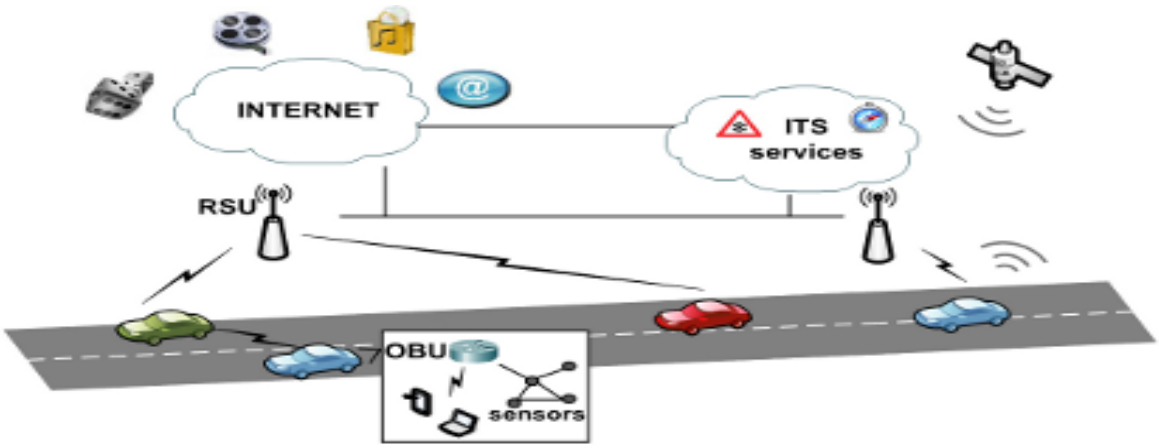


Figure I.2 : Exemple de communications dans les VANET.[1]

I.3.3. Les objectifs de VANET :[1]

- **Sécurité des usagers :**

-Prévention sur les accidents : Les accidents deviennent plus rapidement détectables et l'intervention devient plus rapide, cela peut minimiser le risque de décès après un accident.
-Anticipation du trafic : Les véhicules sont informés par les routes ou il y'a des embouteillages, ils peuvent donc emprunter un autre chemin, cela peut permettre de rendre les routes plus fluides.
-Préventions d'un véhicules prioritaire : Permet d'avisé les conducteurs d'un passage de véhicules prioritaire (Exemple : ambulances, véhicules de police…).

-Anticipation d'un danger quelconque: Les véhicules peuvent s'échanger entre eux des préventions de dangers liés aux routes pour mieux les anticipés.

- **Confort des usagers :**

-Avoir une prévention sur l'itinéraire peut permettre de facilité la conduite et amoindrir les risques d'accidents.
-Les réseaux VANET peuvent communiquer avec les infrastructures externes comme internet, donc la capacité d'accéder à des loisirs comme les téléchargements de flux multimédias, lecture des emails …etc.
-Possibilité de jouer en réseau entre les passagers des voitures, téléchargement et partage de fichier tel que les cartes.
-La régulation des flux de véhicules, permet de réduire le nombre d'embouteillages.

-Le guidage par GPS permettant un déplacement plus facile, et l'auto-localisation qui permet de trouver les véhicules volés.

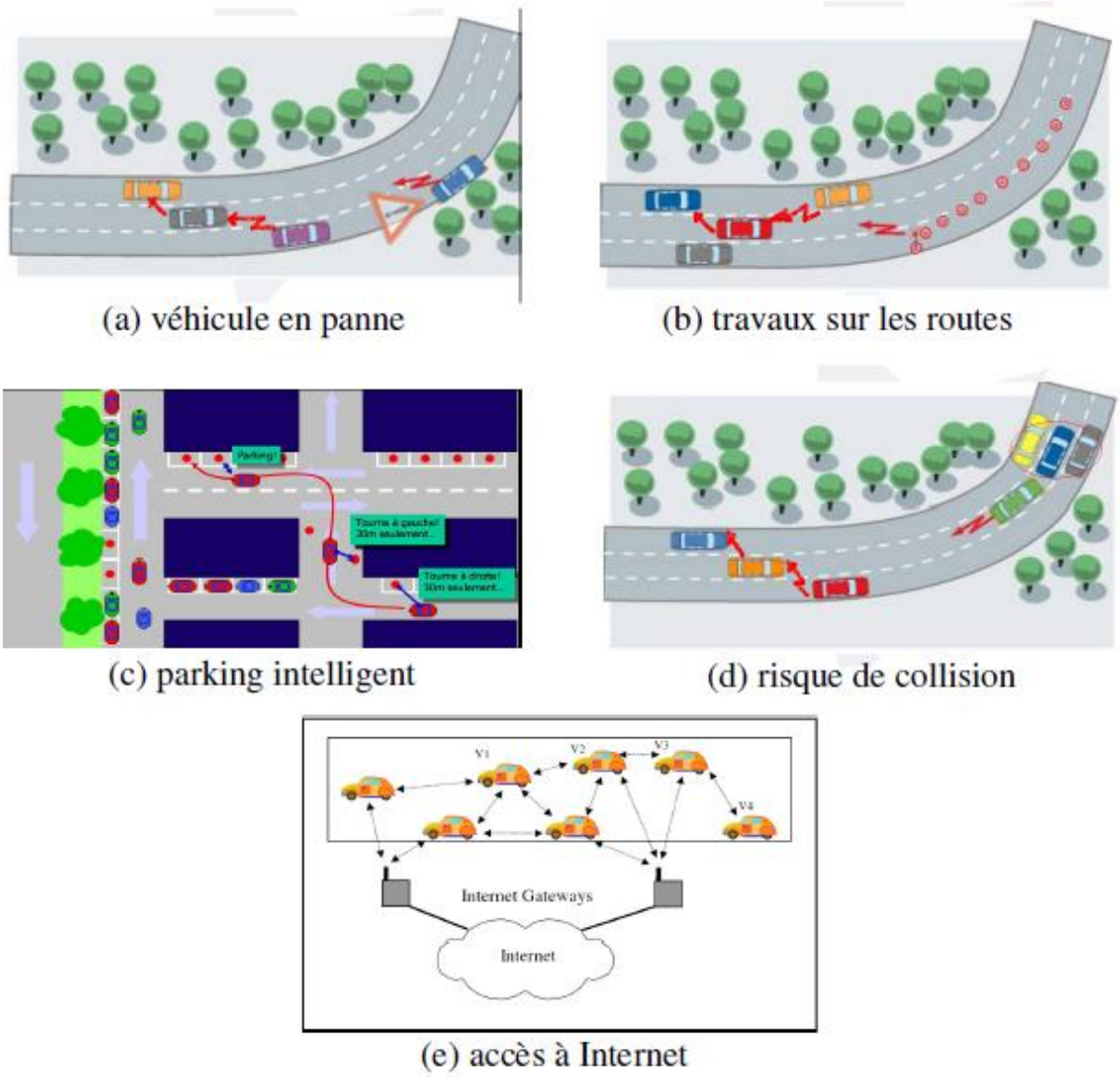


Figure I.3 : Exemples d'utilisation de VANET. [1]

I.3.4. Architectures de communication dans les VANETs : [2]

Dans les réseaux de véhicules, on peut distinguer deux modes de communication, les communications Véhicule-à-Véhicule (V2V) et les communications Véhicule-à-Infrastructure (V2I) (figure I.4). Les véhicules peuvent utiliser un de ces deux modes ou bien les combiner s'ils ne peuvent pas communiquer directement avec les infrastructures de la route.

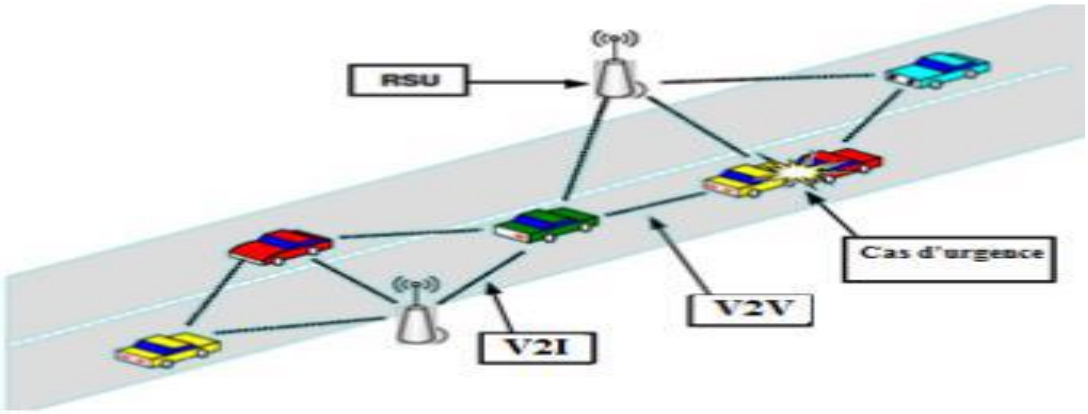


Figure I.4 : Types de communication dans un réseau de véhicules [2]

Dans la section suivante, nous présentons le principe et l'utilité de chaque mode :

I.3.4.1.Communications Véhicule à Véhicule (V2V) : [3]

Avec le développement rapide des technologies de communications sans fil, des systèmes de localisation et de collecte d'information via les capteurs, une nouvelle architecturedécentralisée (ou semi-centralisée) basée sur des communications véhicule à véhicule (V2V, Vehicle to Vehicle) suscite ces dernières années un réel intérêt auprès des constructeurs automobiles, de la communauté R&D et des opérateurs Télécoms.

Ce type d'architecture s'appuie sur un système distribué et autonome et est formé par les véhicules eux même sans l'appui d'une infrastructure fixe pour le relayage des données et des messages On parle dans ce cas d'un réseau ad hoc devéhicules (VANET, Vehicle Ad hoc NETwork), qui n'est autre qu'une applicationdédiée et spécifique des réseaux ad hoc mobiles conventionnels (MANET, Mobile Ad hoc NETwork).

Cette architecture peut être utilisée dans le scénario de diffusion d'alertes (freinage d'urgence, collision, ralentissement…) ou pour la conduite coopérative.

Donc aucune infrastructure n'est utilisée, aucune installation n'est nécessaire sur les routes et tous les véhicules sont équipés pour communiquer directement entre eux n'importe où, que se soit sur les autoroutes, des routes de montagnes ou des routes urbaines, ce qui donne une communication moins coûteuse et plus flexible. Cette approche souffre de certains inconvénients dont nous citons :

- Les délais de communication qui sont élevés, étant donné que la communication se fait en utilisant le multi sauts.

- Les déconnexions fréquentes dues au fait que les véhicules sont mobiles.

- La sécurité réseau est très limitée

Un exemple de réseau VANET V2Vurbain est illustré dans la figure suivante :

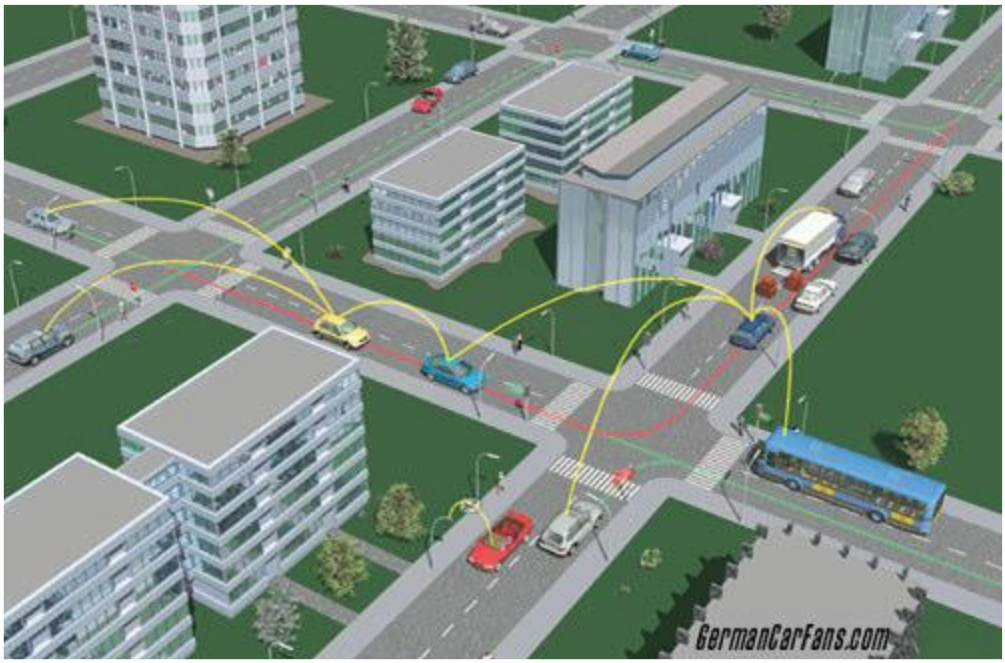


Figure I.5 : Exemple d'un réseau VANETs (mode V2V). [2]

I.3.4.2.Communication de véhicule à infrastructure (V2I) :

Dans cette catégorie, on ne se concentre pas seulement sur de simples systèmes de communications inter véhicules mais aussi sur ceux qui utilisent des stations de bases ou points d'infrastructure RSU (Road Side Units, dénomination proposée par le consortium C2C -CC).

Cette approche (V2I) repose sur le modèle client/serveur où les véhicules sont les clients et les stations installées le long de la route sont les serveurs. Ces serveurs sont connectés entre eux via une interface filaire ou sans fil. Toute communication doit passer par eux. Ils peuvent aussi offrir aux utilisateurs plusieurs services : localisation des stations d'essence et emplacements de parking libre, le chat inter –véhicule, informations climatiques, informations culturelles, échange de données de voiture-à-domicile et même la communication de voiture-à-garage pour le diagnostic distant.

L'inconvénient majeur de cette approche est que l'installation des stations le long des routes est une tâche coûteuse et prend beaucoup de temps, sans oublier les coûts relatifs à la maintenance des stations.

I.3.4.3.Communication hybride :

La combinaison des communications véhicule à véhicules avec les communications de véhicules avec utilisation d'infrastructures, permet d'obtenir une communication hybride très intéressante. En effet, les portées des infrastructures (stations de bases) étant limitées, l'utilisation des véhicules comme relais permet d'étendre cette distance. Dans un but économique et afin d'éviter la multiplication des stations de bases à chaque coin de rue, l'utilisation des sauts par véhicules intermédiaires prend tout son importance.

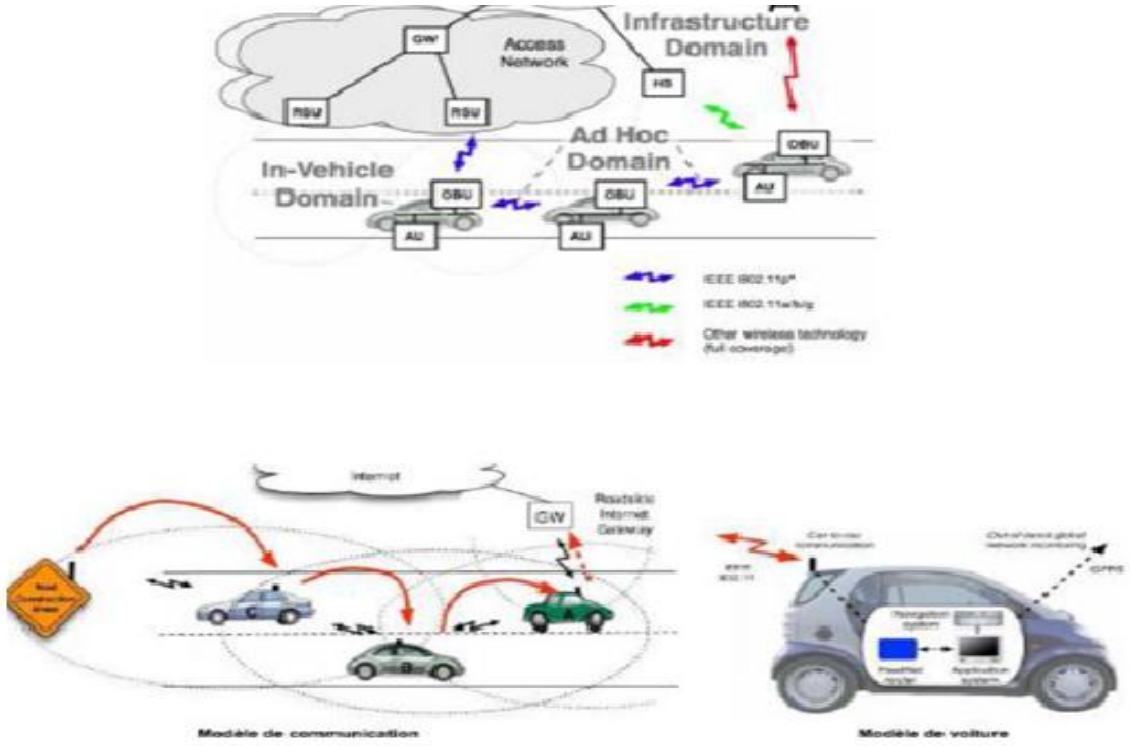


Figure I.6:architecture VANET Hybride C2C**[2]**.

I.3.5. Caractéristiques des réseaux VANET [2][3]:

Les réseaux véhiculaires se distinguent des réseaux sans fil traditionnels par un certain nombre de caractéristiques spécifiques dont on peut citer :

I.3.5.1. Le potentiel énergétique :

À la différence des réseaux sans fil traditionnels où la contrainte d'énergie représente un facteur limitant important, les entités des réseaux véhiculaires disposent de grandes capacités énergétiques qu'elles tirent du système d'alimentation des véhicules**[1]**.

I.3.5.2. Environnement de déplacement et modèle de mobilité :

Les environnements pris en compte par les réseaux Ad Hoc sont souvent limités à des espaces ouverts ou indoor (comme le cas d'une conférence ou à l'intérieur d'un bâtiment). Les déplacements des véhicules quant à eux sont liés aux structures des routes (intersections, panneaux de signalisation, etc…) et aux stations de base routières (infrastructures) que se soit dans les autoroutes ou au sein d'une zone métropolitaine. Les contraintes imposées par ce type d'environnement, à savoir les obstacles radio et les effets de la propagation à trajets multiples (multipath) ou d'évanouissement (fading), affectent considérablement le modèle de mobilité et la qualité des transmissions radio à prendre en compte dans les protocoles de routage. En outre la mobilité est un facteur lié directement au conducteur du véhicule.

I.3.5.3. Le modèle de communication :

Les réseaux véhiculaires ont été imaginés principalement pour les applications liées à la sécurité routière (*ex.* diffusion de messages d'alerte). Dans ce type d'application, les communications se font presque exclusivement par reliages successifs d'une source vers une multiplicité de destinataires. Le modèle de transmission en Broadcast ou en Multicast est donc appelé à dominer largement dans les réseaux véhiculaires, ce qui n'est par exemple pas sans conséquence sur la charge du réseau et le modèle de sécurité à mettre en œuvre.

I.3.5.4.La taille du réseau:

Etant donné les avancées importantes réalisées dans le domaine des communications sans fil et les bas coûts des équipements associés, les véhicules qui intègrent déjà massivement des systèmes GPS et des équipements Bluetooth, seront très probablement équipés et ce, tout aussi massivement, de plateformes de communication leur permettant de constituer de véritables réseaux. Ce faisant, et compte tenu de l'importance sans cesse grandissante de la densité et du parc des véhicules, on peut s'attendre à ce que la taille des réseaux véhiculaires dont les déploiements restent encore très confidentiels, soit d'une tout autre ampleur. L'importance potentielle de la taille des réseaux véhiculaires constitue donc une caractéristique majeure à prendre en compte dans la conception de ces réseaux.

Il y'a d'autre caractéristique comme :

- Communication à courte portée Jusqu'à 1000m.
    - Propagation par trajets multiples.
    - Occupation du canal différente sur autoroute, déserte et centre-ville bondé.

- Les véhicules se déplacent rapidement
  - Changements de topologie.
  - Durée de connexion entre deux nœuds courte.

I.3.6.Avantages et contraintes de VANET :

I.3.6.1. Avantage des réseaux VANET :

❖ **Topologie dynamique :**

Les nœuds des réseaux VANET (véhicules) se déplaçant très rapidement, la topologie des réseaux et à chaque fois modifiée, mais les caractéristiques de VANET permettent le maintien des communications et l'échange de flux d'informations en dépit des changements fréquents des positions des nœuds.

❖ **Echange entre nœuds hétérogènes :**

Les véhicules des réseaux VANET sont de différentes marques et les composants réseaux qui les constituants utilisent différents technique, mais ils peuvent tout de même aboutir à un bon échange d'informations grâce aux protocoles instaurés par les concepteurs du réseau.

❖ **Propagation par trajet multiple :**

Les infos partagées par un véhicule peuvent être reçus par tous les autres véhicules se trouvant dans son entourage.

❖ **Relais d'informations :**

Deux véhicules distants de plusieurs KM peuvent se partager une information, cette information envoyée depuis un nœud A est relié par plusieurs nœuds intermédiaires avant d'arriver au destinataire B.

I.3.6.2. Inconvénients des réseaux VANET :

❖ **Canal radio partagé et limité :**

Un canal radio à fréquences précises est utilisé par tous les nœuds, le flux d'information est donc limité et le débit de transmission diminue surtout dans les centres villes.

Faible bande passante :

Le partage du canal limite la bande passante dont dispose chaque nœud pour partager les informations.

- **Les interférences**

Les réseaux VANET utilisent les transmissions radio pour transmettre l'information, ce qui rend les communications exposées aux interférences radio, ces derniers sont de nature diverse comme : le rapprochement des fréquences d'émission (interférences entre deux nœuds), les bruits de l'environnement (équipements électriques, moteurs), et les phénomènes de réflexion, atténuation et dispersion qui déforment le signal.
Ces interférences font augmenter le taux d'erreurs de transmission, et le rendent incompréhensible par le récepteur.

I.3.7. Les applications des VANETs : [5]

Les communications véhiculaires peuvent être utilisées par de nombreuses applications qui peuvent être classées en applications de sécurité routière, applications pour les systèmes d'aide à la conduite et applications de divertissement qui nécessite une connexion internet. Ci-dessous, nous identifions un ensemble représentatif des applications des VANETs.

I.3.7.1. Système d'alerte de collision :

Les systèmes d'avertissement de collision des véhicules sont l'une des applications les plus intéressantes des VANETs pour l'assistance du conducteur. Ces systèmes avertissent le conducteur pour qu'il change de direction afin d'éviter la congestion routière comme illustré sur la figure suivante :

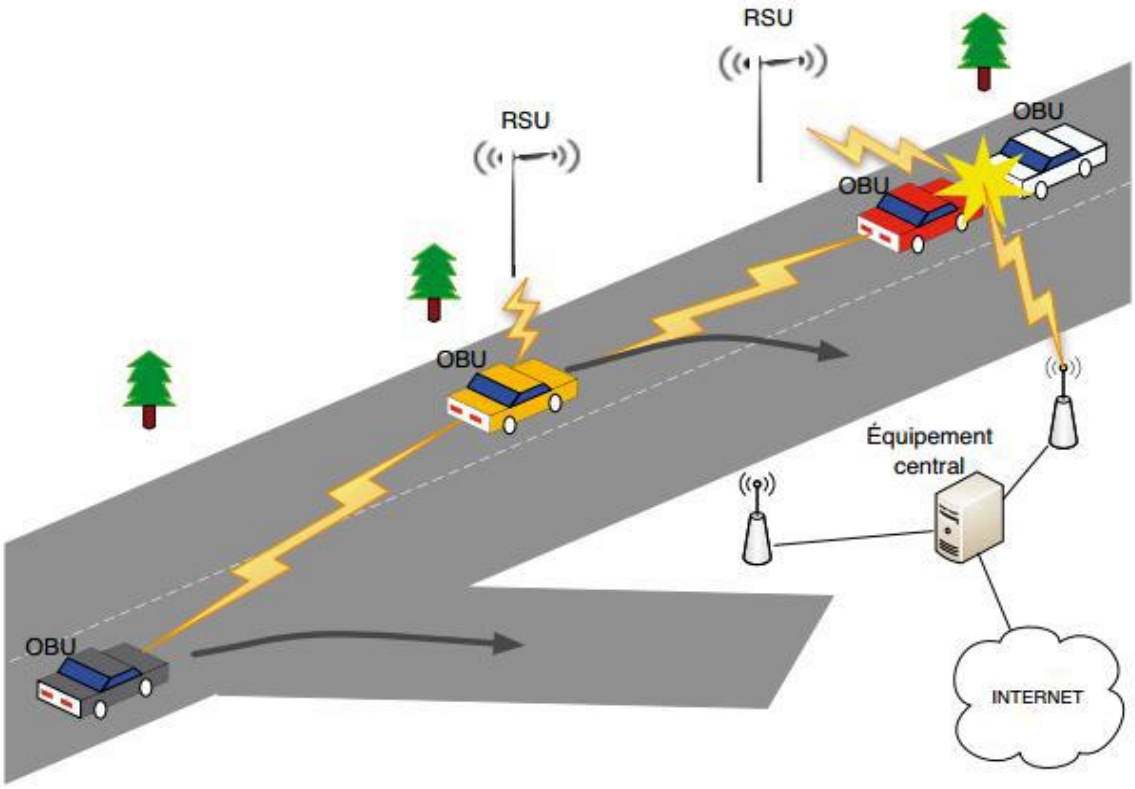


Figure I.7 : Alerte de collision dans les VANETs

I.3.7.2. Conduite coopérative :

Une autre application intéressante dans les VANETs est la sécurité coopérative aux intersections illustrées à la figure I.8, dans laquelle les véhicules échangent des messages afin de rendre le passage plus sécurisé.

Figure I.8 : Sécurité coopérative aux intersections

I.3.7.3. La localisation par carte :

La localisation par carte est une application d'assistance au conducteur, dans laquelle par exemple une direction de trajet entre deux points dans une ville peut être tracée sur une carte afin d'aider des conducteurs perdus dans une partie inconnue de la ville.

I.3.7.4. Parking intelligent :

La localisation des emplacements de parking libre est une application d'assistance au conducteur qui permette de gagner le temps et le carburant gaspillé dans la recherche des emplacements libres pour une voiture.

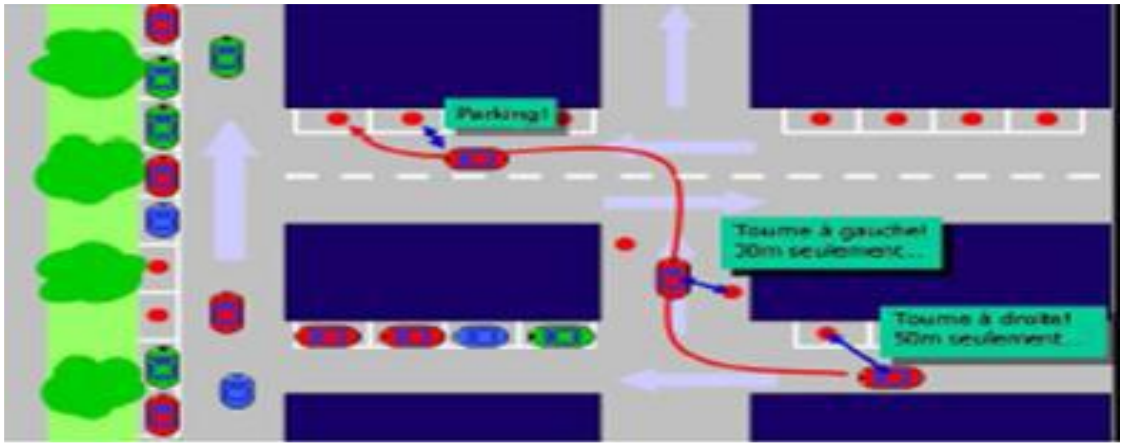


Figure I.9 : Parking intelligent

I.3.7.5. Applications de confort :

Les utilisateurs au bord d'une voiture peuvent avoir accès à plusieurs services de communication et d'informations (en passant d'une voiture à une autre jusqu'au point d'accès le plus proche) comme l'accès mobile à l'Internet pour téléchargements des fichiers MP3, le chat inter-véhicules et les jeux en réseaux.

I.3.8. Le routage dans les réseaux VANET : [1] [5] [6]

I.3.8.1.Introduction :

Le rôle des protocoles de routage est d'acheminer et assuré un échange d'information entre les nœuds d'une manière efficace, comme les réseaux ad hoc sont des réseaux sans infrastructure donc le routage est basé sur des communications multi-saut, cela rend la communication entre deux ou plusieurs nœuds possible même si ils ne sont pas dans la même porté de transmission radio.

La stratégie de routage doit prendre en considération les différentes caractéristiques des réseaux VANET (changements de topologie, la forte mobilité, la capacité limité des liaisons radio…) pour assurer une stratégie sui garantit une connectivité du réseau permanente.

Dans ce qui suit on va classifier les protocoles de routage selonplusieurs critères, et présenté quelques protocoles.

I.3.8.2.Classification des protocoles de routage dans les réseaux VANET : [6]

Les réseaux véhiculaires ont comme caractéristique principale une forte mobilité qui entraîne une topologie très dynamique. Cette caractéristique fait que les protocoles de routage traditionnels des MANETS sont pour la plupart in adaptée aux VANETS. En effet, dans les VANETS, la vitesse peut être beaucoup plus élevée que les MANETS dans certains environnements de communication comme les autoroutes. Dans [Amadou, 2011] [Qabajeh, 2009] Différentes solutions pour le routage dans les réseaux VANET ont été proposées, nous distinguons deux classes de protocoles de routage: les protocoles basés sur la Unicast (topologie) qui sont divisés en protocoles proactifs, réactifs et hybrides et les protocoles basés sur la localisation (géographique) qui utilisent la position physique des nœuds mobiles pour configurer le routage.

I.3.8.2.1. Les protocoles de routage basés sur la topologie :

Ce sont principalement des régimes à base topologique qui utilisent une approche réactive, proactive ou hybride pour créer des itinéraires.

a. Les protocoles réactifs

Les protocoles réactifs construisent des chemins uniquement lorsque ces derniers sont requis par un nœud source et ne gardent que les routes en cours d'utilisation par le processus de routage. On dit alors que la topologie du réseau est découverte à la demande.

Ainsi lorsqu'un nœud cherche à communiquer avec une destination pour laquelle il ne connait pas le chemin, il lance un processus de découverte dans le réseau (généralement par inondation). Cette phase de découverte se termine lorsque le chemin est trouvé. Ces chemins formés sont susceptibles d'être rompus à cause de la haute mobilité des véhicules. Les ruptures de liens sur les chemins sont alors traitées au moyen d'un mécanisme de maintenance, dont le but est de les identifier, puis si possible de les corriger.

Dans ce qui suit, nous allons présenter les protocoles réactifs les plus connus, proposés par certains travaux de recherche pour effectuer le routage dans les réseaux ad hoc

- **Le protocole AODV**

Le protocole de routage AODV (Ad hoc On-demand Distance Vector) [Guizani, 2012] est un protocole décrit dans la [Charles et al, 2003] Ce protocole crée les routes au besoin et utilise le principe de numéro de séquence afin d'utiliser les routes les plus nouvelles, dites encore les plus fraîches. En plus, il utilise le nombre de sauts comme métrique pour choisir entre plusieurs routes disponibles. Trois types de paquets sont utilisés par AODV : les paquets de requête de route RREQ (Route Request Message), les paquets de réponse de route RREP (Route Reply Message) et les paquets d'erreur de route RERR (Route Error Message). En plus de ces paquets, AODV invoque des paquets de contrôle HELLO qui permettent de vérifier la connectivité des routes. AODV repose sur deux mécanismes : découverte de route et maintenance de route. La découverte de route permet de trouver une route pour atteindre une destination et la maintenance de route permet de détecter et signaler les coupures de routes provoquées éventuellement par la mobilité des nœuds.

- **Le protocole DSR**

Le protocole de routage DSR (Dynamic Source Routing) [Guizani, 2012] est un protocole qui est normalisé dans la [Johnson et al, 2007] Ce protocole crée les routes à la demande comme le protocole AODV. Il utilise la technique "routage à la source" dans laquelle la source inclut dans l'entête du paquet la route complète par laquelle un paquet doit passer pour atteindre sa destination. Les noeuds intermédiaires entre la source et la destination n'ont pas besoin de maintenir à jour les informations sur la route traversée puisque la route complète est insérée dans l'entête du paquet. DSR est composé de deux mécanismes : la découverte de route et la maintenance de route. Le premier permet de chercher les routes nécessaires à la demande, tandis que le second permet de s'assurer de la maintenance des routes tout au long de leur utilisation.

On va détailler ce protocole dans le chapitre 3.

b. Les protocoles proactifs

Le principe des protocoles proactifs est de maintenir à tout instant une vue globale et cohérente de la topologie du réseau, et de construire des routes entre les nœuds avant qu'elles ne soient demandées. Ces protocoles exigent que chaque nœud maintienne une table de routage indiquant par quel voisin passer pour atteindre un destinataire. Grâce à ces

informations, chaque nœud dispose à tout instant d'un chemin vers n'importe quel autre nœud du réseau.

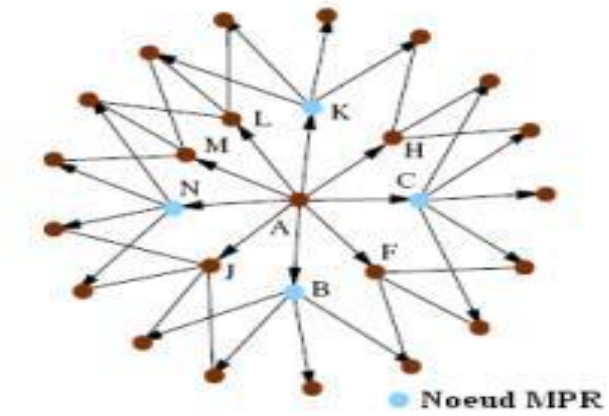
Pour traiter les changements de topologie, les nœuds diffusent des messages de contrôle à travers le réseau. Actuellement, les protocoles Optimized Link State Routing (OLSR) et DSDV ont été proposé pour les réseaux ad hoc véhiculaires

- **Le protocole OLSR**

OLSR est un protocole état de liens, qui optimise la manière de diffusion des messages de contrôle afin d'économiser la consommation de la bande passante, grâce à l'utilisation du concept des "relais multipoints " (MPRs) illustré sur la figure I.10 dans lequel chaque nœud choisit un sous-ensemble de ses voisins pour retransmettre ses paquets en cas de diffusion.

En se basant sur la diffusion en utilisant les MPRs, tous les nœuds du réseau sont atteints avec un nombre réduit de répétitions. Un ensemble de MPRs d'un nœud N est l'ensemble minimal de ses 1-saut voisins qui couvrent (dans le sens de la portée de communication) ses 2-sauts voisins.

Dans OLSR, chaque nœud diffuse périodiquement des messages Hello qui contient l'état de ses liens avec ses 1-saut voisins (unidirectionnel, bidirectionnel ou MPR pour dire que ce voisin est un MPR). Grâce aux messages Hello, un nœud construit sa table des voisins ainsi que la liste des voisins qui l'ont choisi comme MPR dits "MPR-sélecteurs". De plus, un nœud diffuse périodiquement des messages TC (Topology Control) qui contient la liste de ses MPR-sélecteurs.



- **Figure I.10 :** Relais multipoints dans OLSR

En exploitant ces messages, chaque nœud remplit les deux champs nommés "destination" (correspond aux MPR-sélecteurs dans le message TC) et "dernier saut" (prend comme valeur

l'identificateur du nœud émetteur du message TC) d'une table dite de topologie. Les tables de topologie et des voisins sont exploitées pour construire la table de routage.

- **Le protocole DSDV**

Le protocole de routage DSDV (Destination-Sequenced Distance-Vector) [Guizani, 2012] [Charles et al, 1994] est un protocole de routage de type vecteur de distance. Chaque noeud maintient une table de routage contenant des informations sur les destinations accessibles dans le réseau. Ces informations comprennent le noeud suivant utilisé pour atteindre la destination, le nombre de sauts qui sépare le noeud de la destination et le numéro de séquence estampillé par la destinataire. Ce numéro de séquence permet de distinguer les nouvelles routes des anciennes. Chaque noeud envoie périodiquement à ses voisins la totalité de sa table de routage. D'autres paquets de mise à jour sont aussi envoyés à la suite d'un changement dans la topologie du réseau. Ces paquets n'incluent que les entrées de la table affectées par le changement et ont pour objectif de propager les informations de routage aussi rapidement que possible. Quand un noeud reçoit un paquet de mise à jour, il le compare avec les informations existantes dans sa table de routage. Toute entrée dans la table est mise à jour si l'information reçue est plus récente (ayant un numéro de séquence plus grand), ou si elles ont le même numéro de séquence mais avec une distance plus courte.

Dans le protocole DSDV, une unité mobile doit attendre jusqu'à ce qu'elle reçoive la prochaine mise à jour initiée par la destination afin de mettre à jour l'entrée associée à cette destination dans la table de distance. De ce fait, la réaction de DSDV aux changements de la topologie est considérée lente. D'autre part, ce protocole cause une charge de contrôle importante dans le réseau à cause des paquets de mise à jour envoyés périodiquement ou à la suite des événements.

c. Protocoles hybride

Les protocoles hybrides combinent les deux approches précédentes (réactif et proactif). Pour bénéficier de leurs avantages, ils utilisent un protocole proactif, pour connaitre les voisins les plus proches dans le but de réduire le délai et un protocole réactif dans le but de réduire la charge des paquets de contrôles.

Les protocoles hybrides cumulent aussi les inconvénients des protocoles Table-driven et réactifs à savoir, les paquets de contrôles périodiques et le délai de découvertes de routes. Parmi les protocoles hybrides les plus connus on peut citer le protocole : ZRP.

- **Le protocole ZRP**

Le protocole de routage ZRP (Zone Routing Protocol) [Guizani, 2012] [Zygmunt et al, 2003a] est un protocole hybride qui combine les deux approches proactives et réactive. Le protocole ZRP divise le réseau en différentes zones. Pour chaque nœud, il définit une zone de routage exprimée en nombre de sauts maximal σ. Ainsi, la zone de routage d'un noeud inclut tous les noeuds qui sont à une distance au maximum de σ sauts. Les noeuds qui sont exactement à σ sauts sont appelés noeuds périphériques.

À l'intérieur de cette zone, ZRP utilise un protocole proactif et à l'extérieur de cette zone de routage, il fait appel à un protocole réactif.

Le protocole proactif est IARP (IntrAzone Routing Protocol) [Zygmunt et al, 2002d] et celui réactif est IERP (IntErzone Routing Protocol) [Zygmunt et al, 2002c] Chaque noeud doit tout d'abord connaître ses voisins. Pour cela, ZRP utilise soit le protocole de contrôle d'accès au support (MAC) pour connaître les voisins immédiats ou le protocole NDP (NeighbourDiscovery Protocol) pour la transmission et la gestion des échanges de messages HELLO. Par la suite, chaque noeud invoque le protocole IARP pour découvrir les routes vers tous les autres noeuds qui se trouvent dans sa zone de routage. Cependant, le protocole IERP est utilisé à la demande pour chercher les routes entre un noeud et une destination qui se trouvent à l'extérieur de sa zone de routage. Un troisième protocole BRP (BordercastResolution Protocol) [Zygmunt et al, 2002b] est inclus avec IERP pour guider la propagation des requêtes de recherche de route dans le réseau. BRP utilise les données de la topologie fournies par le protocole IARP afin de construire sa liste des nœuds de périphérie et la façon de les atteindre.

Dans le tableau ci-dessous se présente une comparaison entre les protocoles proactifs et les protocoles réactifs.

Routage proactif		Routage réactif	
Avantages	**inconvénients**	**Avantages**	**inconvénients**
La topologie du réseau est connue de tous les mobiles. Les routes sont disponibles immédiatement.	Il faut diffuser régulièrement des informations sur les changements de topologie du réseau.	Les mobiles ne conservent pratiquement aucune information sur la topologie globale du réseau : seules les informations sur les routes actives sont stockées.	
Les protocoles proactifs disposent en permanence d'une route pour chaque destination dans le réseau.	Un volume de signalisations important.	Les protocoles réactifs génèrent à priori un volume plus faible de signalisations.	Les protocoles réactifs engendrent un délai lors de la construction (ou de la reconstruction) des routes et produisent plus difficilement des routes optimales.

Tableau I.1 : analogie entre routage proactif et réactif **[6]**.

I.3.8.2.2.Les protocoles de routage basés sur la géographique :

Les protocoles de routage géographique (ou basés sur la position) utilisent des coordonnées géographiques (par exemple, fournies par un système de géolocalisation tel que le GPS) afin de trouver un chemin vers la destination. Chaque nœud source inclut l'identifiant et la position de la destination dans l'entête de tout paquet à envoyer, les nœuds recevant ce paquet utilisent les informations géographiques incluses dans ce dernier et celles disponibles dans leurs tables de routage pour retransmettre le paquet et répètent le même mécanisme jusqu'à ce que celui-ci atteigne la destination.

L'avantage majeur de ces protocoles par rapport aux protocoles basés topologie, est qu'ils réduisent considérablement les paquets de contrôles, particulièrement dans les réseaux larges et dynamiques. Parmi les protocoles géographiques les plus largement étudiés : GPSRet GyTAR.

- **Greedy Perimeter Stateless Routing (GPSR) :**

Greedy Perimeter Stateless Routing, GPSR est un protocole de routage réactif et efficace pour les réseaux ad hoc véhiculaire qui exploite la correspondance entre la position géographique et la connectivité dans un réseau sans fil afin de prendre des décisions de transfert de paquets**.**

Dans un réseau VANET, les nœuds sont susceptibles de se déplacer. Il est donc nécessaire d'utiliser un mécanisme qui permet à chaque nœud de connaître la position de ses voisins, afin de signaler leur présence et leur localisation, les nœuds inondent le réseau en

envoyant un paquet de signalement (messages« beacon») contenant la position et un identifiant (par exemple, son adresse IP). L'échange périodique de ces paquets permet aux nœuds de construire leur table de position. La période d'émission des messages « beacon » dépend du taux de mobilité dans le réseau ainsi que de la portée radio des nœuds. En effet, lorsqu'un nœud ne reçoit pas de message « beacon » d'un voisin après un temps T, il considère que le voisin en question n'est plus dans sa zone de couverture et l'efface de sa table de position. Un des avantages des messages « beacon » est qu'un nœud n'a pas besoin que des informations sur ses voisins directs, ce qui nécessite peu de mémoire. Alternativement, le protocole GPSR permet au nœud d'encapsuler sur quelques bits leur position dans les paquets de données qu'il envoie, « We encode position as two four-byte floating point quantities, for x and y coordinate values. »**.** Dans ce cas, toutes les interfaces des nœuds doivent être en mode promiscuité afin de recevoir les paquets s'ils se trouvent dans la zone de couverture de l'émetteur.

L'acheminement des paquets par GPSR se fait selon deux modes suivant la densité du réseau : le « Greedy Forwarding » et le « Perimeter Forwarding » (appelés respectivement GF et PF dans la suite).

- **Exemple de scénario :**

Ce protocole détermine la route à suivre en minimisant les distances entre les noeuds et la destination (c'est le mode Greedy Forwarding), mais un second mécanisme est mis en oeuvre en cas de blocage (c'est le mode Perimeter). Dans ce cas, le noeud n'ayant pas de voisin plus proche (en distance) que lui de la destination passe le relais à ses voisins qui eux peuvent avoir un voisin plus proche de la destination (en distance). Sur la **Figure I.22**, le noeud B utilise le mode Perimeter car il n'a pas de voisin plus proche en distance de la destination finale G, ce qui permet de trouver une route passant par le noeud C qui, lui, peut à nouveau utiliser le mode Greedy Forwarding ayant un voisin plus proche de G (en l'occurrence D).

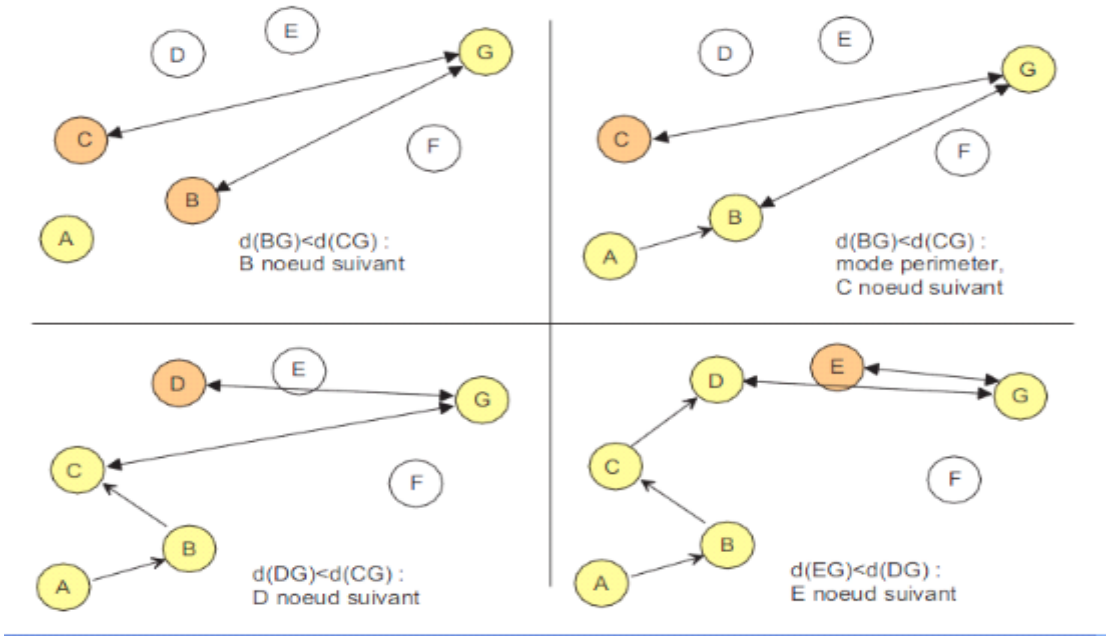


Figure I.11 : Exemple de scénario de protocole GPSR

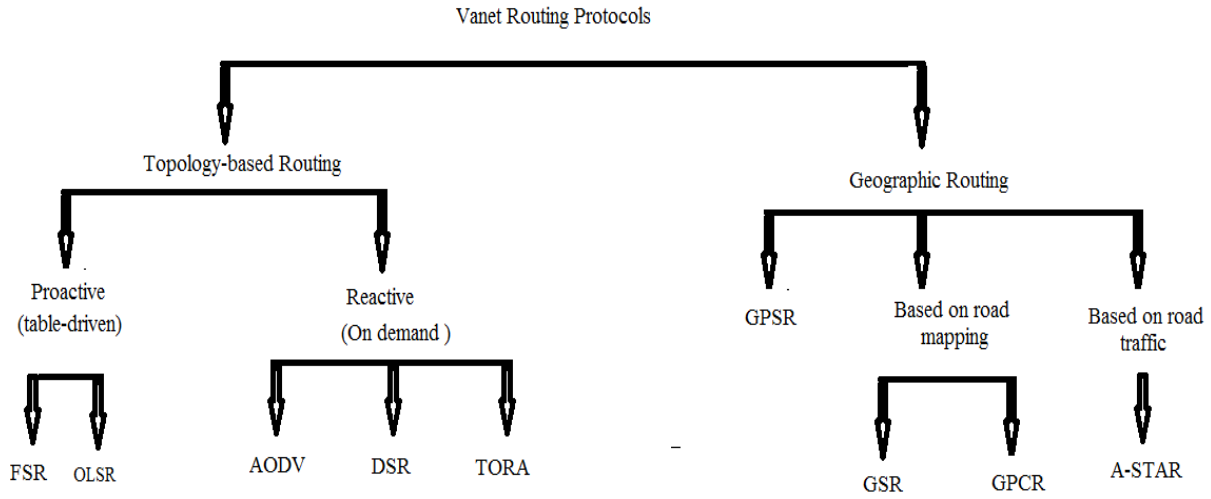


Figure I.12 : illustre la taxonomie des protocoles de routage dans les VANETs.

I.9. Conclusion :

Dans ce chapitre nous avons présenté les réseaux véhiculaires VANET, qui utilisent les mêmes principes des réseaux MANET mais en les appliquant sur des véhicules.

Les réseaux VANET commencent àêtre appréciés et reconnus par les usagers des véhicules et sont de plus en plus utilisés dans différents pays. Ils aident les utilisateurs dans leur conduite et minimisent les risquent d'accidents, et offrent un plus de confort aux usagers et la possibilité d'accès à des loisirs.

Dans la suite de notre travail, nous allons s'intéresser à la sécurité dans les réseaux VANET ,et donner des solutions au projets de partenariat entre les constructeurs automobiles et ceux des appareils de télécommunication ne cessent d'augmenter, dans le but de moderniser les voitures en les équipant des périphériques nécessaires pour intégrer les réseaux VANET, cela va peut-être permettre, dans un avenir proche, de crée un réseau véhiculaire mondial.

II.1. Introduction :

Les communications véhiculaires constitueront dans le futur le plus grand réseau ad hoc viable. De plus, la vie de milliers d'êtres humains sera dépendante des informations échangées entre les véhicules eux-mêmes et avec les infrastructures. A cause de l'importance des informations échangées et du nombre énorme d'utilisateurs, l'environnement des réseaux véhiculaires sera plus qu'hostile. En effet, les messages liés à la sécurité peuvent être falsifiés ou éliminés par des entités malveillantes afin de causer des accidents et mettre en péril la vie des personnes. Donc, avant le déploiement de ces réseaux, des mécanismes de sécurité appropriés doivent être mis en œuvre afin d'éviter ces mauvais scénarios et d'identifier les entités responsables de ces activités malveillantes.

Dans ce chapitre, nous présentons un récapitulatif sur les outils et mécanismes de base de la sécurité en générale, nous passons en revue la sécurité dans les réseaux sans-fil, ensuite nous présentons les problèmes et les mécanismes de base de sécurité dans les VANETs, enfin nous étudions les techniques et solutions de sécurité existantes qui peuvent être mises en œuvre afin de sécuriser les informations échangées à travers ces réseaux.

II.2. La sécurité dans les réseaux sans-fil : [8]

Comme les réseaux VANET peuvent être considérés comme une sous classe des réseaux sans-fil ad hoc, ils en héritent les problèmes de sécurité. Dans cette section, nous nous intéressons à la sécurité des réseaux sans-fil ad hoc de manière générale, nous présentons quelques exemples d'attaques sur ces réseaux, ensuite nous en décrivons les objectifs de sécurité.

II.2.1.Vue globale des besoins de sécurité dans les réseaux sans-fil :

Lors de l'analyse de la nature des communications dans les réseaux ad hoc, des propriétés spécifiques liées à la sécurité et la confidentialité doivent être prises en compte pour la conception des protocoles de communications, à savoir :

- **La mobilité**

La mobilité des nœuds rend la topologie des VANETs instable. Il n'est donc pas facile pour un nœud de connaître correctement son voisinage. Les attaquants peuvent ainsi forger et diffuser des fausses informations de topologie pour construire des routes qui passent par eux et réaliser ainsi des attaques qui visent à causer des accidents ou la congestion de routes. Par ce moyen, un protocole de routage ad hoc non-sécurisé peut facilement être attaqué. De plus, la mobilité des attaquants peut aussi les rendre plus difficiles à détecter ou la localiser.

En comparaison avec les réseaux traditionnels, il n'y a pas autant de mobilité dans les réseaux filaires, et dans les réseaux cellulaires ce sont des infrastructures qui gèrent la mobilité, donc il est nécessaire de construire des protocoles de routage spécialement pour les VANETs capables de découvrir correctement la topologie du réseau même sous attaques.

- **Un support sans fil partagé :**

La nature de transmission radio dans l'air, permet à un intrus d'écouter passivement tous les messages échangés pourvu qu'il se trouve dans la zone d'émission, en opérant en "promiscues mode" et en utilisant un logiciel qui permet de capturer paquets émis (sniffer) . L'adversaire, a donc accès au réseau et peut intercepter aisément les données transmises, sans même que l'émetteur ait connaissance de l'intrusion. L'intrus, en étant potentiellement invisible, peut brouiller le canal radio pour bloquer les transmissions, injecter massive de paquets visant à épuiser les ressources des nœuds, enregistrer, modifier, et ensuite retransmettre les paquets comme s'ils avaient été envoyés par un utilisateur légitime, donc les VANETs ont besoin de nouveau mécanismes afin de sécuriser l'accès au réseau.

- **Les communications multi-sauts :**

Les protocoles de communications multi-sauts sont obligatoires pour avoir des communications sans-fil à longue portée dans les réseaux ad hoc ; cela signifie que tous les nœuds doivent coopérer pour assurer le fonctionnement du réseau. Malheureusement, les nœuds malveillants peuvent exploiter ce principe et mettre en péril la sécurité du réseau, donc des mécanismes de sécurité appropriés doivent être mis en œuvre.

- **La diffusion d'information de la position géographique :**

Avec certains protocoles dans les réseaux ad hoc mobiles, les nœuds sont supposés envoyer périodiquement des messages (balises) indiquant leurs positions courantes ou éventuellement d'autres données nécessaires pour des services spécifiques. Par conséquent, les attaquants peuvent créer un profil sur les trajectoires des nœuds et donc les utilisateurs du réseau.

- **Les opérations autonomes :**

Les nœuds eux-mêmes déterminent leurs états et décident des informations à envoyer de manière autonome. Par conséquent, il est facile pour les entités malveillantes qui ont le contrôle sur un ou plusieurs nœuds d'envoyer des informations falsifiées. Les systèmes de

sécurité, à leur tour, doivent employer des mécanismes qui détectent et empêchent l'utilisation de ces informations.

- **Manque des serveurs centraux**

Dans les VANETs puisqu'il n'y a pas forcément de serveur central, la distribution et la gestion de clés peuvent être difficiles à réaliser. Dans les réseaux traditionnels les solutions de sécurité s'appuient souvent sur des relations de confiance préalablement établies ou des autorités de confiance tierces, et utilisent les primitives cryptographiques pour authentifier les nœuds et sécuriser les échanges de données. Afin d'utiliser ces moyens cryptographiques dans les VANETs, nous devons étudier comment établir des autorités de confiance ou des relations de confiance entre les nœuds sans l'aide d'aucune infrastructure.

- **Nœuds compromis**

Les nœuds dans un VANET sont plus faciles à compromettre que ceux des réseaux traditionnels, parce qu'ils sont de nature mobile, en plus les VANETs peuvent être divisés et/ou fusionnés, les attaquants auront plus de chances d'attaquer (compromettre) des nœuds sans être aperçus. Dans les réseaux ad hoc il y a quelques attaques très sophistiquées, par exemple les attaques de type ‘’wormhole’’ [54] ne peuvent être commises que par des nœuds compromis et sont difficiles à éviter.

L'utilisation de la cryptographie ne permet pas de résoudre le problème de ces nœuds compromis par une simple authentification, car ces nœuds ont été des participants légitimes au processus de routage avant d'être contrôler par des attaquant, c'est pourquoi nous devons donc considérer spécialement d'autres solutions pour ce problème.

II.2.2.Les objectifs de la sécurité : [8]

La sécurisation des communications dans les réseaux sans-fil comme dans les réseaux filaires nécessite la mise en œuvre de mécanismes permettant d'atteindre un certain nombre d'objectifs généraux de sécurité. Ces objectifs comprennent :

- **L'authentification :** cet objectif de sécurité permet aux membres du réseau de s'assurer de la bonne identité des membres avec lesquels ils communiquent.
- **La non-répudiation** : cet objectif de sécurité permet de s'assurer qu'aucun émetteur ne peut nier d'être à l'origine d'un message. Cet objectif est indispensable dans les transactions électroniques et dans toutes les communications sensibles.

- **La confidentialité** : cet objectif de sécurité garantit que seules les parties autorisées peuvent accéder aux données transmises à travers le réseau. Ces données peuvent concerner la couche applicative ou les couches inférieures.
- **L'intégrité :** cet objectif de sécurité permet de s'assurer que les données échangées ne sont pas soumises à une altération volontaire ou accidentelle. Donc, il permet aux destinataires de détecter les manipulations de données effectuées par les entités non autorisées et rejeter les paquets correspondants.
- **La disponibilité :** cet objectif de sécurité vise à garantir aux entités autorisées d'accéder aux ressources du réseau avec une qualité de service adéquate.

II.2.3.Le modèle d'un attaquant :

La première étape pour sécuriser un système est l'identification de la nature des éventuels attaquants. Dans les réseaux ad hoc, nous pouvons classifier un attaquant selon les dimensions suivantes :

- **Interne vs. Externe :** l'attaquant interne est perçu comme un membre normal du réseau et peut communiquer avec les autres membres. La présence des attaques internes est très problématique et difficile à détecter, car elle annule le niveau de sécurité assuré par les techniques cryptographiques. L'attaquant externe est considéré par les nœuds membres comme un intrus et est donc limité dans la diversité des attaques qu'il peut provoquer.
- **Malveillant vs Rationnel** : un attaquant malveillant n'a pas d'intérêts personnels à travers ses attaques et a pour but le dysfonctionnement du réseau. Par conséquent, il peut employer tous les moyens sans tenir compte des coûts correspondants et des conséquences. Par contre, un attaquant rationnel cherche un profit personnel, et ainsi, on peut prévoir les cibles d'attaques et les moyens employés.
- **Passif vs. Actif** : l'attaquant passif écoute simplement les informations qui sont échangées entre les nœuds tandis que l'attaquant actif agit sur les informations qui sont échangées. Il peut les falsifier, les modifier, voire même les détruire.

II.2.4 Les attaques dans les réseaux sans-fil :

Dans les réseaux sans-fil ad hoc, la nature du support de transmission rend ces réseaux plus vulnérables aux attaques qu'un réseau filaire. Un réseau sans-fil qui n'est pas bien sécurisé est exposé à de plusieurs types d'attaques ; nous en citons :

- **L'écoute des communications** (en anglais, *eavesdropping* ou *sniffing*): dans ce type d'attaque, l'adversaire ou l'entité malveillante écoute sur le support de transmission

afin d'extraire des informations sur le trafic échangé dans son voisinage ; il se peut qu'il veuille espionner sur des informations personnelles, ou bien collecter des informations pour les analyser et effectuer ensuite d'autres types d'attaques.

- **L'accès non-autorisé** : dans cette attaque, les entités malveillantes accèdent aux services du réseau sans en avoir les droits ou les privilèges.
- **Le déni de service** (souvent dénoté par DoS abréviation de l'expression en anglais « *Denial of Service* ») : il consiste à rendre les différentes ressources et les services indisponibles pour les utilisateurs dans le réseau ; il est généralement provoqué par d'autres attaques visant la bande passante ou les ressources énergétiques des autres nœuds. La technique la plus naïve pour causer un déni de service dans un réseau sans-fil consiste à causer le brouillage du canal (en anglais *Jamming*); une autre attaque appelée « privation de sommeil » qui consiste à demander un service que le nœud visé offre de manière répétitive afin de lui gaspiller ses ressources systèmes et de l'empêcher de "se reposer".
- **L'usurpation de l'identité d'un noeud** (en anglais, *Spoofing* ou *Impersonation*) : dans ce type d'attaques, l'attaquant essaie de prendre l'identité d'un autre noeud afin de pouvoir recevoir ses messages ou d'avoir des privilèges qui ne lui sont pas accordés.

II.3. concept de base de la sécurité : [9]

- **La cryptographie** : la cryptographie est une des disciplines de la cryptologie s'attachant à protéger des messages en employant souvent des secrets ou des clés. Elle consiste à appliquer des transformations sur le contenu d'un message à l'aide des algorithmes de chiffrement (afin de l'en rendre incompréhensible) et de déchiffrement (afin de reconstruire le message original).

- **La cryptographie symétrique**

  Consiste à utiliser la même clé et le même algorithme de cryptage/décryptage, deux entités peuvent communiquer en toute sécurité tant que leur clé n'est pas compromise. Dès le début, la clé est échangée entre les deux acteurs de la communication et elle reste confidentielle.

- **La cryptographie asymétrique**

  Dans ce type de cryptographie, chaque utilisateur possède une paire de clés :

- Une clé privée qui doit être gardée secrète.

- Une clé publique qui est disponible pour tous les autres utilisateurs.

Ces deux clés sont mathématiquement liées.

Le cryptage asymétrique peut assurer soit la confidentialité, soit l'authentification tout dépend de la façon avec laquelle les deux clés (privée/publique) sont utilisées.

La clé publique sert a crypter les messages tandis que la clé privée sert a les décrypter .une fois le message crypté, seul le destinataire est en mesure de décrypter son contenu secret.

- **Le hachage :**

Il consiste à déterminer une information de taille fixe et réduite (appelée l'empreinte ou le condensé) à partir d'une donnée de taille indifféren**te.**

- **Les fonctions de hachage à sens unique** :

Une fonction de hachage à sens unique est une fonction irréversible qui fournit l'empreinte à partir d'une chaine fournie en entrée. La particularité de cette fonction est qu'il est aisé de calculer l'empreinte d'une chaîne donnée, mais il est difficile de retrouver ou déduire la chaîne initiale à partir de l'empreinte **[8]**.

- **La signature numérique** :

C'est un code numérique associé à un message électronique afin que les destinataires puissent en authentifier les origines et en vérifier l'intégrité. Son implémentation fait appel aux fonctions de hachage et à clé privée du signataire.

- **Le MAC** (Message Authentication Code) :

C'est un code accompagnant des données qui assure les mêmes fonctionnalités de la signature numérique, mais son implémentation se base sur l'utilisation de la clé secrète et sur des fonctions similaires à celles de hachage.

- **Le certificat numérique :**

C'est une structure de données permettant de prouver l'identité du propriétaire d'une clé publique. Les certificats numériques sont signés et délivrés par un tiers de confiance appelé l'**autorité de certification** (AC).

II.4. La sécurité dans les VANETs : [5]

A cause de l'importance de l'information envoyée aux équipements embarqués dans les véhicules, la sécurité des communications dans les VANETs ne consiste pas seulement à assurer les objectifs décrits dans la section II.2.2, mais d'autres objectifs et contraintes doivent

être pris en compte tel que la consistance de données des messages générés par les autres véhicules et l'aspect temps réel des applications liées à la sécurité. Dans cette section, nous présentons des attaques spécifiques sur les VANETs, et les mécanismes de base qui ont été mis en œuvre pour la sécurité de ces réseaux.

II.4.1.Attaques spécifiques sur les VANETs :

Dans cette section, nous passons en revue quelques attaques spécifiques sur les VANETs. Ces attaques comprennent :

- ✓ **L'injection des messages erronés** (figure II.1) :

  Dans cette attaque l'entité malveillante crée des messages contenant des informations erronées afin de causer un accident ou de rediriger le trafic routier de manière permettant la libération de la route utilisée.

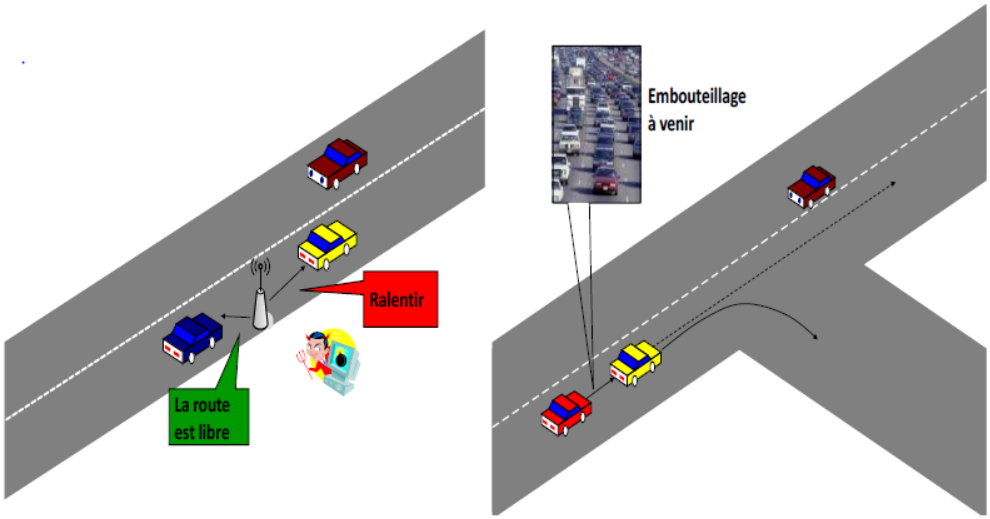


Figure II.1 : Attaques par l'envoi de messages falsifiés**.**

- ✓ **Le déni de service** (figure II.2) **:**

  L'objectif de cette attaque est d'empêcher la réception d'un message lié à la sécurité, donc il vise à annuler les services de sécurité offerts par ces réseaux.

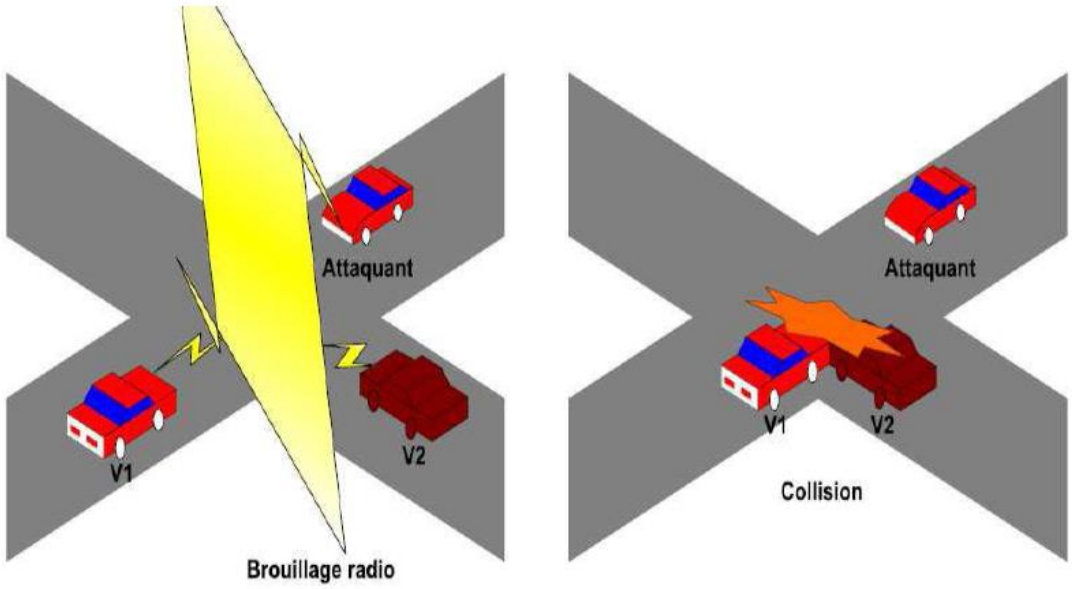


Figure II.2 : attaque déni de service**.**

En utilisant le brouillage du canal l'attaquant empêche V1 et V2 à recevoir les messages liés à la sécurité.

- ✓ **La révélation d'identité et de position géographique des autres véhicules** (figure II.3) **:**

  Dans cette attaque, l'entité malveillante collecte des informations sur les transmissions radio effectuées par le véhicule victime afin de surveiller sa trajectoire. L'utilité de cette attaque est diverse et dépend de l'entité collectant ces informations (il peut être par exemple une entreprise de location de voitures qui veut suivre ses propres véhicules de manière illégitime).

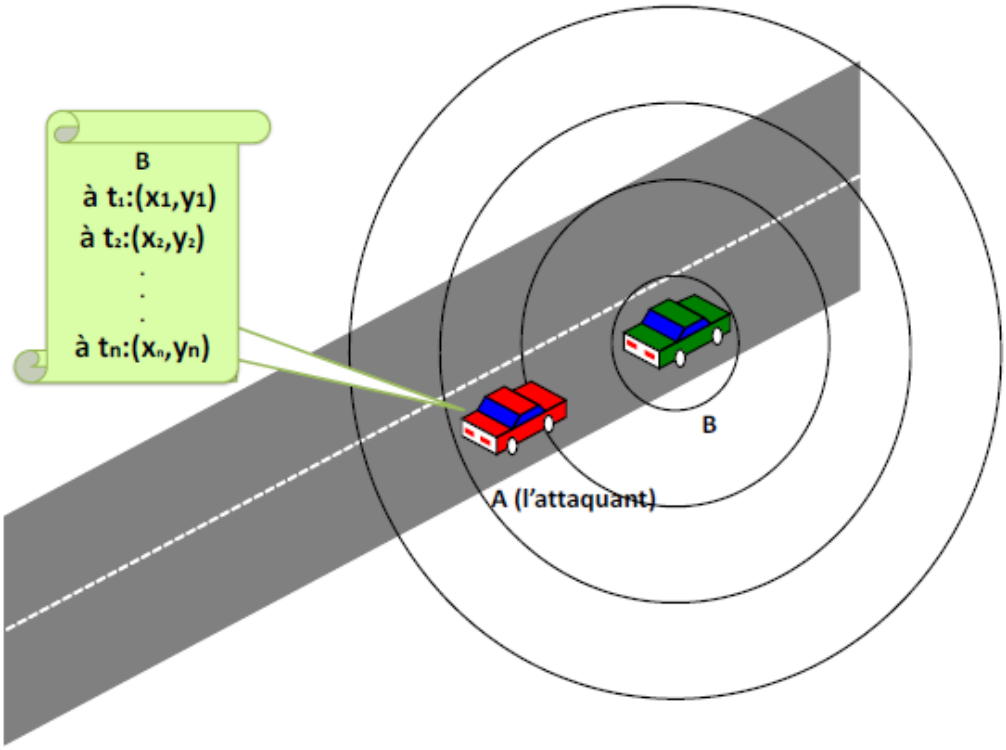


Figure II.3 : Attaque de révélation d'identité et de position géographique d'un véhicule **[10]**.

✓ **Modification des paquets**

Un nœud malveillant peut modifier une partie du paquet qui transite par lui. Par exemple, dans AODV l'attaquant peut diminuer le champ hop count dans un message de découverte de route.

✓ **Suppression des paquets de routage**

Cette attaque peut concerner les paquets de routage et les paquets de données. Un attaquant peut supprimer un paquet qui est envoyé à travers lui, ou un nœud égoïste peut être silencieux quand il doit envoyer des paquets de routage. Par exemple, un nœud peut refuser de rediffuser le RREQ, ou d'envoyer des informations de routage périodiques lorsqu'un protocole de routage proactif est utilisé. La suppression des paquets peut être utilisée pour réaliser des attaques liées à la dégradation des performances et de modification de topologie.

II.4.2.Les éléments de base de la sécurité dans les VANETs :

II.4.2.1. Le TPD (Tamper-Proof Device) :

C'est un dispositif considéré comme inviolable utilisé pour stocker les informations sensibles comme les clés privées et toutes informations confidentielles, et chargé de signer les messages sortants.

Le TPD est conçu de manière à détruire automatiquement toutes les informations stockées lors de la manipulation matérielle. A cet effet, il contient un ensemble de capteurs qui lui permettent de détecter ces manipulations et effacer toutes les informations stockées afin de les empêcher d'être compromises. Ce module est connu aussi sous le nom de HSM (Hardware Security Module).

II.4.2.2. Les certificats dans les VANETs :

Pour assurer les objectifs de sécurité dans ces réseaux, des outils cryptographiques doivent être mis en œuvre. La cryptographie asymétrique présente des solutions possibles pour les VANETs et parait plus adéquate aux caractéristiques et exigences de ces réseaux. En effet, grâce à la cryptographie asymétrique, il est possible d'utiliser des certificats numériques pour identifier les véhicules de façon unique.

Dans les VANETs il existe deux types de certificats :

- **Le certificat à long terme**

  Chaque véhicule doit avoir un certificat indiquant le véhicule et son propriétaire de manière permanente **;** ce type de certificat contient d'autres informations en plus comme celles concernant les caractéristiques des équipements du véhicule. Il peut être utilisé pour établir une communication sécurisée avec l'AC et renouveler les certificats à court terme.

- **Le certificat à court terme**

  Comme son nom l'indique, la durée de vie de ce certificat est très courte (d'environ une minute**)** ; il ne doit pas contenir lesinformations indiquant le propriétaire du véhicule ; à cet effet il utilise un pseudonyme qui permet d'identifier le véhicule de façon unique. Ce type de certificat est utilisé généralement dans les protocoles de routage.

Il faut souligner que chaque véhicule possède un seul certificat à long terme et plusieurs certificats à court terme. Ainsi, toutes les clés privées correspondantes aux clés publiques sont stockées dans le TPD, donc le TPD doit avoir une grande capacité de stockage afin que les

véhicules puissent communiquer de manière sécurisée même en absence de connectivité avec l'AC pour des périodes très longues.

II.4.2.3. La sécurité du système de balisage :

Le balisage (en anglais *Beaconing*) consiste en la diffusion périodique aux voisins aun saut d'un paquet spécifique contenant des informations utiles pour les applications ou les protocoles exécutés au niveau des noeuds voisins. Généralement, les informations incluses dans les balises (en anglais *Beacons*) comprennent des informations sur le noeud tels l'identifiant, les coordonnées géographiques et la vitesse de déplacement. La fréquence des balises varie de 1HZ à 10HZ dans la plupart des cas.

Afin de sécuriser l'opération de balisage, chaque noeud V calcule la signature numérique sig(E, m) sur les différents champs du paquet (m dénote les champs qui correspondent aux informations énoncées ci-dessus et E l'entête du paquet) à envoyer en utilisant sa propre clé privée CPrV qui correspond à sa clé publique CPuV. La signature numérique sig(E, m) est ensuite ajoutée au message qui sera envoyé conjointement avec son propre certificat numérique CRTV.

Les noeuds recevant ce message peuvent authentifier la source du message grâce à la clé publique CPuV incluse dans le certificat numérique CRTV. Le format d'un paquet balise est illustré dans la figure ci-dessous.

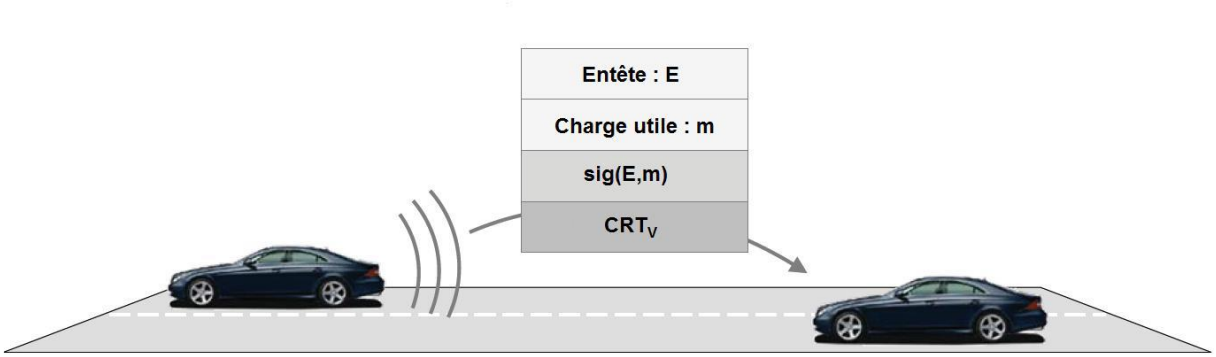


Figure II.4 : Format d'un paquet balise.**[11]**

II.4.3.La confidentialité dans les VANET :

La confidentialité de l'identité et de la localisation sera parmi les préoccupations des propriétaires de véhicules et les techniques cryptographiques seules ne peuvent pas assurer cet

objectif. La plupart des recherches établies à ce stade proposent l'utilisation de mécanismes de changement de pseudonymes qui assurent cet aspect de confidentialité à une certaine mesure, mais ces solutions ont toujours leur impact négatif sur la performance des protocoles de routage dans les VANETs.

II.5. La sécurité de routage dans VANETs :

II.5.1 Les attaques contre les protocoles de routage :

II.5.1.1Pour quoi attaquer les protocoles de routage ?

Généralement, dans le contexte ad hoc, les attaques contre les protocoles de routage peuvent avoir les trois objectifs suivants **:**

- ✓ Avoir plus de contrôle sur les communications entre certains nœuds.
- ✓ Dégrader la qualité de service fourni par le réseau (par exemple en termes de délais et de taux d'acheminement de paquets).
- ✓ Epuiser les différentes ressources critiques comme la bande passante, l'énergie et la capacité du traitement de quelques nœuds.

Il faut noter que ces objectifs ne sont pas entièrement indépendants les uns des autres, et parfois le fait d'atteindre l'un de ces objectifs mène directement aux autres.

II.5.1.2.Les mécanismes d'attaques contre les protocoles de routage :

Dans les réseaux ad hoc, une attaque n'est qu'une combinaison spécifique de quelques mécanismes d'attaques ayant pour but d'atteindre un ou plusieurs objectifs présentés dans la section précédente. Ces mécanismes d'attaques comprennent des actions élémentaires comme l'écoute du trafic, le rejeu (*Replaying*), la modification et l'élimination des paquets de contrôle (c'est-à-dire les paquets utilisés dans l'opération de routage qui ne sont pas des paquets de données). En plus, l'attaquant peut essayer de forger des paquets de contrôle falsifiés (en anglais *Packet forgery*), ou bien créer des paquets de contrôle sous une fausse identité (en anglais *Spoofing*) [1].

Les attaquants peuvent rejouer et modifier les paquets de données, mais ces manipulations ne sont pas considérées typiquement comme une problématique de sécurisation de routage, et doivent être détectées à l'aide d'outils cryptographiques au niveau des couches supérieures (le modèle de vérification de bout en bout) ou des couches inférieures (le modèle de vérification de saut par saut).

II.5.1.3.Exemples d'attaques contre les protocoles de routage :

Dans cette section, nous décrivons quelques attaques connues sur les protocoles de routage dans les réseaux ad hoc :

- ✓ **Attaque Blackhole** : dans cette attaque, le nœud malveillant élimine tous les paquets de données passant par lui.
- ✓ **Attaque Grayhole** : c'est une variante de l'attaque *Blackhole* qui consiste à éliminer seulement les paquets de données de certaines applications qui sont vulnérables à la perte de paquets.
- ✓ **Attaque Sinkhole :** dans cette attaque un noeud malveillant essaie d'attirer les paquets de ses voisins à passer par lui, ce qui lui permet par la suite de modifier leurs contenus ou les éliminer. Donc, l'attaque *Sinkhole*peut être utilisée pour monter d'autres attaques comme le *Blackhole* et le *Grayhole*.
- ✓ **Attaque Flooding :** elle consiste à inonder le réseau par un nombre élevé de paquets (paquets de demande de route, de formation de groupes...etc.) afin de générer un trafic supplémentaire important et causer une dégradation de performance du protocole de routage.
- ✓ **Attaque Sybil :** c'est une variante de l'attaque *Spoofing*, où un seul noeud prétend être plus qu'un seul en utilisant simultanément plusieurs identités différentes dans le réseau afin d'avoir la capacité de monter plus aisément des attaques. Cette attaque est très dangereuse sur le routage géographique, car un noeud peut prétendre être sur plusieurs positions stratégiques à la fois, afin d'être choisi comme relai pour l'acheminement de leurs paquets ; ce qui donne lieu à une attaque *Sinkhole*.

II.5.2. Mécanismes de sécurité de routage dans VANET :

Pour faire face aux attaques décrites dans la section précédente, de nombreux mécanismes de sécurité de routage ont été proposés dans la littérature qu'on peut les classer dans les trois catégories suivantes :

• Les mécanismes de routage sécurisé garantissent l'authentification, la confidentialité, l'intégrité et finalement la non-répudiation dans les deux phases de routage : découverte de route et la transmission des données.

• Les systèmes détection d'intrusion pour surveiller les noeuds des VANETs.

• Les mécanismes de gestion des clés qui traitent l'identification et toutes les questions concernant (création, la distribution, la révocation, le renouvellement et l'échange des clés).

II.5.2.1. Les protocoles de routage sécurisés :

Les protocoles de routage ad hoc présentés dans le chapitre 1 sont spécifiés sans aucune mesure de sécurité, cependant la sécurité de ce service est identifiée comme essentielle pour assurer un large déploiement de ces réseaux et susciter leurs intérêts dans la sécurité routière. Dans cette section, nous présentons certains protocoles de routage sécurisés qui sont en effet des protocoles de routages existants renforcés par des mécanismes de sécurité supplémentaires.

- **ARIADNE**

Le protocole Ariadnea été proposé par Hu et al. Comme extension au protocole DSR en se basant sur la cryptographie symétrique utilisée par la technique d'authentification TESLA (Timed Efficient Stream Loss-tolerant Authentication).

- **Secure Routing Protocol**

Le protocole SRP (Secure Routing Protocol) est conçu comme une extension sécurisée de protocole de routage réactif DSR en se basant sur la cryptographie symétrique.

- **Secure Ad-Hoc On demand Distance Vector**

SAODV est une extension du protocole AODV pour assurer l'authenticité et l'intégrité des messages de routage, et pour éviter les manipulations de la valeur de nombre de sauts (HOP-COUNT).

La figure II.5 : illustre les techniques cryptographiques utilisées dans ces protocoles.

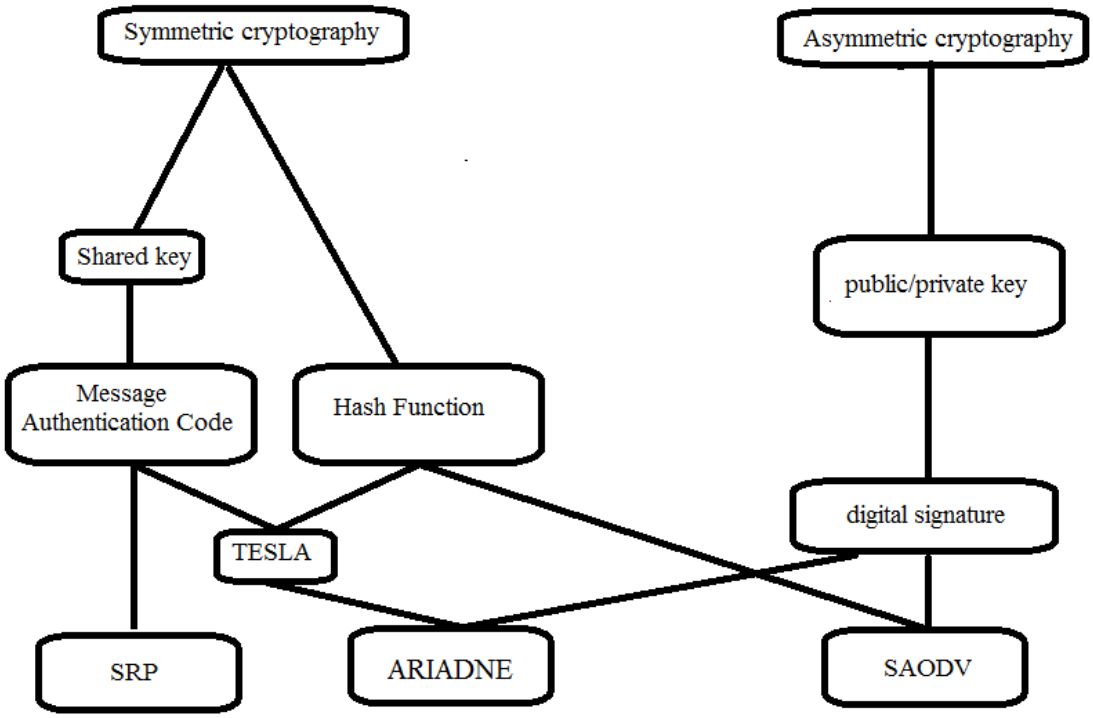


Figure II.5 : Protocoles de routage sécurisés

II.5.2.2.Les systèmes de détection d'intrusions :

L'utilisation des techniques comme la cryptographie n'offre pas la possibilité de détecter de nouvelles attaques, ni même de défendre le réseau contre des nœuds internes compromis.

Toutefois, ce type de système est utilisé comme première ligne de défense alors que la deuxième ligne de défense est occupée par les systèmes de détection d'intrusion communément désignés par son acronyme anglais IDS (Intrusion Detection System). Un IDS fonctionne de trois phases : une phase de collection de données suivie d'une phase d'analyse et enfin une phase de réponse pour prévenir ou minimiser l'impact sur le système. Le système IDS est implanté au niveau de certains noeuds spéciaux appelés moniteurs ou noeuds de surveillance. Le déploiement de ces noeuds diffère en fonction du type protocole et de l'architecture de l'IDS.

Les IDS peuvent être classifiés selon les techniques de détection utilisées :

• système de détection d'anomalie : le système détecte tout comportement qui dévie le comportement normal préétabli et déclenche une réponse.

• système basé sur les signatures : le système possède une base de données de certaines attaques avec laquelle sont comparées les données collectées. Une attaque est détectée si les données collectées coïncident avec un comportement malicieux déjà enregistré.

• système basé sur les spécifications : le système définit un ensemble de conditions qu'un protocole doit satisfaire. Une attaque est détectée si le programme ou le protocole ne respecte pas les conditions établies du bon fonctionnement.

Les IDS peuvent aussi être classés selon l'architecture en : autonome, distribuée et coopérative et hiérarchique.

- **Watchdog and Pathrater**

Marti, Giuli, and Baker ont présenté une solution pour détecter les noeuds malicieux qui suppriment les paquets (de façon sélective ou aléatoire) passant par ce noeud de transit. Cette solution nommée Watchdog consiste en effet, à surveiller le comportement de tous les noeuds d'une part, et choisir la route la plus sécuritaire grâce au module nommé Pathrater d'une autre part. De ce fait, tous les noeuds du réseau se surveillent les uns les autres sous forme d'architecture maillée.

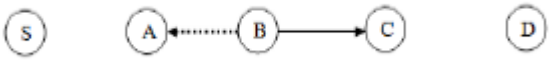


Figure II.6 : Le principe du Watchdog

- **CONFIDANT : un système basé sur la réputation**

Buchegger et Le Boudec ont proposé une extension au protocole de routage DSR appelé CONFIDANT (Cooperation Of Nodes : Fairness In Dynamic Ad-hoc NeTworks), utilisant un mécanisme similaire au mécanisme Watchdog et Pathrater ou chaque noeud observe le comportement de ses voisins. Une fois qu'un comportement malicieux est détecté, le noeud malicieux est exclu de tous les services offerts par le réseau (retransmission des paquets par exemple) et l'isole grâce à un système de réputation en alertant les autres noeuds par la diffusion d'un message d'alarme.

II.5.2.3.Mécanisme de gestion de clés :

Un système de distribution de clé dans les réseaux ad hoc peut être asymétrique ou symétrique, dans le premier cas chaque noeud possède une paire de clés publique / privé géré par une PKI Public Key Infrastructure, dans le second cas, il utilise soit une clé symétrique partagée par tous noeuds d'un réseau, ou plusieurs paires de clés symétriques partagées par chaque deux ou plusieurs noeuds.

- **Gestion de clés asymétriques**

Le déploiement des PKI traditionnelles dans les réseaux ad hoc est problématique, puisqu'un tel système a besoin d'une autorité de certification (CA) qui est un serveur central qui assure la livraison et la révocation de certificats en permanence, en plus le CA doit être toujours connecté et accessible par les noeuds. Ces contraintes font du PKI traditionnelle inadaptée à un environnement VANET.

PKI Auto-organisée

Capkun and Hubeau ont proposé une infrastructure à clé public auto-organisée inspirée du PGP pour authentifier les noeuds d'un réseau mobile ou les certificats numériques sont créés, signés, émis et enregistrés par les noeuds eux-mêmes. Dans cette PKI chaque noeud établi des certificats pour les noeuds en qui il a confiance, et si deux noeuds veulent communiquer sans connaissance préalable l'un à l'autre, ils s'échangent leur liste de certificat afin de créer un certificat entre eux.

Par exemple si un nœud A veut communiquer avec un nœud C, et que le nœud A fait confiance en un troisième noeud B comme le noeud C, alors A peut établir une chaine de confiance à travers B.

- **Gestion de clé Symétrique**

Le but d'échange de clés symétriques est d'établir une clé secrète commune entre les parties communicantes sans avoir aucune information préalable l'une sur l'autre. Parmi les protocoles d'échanges de clés on peut citer celui inventé par Diffie et Hellman.

II.6. Le protocole d'échange de clé Diffie – Hellman : [11]

II.6.1. Introduction :

Depuis la nuit des temps des empires cherchent à cacher des informations à leurs ennemis. Ils ont pour cela développe des méthodes pour encoder et décoder leurs données. Aujourd'hui avec l'ère de l'informatique et la puissance de calcul qui en d´écoule, la cryptologie est une science primordiale pour les services secrets, mais également pour le secteur bancaire et plus généralement les entreprises.

Dans cette section nous expliquerons les fondements théoriques du protocole de Diffie-Hellman. Pour cela nous démontrerons les résultats d'algèbres nécessaires à son fonctionnement.

II.6.2. Histoire : Les créateurs :

- **Whitfield Diffie :**

Bailey Whitfield Diffie, né le 5 juin 1944, est un cryptologue américain. Il est l'un des pionners de la cryptographie asymétrique (utilisation d'une paire de clés publiques et privées) en collaboration avec Martin Hellman et Ralph Merkle.

En 1965, il reçoit un Bachelor en mathématiques au MIT. En 1976 avec l'aide de Martin Hellman, il publie New Directions in Cryptography. La méthode révolutionnaire décrite dans cet article permet de résoudre un problème fondamental en cryptographie : la distribution des clés.

Cette méthode sera par la suite renommée en méthode d'échange de clés Diffie-Hellman et c'est elle que nous allons présenter dans le titre qui suit. Ce principe est aussi à l'origine de méthodes de chiffrage asymétrique plus évoluées comme le RSA ou El Gamal.

Diffie continua ses recherches au sein de Nortel Telecom où il s'occupa de l'architecture du système de sécurité PDSO pour les réseaux X.25. En 1991, il rejoint Sun Microsystems en Californie où il continue à s'occuper des problèmes de sécurité et de cryptographie. En 1992, il est nommé Docteur Honoris Causa par l'Ecole Polytechnique Fédérale de Zurich. Il est aussi membre de la fondation Marconi. En 1998, il a écrit avec Susan Laudau le livre « Privacy on the Line » sur les écoutes téléphoniques et les enjeux politiques liés à la cryptographie.

- **Martin Hellman :**

Martin E. Hellman, n´e le 2 octobre 1945, est aussi un cryptologue américain. Il à lui aussi développe la cryptographie asymétrique (découverte faite en collaboration avec Ralph Merkle et Whitfield Diffie). Hellman est aussi à l'origine d'une attaque avec compromis temps/ mémoire notamment utilisée.
Pour trouver des mots de passe. Cette technique a par la suite été améliorée par Philippe Oechslin.

En 1966, Martin Hellman obtient un bachelor à l'Université de New York, suivi d'un master à l'Université de Stanford en 1967 et un doctorat en 1969. De plus 1968 à 1969, il travaille chez IBM où il rencontre un autre cryptologue très connu, Horst Feistel. De 1969 à 1971, il est professeur assistant au MIT. En 1971, il retourne à Stanford pour poursuivre ses recherches. Il est actuellement à la retraite.

II.6.3. Le protocole de Diffie-Hellman :

II.6.3.1.Principe de l'échange de clé de Diffie-Hellman :

L'échange de clé de Diffie-Hellman a été développé par ces deux auteurs en 1976 et publié dans l'article : W. Diffie and M.E. Hellman, New directions in cryptography, IEEE Transactions on Information Theory 22 (1976), 644-654.

Nous rappelons ici que l'échange d'une clé secrète est fondamental en cryptographie. En effet tout chiffrement d'une grande quantité de données ne peut se faire qu'avec du chiffrement à clé secrète, surtout si cet échange a lieu en temps réel, en raison de la lenteur relative des chiffrements à clé publique.

Il s'agit donc, comme il est exigé par de nombreux protocoles, d'échanger entre deux interlocuteurs A et B une clé secrète K de taille t octets. Pour cela A et B disposent d'un groupe cyclique fini G et d'un générateur a de ce groupe (les éléments de G sont donc, si on

note multiplicativement l'opération du groupe, 1, a, a2, · · ·, as−1 où s est l'ordre de G). Prenons par exemple pour G le groupe multiplicatif (Z/pZ) _ où p est un grand nombre premier et a un élément générateur de ce groupe (mais ça pourrait être aussi un générateur d'un grand sous-groupe de (Z/pZ) _).

Voici comment se passe (de manière schématique) l'échange. Les calculs indiqués sont faits dans le groupe G, donc dans notre exemple modulo p.

- Données publiques : le groupe $G = (\mathbb{Z}/p\mathbb{Z})^*$, un générateur a de ce groupe, un générateur de masque h.
- A tire au sort un entier n tel que $1 < n < p - 1$ et le garde secret.
- A envoie a^n à B (calcul fait dans le groupe, donc ici modulo p).
- B tire au sort un entier m tel que $1 < m < p - 1$ et le garde secret.
- B envoie a^m à A.
- A calcule $s = (a^m)^n$.
- B calcule $s = (a^n)^m$.
- A et B disposent maintenant même s

Rappelons que les calculs sont faits modulo p bien sûr. Le nombre **s** est à peu près de la taille de p et doit certainement être adapté à la taille de la clé commune convoitée. Ceci est fait grâce au générateur de masque :

K = h(s, t) où t est la taille en octets de la clé secrète cherchée K.

II.6.3.2. A quoi sert le protocole :

La cryptologie est une science qui englobe d'une part la cryptographie, c'est `à dire l'écritures secrète des données, et la cryptanalyse qui est l'analyse de cette dernière. En règle générale les méthodes de cryptage font appellent a des clés de cryptage. Une clé de cryptage est un ensemble de valeurs qui permet d'encoder et de décoder des données.

Des algorithmes tels que le RSA utilisent plusieurs nombres premiers qui doivent être tenus secrets. Or la difficulté est de pouvoir communiquer la clé de cryptage d'un émetteur A à un destinataire B sans qu'une tiers personne E ne puisse l'intercepter.

C'est la que le protocole Diffie-Hellman intervient. Il propose à A et B de pouvoir définir une clé secrète même si E écoute leur communication.

II.6.3.3. Pourquoi cette méthode est sécurisée :

Supposons qu'une troisième personne, disons E, écoute les transmissions de A et B. Dans ce cas E n'accès qu'a p, g, g^a mod p et g^b mod p.

Dans ce cas, on peut se demander pourquoi il n'est pas possible a E de calculer a ou b afin d'obtenir la clé secrète. Il peut, en apparence, paraitre simple de calculer a = $\log_g (g^a)$ ou

$b = \log_g (g^b)$. Mais ce n'est pas le cas car on travaille ici en mod p. Ce qui implique de calculer un logarithme discret. Or d'après la littérature, il n'existe pas à ce jour de solution rapide pour le calculer. E est donc dans l'impossibilité de déterminer (g^a mod p) b mod p.

Notons tout de même qu'il faut que p soit suffisamment grand pour éviter que E tente une recherche exhaustive. Actuellement, en utilisant un nombre premier p de l'ordre de 500 à 1024 chiffres et a et b de l'ordre de 100 chiffres, il est impossible de déterminer la clé secrète, même avec les meilleurs algorithmes de résolution de logarithme discret.

Cependant comme de nombreux algorithmes utilisent le protocole de Diffie-Hellman, si une solution pratique pour résoudre un logarithme discret était découverte, elle rendrait ceux-ci inutiles. Sachant que ce protocole est notamment utilise pour les connections s´sécurisées Internet, on laisse au lecteur le soin d'imaginer les problèmes qui en résulteraient.

II.6.4. exemple d'échange Diffie-Hellman entre deux nœuds :

Voici comment se passe l'échange Diffie-Hellman. Les calculs indiqués sont faits dans le groupe cyclique fini qui possède g comme générateur.

1. Le noeud M1 tire au hasard un entier a tel que $1 < a < P - 1$ et le garde secret.

2. Le noeud M1 envoie à M2 **$A=g^a \bmod p$**

3. Le noeud M2 choisit un nombre b tel que $1 < b < P - 1$ et le garde secret.

4. Le noeud M2 envoie à M1 **$B= g^b \bmod p$**

5. Le noeud M1 a reçu B et calcul **$B^a \bmod p$** (c'est-à-dire en passant par **$(g^b)^a \bmod p$** Mais il ne connaît pas B) : **$S = B^a \bmod P$**

6. Le noeud M2 a reçu A et calcul **$A^b \bmod p$** (c'est-à-dire en passant par, **$(g^a)^b \bmod p$**, mais il ne connaît pas A) : **$S= A^b \bmod P$**

M1 et M2 obtiennent à la fin de leurs calculs respectifs le même nombre qui n'a jamais été exposé à la vue des indiscrets : c'est la clé S.
La figure II.2 présente le processus d'échange de clé Diffie Hellman.

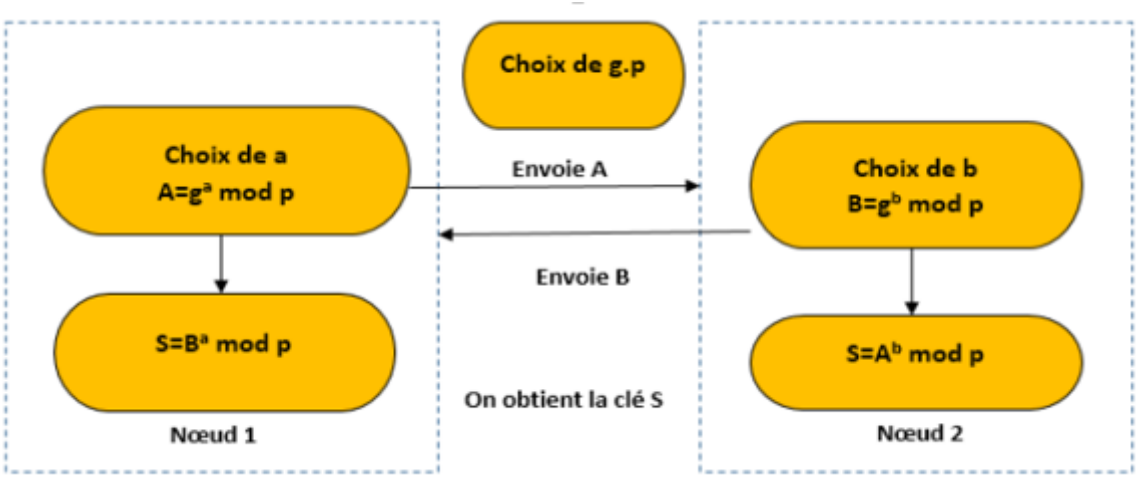


Figure II.6 : Echange de clés Diffie-Hellman

D'après cette étude, on peut constater que la solution symétrique semble être la plus adaptée pour les VANETs pour sa facilité de déploiement et sa rapidité de calcul.

II.7. Conclusion :

Dans un réseau ad VANET tous les nœuds doivent coopérer dans les opérations de routage, en gérant entre autre l'établissement des chemins, la dissémination de notifications de ruptures de chemins et la retransmission des données.

Etant donné cette caractéristique, il devient relativement facile de mener des d'attaques, qui visent à modifier la topologie du réseau, comme la déclaration de faux voisins, l'utilisation de fausses identités (attaque sybil), ou à dégrader ses performances via des attaques comme l'attaque blackhole.

Dans ce chapitre nous avons étudié les notions et les mécanismes de sécurité dans les réseaux ad-hoc en générale ainsi dans les réseauxvanetsprécisément, ensuite nous avons présenté la sécurité de routage dans vanets et les attaques spécifient aux protocoles de routages.

Aussi nous avons étudié Dans ce chapitre les mécanismes de sécurisation de routage par des protocoles sécurisées ou par des mécanismes de gestion des clés.Finalement nous avons bien détaillé le protocole diffie-hellman qui est un mécanisme de gestion de clé symétrique.

Dans le chapitre suivant on va étudier le protocole DSR et leur fonctionnement,Ensuite nous avons présenté une méthode de sécurisation de ce protocole à l'aide de protocole Diffie-Hellman.

III.1. Introduction :

Dans ce chapitre nous allons présenter le protocole DSR et ses mécanismes de fonctionnement, Nous décrivons, en plus les différentes contributions visant à résoudre le problème relatif à la sécurité du protocole de routage DSR dans les réseaux VANETs soulevés dans les chapitres précédents, Donc on va présenter notre solution pour sécuriser le protocole DSR avec le mécanisme d'échange de clé Diffie-Hellman.

III.2. le protocole DSR : [12][13][14]

III.2.1. Définition du protocole DSR :

Le protocole "Routage à Source Dynamique" (DSR : Dynamic Source Routing) est un protocole de routage réactif unicast, à chemin unique, simple, efficace et dédié aux réseaux Ad Hoc mobile multi-sauts.

Ce protocole est basé sur l'utilisation de la technique "routage source". Avec cette technique, la source des données détermine la séquence complète des nœuds à travers lesquelles, les paquets de données seront envoyés.

Afin d'envoyer un paquet de donnée à un autre nœud, l'émetteur construit une route source et l'inclut dans l'entête du paquet. La construction se fait en spécifiant l'adresse de chaque nœud à travers lequel le paquet va passer pour atteindre la destination.

Par la suite, l'émetteur transmet le paquet au premier nœud spécifié dans la route source. Un nœud qui reçoit le paquet, et qui est différent de la destination, supprime son adresse de l'entête du paquet reçu le transmet au nœud suivant identifié dans la route source.

Ce processus se répète jusqu'à ce que le paquet atteigne sa destination finale. Enfin, le paquet est délivré à la couche réseau du dernier hôte.

III.2.2. Le mécanisme de fonctionnement du protocole DSR :

Le protocole DSR doit résoudre deux problèmes :

- la découverte de la route.
- l'entretien (La maintenance) de cette route.

Le premier mécanisme permet de déterminer automatiquement les routes nécessaires à la communication entre nœuds, tandis que le second permet de s'assurer de la correction des routes tout au long de leur utilisation. Nous allons décrire ces deux mécanismes ci-dessous.

III.2.2.1. Mécanisme de découverte des routes :

DSR étant un protocole réactif, un nœud source S va rechercher une route uniquement s'il veut émettre un paquet vers un nœud destinataire D, et qu'il ne possède aucune route vers celui-ci dans son cache. Le nœud A veut trouver la route qui mène au nœud E suivant la figure ci-dessous.

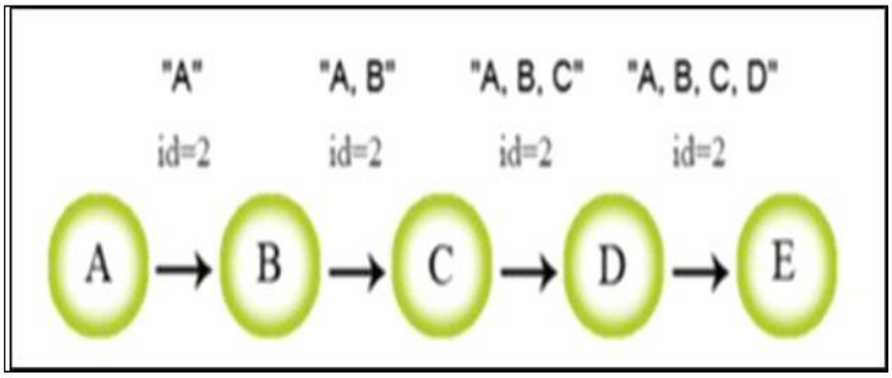


Figure III.1 : La Découverte de chemin dans le DSR

- **Découverte de la route**

La procédure est la diffusion d'une requête de route et l'émission d'une réponse Lorsqu'un nœud source S souhaite envoyer des données à un nœud destination D et n'a pas de route vers ce nœud D.

Le nœud S envoie un paquet Route Request (requête de recherche de route) à destination du nœud D, ce paquet se propage dans le réseau (voir figure III.2), cette propagation se termine lorsque le nœud D ou un nœud possédant un chemin vers celui-ci dans son cache est atteint.

Le paquet contient l'adresse source, l'adresse de destination, un numéro d'identification et un champ Route Record dans lequel sera enregistrée la séquence des nœuds visités durant l'inondation du paquet Route Request dans le réseau.

Quand un nœud reçoit un paquet Route Request, il vérifie s'il connait un chemin vers la destination, si ce n'est pas le cas, il ajoute son adresse dans le champ Route Record du paquet Route request et transmet ce paquet à ses voisins.

Pour éviter les boucles et les multiplications des paquets Route Request, ce transfert ne se fait que si l'adresse du nœud n'apparait pas déjà dans le champ Route Record.

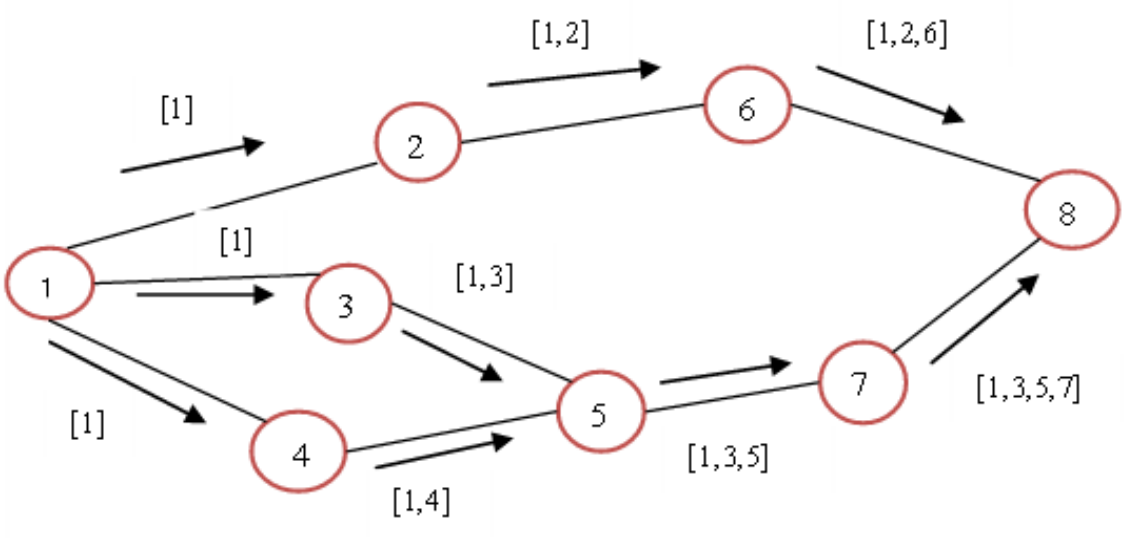


Figure III.2 : Construction de l'enregistrement de route dans DSR

Pour conclure, le protocole DSR initie une Route Discovery en émettant un paquet en diffusion (broadcast) d'en-tête Route Request (RReq), qui va inonder le réseau. Ce paquet va donc être reçu par tous les nœuds intermédiaires du réseau.

La Route Request contient :

- la source S et la destination D de la Route Discovery.
- un numéro unique de la requête (l'ID, 2 dans l'exemple de la figure II.1).
- un enregistrement qui liste les adresses de chaque nœud intermédiaire à travers lesquels la copie de cette Route Request a été transmise.

Le format de paquets RREQ :

Le paquet RREQ de demande de route ou bien route discovery qui est diffusé en broadcast par le protocole de routage DSR, est le message d'interrogation des routes disponibles. Il est constitué d'une trame de 24 octets :

- Les quatre premiers octets sont constitués du champ Type sur 8 bits forcé à 1 indiquant qu'il s'agit d'un message RREQ. Les bits suivants : J, R, G, D, U décrits plus bas indiquent les différentes utilisations du message. Un champ *reserved* sur 11 bits mis à 0 laisse la possibilité d'évolution ultérieure. Puis un champ de 8 bits indique le nombre de sauts.

- Les quatre octets suivants portent l'identification du message.
- Les quatre suivants l'adresse IP de la destination.
- Les quatre suivants le numéro de séquence de la destination.
- Les quatre suivants l'adresse IP d'origine.
- Les quatre derniers sont destinés à l'identification de la transaction par la source IP.

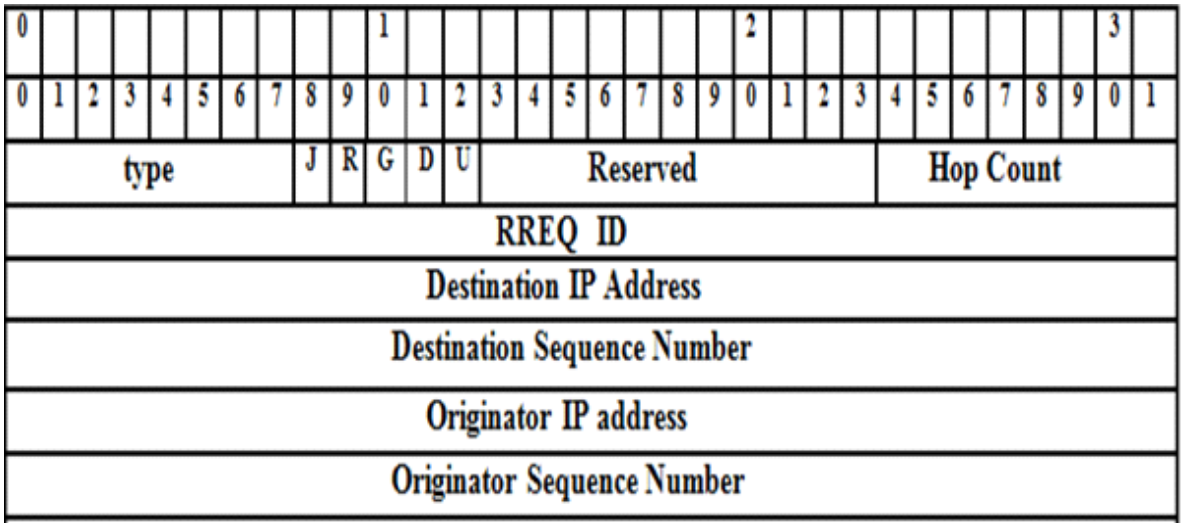


Figure III.3 : Format de paquet RREQ

- Le champ Type est forcé à 1.
- Le bit J : est positionné à 1 sur le message sortant en cas d'utilisation de multicast.
- Le bit R : est positionné à 1 sur le message retour en cas d'utilisation de multicast.
- Le bit G : Drapeau RREP gratuite, indique si un RREP gratuite doit être envoyé en unicast vers le nœud spécifiée dans le champ Adresse IP destination.
- Le bit D : Drapeau de destination seulement, indique que la destination seule peut répondre à cette RREQ.
- Le bit U : Numéro de séquence inconnue, indique à la destination que le numéro de séquence est inconnu.
- Reserved : Envoyés à 0, ignoré à la réception.
- Hop count : Le nombre de sauts à partir de l'adresse IP origine jusqu'au nœud de traitement de la demande.

- RREQ ID : Un numéro de séquence unique permet d'identifier les RREQ notamment lorsqu'elle est prise en association avec l'adresse IP d'origine.
- Destination IP Address : L'adresse IP destinataire pour laquelle une route est souhaitée.
- Destination Sequence Number : Le dernier numéro de séquence reçu par l'émetteur pour toute route vers le destinataire.
- Originator IP Address : L'adresse IP du nœud à l'origine de la demande de route.
- Originator Sequence Number : le numéro de séquence actuelle utilisée dans l'itinéraire pointant vers l'initiateur de la route demandée.

- **Le renvoi du chemin :**

Lorsque le paquet Route request atteint la destination (ou bien une station intermédiaire ayant une route vers la destination), la destination (ou le nœud intermédiaire) envoie un paquet de réponse Route reply (voir figure III.4) via le chemin donné dans le Route Record si les liaisons sont bidirectionnelles (symétriques) ou via un autre chemin (utilisant éventuellement une découverte de la route).

Afin de diminuer le coût de la recherche de route, chaque nœud peut garder en mémoire les routes qu'il a apprises dans le cache de route.

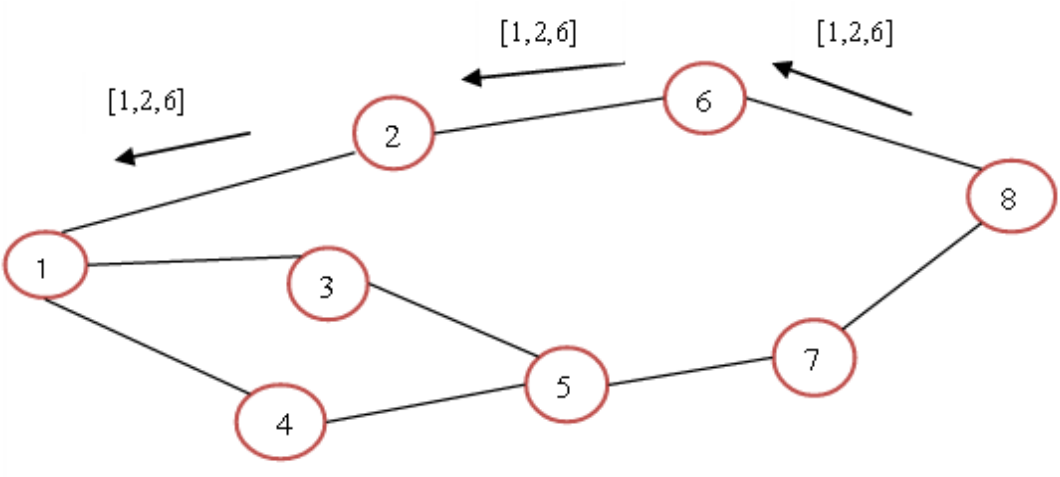


Figure III.4 : L'envoi du chemin ou de la route replay (RREP)

Le format de paquets RREP :

Quand le destinataire D reçoit cette Route Request, il retourne à la source S un message RREP de réponse à la demande de route, indiquant au demandeur les routes disponibles. Il est constitué d'une trame de 24 octets :

- Les quatre premiers octets sont constitués du champ Type sur 8 bits forcé à 2 indiquant qu'il s'agit d'un message RREP. les deux bits R et A décrits plus bas indiquent pour le premier qu'une route défectueuse est réparée, le deuxième pour sa part indique que ce message sera suivi d'un RREP-ACK. Puis 12 bits réservés pour évolution mis à 0. 5 bits de préfixe référençant le nœud pour les routages. Puis un champ de 8 bits indique le nombre de saut.
- Les quatre octets suivants portent l'identification du message.
- Les quatre suivants l'adresse IP de la destination.
- Les quatre suivants le numéro de séquence de la destination.
- Les quatre suivants l'adresse IP d'origine.
- Les quatre derniers sont destinés pour le temps en millisecondes pendant lequel les nœuds recevant RREP.

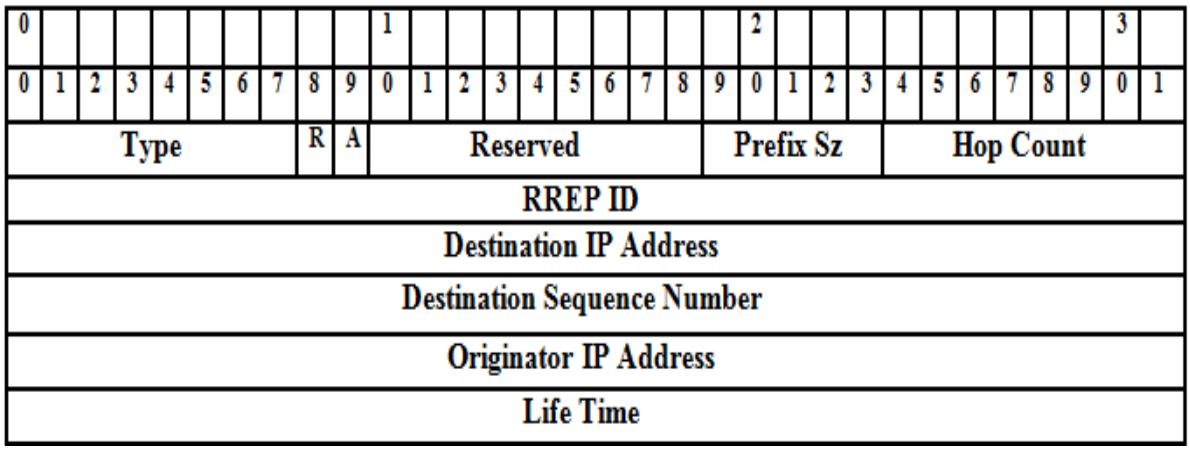


Figure III.5 : Format de paquet RREP

- Le champ Type : forcé à 2.
- Le bit R : Indicateur de réparation ; utilisée pour le multicast.

- Le bit A : acquittement requis.
- Reserved : Envoyé à 0, ignoré à la réception.
- Prefix Size : Si non nul, la taille de préfixe à 5 bits indique que le saut suivant peut être utilisé pour tous les nœuds avec le même préfixe de *routage (tel que défini par le préfixe Taille) pour la destination demandée.
- Hop Count : Le nombre de sauts à partir de l'adresse IP origine à l'adresse IP de destination. Pour une route de multicast demandée, indique le nombre de sauts à effectuer à chaque membre de l'arbre en envoyant l'RREP.
- Destination IP : L'adresse IP de la destination pour laquelle une route est fournie.
- Destination Sequence Number : Le numéro de séquence de la destination associée à l'itinéraire.
- Originator IP Address : L'adresse IP du nœud à l'origine du RREQ pour laquelle l'itinéraire est fourni.
- Lifetime : Le temps en millisecondes pendant lequel les nœuds recevant RREP peuvent considérer la route comme étant valide.

- **La notion de cache :**

Dans le réseau, les nœuds peuvent enregistrer dans leur cache des informations de routage obtenues au travers des différents paquets Route Discovery reçus et des paquets de données.

De plus, si un nœud intermédiaire qui reçoit un message Route Request possède en cache une route vers la destination D, alors il envoie un Route Reply à S en ajoutant la route connue.

Si un nœud recevant un message Route Request a récemment vu un autre message Route Request contenant le même ID et la même adresse de destination, ou si la propre adresse du nœud est déjà listée dans la Route Request, alors le nœud supprime la requête.

- **L'envoi du message :**

Finalement, le nœud source obtient plusieurs routes pour atteindre le destinataire. Une fois ces routes connues, le nœud va pouvoir envoyer des paquets d'option Source Route (SrcR) contenant les données à échanger.

III.2.2.2. Mécanisme de maintenance de route :

- **Le message RERR :**

Le message RERR indiquant une route en erreur est le message retour signalant au demandeur les routes en erreur. Il est constitué d'une trame de 20 octets :

- Les quatre premiers octets sont constitués du champ Type sur 8 bits forcé à 3 indiquant qu'il s'agit d'un message RERR. le bit N décrit plus bas indique qu'une route défectueuse est réparée.
- Puis 15 bits réservés pour évolution mis à 0. Les 8 bits suivants indiquent le numéro de destination inaccessible.
- Puis sur quatre octets l'adresse IP de destination inaccessible.
- Les huit suivants indiquent si besoin le complément d'adresse.

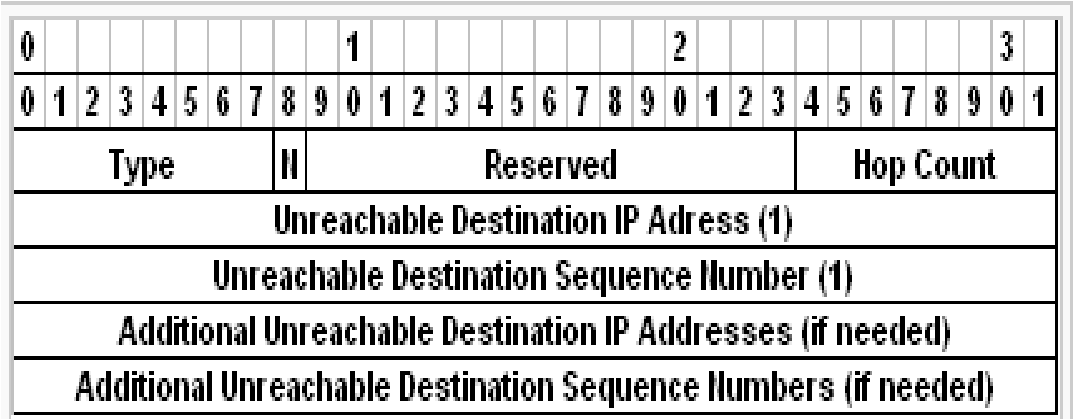


Figure III.6 : Message RERR

- Le champ Type : forcé à 3.
- le bit N : Indicateur de non-suppression. Quand un nœud a effectué une réparation locale d'un lien. Les nœuds amont ne doivent pas supprimer la route.
- Reserved : Envoyé à 0, ignoré à la réception.
- DestCount : Le nombre de destinations inaccessibles inclus dans le message, doit être d'au moins de un.
- Unreachable Destination IP Address : L'adresse IP destinatrice qui est devenue inaccessible en raison d'une coupure de liaison.

- Unreachable Destination Sequence Number : Le numéro de séquence dans l'entrée de la table de routage pour la destination indiquée dans le champ précédent Destination IP inaccessible.

- **Maintenance de route :**

Lorsqu'un nœud transmet un paquet, il est responsable de confirmer sa bonne réception par son prochain saut vers la destination tout au long de la route source.

Par exemple, dans la figure III.7 le nœud (A) a transmis un paquet au nœud (F) en utilisant les nœuds (B), (C), (D) et (E) comme des nœuds intermédiaires. Dans ce cas le nœud (A) est responsable de la réception du paquet par (B) qui est lui-même responsable de sa réception par (C). La mobilité des nœuds dans les réseaux Ad Hoc nécessite de vérifier, après l'envoi d'une donnée, que la topologie est toujours la même et que la source peut utiliser une source pour atteindre la destination en utilisant une procédure de maintenance de route.

Quand un nœud détecte une erreur de transmission, un paquet route error (erreur de route) contenant l'adresse du nœud qui a détecté l'erreur et celle du nœud qui le suit dans le chemin, est envoyé à l'émetteur original du paquet. Lors de la réception de ce paquet par la source, le nœud concerné par l'erreur est supprimé du chemin sauvegardé, et tous les chemins qui contiennent ce nœud sont tronqués à ce point-là. Par la suite une nouvelle opération de découverte de routes vers la destination est initiée par l'émetteur.

Reprenons la figure III.7 le nœud (D) est incapable d'envoyer le paquet au prochain saut (E) (à cause de la coupure des liens entre D et E), alors il retourne un paquet erreur de route (Route Error) à (A) en signalant une coupure de lien entre lui et (E). Quand le nœud source (A) et les nœuds intermédiaires (B) et (C) reçoivent ce paquet, ils suppriment la route de leur cache. Si (A) à une autre route vers (E) dans son cache, il peut l'utiliser pour envoyer le paquet immédiatement, sinon il initie une nouvelle requête de route.

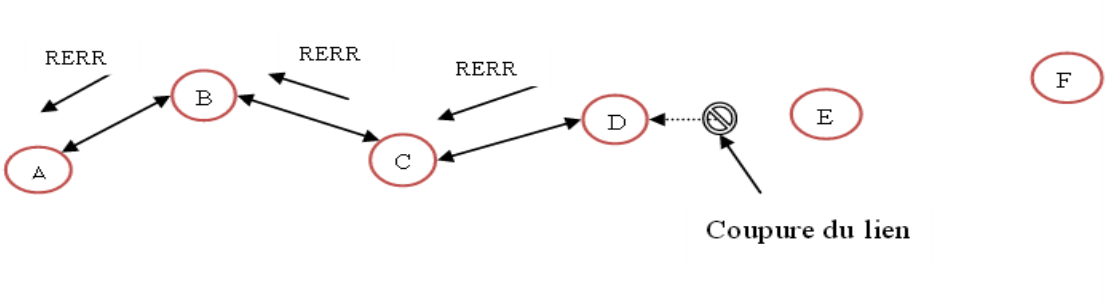


Figure III.7 : Erreur dans DSR (envoi de Route error) RRER.

Fonctionnement de mécanisme de maintenance des routes :

Dans un réseau Ad Hoc, les nœuds étant mobiles, il faut vérifier, après l'envoi d'une donnée, que la topologie est toujours la même et que la source S peut utiliser une route pour atteindre la destination D. Pour ce faire, DSR utilise le mécanisme de Route Maintenance qui est une succession de trois procédures conditionnelles.

- **Accusés de réception :**

Dans cette situation le nœud A généré un paquet pour F en utilisant une route passant par B, C et D. Dans ce cas, A est responsable du lien entre A et B, B est responsable du lien entre B et C... .

Un accusé de réception permet de confirmer que le lien est fonctionnel. Il peut souvent être généré sans coût en utilisant des standards existant de la couche MAC : IEEE 802.11 ou acquittement passif (DSR va écouter tous les paquets dans sa portée radio. Chaque paquet est examiné pour savoir si le paquet est bien retransmis par le nœud suivant.).

Si ces standards ne sont pas pris en charge par l'adaptateur sans fil, l'expéditeur du paquet peut explicitement demander un acquittement ou Acknowledgment Request auquel le nœud suivant devra répondre par un paquet d'acquittement. Si le lien entre ces deux nœuds est unidirectionnel, l'acquittement peut emprunter une route différente. Après réception d'un accusé, le nœud peut choisir de ne pas en demander de nouveau pendant un temps bref pour tous les messages à destination du nœud suivant.

- **En cas d'échec... :**

- En cas d'échec du Route Maintenance, le nœud va envoyer des demandes d'acquittements de type Acknowledgment Request un nombre prédéfini de fois (dans des messages de data, dans des paquets vides ou par retransmission).

- **Recherche d'une solution :**

Si aucun accusé n'est reçu, le nœud détectant la rupture du lien mettra à jour son cache de route et enverra un paquet de type Route Error en direction de la source. Celle-ci pourra choisir une nouvelle route ou recommencer une procédure de Route Discovery.

III.2.3. Structures de données conceptuelles associées aux nœuds dans DSR :

Dans un réseau VANET, chaque nœud participant au routage DSR a besoin de quatre structures de données conceptuelles

· Un cache de routes.

· Une table de requête de route.

· Un tampon d'émission.

· Un tampon de retransmission.

- **Le cache de routes :**

Chaque nœud dans le réseau maintient un cache de routes où toute information de routage sera sauvegardée. Quand un nœud apprend de nouvelles routes à travers des paquets de réponse de route ou les en-têtes de routage des paquets, il les ajoute à son cache. De la même façon il peut les supprimer quand il apprend qu'elles ne sont plus valides.

Par exemple à travers des paquets d'erreur de route (RERR) qui annoncent une coupure de liens entre les nœuds. Le cache de routes doit supporter de sauvegarder plus d'une route source pour chaque destination c'est pour cela que le DSR est appelé protocole à chemins multiples (multi-path).

Le cache de routes supporte les opérations suivantes :

- **Void Insert (Route RT)** : Insère l'information extraite de la route RT dans la cachette de route.

- **Route Get (Node DEST)** : Retourne une route source de ce nœud vers cette destination.

- **Void Delete (Node FROM, Interface INDEX, Node TO)** :

Enlève de la cachette de route toutes les routes qui supposent qu'un paquet transmis par un nœud From sur son interface avec l'INDEX donné et qui sera reçu par le nœud TO.

S'il y a des cachettes de route multiple à une destination, l'opération Get () d'une route devrait préférer les routes qui mènent directement au nœud de la cible sur des routes qui entreprennent à atteindre la cible par toute infrastructure Internet connecté à un réseau VANET.

La politique de la mise en place de la cachette de route devrait permettre aux routes d'être classé par catégories basé sur la préférence, où les routes avec une plus haute préférence est moins possible d'être enlevées de la cachette. Par exemple, un nœud pourrait préférer des routes pour qu'il commence une découverte de route que celles qu'il a apprises comme le résultat d'une recherche illégale sur les autres paquets. Particulièrement, un nœud devrait préférer des routes qui son utilisé actuellement sur celles qui ne le sont pas.

- **La table requête de route « Route Request » :**

La table de requête de route est une collection de registres contenant des informations concernant les paquets de requête de route qui sont récemment diffusés par un nœud (S) afin de trouver une route vers une destination (D) quelconque .

Un registre contient :

- Le temps où (S) a commencé une découverte de route vers (D).
- L'intervalle de temps durant lequel (S) doit attendre avant d'initier une nouvelle découverte de route vers (D).
- TTL est le champ dans l'en-tête IP de la dernière requête de route initiée par (S).
- Un identificateur unique de cette requête de route RREQ-ID.

- **Le tampon de transmission « Send Buffer » :**

Le tampon de transmission associé aux paquets qui ne peuvent pas être transmis par un nœud à cause de l'invalidité de la route vers les destinations. Chaque paquet dans le tampon de transmission contient son temps de placement dans le tampon, et devrait être enlevé du tampon de transmission après l'expiration du compteur « SEND-BUFFER-TIMEOUT ». Une stratégie FIFO est utilisée pour acheminer les paquets qui permettre d'éviter la saturation du tampon.

Une découverte de route devrait être commencée aussi tôt que possible vers la destination de tous les paquets qui résident dans le tampon de l'émetteur.

- **Le tampon de retransmission « Retransmission Buffer » :**

Le tampon de retransmission d'un nœud associé aux paquets envoyés par ce nœud qui attendent la réception d'acquittements.

Un nœud maintient pour chaque paquet dans son tampon de retransmission :

- Un compte qui contient Le nombre de retransmissions du paquet.
- Le temps de la dernière retransmission.

Les paquets sont enlevés du tampon lorsque le nœud reçoit des acquittements (ACK), ou quand le nombre de retransmissions dépasse DSR-MAXRXTSHIFT (le nombre de tentatives de retransmission des paquets atteint un maximum).

III.2.4. L'en-tête de routage DSR "DSR Routing Header" :

Un en-tête de routage est utilisé pour lister les nœuds intermédiaires qui seront visités tout au long du chemin emprunté par un paquet pour atteindre une destination. L'en-tête de routage possède le format suivant :

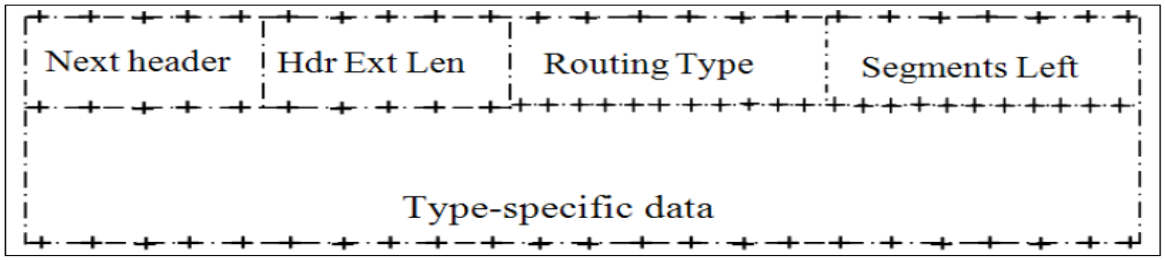


Figure III.8 : Format de l'en-tête de routage.

- **<u>L'en-tête de routage est constitué des champs suivants :</u>**

-Next Header : Sur 8 bits, il identifie le type d'en-tête qui suit immédiatement cet en- tête de routage.

-Hdr Ext Len : Sur 8 bits, il indique la taille de l'en-tête de routage sans compter les premiers 8 octets.

-Segments Left : C'est le nombre de nœuds intermédiaires restant avant d'atteindre la destination finale.

-Le champ « type-specific data » :

Le champ « type-specific data » est structuré sous la forme suivante :

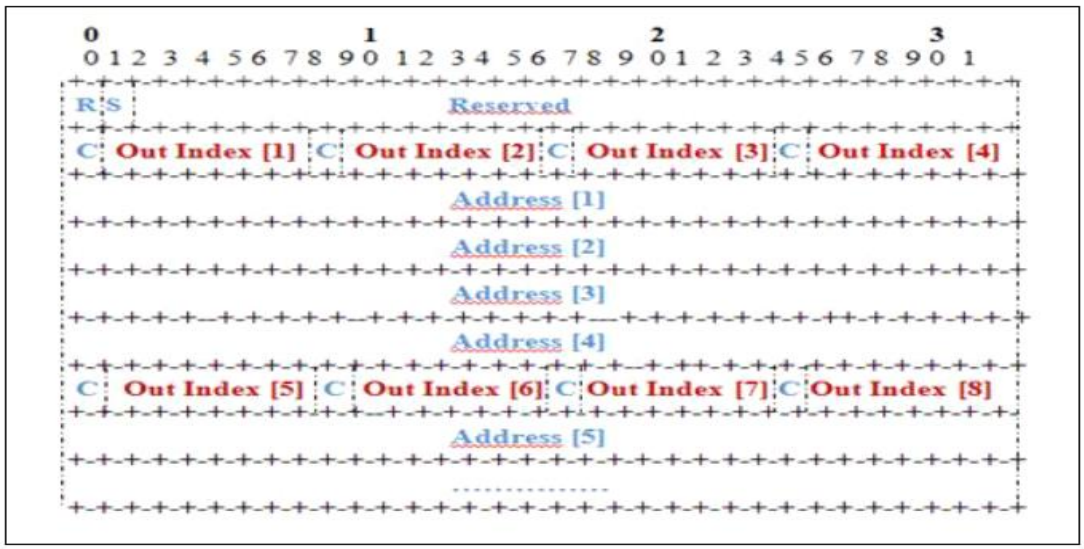


Figure III.9 : Format du champ type-specific data.

- **Acknowledgment Request (R)** : L'indicateur R est activé pour demander un acquittement explicite de la part du prochain saut vers lequel le paquet sera transmis.
- **Le Paquet sauvé Salvaged Packet (S)** : Indique que ce paquet a étés sauvé par un nœud intermédiaire.
- **Reserved** : Envoyé comme 0, ignoré en réception.
- **Change Interface (C) bit [1...n]** : Si l'indicateur C est actif cela implique que le nœud correspondant va transmettre le paquet sur une interface différente de celle sur laquelle il a reçu ce paquet.
- **OUT Index [1...n]** : C'est l'index de l'interface que le nœud d'adresse [i-1] doit utiliser en transmettant le paquet au nœud d'adresse [i].
- **Adresse [1...n]** : c'est l'adresse du Ime saut listé dans l'en-tête de routage.

III.2.5. Avantages et inconvénients du protocole DSR :

Comme les informations de routage sont contenues dans les entêtes des paquets, il n'est pas nécessaire de maintenir les informations de routage par des envois périodiques de paquets de contrôle au niveau chaque nœud intermédiaire. En effet, les routes ne sont maintenues qu'entre les nœuds qui communiquent ce qui réduit les coûts engendrent par les messages de contrôle. L'utilisation du cache au niveau des nœuds intermédiaires permet d'une part d'accélérer la découverte d'une route et d'autre part de réduire le nombre de RREQ.

L'inconvénient du protocole est que l'en-tête des paquets augmente en fonction de la taille de la route ce qui pose des problèmes de passage à l'´échelle. Le processus de découverte par inondation peut être coûteux car il peut atteindre tous les nœuds du réseau, Des collisions peuvent avoir lieu lors de la transmission d'un paquet par deux voisins simultanément.

La cohérence des caches est difficile à maintenir (peuvent être rendues invalides) à cause des mouvements fréquents des nœuds. Des risques d'engorgement sont possibles en cas de trop nombreuses réponses obtenues à partir des caches (ceci peut être évité si les nœuds écoutent en mode promiscuités les transmissions des RREP et s'ils transmettent une RREP uniquement lorsqu'ils disposent d'une route plus courte). Un nœud peut transmettre une route erronée et ainsi polluer les caches de ses voisins (ou cache poisonning).

III.3. Sécurisation du protocole DSR :

III.3.1. Introduction :

Le protocole DSR ne spécifie aucune mesure de sécurité. Il est pourtant, la principale cible pour des attaques qui visent l'acheminement des alertes de sécurité routière dans réseau VANET.

Avant de présenter notre solution de sécurité, nous présentons ici un résumé des failles et des besoins de sécurité du protocole DSR.

Puisque DSR ne dispose d'aucun mécanisme de sécurité, des nœuds malicieux peuvent mener des attaques par des moyens tels que :

- un nœud malveillant peut bloquer tous les paquets d'une victime, en réalisant ainsi un blackHole, ou juste sélectionner les paquets à bloquer en réalisant un Greyhole.
- Un attaquant peut générer de faux messages d'alerte pour les diffuser.
- Un attaquant peut facilement modifier le contenu des paquets et les rediffuser

Suite à ces attaques les besoins en sécurité dans DSR sont énumérés comme suit :

-Vérifier que l'émetteur des paquets de routage est bien celui qu'il prétend être.

-Vérifier que les paquets de routage arrivés, n'avaient pas été altérés durant le transfert.

Nous allons présenter par la suite notre contribution de sécurisation qui consiste a implémenté le protocole d'échange de clé Diffie-Hellman dans les paquets RREQ et RREP afin de sécurisé le protocole DSR.

III.3.2. DSR et la sécurité :

Les recherches récentes sur les réseaux vanet ne se focalisent que très peu sur les aspects sécurité. Pourtant leurs spécificités montrent à quel point les réseaux vanet sont 99vulnérables. Parmi ces vulnérabilités figurent :

- la transmission en milieu ouvert ;
- les problématiques de topologies dynamiques ;
- l'absence d'autorité centrale ;
- la nécessité d'une bonne coopération des nœuds ;
- l'hétérogénéité des participants avec pour certains des capacités restreintes.

Pour donner un exemple de vulnérabilité sur une transmission en milieu ouvert (sans fil), on peut mettre en avant l'exposition des nœuds à des problèmes d'intégrité physique. Une surveillance sismique par exemple, nécessite de lâcher des capteurs dans la nature. Ils deviennent alors physiquement accessibles. Un moyen de contourner ce problème est de mettre en évidence une attaque physique sur un élément. Autre exemple concret, le fait que les nœuds utilisent une transmission sans fil les rend également très sensibles à une Attaque par déni de service sur le canal radio.

Les autres vulnérabilités précédemment citées nous amènent à faire un focus sur le routage des réseaux vanets. Il est identifiée comme particulièrement sensible. Son fonctionnement nécessite entre autres, la bonne coopération de tous les nœuds, ce qui présente un risque s'il n'y a aucun contrôle des participants. Par conséquent l'authentification, l'intégrité, la confidentialité et la disponibilité doivent faire l'objet d'une attention particulière.

Parmi les attaques liées aux problèmes d'authentification on peut citer le trou noir (blackhole), Cette attaque consiste à insérer un nœud malicieux ayant la capacité d'usurper l'identité d'un nœud valide. Le nœud en question pourra ainsi ignorer les données qu'il est

censé faire transiter. L'attaque «grey hole», qui en est une variante, pourra ignorer seulement certain type de paquets. La figure ci-dessous décrit une attaque de type blackhole.

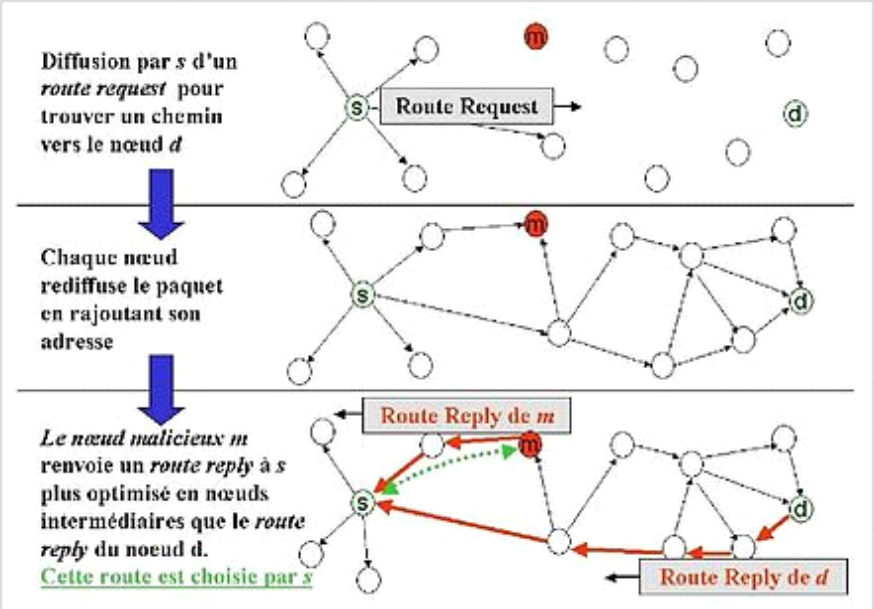


Figure III.10 : Description d'une attaque BlackHole issue de l'étude de Gayrault

III.3.3. Implémentation de protocole d'échange de clé Diffie-Hellman dans le protocole DSR :

Notre première contribution dans ce travail consiste à proposer une approche qui permet d'intégrer le protocole de clés secrètes Diffie-Hellman aux nœuds pour renforcer la sécurité afin de définir une clé secrète entre deux nœuds source et destination.

Par la suite on va ajouter un champ dans l'entête des paquets RREP et RREQ qui contient les paramètres de protocole Diffie-Hellman.

III.3.3.1. Intégration de protocole d'échange de clé Diffie-Hellman aux nœuds mobile :

Le protocole de routage DSR et le protocole de sécurité Diffie-Hellman peut être installé sur différents équipements comme les micro-capteurs de Réseau de capteurs sans fil mais aussi sur des PDA (Personal Digital Assistant) , des ordinateurs portables...,aussi dans les véhicules.

Gestion de clé symétrique $\Longleftrightarrow$ clé secrète commune

Le protocole d'échange de clés de Diffie-Hellman, repose sur une fonction de la forme :

K=g*mod P avec ***P*** premier et **g < *P*.**

Une telle fonction est très facile à calculer, mais la connaissance de *K* ne permet pas d'en déduire facilement *X*. Cette fonction est publique, ainsi que les valeurs de *g* et *P*.

Voici comment se passe l'échange Diffie-Hellman. Les calculs indiqués sont faits dans le groupe cyclique fini qui possède g comme générateur.

- **Si un nœud émetteur :**
    - Le nœud tire au hasard un entier **a** tel que **1 < a < P − 1** et le garde secret.
    - Le nœud calcul A = $\mathbf{g^a \bmod p}$.
    - ce nœud envoie les valeurs A, g, p.
    - ensuite ce nœud attend la valeur B calculé par la destination.
    - Le nœud a reçu B et calcul $\mathbf{B^a \bmod p}$ (c'est-à-dire en passant par $\mathbf{(g^b)^a \bmod p}$ Mais il ne connaît pas B) : $\mathbf{S = B^a \bmod P}$
- **Si un nœud destinataire :**
    - ce nœud reçoit les valeurs A, g, p de la part de nœud émetteur.
    - calcul la valeur $\mathbf{B= g^b \bmod p,}$ et l'envoyer au nœud émetteur.
    - Le nœud a reçu A et calcul $\mathbf{A^b \bmod p}$ (c'est-à-dire en passant par, $\mathbf{(g^a)^b \bmod p}$, mais il ne connaît pas A) : $\mathbf{S= A^b \bmod P}$.

Les deux nœuds obtiennent à la fin de leurs calculs respectifs le même nombre qui n'a jamais été exposé à la vue des indiscrets : c'est la clé **S.**

III.3.3.2. Intégration des paramètres de Diffie-Hellman au protocole DSR "DH-DSR" [15] :

Nous avons vu que , à cause de l'aspect décentralisé des réseaux ad hoc où chaque nœud a besoin de la collaboration des autres nœuds, la sécurisation d'un protocole de routage ad hoc est un problème délicat et difficile à résoudre; pour cette raison nous avons proposé d'ajouter d'autres solutions visant à améliorer la sécurité.

- **intégration des paramètres de Diffie-Hellman au paquet RREQ**

Pour sécuriser la requête RREQ, on insère a l'entête du cette paquets un champ qui contient les clés de chiffrements calculée par l'algorithme d'échange de clé Diffie-Hellman, (qui est implémenter dans chaque nœud reçoit cette requête).

Ce champ appelée "DH" prend un espace de 3 octets, et contient :

- 8 bits pour Le nombre A = $\mathbf{g^a \bmod p}$ calculé par la source.
- 8 bit pour le nombre premier p.
- 8 bit pour le nombre g tel que g < p.

Donc on a ajouté 3 octets en plus à la requête RREQ, et devient une trame qui constitué de 27 octets.

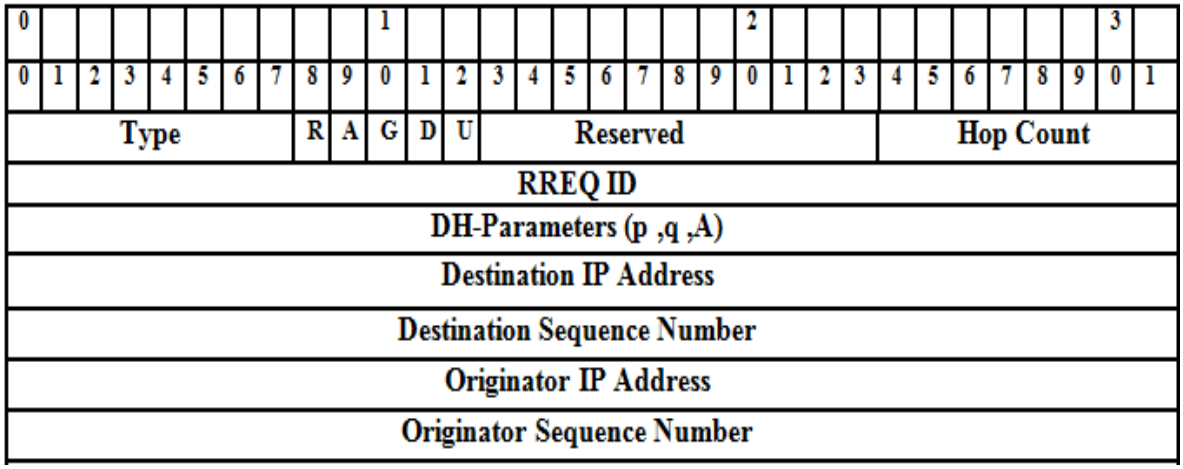


Figure III.11 : paquet RREQ après sécurisation

- **intégration des paramètres de Diffie-Hellman au paquet RREP**

chaque fois le paquet RREQ sécurisé arrive au nœud destiné ,ce dernier va prend les nombres A , p , g envoyer par le nœud source et calculé B , ensuite la destination doit modifier le champ de DH (maintenant il contient le nombre B seulement sur 1 octet) dans le paquet RREP .

Donc ce message RREP sera constitué d'une trame de 25 octet.

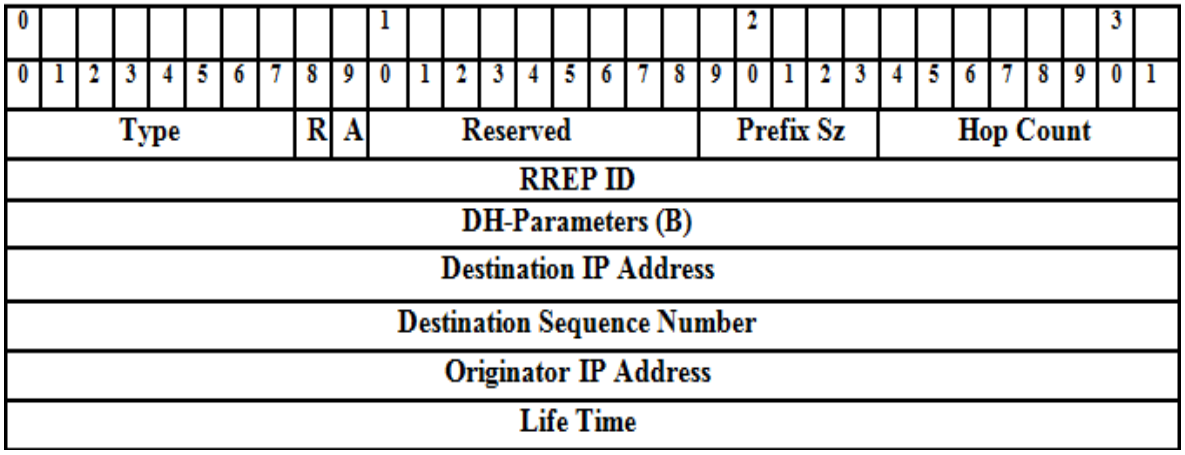


Figure III.12 : Paquet RREP après sécurisation

- **Mise à jour des paramètres de DH**

Après l'insertion des paramètres de Diffie-Hellman aux requêtes RREQ et RREQ, la source va prendre le nombre B et calculé la clé secrète $K=B^a \bmod p$, ensuite la destination calcule **$K=A^b \bmod p$** avec le nombre A.

Cette clé secrète commun qui chiffre les données à la source et les déchiffres à la destination doit être résiste un certain temps aux attaques visé a obtenue celui-ci afin de déchiffre les données.

Supposons qu'une troisième personne, disons ATQ, écoute les transmissions de la source et la destination. Dans ce cas ATQ n'accès qu'a p, g, $A=g^a \bmod p$ et $B=g^b \bmod p$.
Dans ce cas, on peut se demander pourquoi il n'est pas possible à ATQ de calculer a ou b afin d'obtenir la clé secrète.
Il peut, en apparence, paraitre simple de calculer $a = \log_g (g^a)$ ou $b = \log_g (g^b)$, Mais ce n'est pas le cas car on travaille ici en mod p. Ce qui implique de calculer un logarithme discret.
Or d'après la littérature et plusieurs documents, il n'existe pas à ce jour de solution rapide pour le calculer :

- DH-512 : des paramètres de tailles maximum de 512 bits, il faut 2 jours avec les meilleurs algorithmes de résolution de logarithme discret (ça veut dire pour obtenue $a = \log_g (g^a)$ ou $b = \log_g (g^b)$ qui permet de calculer $K=A^b \bmod p$).

- DH-24 : c'est notre proposition, qui consiste à minimiser la taille des paramètres afin de réduire la taille des requêtes RREQ et RREP pour que l'augmentation de la taille

des que l'augmentation de la taille des paquets n'influence pas d'une manière significatif sur le fonctionnement du protocole (augmentation de délai) paquets n'influence pas d'une manière significatif sur le fonctionnement du protocole (augmentation de délai). cette proposition est faible face aux attaquants car on a minimisant la taille des paramètres, donc l'attaquant a besoin peu de temps environ 45 seconde pour calculé l'algorithme discret et obtenir la clé K.

Pour sécuriser notre proposition, il faut faire une mise à jour des paramètres de DH, avec le mécanisme de maintenance de route.

pour ce faire il faut à chaque 40s après l'échange des données , le nœud source génère a nouveau les paramètres DH et envoyé les nouveaux paramètres A ,p ,g vers la destination utilisant la même route sauvegarder dans la table de routage avec le message MAJ , aussi la destination reçoit ces paramètres et exécuté à nouveau l'algorithme DH et envoyé le nouveau nombre B vers la source avec le message MAJ-REP, donc les deux extrémités obtient un nouveau clé secret commun utilisé pour chiffré et déchiffré les données .

Ce mécanisme de mise à jour se fait sans perturber l'envoie de données et l'ancienne clé commune sera remplacée par le nouveau automatiquement.

Le format de message MAJ est présenté dans la figure suivante :

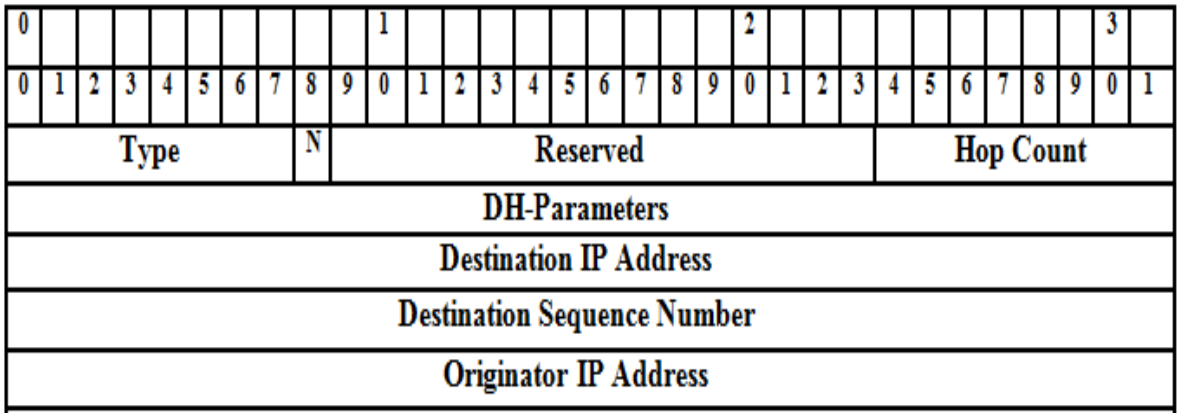


Figure III.13 : Format de message MAJ

Le message MAJ indiquant une mise à jour des paramètres de protocoles de Diffie-Hellman, pour sécuriser les donnes échanger sur le canal non protéger et interdire à un attaquant de calculé l'algorithme discret et obtenir la clé K.

Ce message a été envoyé chaque 40s après l'échange des données par les deux extrémités. Il est constitué d'une trame de 20 octets :

- Les quatre premiers octets sont constitués du champ Type sur 8 bits forcé à 3 indiquant qu'il s'agit d'un message MAJ.

  Le bit N indique qu'une nouvelle mise à jour des paramètres sera effectuée.

  Puis 15 bits réservés pour évolution mis à 0. Les 8 bits suivants indiquent le numéro de destination.

- Puis sur quatre octets l'adresse IP de destination.
- Les quatre suivants indiquent Le numéro de séquence dans l'entrée de la table de routage pour la destination indiquée dans le champ précédent Destination IP.
- Enfin quatre octets pour l'adresse IP source.

Le format de message MAJ-REP est présenté dans la figure suivante :

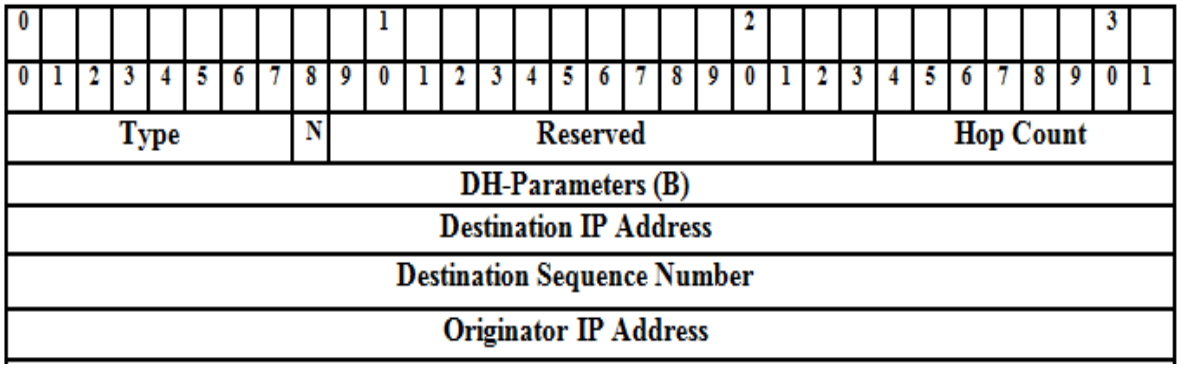


Figure III.14 : format de message MAJ-REP

- **Implémentation du l'algorithme Diffie-Hellman en java : [16] [17]**

Nous allons vous présenter un algorithme d'échanger de clé DH entre deux nœuds utilisant java.

Cet algorithme permet d'échanger des clés entre deux véhicules.

Considérons deux individus V1 et V2, qui souhaitent s'envoyer un message.

Pour le noeud V1 :

- Tout d'abord V1 choisis un nombre premier aléatoires p d'une valeur maximal de 1 octet depuis d'un tableau des nombres premiers.

LISTE {2, 3,5...251}

p=aléatoire(LISTE).

Afficher (p).

- Puis il choisit un nombre g tel que g<p.

g=aléatoire(LISTE)

Si g>=p faire g=aléatoire(LISTE)

Sinon

Afficher (g).

- V1 choisis un nombre secret $0 \leq a \leq p-1$ et le garde secret.
- V1 envoie les valeurs p, g, $A=g^a \bmod p$ à V2.
- V1 peut désormais calculer la clé secrète key= $(g^b \bmod p)^a \bmod p$.

Pour le noeud V2 :

- V2 choisi un nombre secret $0 \leq b \leq p-1$.
- V2 envoie la valeur $B=g^b \bmod p$ à V1.
- V2 procède de manière analogue et obtient la même clé que V1 :

Key= $(g^a \bmod p)^b \bmod p$.

A et B sont alors en possession chacun de la même clé secrète (key) et peuvent ainsi utiliser un simple algorithme de clé privée.

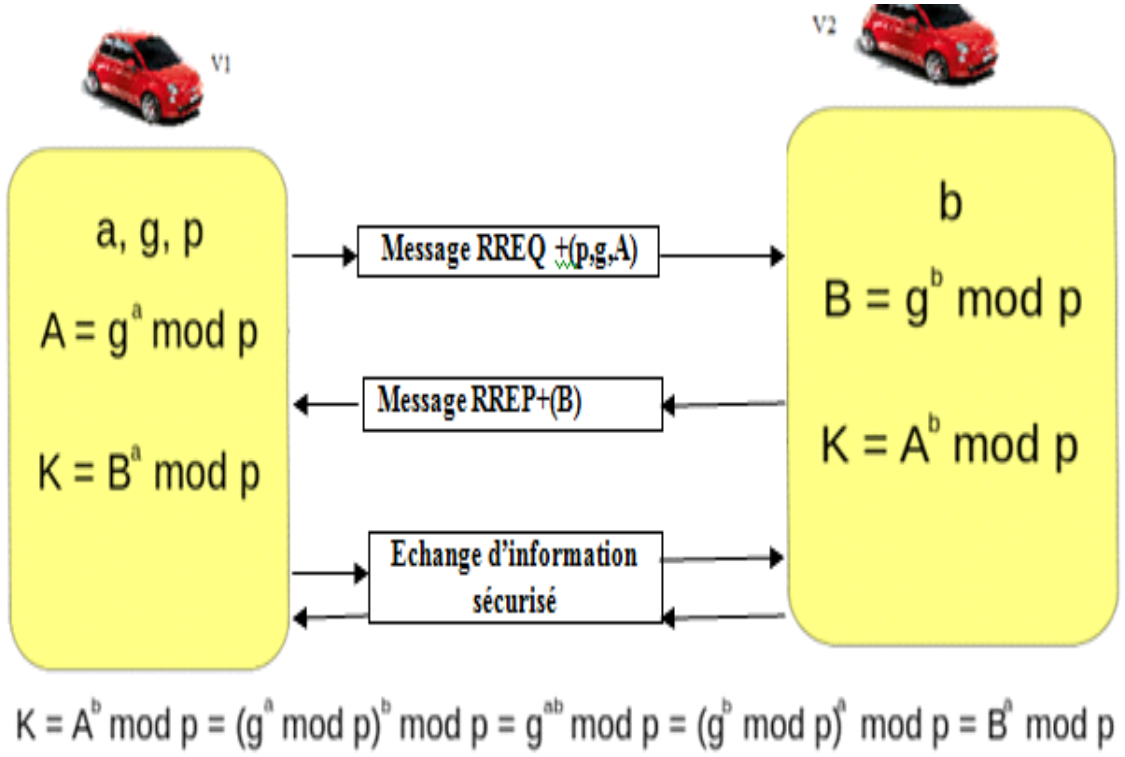


Figure III.15 : Echange d'information sécurisée avec DH-DSR

III.3.5. Résolution de principal problème de DH "man in the middle" avec la génération de la signature numérique :

- **L'attaque de l'homme au milieu :**

L'échange de clé de Diffie-Hellman est sensible à l'attaque de l'homme au milieu. Cette attaque permet à un attaquant actif O de s'intercaler dans la communication entre A et B et de créer avec A une clé commune, de faire de même avec B. Ainsi A et B pensent communiquer directement alors qu'en réalité, chacun communique avec O. C'est donc une attaque qui exploite le défaut d'identification de A vis à vis de B et de B vis à vis de A.

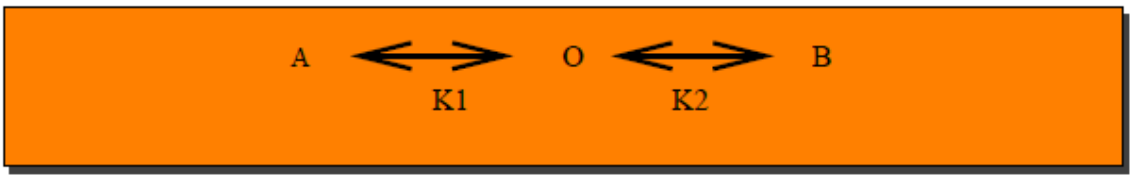


Figure III.16 : Attaque de l'homme au milieu

Pour assurer l'authentification mutuelle des deux interlocuteurs de manière à éviter l'attaque de l'homme au milieu il faut combiner l'échange de clé de Diffie-Hellman avec la signature à clé publique.

- **la génération de la signature numérique : [5]**

La signature numérique symétrique sera le mécanisme permettant de garantir l'intégrité des paquets DSR échangés entre les nœuds et de les authentifier. En effet la mobilité des nœuds impose que le temps de routage de paquet de la source à la destination soit minimal, c'est pourquoi on propose d'utiliser l'algorithme de chiffrement AES.

Lorsqu'on souhaite signer un paquet P, on utilise le processus suivant :

- On applique la fonction de hachage MD5 aux données du paquet DSR, que nous avons choisi car elle donne un condensé réduit à 128 bits, notons HS(P) le résultat de cette opération, Ce condensé sera ajouté comme champs authentification au paquet DSR et permet de s'assurer de son l'intégrité, et qu'il est bien entier et sans erreur.

- On chiffre ce condensé AES (HS(P)) Utilisant la clé secrète Diffie-Hellman, le résultat de cette opération constitue la signature S(P) du paquet.

Lorsque la destination reçoit le paquet, il vérifie son authenticité avec la procédure suivante :

- Le condensé HD(M) du paquet à la destination est généré de la même manière au moyen de la fonction de hachage MD5.
- Parallèlement, la signature S(P) est déchiffrée au moyen de la clé secrète Diffie-Hellman déjà établie lors de la découverte de route. Le condensé censé avoir été généré par la source est ainsi retrouvé par le destinataire $AES^{-1}(S(P))$.
- Le condensé HD(P) est comparé avec celui déchiffré depuis la signature.
  - En cas d'égalité, le paquet P est authentifié.
  - Si les deux sont différents, soit le paquet a été altéré, soit il n'a pas été rédigé par le voisin.

Ce processus est présenté par les figures suivantes :

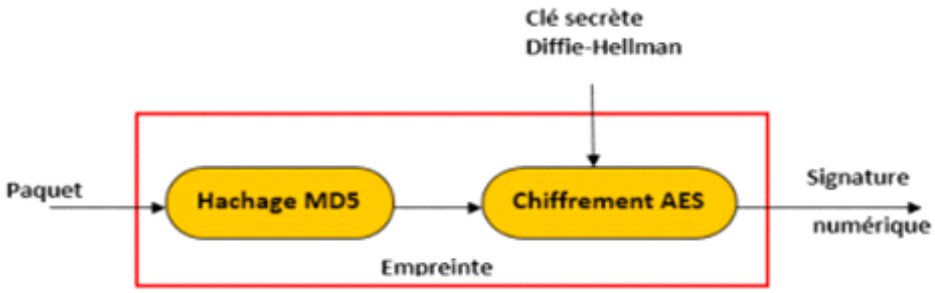


Figure III .17 : Processus de création de la signature numérique

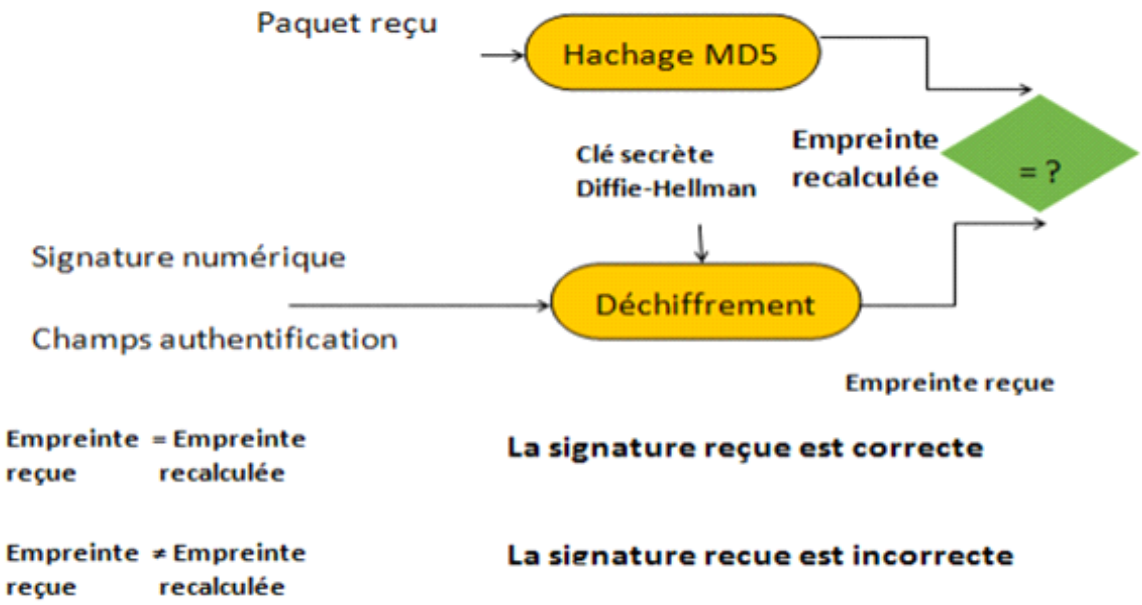


Figure III.18 : Processus de vérification de la signature numérique

Dans cette proposition de sécurité du protocole de routage DSR, nous avons trouvé une solution de signature qui protège le trafic de routage contre les attaques qui visent l'authentification et l'intégrité du contenu.

III.4. Conclusion :

Dans ce chapitre, nous allons présenter le protocole DSR ainsi le protocole Diffie-Hellman, et nous allons proposer une méthode de sécurisation du protocole DSR à l'aide des deux propositions suivantes :

(i) Etablir un champ qui contient des clés secrètes de Diffie-Hellman dans les entête des paquets RREQ et RREP. L'idée consiste à avoir des tables de nœud qui contiennent des clés secrètes qui seront utilisées comme des clés de chiffrement/déchiffrement.

(ii) Ajouter au paquet DSR une signature numérique basée sur la cryptographie symétrique générée à l'aide de l'algorithme AES et la fonction de hachage MD5 ce qui minimise le temps de calcul de la signature numérique. Cette solution proposée est spécialement conçue pour éviter les attaques d'usurpation d'identité et de modification de paquets.

Conclusion générale

Les réseaux VANETs qui avaient initialement pour objectif d'apporter des solutions de sécurité et de gestion de trafic routier, permettent actuellement le développement de nouveaux services aux usagers de la route : localisation des stations d'essence, emplacements de parking libre, le chat inter –véhicule, informations climatiques, informations culturelles, etc.

Le réseau VANET est un ensemble de nœuds communiquant entre eux par ondes radio. Ils sont caractérisés par une topologie dynamique au gré d'ajout (portée radio) ou de départ (le signal radio ne peut être capté) d'un véhicule du réseau. Les VANETS sont globalement des réseaux ad hoc, c'est-à-dire des réseaux sans infrastructure de gestion et de contrôle de la communication.

Dans cette thèse nous avons traité le problème de la sécurité des communications entre les nœuds. Plus exactement, nos travaux permettent de sécuriser les protocoles de routage de ces réseaux. Car en effet, cette fonctionnalité (le routage) est fondamentale pour le bon fonctionnement de tout réseau et de surcroît d'un réseau VANET. Toute attaque sur la fonction de routage peut conduire à des situations catastrophiques (accidents, congestion, etc.) dans le cas d'un VANET réel.

Pour atteindre cet objectif nous avons commencé par un état de l'art sur les réseaux vanet, puis une étude exhaustive des protocoles de routages des réseaux ad hoc qui spécifient la manière avec laquelle les entités communiquent pour échanger les données.

Ensuite nous avons étudié les notions et les mécanismes de sécurité dans les réseaux ad-hoc en générale ainsi dans les réseaux vanet précisément et nous avons présenté la sécurité de routage dans vanet et les attaques spécifient aux protocoles de routages.

Aussi nous avons étudié les mécanismes de sécurisation de routage par des protocoles sécurisées ou par des mécanismes de gestion des clés.et nous avons bien détaillé le protocole diffie-hellman qui est un mécanisme de gestion de clé symétrique.

Finalement nous avons étudié le protocole DSR et leur fonctionnement, Ensuite nous avons présenté une méthode de sécurisation de ce protocole à l'aide de protocole Diffie-Hellman.

En résumé, et afin d'éviter les attaques d'usurpation d'identité et de modification de paquets nous avons proposé d'ajouter deux extensions au protocole DSR : (1) établir des clés

secrètes de Diffie-Hellman entre deux véhicules au moment de découverte de route ; (2) Ajouter au paquet DSR, une signature numérique basée sur la cryptographie symétrique générée à l'aide de l'algorithme AES et la fonction de hachage MD5 ce qui minimise le temps de calcul de la signature numérique.

Liste des abréviations

AES	**A**dvanced **E**ncryption **S**tandard
AODV	**A**d-hoc **O**n **D**emand **D**istance **V**ector
A-STAR	**A**nchor-based **S**treet and **T**rafic **A**ware **R**outing
CA	**C**ertificate **A**utority
DSDV	**D**estination-**S**equenced **D**istance-**V**ector
DSR	**D**ynamic **S**ource **R**outing
DH	**D**iffie-**H**ellman
FSR	**F**isheye **S**tate **R**outing
FTP	**F**ile **T**ransfer **P**rotocol
GF	**G**reedy **F**orwarding
GPS	**G**lobal **P**ositioning **S**ystem
GPSR	**G**reedy **P**erimeter **S**tatless **R**outing
IDS	**I**ntrus **D**etection **S**ystem
IEEE	**I**nstitute of **E**lectrical and **E**lectronic **E**ngineers
IP	**I**nternet **P**rotocol
MANET	**M**obile **A**d-hoc **NET**works
MD5	**M**essage **D**igest 5
OLSR	**O**ptimized **L**ink **S**tate **R**outing
PF	**P**erimeter **F**orwarding
PKI	**P**ublic **K**ey **I**nfrastructure
RREP	**R**oute **REP**ly
RREQ	**R**oute **REQ**uest
RRER	**R**oute **ERR**OR
SAODV	**S**ecure **A**d-hoc **O**n **D**emand **D**istance **V**ector
SOLSR	**S**ecure **O**ptimized **L**ink **S**tate **R**outing
SRP	**S**ecure **R**outing **P**rotocol
TCP	**T**ransport **C**ontrol **P**rotocol
V2I	**V**éhicule **à I**nfrastructure
V2V	**V**éhicule **à V**éhicule
VANET	**V**éhicule **A**d-hoc **NET**work
ZRP	**Z**one **R**outing Protocol

Bibliographie :

[1] : Mme BOUZIANE Nabila Maitre-Assistant ‘’ Les Réseaux Véhiculaires VANET ‘’ UNIVERSITÉ DES SCIENCES ET DE LA TECHNOLOGIE HOUARI BOUMEDIENE, Algérie 2015.

[2] : Mr Sidi Mohammed Senouci professeur a l’institut Supérieur Automobile et Transport) ‘’Chapitre : Communications véhicule à véhicule ‘’.

[3] : Mr. MERAIHI Yassine ‘’ ROUTAGE DANS LES RESEAUX VEHICULAIRES (VANET) CAS D’UN ENVIRONNEMENT TYPE VILLE ’’, Thèse de MAGISTER EN GENIE ELECTRIQUE Option : INFOTRONIQUE, UNIVERSITE M’HAMED BOUGARA – BOUMERDES –ALGERIE 2011.

[4] : Shengjing MA –Benoit DE MIANVILLE ‘’QOS DANS LES RESEAUX VEHICULAIRES ‘’.

[5] : MR Mohammed ERRITALI ‘’ Contribution à la sécurisation des réseaux ad hoc véhiculaires’’, Thèse de doctorat, UNIVERSITÉ MOHAMMED V –AGDAL-RABAT-MAROC.

[6] : Ayoub Benchabana et Ramla Bensaci ‘’Analyse des protocoles de routages dans les réseaux vanet ’’, thèse de master, Université Kasdi Merbah-Ouargla-Algérie 2014.

[7] : Mlle BOUZEBIBA Hadjer et Mlle BOUIZEM Yasmina ‘’ Impact des modèles de mobilités sur les performances des protocoles de routage en milieu urbain réaliste dans les réseaux VANET (V2V) ‘’, Master en Informatique, Université Abou Bakr Belkaid–Tlemcen-Algérie 2015.

[8] : Noureddine CHAIB ‘’ La sécurité des communications dans les réseaux VANET ‘’, thèse de Magister en Informatique, UNIVERSITE ELHADJ LAKHDER – BATNA-Algérie.

[9] : Yahiatene Youcef ‘’ trafic encryption keys distribution models in Manet (distribution de clé dans un réseau dynamique) ‘’, thèse de magister, université M’hamed bougara – boumerdes –Algérie 2011.

[10] : Renaud Dumont ‘’Cryptographie et Sécurité informatique ‘’, Notes de cours provisoires, Université de Liège 2009 - 2010.

[11] : Le protocole Diffie-Hellman, José R. Beuret et Gwenol Grandperrin, Juin 2006.

[12] : S. Maag — C. Grepet — A. Cavalli ‘’ Un Modèle de validation pour le protocole de routage DSR‘’ Institut National des Télécommunications, le 18 janvier 2005.

[13]: David B. Johnson et David A. Maltz et Josh Broch ‘’: DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks’’, Computer Science Department, Carnegie Mellon University.

[14] : Dynamic Source Routing (DSR) Protocol Drs. Baruch Awerbuch & Amitabh Mishra “Advanced Topics in Wireless Networks” , Department of Computer Science Johns Hopkins.

[15]: Https://www.youtube.com/watch?v=TfK5tf3ScR4, ‘’Logjam: Diffie-Hellman, discrete logs, the NSA ‘’.

[16]: Java socket programming: create client/server chat application.
http://javahow87.blogspot.com/2015/07/how-to-create-java-chat-using-sockets.html

[17]: Implementing the Diffie-Hellman key exchange: Diffie Hellman « Security « Java Tutorial.
http://www.java2s.com/Tutorial/Java/0490__Security/ImplementingtheDiffieHellmankeyexchange.htm.

Annexe 1 : Fonctions de hachage

Pour assurer l'authentification et l'intégrité des données, on utilise les fonctions de hachage. Une fonction de hachage produit à partir d'un texte de taille variable une sortie de taille fixe. Cette sortie est souvent appelée résumé ou empreinte. Une fonction de hachage doit être sans collision, c'est-à-dire qu'il n'y a pas deux messages ayant la même empreinte. Cela signifie que la moindre modification du message entraîne la modification de son empreinte, ce qui assure l'intégrité. De plus une fonction de hachage est à sens unique car il est facile de calculer l'empreinte mais il est impossible de retrouver le message à partir de son empreinte.

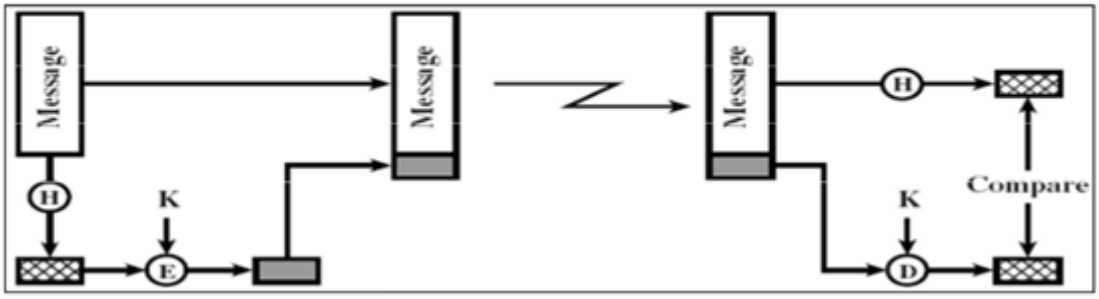


Figure 1 : Fonction de hachage avec chiffrement symétrique

L'authentification peut être assurée en utilisant le chiffrement symétrique (figure 1) et dans ce cas on parle de MAC (MessageAuthentification Code). Le résumé de message peut aussi être chiffré en utilisant le chiffrement à clef publique et le résultat de cette opération est appelé signature numérique (figure 2). En effet seul le propriétaire de la clef privée peut générer la signature et tout le monde peut la vérifieren utilisant la clef publique.

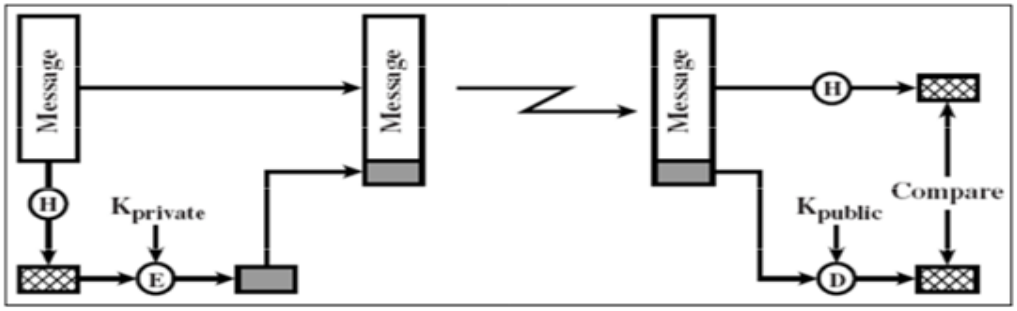


Figure 2 : Fonction de hachage avec chiffrement à clef publique

La figure 3 présente une technique qui utilise une fonction de hachage avec aucun chiffrement pour l'authentification de message. Cette technique suppose que les parties en communication, soient A et B, partagent une valeur secrète SAB. Lorsque A veut envoyer un message à B, il applique la fonction de hachage sur la concaténation de la valeur secrète et du message : le résultat MD=H(SAB||M) sera envoyé avec le message. Puisque B possède SAB il peut recalculer H (SAB || M) et vérifie MD.

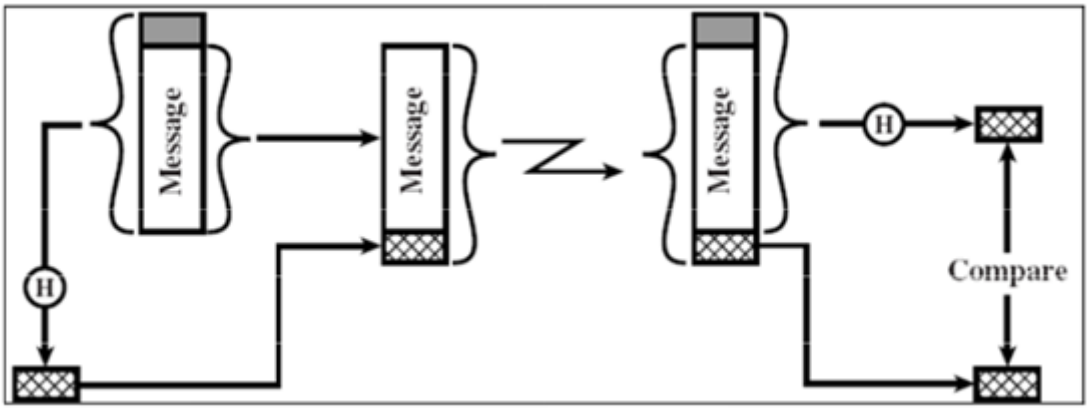


Figure 3 : Fonction de hachage sans chiffrement

Les algorithmes de hachage les plus utilisés actuellement sont :

-MD5 (MD signifiant Message Digest) créant une empreinte digitale de 128 bits.

-SHA (Secure Hash Algorithm, pouvant être traduit par Algorithme de hachage sécurisé) créant des empreintes d'une longueur de 160 bits.

Annexe 2 : La signature numérique :

La signature numérique est un système assurant l'intégrité, l'authentification et la non-répudiation des données, il repose sur la cryptographie asymétrique. L'émetteur crée une empreinte de son message, chiffre l'empreinte avec sa clé privé puis il envoie le message et la signature, le récepteur utilise la clé publique de l'émetteur pour déchiffrer la signature, il recalcule l'empreinte du message et la compare avec celle reçue. Si le condensât nouvellement calculé égale au condensât accompagnant le message alors le message n'a pas été modifié et il est prouvé authentique.

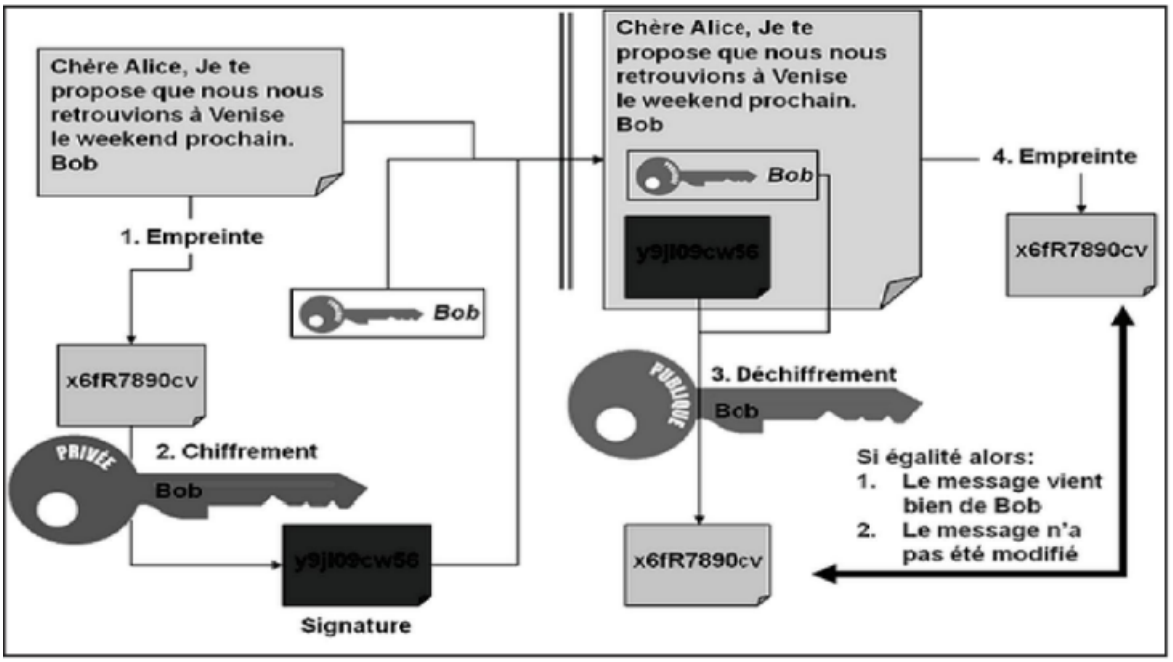


Figure 4 : procédure de La signature numérique

Annexe 3: AES (Advanced Encryption Standard)

En octobre 2000, un nouveau standard de chiffrement à clef secrète fut élu parmi 15 candidats par le NIST (National Institute of Standards and Technology) afin de remplacer le vieillissant DES dont la taille des clefs devenait trop petite. L'algorithme choisi pour devenir l'AES est le Rijndael, du nom condensé de ses concepteurs, Rijmen et Daemen. Celui-ci est un système de chiffrement par blocs. Les messages sont chiffrés par blocs de 128 bits.

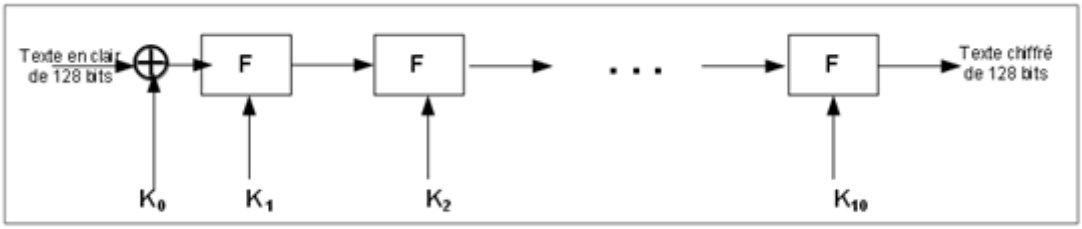


Figure 5 : les itérations de l'AES

Le principe de fonctionnement de l'AES est décrit dans la figure 7. En premier lieu, on ajoute bit à bit le message avec la clef secrète K0. Puis, comme pour tous les algorithmes de chiffrement par blocs, on itère une fonction F, paramétrée par des sous-clefs qui sont obtenues de la clefmaître par un algorithme de cadencement de clefs. Dans le cas d'AES, on itère 10 fois la fonction F. La fonction F, itérée lors du chiffrement, prend en entrée des blocs de 128 bits répartis sur 16 octets. Tout d'abord, on applique à chaque octet la même permutation S. Ensuite on applique aux 16 octets une seconde permutation P. Au résultat obtenu, on ajoute alors bit à bit la sous-clef de 128 bits obtenue par l'algorithme de cadencement de clef.

Annexe 4 : Algorithme d'échange des clés Diffie-Hellman en java

Client V1 :

```
package communication_voiture;

import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.IOException;
import java.io.InputStreamReader;
import java.io.OutputStreamWriter;
import java.io.PrintWriter;
import java.net.InetAddress;
import java.net.Socket;
import static java.time.Clock.system;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.Random;
import java.util.Scanner;

public class V1_client {

    static final int port = 2004;

    public static void main(String[] args) throws Exception {
        try {
            Socket socket = new Socket(InetAddress.getLocalHost(), port);
            BufferedReader lecture = new BufferedReader(new InputStreamReader(socket.getInputStream()));
            PrintWriter ecriture = new PrintWriter(new BufferedWriter(new OutputStreamWriter(socket.getOutputStream())), true);
            ArrayList<Integer> pgTab = new ArrayList();
//            ArrayList<Integer> aTab = new ArrayList();
            pgTab.addAll(Arrays.asList(2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109,
113, 127, 131, 137, 139, 149, 151, 157, 163, 167, 173, 179, 181, 191, 193, 197, 199, 211, 223, 227, 229, 233, 239, 241, 251));
//            aTab.addAll(Arrays.asList(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15));
            Random randomGenerator = new Random();
```

```
    int p = randomGenerator.nextInt((pgTab.size()));
    System.out.println("la valeur de p est:" + pgTab.get(p));
    int g = randomGenerator.nextInt((pgTab.size()));
    System.out.println("la valeur de g est:" + pgTab.get(g));
      int a = randomGenerator.nextInt(aTab.size());
      System.out.println("la valeur de a est:" + aTab.get(a));
    double af =6.0;
     System.out.println("la valeur de a est:" +af);
    double gf = pgTab.get(g);
    double pf = pgTab.get(p);

    while (pgTab.get(p) <= pgTab.get(g)) {
        p = randomGenerator.nextInt((pgTab.size()));
        pf = pgTab.get(p);
        gf = pgTab.get(g);
        System.out.println("la valeur de p=" + pf + "la valeur de g est:" + gf );
    }

    Scanner sc = new Scanner(System.in);
    double A = (Math.pow(gf, af)) % pf;
    System.out.println("entrer votre message");
    String msg = sc.nextLine();
    String messageV1 = gf + "," + pf + "," + A + "," + msg;

    ecriture.println(messageV1);

    String message_recu = lecture.readLine();
    String[] message = message_recu.split(",");
    String B = message[0];
    String msgV2 = message[1];
            System.out.println("Voiture1:j'ai bien reçu ton message voiture2:\n" + "B=" + B + "\n" + "votre message est:" + msgV2);
            System.out.println(af);
            double K = (Math.pow(Double.parseDouble(B), af) % pf);
            System.out.println("la valeur de K est:\n" + K);

        } catch (Exception e) {

            System.out.println("problem de connexion");
        }
    }
}
```

Server V2 :

```
package communication_voiture;

import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.IOException;
import java.io.InputStreamReader;
import java.io.OutputStreamWriter;
import java.io.PrintWriter;
import java.net.ServerSocket;
import java.net.Socket;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.Random;
import java.util.Scanner;

public class v2_Server {

    static final int port = 2004;

    public static void main(String[] args) throws Exception {
        try {
            ServerSocket s = new ServerSocket(port);
            Socket soc = s.accept();
            String message_reçu;

            BufferedReader lecture = new BufferedReader(new InputStreamReader(soc.getInputStream()));
            PrintWriter ecriture = new PrintWriter(new BufferedWriter(new OutputStreamWriter(soc.getOutputStream())), true);
            while (true) {
                message_reçu = lecture.readLine();
                String[] message = message_reçu.split(",");
                String g = message[0];
            String p = message[1];
            String A = message[2];
            String msgv1 = message[3];
              ArrayList<Integer> bTab = new ArrayList();
              bTab.addAll(Arrays.asList(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15));
            Random randomGenerator = new Random();
              int b = randomGenerator.nextInt(bTab.size());
              System.out.println("la valeur de b est:" + bTab.get(b));
            double bf = 5.0;
             System.out.println("la valeur de b est:" +bf);
            double pf = Double.parseDouble(p);

            System.out.println("voiture2:jai bien reçu ton message voiture1:\n g=" + g + "\n" + "p=" + p + "\n" + "A=" + A + "\n" + "votre message est:"
            Scanner sc = new Scanner(System.in);

            double B = Math.pow(Double.parseDouble(g), bf) % Double.parseDouble(p);
            System.out.println("taper votre message:");
            String msgV2 = sc.nextLine();
            System.out.println("envoie de message a la voiture1 en cours...");
            String messageV2 = B + "," + msgV2;
            ecriture.println(messageV2);
            double K = (Math.pow(Double.parseDouble(A), bf) % Double.parseDouble(p));
            System.out.println("la valeur de K est:\n" + K);

        }
    } catch (Exception e) {
        System.out.println("problem de connexion");

    }
```