République Algérienne Démocratique et populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université M'Hamed BOUGARA de BOUMERDES

Mémoire

Faculté des Sciences

Département d'informatique

MEMOIRE DE MAGISTER

Spécialité : informatique de gestion et spécification de logiciels

Option : spécification de logiciels et traitement de l'information

Ecole doctorale

Présenté par :

YAHIATENE Youcef

Thème

*Traffic Encryption Keys distribution models in Mobile Ad hoc Networks (*Distribution de clés dans un réseau dynamique)

Devant le jury de soutenance composé de :

Mr. M. MEZGHICHE	UMBB	Professeur	Président
Mr. K. TAMINE	Université de LIMOGE	Maître de conférences	Promoteur
Mr. M.AHMED NACER	USTHB	Professeur	Examinateur
Mr. A.ABDELLI	USTHB	Maître de recherches	Examinateur

Année Universitaire : 2010/2011

Table des matières

Remerciements

Je tiens à remercier, Monsieur TAMINE KARIM, pour ses conseils judicieux, sa grande disponibilité et les précieuses discussions que nous avons eux ensemble. Je lui exprime ma profonde gratitude pour m'avoir fait profiter de ses connaissances, mais aussi de ses méthodes de travail. Grâce à lui, j'ai découvert un domaine de recherche qui aujourd'hui me passionne.

J'adresse mes plus vifs remerciements à Monsieur M. MEZGHICHE Professeur à l'UMBB, pour avoir accepté de présider ce jury et pour m'avoir accueilli dans son laboratoire LIFAB.

Je remercie également et pleinement Monsieur AHMED NACER Professeur à l'université de Bab Ezzouar et Monsieur ABDELLI Abdelkrim Maître de conférences à l'université de Bab Ezzouar qui m'on fait l'honneur d'accepter de juger ce modeste travail.

Enfin je souhaite aussi témoigner toute mon amitié à l'ensemble de mes collègues de poste graduation.

Dédicaces

Je dédie ce travail à ma femme Ahlem

A mes parents

A mes beaux parents

A la mémoire de mon oncle YAHIATENE Mourad

A mes chers frères Adel, Sofiane, Mahfoud et Abdel Hadi

A mes chères sœurs Ibticem, Nawel, Amina, Sarah, Hadjer et Akila

A toute la famille YAHIATENE, OURAGH, BELGACEM, HIRECHE, AIT RACHED

A mes oncles et mes tantes

A tous mes cousins et cousines

A tous mes amis

A toute l'Equipe d'Emploitic

Résumé

Les réseaux ad hoc sont composés d'unités mobiles communiquant via un média sans fil, sans la nécessité d'infrastructure physique. Dans ce genre de topologie, tous les nœuds coopèrent afin d'assurer la bonne gestion du réseau (contrôle, routage,…). La nature complètement distribuée de ce type de réseau pose le problème de performances (dues aux calculs des routes) ainsi que les problèmes liées à la sécurité des échanges entre les nœuds.

L'absence d'une gestion centrale des fonctionnalités du réseau rend ces derniers beaucoup plus vulnérables aux attaques que les réseaux sans fil, les protocoles de sécurité qui existent actuellement ne sont pas conçus pour un tel environnement (dynamique). Ils ne prennent pas la contrainte des ressources en considération car non seulement l'environnement est dynamique, mais les ressources sont aussi limitées (mémoire, capacité de calcul et surtout énergie), ce qui complique davantage la problématique, Cependant, en raison de l'importance des domaines d'application des réseaux mobiles Ad hoc comme les opérations militaires (communication entre les avions, les voitures et le personnel, les opérations de secours et les situations d'urgence en cas de sinistre, etc.), En effet, la confidentialité, l'intégrité et l'authentification sont des problèmes plus saillants dans les réseaux ad hoc que dans les réseaux classiques. Une stratégie efficace pour résoudre ces problèmes consiste en l'utilisation des cryptosystèmes symétriques ou asymétriques, ces derniers permettent aux membres d'un réseau de chiffrer le flux de données lors de l'émission et de le déchiffrer au moment de la réception, grâce à un ensemble de clés dynamiquement calculées et distribuées entre ces membres.

L'objectif principal de ce mémoire consiste à étudier les solutions qui existent sur les modèles de distributions de clés d'échanges dans les réseaux dynamiques, et nous proposons une architecture de gestion de clé distribuée. Cette architecture doit supporter les différentes caractéristiques de ces réseaux (absence d'une unité centrale de gestion de réseau, topologie réseau dynamique, etc.).Et nous terminons par modéliser et implémenter cette nouvelle méthode de distribution de clé en utilisant le principe de L'interpolation polynomiale

Mot clés : *clés de chiffrement de trafic, modèle d'authentification dans réseau ad hoc, TEK …*

Abstract

The ad hoc networks are composed of mobile units communicating via a wireless media, without the need for physical infrastructure. In this kind of topology, all the nodes cooperate to ensure the proper management of the network (control, routing,…).The fully distributed nature of this type of network poses the problem of performance (due to the calculations of roads) as well as problems related to the security of exchanges between the nodes.

The absence of a central management of the functionalities of the network makes them much more vulnerable to attacks than wireless networks Unfortunately, the protocols of security which exist nowadays are not conceived for such environment (dynamics). They do not take in consideration the shortage of means because not only the environment is dynamic, but means are also restricted (memory, capacity of calculation and especially energy), which make the problems more complicated .

However, owing to the importance of the domains of application of Ad hoc mobile networks in the military (communication between planes, cars and personnel and operations of assistance, urgent situations in case of disaster, etc), In fact the confidentiality, integrity and authentication issues are most salient in ad hoc networks than in traditional networks. An effective strategy to solve these problems is the use of symmetric or asymmetric cryptosystems, they allow members of a network to encrypt the data stream on the issue and to decipher the time of receipt, with a set of keys dynamically calculated and distributed among members.

The main objective of this paper is to study the solutions that exist on the distribution model key exchange in dynamic networks by proposing a distributed architecture. This architecture must support the different characteristics of these networks (absence of a central processing unit of management of network, dynamic network topology, etc). To this end, a trust model adapted to a dynamic environment, to ensure the evolution of the trust levels of the nodes, is established. And we end with model and implement this new method of key distribution using the principle of interpolation polynomial

Key words : *Traffic encryption keys, Distribution model keys , TEK …*

ملخص

تتكون شبكات Ad Hoc من وحدات متنقلة تتصل عبر وسائط لاسلكية، من دون الحاجة إلى البنية التحتية المادية. في مثل هذا النوع من الطوبولوجيا ، كل العقد تتعاون لضمان الإدارة السليمة للشبكة (الرقابة ، التوجيه ،...). طبيعة التوزيع الكامل لهذا النوع من الشبكة يطرح مشكلة الأداء (نظرا لحسابات الطرق)، بالإضافة إلى المشاكل المتعلقة بأمن تبادل المعلومات بين العقد, نظرا لعدم وجود إدارة مركزية ، الشيء الذي يجعل من هذه الأخيرة أكثر عرضة للهجوم مقارنة بالشبكات الأخرى(السلكية واللاسلكية). للأسف، بروتوكولات الأمن والحماية الموجودة حاليا ليست مصممة لمثل هذا النوع من الشبكات) محيط ديناميكي متحرك، أضف إلى ذلك محدودية الطاقة والذاكرة وضعف القدرة على الحساب ,وذلك مما يزيد من تعقيد مشكلة الأمن في هذه الشبكات .ونظرا لأهمية استخدام هذه الشبكات في عدة مجالات مثل العمليات العسكرية، الاتصالات بين الطائرات والسيارات والأفراد وعمليات الإغاثة في حالات الطوارئ والكوارث، وما إلى ذلك ,فإنه من الضروري أن يكون الهدف الرئيسي من هذا العمل هو وضع آلية أمنية مضمونة وذلك من خلال اقتراح بنية هرمية توزيعية والتي تسمح باستخدام هيكل مبني على مفتاح عام.

في الواقع، السرية, سلامة وإثبات صحة المستندات هي المشاكل الأكثر بروزا في الشبكات Ad Hoc مقارنة بالشبكات التقليدية .هناك إستراتيجية فعالة من أجل حل هذه المشاكل هو استخدام نظم الترميز غير المتناظرة أو متماثل،(cryptosystèmes symétriques ou asymétriques)فهي تسمح لأعضاء شبكة لتشفير البيانات عند إرسالها وإلى فك الترميز

وقت الاستلام، هذا باستخدام مجموعة من مفاتيح محسوبة وموزعة بشكل حيوي على أعضاء الشبكة

الهدف الرئيسي من هذا التقرير الذي يتضمن دراسة الحلول الموجودة في توزيع نماذج في الشبكات Ad Hoc. و في النهاية نقترح نموذجا وطريقة جديدة لتوزيع المفاتيح لتشفير البيانات من خلال استخدام مبدأ polynomial الاستقراء

كلمات : مفاتيح التشفير ، ونظام مصادقة مخصصة في الشبكة، نظم الترميز غير المتناظرة أو متماثل ,تك ...

Introduction Générale:

Les réseaux ad hoc sont composés d'unités mobiles communiquant via un média sans fil et ne nécessitant pas une infrastructure physique. Dans ce genre de topologie, tous les nœuds coopèrent afin d'assurer la bonne gestion du réseau (contrôle, routage,…). Les réseaux ad hoc sont idéal pour des applications liées à des opérations de secours (militaires, pompiers, tremblement de terre, etc..) ainsi que les missions d'exploration.

La nature complètement distribuée de ce type de réseau pose le problème de performances (due aux calculs des routes) ainsi que les problèmes liés à la sécurité des échanges entre les nœuds.

Concernant le routage des paquets, les protocoles actuels se divisent en deux catégories, les protocoles proactifs et les protocoles réactifs [44].

Les protocoles proactifs établissent les routes à l'avance en se basant sur l'échange périodique des tables de routage comme dans le cas des réseaux classiques, alors que les protocoles réactifs cherchent les routes à la demande [44].

Les protocoles réactifs soufrent de coûts élevés lors de l'établissement des routes. Ceci provient en grande partie du caractère massif des inondations des paquets de recherche de routes. Le trafic généré pour le contrôle et la mise à jour de la table de routage dans les protocoles proactifs peut être important et partiellement inutile et le coût en bande passante est important.

Une troisième catégorie de protocoles de routage, dite hybrides, combinant les deux approches (réactive et proactive) a aussi été développée.
En ce qui concerne l'aspect sécurité, l'essentiel des propositions de protocoles de routage sécurisés présupposent une phase de distribution de clefs pour protéger le routage. Cependant, ces travaux reposent implicitement sur une infrastructure de sécurité, donc chaque nœud détient une clé publique et une clé privée que l'autorité de certification CA (Certificat Authority) lui délivre, la disponibilité de CA dans le réseau est importante puisque les clés doivent être mises à jour périodiquement pour diminuer les risques d'attaques .

Mais dans les réseaux Adhoc avoir un seul CA présente un point de vulnérabilité, s'il n'est plus disponible alors les nœuds ne pourront plus prouver leur authenticité des clés publiques des autres nœuds et ne pourront plus mettre en place une communication sécurisée.

Les travaux pour sécuriser le routage dans les réseaux ad hoc se heurtent souvent à la même difficulté qui est de proposer des mécanismes relativement robustes face aux différentes attaques possibles, sans pour autant affecter les performances du réseau ad hoc de manière trop prononcée. En effet beaucoup de propositions s'appuient sur un protocole de routage ad hoc existant; puis pour assurer la sécurité des échanges on y ajoute à posteriori des mécanismes de sécurité. Cependant, dans un but de simplification, les auteurs de ces travaux sont amenés à supprimer des optimisations présentes dans le protocole initial au détriment des performances du protocole résultant.

Due à la nature dynamique et flexible des réseaux ad hoc, la sécurité des communications dans ce type de réseaux présente des nouveaux défis. En effet, la confidentialité, l'intégrité et l'authentification sont des problèmes plus saillants dans les réseaux ad hoc que dans les réseaux classiques. Une stratégie efficace pour résoudre ces problèmes consiste en l'utilisation des cryptosystèmes symétriques ou asymétriques, ces derniers permettent aux membres d'un réseau ad hoc de chiffrer le flux de données lors de l'émission et de le déchiffrer au moment de la réception, grâce à un ensemble de clés dynamiquement calculées et distribuées entre ces membres. Une telle distribution doit être à la fois optimale pour éviter la surcharge du réseau, et efficace pour éviter une telle attaque, et d'ailleurs c'est ça notre problématique.

Dans le cadre des réseaux ad hoc, plusieurs modèles de distribution de clés TEKs(Traffic Encryption Key) ont été proposés, citant par exemple les modèles avec et sans contribution tels que [1] et [2] , les modèles avec et sans clustérisation : [3] et LKH [4], les Group KeyAgreement : [5] et DMGSA [6].

La première partie de notre travail consistera à faire un état de l'art des différents protocoles de routage pour les réseaux ad hoc, à étudier les performances de ces différents protocoles et enfin à étudier les notions et les méthodes qui ont été utilisées pour sécuriser les échanges dans les réseaux Ad hoc.

Dans la deuxième partie de ce mémoire, nous présenterons notre contribution : la proposition d'un modèle d'authentification et de distribution des clés de chiffrement de trafic TEKs dans le protocole de routage ad hoc AOMDV. Ce modèle permet une authentification individuelle des nœuds les uns auprès des autres, ainsi l'utilisation d'un système cryptographique symétrique pour sécuriser les échanges de données.

Ce mémoire est organisé de la manière suivante :

Première partie : Les réseaux Adhoc

Principe de fonctionnement des réseaux Adhoc.

Les protocoles de routages dans les réseaux Adhoc.

Les problèmes de sécurité liés aux réseaux Adhoc .

Deuxième partie Problème d'échanges de clé dans les réseaux Adhoc : Etat de l'art

Troisième partie : Contribution une méthode d'échanges de clé pour les réseaux adhoc basé sur l'interpolation Polynomiale

Principe de l'interpolation polynomiale.

Application : Description d'un nouveau protocole d'échange de clé pour les réseaux Adhoc.

Quatrième partie : Simulation du protocole et analyse de la sécurité.

Environnement de simulations.

Résultats des tests du protocole.

Résistance du protocole à l'attaque Man-In-The-Middle .

(a) Principe du protocole Diffie-Hellman.

(b) Principe de l'attaque Man-In-The-Middle.

(c) Notre nouveau Protocole d'échange de clé.

i. Dans cette partie on va décrire un nouveau protocole qui consiste :

ii. Le nœud A envoie un demi secret à B (par la méthode précédente (interpolation polynomiale)) .

iii. Le nœuds B envoie l'autre demi secret (par la méthode précédente) .

iv. A et B reconstituent le secret grâce à la méthode Diffie-Helman

(d) Analyse de l'attaque Man-In-The-Middle.

Première partie
Les réseaux Ad hoc

2 Principe de fonctionnement des réseaux Ad hoc

La technologie des réseaux informatiques sans fil est en pleine expansion, suscite actuellement un grand intérêt vu les facilités qu'elle apporte : grande liberté de mouvement, économie d'installation en coût et en temps, interopérabilité et fusion avec le monde de la téléphonie mobile, etc... Le but ultime des réseaux c'est : «d'offrir l'accès à l'information n'importe où et n'importe quand». Le concept des réseaux mobiles ad hoc essaie d'étendre les notions de mobilité à toutes les composantes de l'environnement. Aucune administration centralisée n'est disponible, ce sont les hôtes (noeuds) mobiles eux mêmes qui forment d'une manière ad hoc l'infrastructure du réseau.

La famille de réseaux se divise en deux classes de topologies :

Les topologies avec infrastructure : Elles sont constituées d'un ensemble de stations de bases fixes connectées par un réseau filaire. La zone de couverture de chaque station de base définit une cellule. Les hôtes mobiles communiquent entre eux via le réseau des stations de base. Le réseau GSM est un exemple typique des réseaux sans fil avec infrastructure. Les réseaux WLAN basés sur la norme IEEE802.11 sont un autre exemple plus récent de cette famille de réseaux.

Les topologies sans infrastructure : Ces réseaux sont constitués d'unités mobiles communiquant entre eux sans l'aide d'infrastructure fixe. Appelées communément ad hoc, elles ne nécessitent aucune structure physique (backbone) pour être déployées et elles sont opérationnelles instantanément. Dans ce type de réseaux, tous les hôtes doivent coopérer pour gérer les communications entre eux (routage, contrôle de l'accès au media, etc...).

Dans ce chapitre, nous allons présenter le concept des réseaux mobiles ad hoc, leurs caractéristiques et quelques exemples d'applications.

2.1 Définition des MANETs:

Un réseau mobile ad hoc est appelé MANET (Mobile Ad hoc Network),le terme MANET est le nom d'un groupe de travail de l'IETF (Internet Engineering Task Force) [7] ,est un groupe de noueds capable de s'auto organiser, et qui communiquent les uns avec les autres sans le support d'une quelconque infrastructure de gestion prédéfinie,les MANETS sont des réseaux pair à pair, multi-sauts, qui s'appuient sur les nœuds intermédiaires en tant que relais pour acheminer les paquets. Ceci signifie que les nœuds jouent à la fois le rôle d'hôte et de routeur : ils sont d'une part responsables de l'émission et de la réception de leurs propres données, et ils assurent d'autre part la retransmission du trafic des autres nœuds.

Formellement, un réseau ad hoc peut être modélisé par un graphe

Gt=(V_t , E_t), où V_t représente l'ensemble des noeuds et E_t modélise l'ensemble des connexions qui existent entre ces noeuds. Si u et v appartiennent à Vt et e=(u, v) appartient à E_t cela veut dire que les noeuds u et v sont en mesure de communiquer ensemble directement à l'instant t . Les applications des réseaux ad hoc représentent l'une des causes principales de l'importance de la sécurité dans de tels réseaux. A cause de leurs flexibilités, comme on va le voir par la suite, les réseaux ad hoc peuvent être déployés dans des scénarios critiques et de secours tels que les opérations militaires ou catastrophes naturelles, etc.

Pour préciser les domaines d'utilisation, nous donnons ci-après les principales caractéristiques des réseaux ad hoc:

2.2 Les caractéristiques des MANETS:

Un réseau ad hoc est donc un système autonome constitué de nœuds mobiles. Ces derniers communiquent avec leurs voisins par des liaisons sans fil point à point. Quand les zones d'émission/réception de deux nœuds en communication sont disjointes, les nœuds intermédiaires sont alors sollicités pour assurer le routage. A partir de cette définition générale nous présentons les caractéristiques principales qui différencient un réseau ad hoc d'un réseau doté d'une architecture fixe.

1.La mobilité de tous les nœuds (topologie dynamique): est une caractéristique intrinsèque des MANET. Le déplacement des nœuds provoque des modifications aléatoires et non prédictibles de l'architecture du réseau.

2.Équivalence des noeuds du réseau: Dans un réseau classique, il existe une distinction nette entre les noeuds terminaux (stations, hôtes) et les noeuds internes (routeurs par exemple) du réseau. Cette différence n'existe pas dans les réseaux ad hoc car tous les noeuds peuvent être amenés à assurer des fonctions de routage.

3.Le nombre de nœuds mobiles présents dans un MANET varie selon les besoins ou la position de chaque nœud. d'une façon plus générale aucune limitation n'est faite sur la taille ou le nombre de nœuds d'un réseau ad hoc.

4.Les ressources énergétiques des nœuds mobiles, alimentés par des sources d'énergies autonomes (batteries) sont limitées tel que les Personal Digital Assistant (PDA) qui sont des équipements typiquement limités en énergie, en puissance de calcul et en capacité mémoire. Ces équipements intègrent des modes de gestion d'énergie et il est important que les protocoles misent en place dans les réseaux ad hoc prennent en compte cette caractéristique.

5. L'absence de serveur centralisé rend complexe le contrôle et la gestion d'une architecture qui se forme et évolue au gré de l'apparition et des déplacements des nœuds. En conséquence, il n'existe aucune hiérarchie entre les nœuds et aucun service réseau ne peut prétendre être centralisé.

6. Les liaisons physiques s'appuient sur les technologies de communications sans fil, indispensables à la mise en place d'un réseau ad hoc. Malgré des progrès très importants, leurs performances sont encore aujourd'hui en dessous de celles des technologies des réseaux LAN filaires.

7. Sécurité physique limitée: les réseaux ad hoc ont généralement un degré faible de sécurité physique dû à l'utilisation de médium radio. Les possibilités de s'insérer dans le réseau sont plus grandes, la détection d'une intrusion ou d'un déni de service est plus délicate et l'absence de centralisation rend plus complexe la collecte d'informations pour la détection d'intrusions.

8. Une bande passante limitée : Un des caractéristiques primordiales des réseaux basés sur la communication sans fil et l'utilisation d'un médium de communication partagé. Ce partage fait que la bande passante réservée à un hôte soit modeste.

Communications multi-sauts : Dans un MANET, les nœuds qui ne peuvent directement atteindre les nœuds destinations auront besoin de relayer leurs données via d'autres nœuds.

2.3 Sélection de relais radio :

Les algorithmes de sélection se basent généralement sur une règle simple, c'est ce qu'on appelle la règle du multipoint, où chaque noeud sélectionné comme relais un sous-ensemble de ses voisins «directs» qui couvre l'ensemble de ses voisins à deux sauts. De cette manière, une information diffusée par un noeud est relayée de proche en proche par les relais, sélectionnés saut par saut, et tout le réseau finit par être couvert avec un nombre de retransmissions réduit. Les relais d'une diffusion ne seront pas forcément les mêmes, parce que chaque noeud sélectionne ses relais comme bon lui semble, tant que la règle du multipoint est respectée. Ceci aboutit à une distribution des rôles de relais et du trafic de diffusion.

Quelles sont les contraintes et les problèmes lies aux Réseaux Ad hoc, c'est ce que nous allons voir dans la section suivante :

2.4 Problèmes et contraintes liés aux réseaux sans fils :

les contraintes que nous allons voir n'ont pas d'équivalence dans les réseaux filaire

1. Une atténuation rapide du signal en fonction de la distance bien plus rapide que sur un câble, cela induit l'impossibilité pour un émetteur de détecter une collision au moment même où il transmet. Dans un réseau radio, un signal venant d'un autre noeud est tellement atténué par la distance qu'il ne provoquera que des perturbations négligeables par rapport au signal émis localement. Par exemple, sur la figure 1 (atténuation du signal), au niveau du noeud B, le signal émis par B lui-même est de très loin supérieur à celui qu'il reçoit du noeud A. Par conséquent, le signal du noeud A est complètement ignoré par B qui croit qu'il n'y a pas de collision. Malheureusement, au niveau du noeud C, les deux signaux ont des puissances

comparables et il y a bien collision du point de vue récepteur

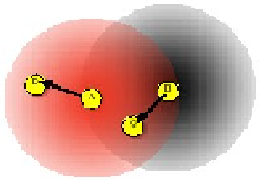


FIGURE **1:** Atténuation du signal

2. Les interférences Il est évident que les liens radios ne sont pas isolés, de plus le nombre de canaux disponibles est limité. Il faut donc les partager. Les interférences peuvent être de natures diverses. Par exemple, des émetteurs travaillant à des fréquences trop proches peuvent interférer entre eux. L'environnement lui-même peut également produire des bruits parasites (certains équipements électriques, certains moteurs, ...) qui interfèrent avec les communications.

3. La puissance du signal non seulement elle est rapidement atténuée avec la distance, mais elle est également limitée par des réglementations très strictes. Un émetteur ne peut donc dépasser une certaine puissance à l'émission.

4. L'autonomie Les applications relatives aux réseaux sans fil ont en général un caractère nomade et tirent leur autonomie de batteries. En effet, émettre ou recevoir des données consomme de l'énergie; et là encore la puissance d'émission à un impact important sur la quantité d'énergie utilisée qu'on essaie si possible de la limiter à ce qui est strictement nécessaire.

5. Problème lie à la sécurité Les protections ne peuvent pas se faire de manière physique (en général, il est difficile voir impossible d'empêcher quelqu'un de placer discrètement une antenne réceptrice très sensible dans le voisinage), c'est à dire faire du l'espionnage d'une manière passive.

6. La mobilité : Les terminaux dans le réseau Ad hoc sont amenés à se déplacer donc la topologie du réseau change fréquemment.

2.5 Les applications des Réseaux Ad hoc :

Les projets relatifs aux réseaux sans fil en général et aux réseaux Ad hoc en particulier aient débuté dans le domaine militaire, leurs domaines d'application s'étendent bien au-delà du cadre militaire. Les réseaux sans fil sont en fait plus faciles à déployer dans des bâtiments où il est impossible d'installer des câbles convenablement, parce qu'ils offrent plus de flexibilité et une rapidité de déploiement.

On distingue une variété d'applications pour les réseaux Ad hoc, on cite entre autres [8] :

Applications militaires : les réseaux Ad hoc peuvent être utilisés à la surveillance des activités des forces ennemies, à l'analyse du terrain et a la détection d'agents chimiques ou de radiations avant d'y envoyer des troupes.

Applications à la sécurité : un réseau Ad hoc peut constituer un système d'alarme distribué qui servira à détecter les intrusions sur un large secteur.

Applications environnementales : des thermo-capteurs dispersés sur une forêt peuvent signaler un éventuel début d'incendie; ce qui permettra une meilleure efficacité pour la lutte contre les feux de forêt. Sur les sites industriels, les centrales nucléaires ou dans les sites pétroliers, des capteurs peuvent être déployés pour détecter des fuites de produits toxiques (gaz, produits chimiques, éléments radioactifs, pétrole, etc.) et alerter les utilisateurs dans un délai suffisamment court pour permettre une intervention efficace.

FIGURE **2:** Application des Réseaux Ad hoc

D'une façon générale, les réseaux Ad hoc sont utilisés dans toute application où le déploiement d'une infrastructure réseau filaire est trop contraignant, soit parce que il est difficile à mettre en place, soit parce que la durée d'installation du réseau ne justifie pas de câblage à demeure

Les caractéristiques des réseaux Ad Hoc rendent le routage très complexe. Dans la partie qui suit nous présentons le principe de routage dans ces réseaux

3 Les protocoles de routages dans les réseaux Adhoc

Un mobile dans un réseau Ad hoc veut communiquer avec un autre qui n'est pas dans sa portée de communication directe,ces messages vont devoir être transmis de proche en proche jusqu'à la destination : c'est ce que l'on appelle "le routage" Un protocole de routage à pour fonction de déterminer le chemin entre deux nœuds en fonction d'une stratégie prédéfinie. Dans les réseaux multi-sauts1 , les fonctions fondamentales du routage servent à trouver et à maintenir les routes entre une source et une destination pour permettre les communications.

L'environnement Ad hoc où la gestion du réseau est distribuée sur l'ensemble des éléments constituant le réseau, la portée de transmission de chaque mobile est limitée etc., les algorithmes de routage doivent résoudre ces problèmes, supporter la maintenance et prendre en charge en un temps limité la reconstruction des routes en plus de leur rôle principal qui consiste à établir des routes efficaces entre une paire de nœuds.

3.1 Définition du Routage :

Le Routage consiste à faire transiter une information depuis un émetteur vers une destination.Le rôle principale d'un protocole de routage consiste à établir des routes efficace entre une paire de noeuds pour que les messages puissent être livrés au moment opportun .La construction des routes doit être faites avec un minimum de charge de contrôle et de consommation de bande passante.

3.2 Contraintes de routage dans les réseaux Ad hoc et la limite des protocoles conventionnels :

Comme nous l'avons vu précédemment les caractéristiques des Réseaux Ad hoc : changements de topologie fréquents, bande de transmission limitée, mobilité des nœuds, atténuation du signal, réflexion, chemins multiples, et autres caractéristiques dues aux équipements tel que les ressources limitées (capacité de CPU et de stockage limitée, faible batterie etc.)

Les protocoles de routages utilisés dans les réseaux conventionnels (filaires) ont été conçus avec l'hypothèse que les changements topologiques sont de nature simple et n'ont lieu que rarement., et sont exécutés par des nœuds dotés de suffisamment de ressources. De ce fait, il est très important de signaler que les méthodes et les approches utilisées pour l'acheminement des paquets dans le réseau Ad hoc sont évidemment différentes et plus complexes à mettre en œuvre par rapport à celles utilisées dans les réseaux classiques. Ce qui nous pousse à postuler que la gestion du routage dans un environnement Ad hoc diffère de loin de celle utilisée dans les réseaux filaires [9].

1 Le chemin entre les nœuds source et destination nécessite la présence de plusieurs nœuds intermédiaires

3.3 Notions fondamentales sur le Routage

3.3.1 L'inondation

L'inondation ou le broadcasting2 consiste à faire propager un paquet (de donnée ou de contrôle) dans le réseau en entier.le noeud diffuse le paquet à ses voisins directs, chaque voisin recevant le paquet le rediffuse à son tour jusqu'a ce que le paquet arrive à tous le réseau, ya un certain traitement qui s'applique aux paquets pour éviter le boucle à l'infini.

3.3.2 **Etat de lien (Link state):**

L'état de lien consiste que chaque noeud maintient une vision globale de la topologie du réseau, en diffusant d'une manière périodique des requêtes portant l'état de ses liens, afin que les autres hôtes puissent mettre à jour leurs tables de routage.

3.3.3 Vecteur de distance :

Chaque nœud diffuse à ses nœuds voisins, sa vision des distances qui le séparent de tous les hôtes du réseau. En se basant sur les informations reçues par tous ses voisins, chaque nœud de routage fait un certain calcul pour trouver le chemin le plus court vers n'importe quelle destination. Le processus de calcul se répète, s'il y a un changement de la distance minimale séparant deux nœuds, jusqu'à ce que le réseau atteigne un état stable. Cette technique est basée sur l'algorithme distribué de Bellman Ford.

3.4 Classification des protocoles de routages:

Selon la manière de création et de mise à jour de routes lors de l'acheminement des données, les protocoles de routage peuvent être classés en trois catégories : les protocoles proactifs et les protocoles réactifs et les protocoles hybrides .

3.5 Les protocoles proactifs:

Dans ce type de protocole le nœud maintient d'une manière permanente des routes pour toutes les destinations du réseau, donc la recherche des routes se fait dans un intervalle régulier.le principe de base est de maintenir à jour les tables de routage, de sorte que lorsqu'une application désire envoyer un paquet à un autre mobile, une route soit immédiatement connue.L'avantage premier de ce type de protocole est d'avoir les routes immédiatement disponibles quand les applications en ont besoin, mais cela se fait au coût d'échanges réguliers de messages (consommation de bande passante) qui ne sont certainement pas tous nécessaires (seules certaines routes seront utilisées par les applications en général) [45].

2 diffusion du paquet dans tous le réseau

3.6 Les protocoles réactifs:

A la différence des protocoles proactifs, les routes ne sont pas calculées que sur demande. Les protocoles réactifs ou encore ”on demand” entreprennent la recherche d’une route uniquement avant de transmettre un paquet. Leur principe est de ne rien faire tant qu’une application ne demande pas explicitement d’envoyer un paquet vers un noeud distant.L’avantage majeur de cette méthode est qu’elle ne génère du trafic de contrôle que lorsqu’il est nécessaire et donc elle réduit de manière drastique la quantité d’information de routage à transmettre [45].

3.7 Les protocoles hybrides:

Dans cette catégorie de protocole on utilise les techniques des deux protocoles qu’on a vu (protocole ré-actif et proactif) ,le réseau est décomposé en un ensemble de zone ou il utilise le principe des protocoles proactifs pour avoir des informations sur les voisins les plus proches (deux à trois saut) et les autres zones fait appel au protocole réactif .Ce type de protocoles peut combiner les avantages des deux méthodes : échange de paquets de contrôle réguliers et inondation de l’ensemble de réseau pour chercher une route vers un noeud éloigné [45].

3.8 Description de quelques protocoles de Routage:

3.8.1 Protocole de la famille proactif :

DSDV (Dynamic Destination-Sequenced Distance-Vector) [45]: Ce type de protocole utilise le principe de Bellman-Ford, chaque noeud sauvegarde une table de routage contenant toutes les destinations possibles avec trois entrées : adresse de la destination, nombre de sauts, numéro de séquence (NS). Le numéro de séquence est utilisé pour connaître l’information la plus récente. La mise à jour de la table de routage s’effectue périodiquement et après chaque changement de topologie du réseau, ce protocole utilise deux types de paquets "**full dump**" qui contient toutes les informations de routage et un paquet plus petit dit "**incrémental**", ce dernier contenant les informations du routage changées depuis la dernière mise à jour .

FSR (Fisheye State Routing) [45]:Ce protocole de routage est basé sur la technique nommée **"œil de poisson"** proposée pour réduire la taille des informations dans la représentation des données graphiques. Pour réaliser cette technique, FSR introduit la notion de portée. La portée d’un nœud i est l’ensemble des nœuds qui peuvent être atteints en passant par h sauts à partir de i. Un nœud dans FSR effectue des mises à jour pour les nœuds qui résident dans sa portée plus fréquemment que pour les nœuds qui n’appartiennent pas à sa portée. Cette technique permet de réduire considérablement la charge de routage mais au prix d’utilisation de chemins non optimaux. La figure 3 illustre le principe de la technique de protocole FSR .

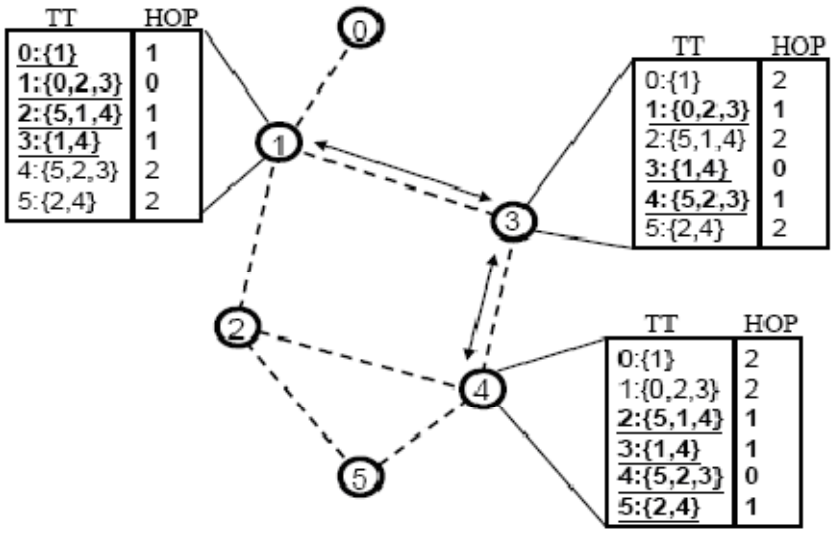


FIGURE **3:** Protocole FSR

3.8.2 Protocole de la famille ré-actifs :

1. AODV (Ad hoc On-demand Distance Vector / Routage avec Vecteur de Distance à la Demande)[45]:Ce protocole fait partie des protocoles à vecteur de distance, qui sont en général sujets aux problèmes de boucle et de comptage à l'infini de l'algorithme de Bellman-Ford, mais dans le cas de AODV ces problèmes sont résolus par l'utilisation de numéros de séquence pour les messages de contrôle.Si une application dans le réseau veut envoyer des packets et que une route est disponible dans sa table de routage le protocole AODV ne joue aucun rôle. S'il n'y a pas de routes disponibles, il va par contre en rechercher une. Cette recherche commence par une inondation de paquets Route Request (RREQ). Chaque noeud traversé par un RREQ en garde une trace dans son cache et le retransmet. Quand les paquets de recherche de routes arrivent à destination (ou un noeud intermédiaire qui connaît lui-même une route valide jusqu'à la destination), alors un paquet de réponse est généré (RREP) et il est envoyé par le chemin inverse grâce aux informations gardées dans les caches des noeuds traversés par les RREQ .Dans la figure 4 illustre le principe de découverte de chemins dans le protocole AODV.

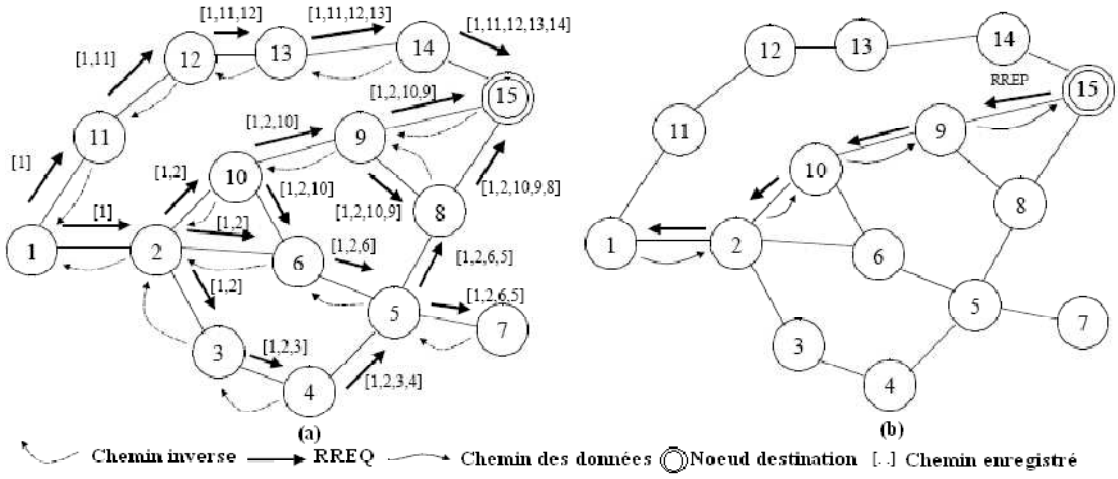


FIGURE **4:** Phase de découverte de chemins dans AODV

Ils existent d'autres protocoles de routages ré-actifs citons :

2. DSR « Dynamic Source Routing / Routage à Source Dynamique »:

Le protocole "Routage à Source Dynamique" (DSR : Dynamic Source Routing) [45], est basé sur l'utilisation de la technique "routage source". Dans cette technique, la source de données détermine la séquence complète des nœuds à travers lesquelles, les paquets de données seront envoyés. Les deux opérations de base dans DSR sont :

- découverte de route .
- maintenance de route .

Dans la première opération, un hôte diffuse un paquet de type 'requête de route'. Cette requête est propagée dans le réseau jusqu'à ce qu'elle atteigne la destination qui répond par un paquet 'réponse de route' avec la séquence des nœuds visités. Une fois la route construite, elle est incluse en tête du paquet de donnée. Un nœud qui reçoit le paquet de données supprime son adresse de l'entête du message reçu puis le transmet au nœud suivant. La deuxième opération assure la validité des chemins utilisés en exécutant une procédure de maintenance de route. Quand un nœud détecte un problème fatal de transmission il envoie un message 'erreur de route' à l'émetteur original. Ce dernier doit initier une nouvelle opération de demande de route.

3. TORA (Temporary Ordering Routing Algorithme / Algorithme de Routage Ordonné Temporairement) [45]:

L'Algorithme de Routage Ordonné Temporairement ou TORA (Temporary Ordering Routing Algorithme) a été conçu principalement pour minimiser l'effet des changements de la topologie qui sont fréquents dans les réseaux ad hoc. L'algorithme s'adapte à la mobilité de ces environnements en stockant plusieurs chemins vers une même destination, ce qui fait que beaucoup de changements de topologie n'auront pas d'effets sur le routage des données, à moins que tous les chemins qui mènent vers la destination seront perdus (rompus). La

principale caractéristique de TORA, est que les messages de contrôle sont limités à un ensemble réduit de nœuds. Cet ensemble représente les nœuds proches du lieu de l'occurrence du changement de la topologie .

Et pour terminer cette partie nous allons faire une petite comparaison entre les protocoles pro-actifs et ré-actifs

Protocole pro-actifs	Protocol ré-actifs
• Evaluation des routes d'une maniere continue dans le réseau • Route existe dans la table de routage transmission directe des données • Gain de temps routes connues au préalables • Gaspillage du capacité du reseau à cause du changement frequent de la topologie sans l'utilisation de l'information • Emission réguliere des messages dans le réseau pour le mentient de la table de routage	• Route determine à la demande • Delais important pour trouver une route • Pas de surconsommation de la bande passante • Pas de gaspillag de ressource

TABLE **1:** Protocoles pro-actifs vs ré-actifs

3.8.3 Protocole de la famille hybride:

Le protocole ZRP (Zone Routing Protocol) :Dans ce protocole le réseau est décomposé en plusieurs zones ,le routage interzone est assuré par un protocole de routage proactif ,et le routage intra zone est assuré par un protocole de routage réactif,les nœuds qui se trouvent exactement à une distance égale au rayon de la zone sont appelés "**nœuds périphériques**". Les performances de ZRP dépendent de la valeur choisie pour le rayon des zones. Pour des grandes valeurs, ZRP se comporte comme un protocole de routage purement proactif tandis que pour des petites valeurs, il se comporte comme un

protocole de routage purement réactif [10] .

Dans la partie qui suit nous allons faire une comparaison entre quelques protocoles de routages:

Protocoles / propriétés	DSDV	OLSR	AODV	DSR	ZPR	TORA
Sans boucle	Oui	Oui	Oui	Oui	Oui	Oui
Routes multiples	Non	Non	Non	Oui	Non	Oui
Distribué	Oui	Oui	Oui	Oui	Oui	Oui
Réactif	Non	Ono	Oui	Oui	Hybride	Oui
Supporte les liens unidirectionnels	Non	Non	Non	Oui	Non	Non
Qualité de Service supportée	Non	Non	Non	Non	Non	Non
Multicast	Non	Oui	Oui	Non	Non	Non
Sécurité	Non	Non	Non	Non	Non	Non
Conservation d'énergie	Non	Non	Non	Non	Non	Non
Braodcast périodique	Oui	Oui	Oui	Non	Non	Non

Tous les protocoles de routage ad hoc sont distribués. Cela signifie que le routage est effectué par les entités elles-mêmes et non pas par des routeurs dédiés. On remarque aussi que la sécurité, la conservation d'energie et la qualité de service n'ont pas été pris en considération lors de la conception de ces protocoles. Les protocoles de routage ad hoc diffèrent dans leur schéma de base : il y a des protocoles qui fournissent plusieurs routes vers une même destination tandis que les autres ne garde qu'un seul chemin (le plus court chemin en termes de nombre de sauts). Le multicast est un procédé de routage des données qui consiste à envoyer un même paquet à plusieurs destinations. Il est supporté par les deux protocoles AODV et OLSR. Les protocoles non réctifs (proactifs) se base sur la diffusion périodique (broadcast) des paquets de contrôle afin de garder une vue globale du réseau au niveau de chaque entité contrairement aux protocoles réactifs qui minimisent l'overhead des écahnges en établissant les chemins lors du besoin seulement.

3.9 Conclusion:

Après notre étude des différents protocoles de routage qui existent, nous avons vu que ces protocoles utilisent une variété de techniques afin de résoudre le problème de routage. Une technique proactive maintient les routes fraiches périodiquement tandis qu'une technique réactive établit des découvertes de routes sur demande. Chaque technique peut donner des résultats très satisfaisants dans un réseau particulier et ne pas convenir du tout pour d'autres réseaux ad hoc. En résumé, cette étude théorique nous a permis de nous familiariser avec un environnement de routage récent. Elle présente le point de départ vers l'étude de mécanismes de sécurité appliqués dans le routage MANET.

4 Les problèmes de sécurité liés aux réseaux Adhoc

Le développement des protocoles de routage et la communication sans fil a permis le traitement de l'information à travers des unités mobiles qui ont des caractéristiques particulières, donc ces unités portables peuvent s'interconnectées par liaison sans fil et formant un réseau sans aucune infrastructure ou une administration centralisée. La sécurité dans ce type de réseau est l'un des obstacles à un large déploiement des réseaux Ad hoc, sécuriser un réseau sans fil revient à instaurer les différents services de sécurités en prenant compte les caractéristiques de ces réseaux ,la communication dans le réseau s'effectue à travers des liens sans fil, ces derniers les rendent exposés à des attaques passives comme les écoutes clandestines qui permettent à un adversaire d'avoir accès à des informations secrètes, et des attaques actives, quant à elles peuvent détruire, injecter, modifier des messages et usurper l'identité d'un noeud, et par conséquent violer la disponibilité, l'intégrité et l'authentification, qui sont des éléments de base de sécurité des réseaux.

4.1 Difficultés pour la sécurité dans les Réseaux Ad hoc:

Comme nous l'avons vu dans les propriétés des réseaux Ad hoc totalement distribué, ces obstacles doivent être pris en compte lors de l'établissement d'une politique de sécurité:

L'absence d'une infrastructure : l'absence d'une administration centralisée rend la tâche difficile pour gérer les différents nœuds et sécuriser et concentrer les accès au réseau en un seul point .

La taille du réseau ad hoc peut être très importante. Donc, on ne peut pas contrôler le nombre de membres ni la fréquence d'adhésion au réseau.

La mobilité des noeuds ad hoc doit être prise en compte pour assurer des communications sécurisées. En effet, quand un noeud se déplace dans le réseau, il peut perdre sa connectivité avec le réseau sans vouloir le quitter. Il ne doit pas être obligé à chaque fois de se ré-authentifier. De plus, ce mécanisme de ré-authentification doit être léger et efficace et nécessiter le moins de messages transmis possible.

Il est difficile d'établir une confiance entre noeuds, les uns dans les autres, en même temps, il est très facile d'intercepter les informations qui circulent entre noeuds en plaçant par exemple une antenne réceptrice très sensible dans le voisinage.

La solution la plus appropriée pour assurer les services de sécurité requis dans les MANETs est l'établissement d'un protocole de gestion de clé. Ce protocole doit assurer la distribution de la clé de chiffrement de données TEK (Traffic Encryption Key) à la source pour chiffrer les données et aux récepteurs pour les déchiffrer. Il est également responsable de sécuriser la distribution de la TEK grâce au déploiement des clés de chiffrement de clés KEKs (Key Encryption Key) et du renouvellement de clés selon la politique de sécurité adoptée par l'application en question.

4.2 Concepts de base sur la sécurité:

Nous présentons dans ce qui suit quelques notions utilisées dans le domaine de la sécurité des réseaux en général et des réseaux mobiles en particulier :

4.2.1 Cryptographie symétrique:

La technique de 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. Dés le début, la clé est échangée entre les deux acteurs de la communication et elle reste confidentielle.

4.2.2 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 la quelle les deux clés (privée/publique) sont utilisées [29].

La clé publique sert à crypter les messages tandis que la clé privée sert à les décrypter. Une fois le message crypté, seul le destinataire est en mesure de décrypter son contenu secret [29].

4.2.3 Fonction de hachage :

La fonction de hachage est utilisée pour réduire des données de n'importe quelle taille en un condensé de taille fixe. Le résultat de cette fonction est envoyé avec le message pour assurer l'intégrité des données.

L'intérêt et le point fort de la fonction de hachage est qu'elle peut être facilement calculée mais difficilement inversée. Et elle a la particularité que le moindre changement du message original entraînera un changement dans le condensé .

Grâce aux propriétés de la fonction de hachage, cette dernière a été employé dans les mécanismes de sécurité tels que la signature numérique.

4.2.4 Chaînes de hachage :

la chaîne de hachage est obtenue en appliquant plusieurs fois la fonction de hachage h sur une donnée initiale x_0 , La séquence des x_i obtenues peut être utilisée par exemple comme des mots de passe afin de s'authentifier auprès d'un serveur.

4.2.5 Signature numérique

La signature numérique est un mécanisme qui permet d'authentifier un message, autrement dit de prouver qu'un message provient bien d'un expéditeur donné, à l'instar d'une signature sur un document papier. Pour signer un message, on peut le chiffrer avec la clé privée. Le déchiffrement avec la clé publique prouve que seul le détenteur de la clé privée à créer la signature .

4.3 Conditions de sécurité:

Le service de sécurité dans un réseau mobile ad hoc n'est pas différent de ceux des autres réseaux. Le but c'est de protéger l'information et les ressources des attaques et des mauvais comportements[30].

4.3.1 La confidentialité des données :

Le principe de la confidentialité assure que seul les acteurs de la transaction sont en mesure de comprendre les données secrètes échangées. Des contrôles d'accès stricts doivent être mis en place pour garantir la confidentialité des données .

4.3.2 L'intégrité des données:

L'intégrité des données garantit que les données échangées n'ont pas été altérées ou modifiées d'une manière inattendue.

4.3.3 L'authentification :

L'authentification assure que la communication entre l'émetteur et le récepteur se faite d'une manière authentique pour interdire aux intrus d'injecter des messages falsifiés (s'assurer qu'il n y a pas de nœud malveillant masqué "usurpation d'identité").

4.3.4 La non-répudiation :

La non-répudiation assure qu'un message envoyé ne sera pas nié par son expéditeur, cette propriété est réalisé en appliquant une méthode basée sur la signature électronique.

4.3.5 La disponibilité :

Ce principe permet de s'assurer que les services réseau désirés sont toujours disponibles, le système qui assure la disponibilité dans un réseau ad hoc cherche à combattre les dénis de service et les nœuds qui se comportent mal tels que les nœuds égoïstes

4.4 Les attaques dans les réseaux ad hoc :

Les réseaux Ad hoc sont dynamiques et auto-organisé donc n'importe quel noeud peut participer au routage et comme il utilise le medium sans fil pour envoyer des paquets donc il n'y a pas de barrières pour qu'un nœud malicieux causera des perturbations dans le trafic circulant, dans la partie qui suit nous allons présenter quelques type d'attaque:

4.4.1 Nœud malicieux :

Un nœud malicieux n'est qu'une unité mobile malveillante, ayant pour but d'écouter illégalement le trafic, et qui peut même supporter des coûts énergétiques afin de lancer des attaques qui perturbent le fonctionnement correct du réseau [31]

4.4.2 Attaquant actif-n-m :

Un attaquant actif-n-m est un attaquant qui possède m nœuds (m nœuds malicieux) et qui compromis n nœuds. Par exemple, actif-0-1 est un attaquant qui possède un seul nœud, et actif-1-2 est un attaquant qui possède deux nœuds et qui compromis un seul.

4.4.3 Les attaques de type dénis de service DoS :

Ce type d'attaque vise de rendre l'application incapable de répondre aux demandes des utilisateurs, cette catégorie d'attaque est connue sous plusieurs forme, la plus connue est l'envoi d'une quantité excessive de données a fin de surcharger le réseau.De nombreux type d'attaques de dénis de services existent:

Brouillage du canal radio pour empêcher les communications.

Tentative de débordement de la table de routage des nœuds relais.

Non-coopération d'un nœud au bon fonctionnement du réseau dans le but de préserver son énergie. Cet égoïsme peut mettre tout le réseau en péril.

Tentatives de gaspillage d'énergie. Cette attaque oblige un nœud qui est censé rester en veille le plus longtemps possible à passer en mode actif et donc à consommer toute son énergie. Ross Anderson et Franck Stajano la présentent sous l'appellation de « sleep deprivation torture attack » ou scénario de torture par privation du sommeil. [11] .

Dispersion et suppression de trafic en jouant sur les mécanismes de routage.

4.4.4 Modification des champs de messages de contrôle

Les nœuds malicieux peuvent altérer les différents champs d'un paquet. Chaque modification rapportée a un effet particulier sur le routage et un but précis :

Modification du numéro de séquence le SN (Sequence Number) : un chemin avec un SN très haut est automatiquement choisi pour l'acheminement des données. En effet, un adversaire peut incrémenter ce champ pour se présenter comme relai à faible coût. Il oblige donc les paquets à le traverser. Supposons qu'un nœud malicieux M reçoit un RREQ originaire de S afin d'atteindre la destination X, après avoir été acheminé par B. M peut réorienter le trafic à travers lui-même en répondant à B par un RREP qui contient un numéro de séquence pour X plus grand que le dernier déclaré par X même. Eventuellement, le RREQ inondée par B atteint un nœud qui a une route valide vers X et par conséquent, un RREP valide va être envoyé vers S. Cependant, B a déjà reçu le faux RREP de M, si le numéro de séquence pour X (que M a utilisé) est plus grand que le numéro du vrai RREP, B ignore ce vrai RREP. Cette route ne va pas être corrigée jusqu'à ce qu'un vrai RREQ ou bien un vrai RREP pour X, avec un numéro de séquence plus grand, rentre au réseau.

Modification du nombre de sauts : Il suffit donc qu'un nœud positionne ce champ à zéro ou à l'infini (une très grande valeur) pour que le routage soit complètement perturbé. Par exemple le protocole AODV utilise le nombre de sauts pour déterminer le chemin optimal et le plus court, par conséquent, les nœuds malicieux peuvent augmenter leur chance d'être inclus dans le nouveau chemin crée en mettant un zéro dans le champ de calcul de sauts du RREQ qu'ils acheminent.

Modifier les routes vers la source : en particulier dans les protocoles qui mettent la route explicitement dans les paquets (exemple : DSR). Une simple attaque qui altère ce champ délicat entraine des boucles dans le routage.

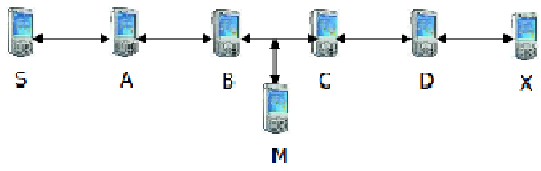


FIGURE 5: Attaque en utilisant des modifications

4.4.5 Attaques en utilisant la fabrication:

Cette catégorie d'attaque inclue les attaques basées sur la génération de faux messages de routage. Des attaques pareilles sont difficiles à détecter [30].

Falsifier les routes erreur : Dans certain protocoles de routage à la demande comme AODV et DSR implémentent des mécanismes pour la maintenance des chemins et pour éliminer les routes inexistantes causés par la mobilité des nœuds. Quand une route active de S vers D est défaillante, le nœud prédécesseur du lien en question envoie un paquet RERR à S, si ce dernier n'a pas d'autres routes vers D et il a toujours besoin de cette route, il

lance une nouvelle opération de découverte. La vulnérabilité c'est que cette attaque peut être lancée en disséminant un faux RERR, ce qui donne une destruction de routes valides qui cause un déni de service et privation de sommeil.

Diffusion des routes falsifiées :Le but de cette attaque est de surcharger la table de routage des nœuds avec des routes erronés, l'attaquant peut exploiter cette méthode de route et empoisonne les caches de ses voisins en diffusant des paquets contenants de fausses routes.

4.4.6 Les attaques passives:

Attaque passive, ou se qu'on appelle le sniffing, à cause de la nature physique des ondes radio (un support de transmission partagé). Un noued peut écouter le réseau et récupérer des données illégalement et qui peuvent être confidentielles.Ce type d'espionnage est très fréquent dans les réseaux Ad hoc,un sniffer peut analyser les paquets ,il trouvera la structure du réseau donc il peut préparer une attaque active pour nuire le réseau.

4.4.7 L'usurpation de l'identité d'un noeud :

Ce type d'attaque est catégorisé dans les attaques active c'est ce qu'on appelle la mystification, consiste à se faire passer pour une entité connue et fiable auprès des autres noeuds dans le réseau à fin de récupérer illégalement des informations confidentielles ou injecter des messages dans le réseau.

4.5 Classification des attaques:

On peut classifier les attaques selon plusieurs critères

4.5.1 Attaque interne ou externe :

Si l'attaquant ou le noeud malicieux se trouve dans le réseau on parle d'une attaque interne, dans le cas ou le nœud malicieux se connecte depuis l'extérieur on dira que l'attaque est externe.

4.5.2 Attaque passive ou active :

Dans le cas d'une attaque active le noeud essaie d'altérer le message envoyé d'un noued à un autre dans le but de modifier le protocole ici l'intégrité du message n'est pas protegé.Par contre, une attaque passive prive le réseau de la confidentialité des messages échangés.

4.5.3 Attaque individuelle ou attaque distribuée :

Dans l'attaque individuelle, une seule entité est utilisée.par contre il existe des attaques qui utilisent plusieurs nœuds c'est se qu'on appelle attaque distribuées, ce genre d'attaques est plus dangereux et difficile à détecter.

Voici quelques attaques pour les réseaux Ad hoc :

Black hole :Elle est catégorisé comme attaque interne où l'intrus absorbe les paquets qui le traversent et crée ce qu'on appelle un trou noir dans le réseau. [12] [13]

Green hole : Au contraire du black-hole qui absorbe tous, ici l'adversaire diffuse une partie des messages reçus et bloque le reste. [13]

Traffic jamming (brouillage) : L'attaquant paralyse le réseau en générant un signal sur les mêmes fréquences du réseau. [14]

Spoofing ou usurpation d'identité : consiste à se faire passer pour quelqu'un d'autre en utilisant son identité. L'attaquant se présente en utilisant l'identité d'un nœud légitime et peut ainsi communiquer avec les nœuds du réseau sans être rejeté. [13]

Flood ruching attack (précipitation) :l'attaquant inonde le réseau par des requêtes pour augmenter la charge de réseau. [32]

Wormhole ou Attaque tunnel :Dans cette attaque deux nœuds malicieux coopèrent et falsifient le nombre de sauts en annonçant un raccourci fictif entre deux parties du réseau .Dans la figure suivante la source choisit d'acheminer les données par {S,M1,M2,D} au lieu de {S,A,B,C,D} car elle la considère comme une route plus courte mais en réalité les attaquants utiliseront un chemin plus long {S,M1,A,B,C,M2,D} vu que le lien déclaré entre M1 et M2 n'est pas réel.

Attaque sybille : L'attaque sybille consiste pour un attaquant unique à simuler le comportement de plusieurs participants. Le nœud malicieux crée un ensemble de nœuds fictifs ou bien usurpe l'identité des nœuds réels.

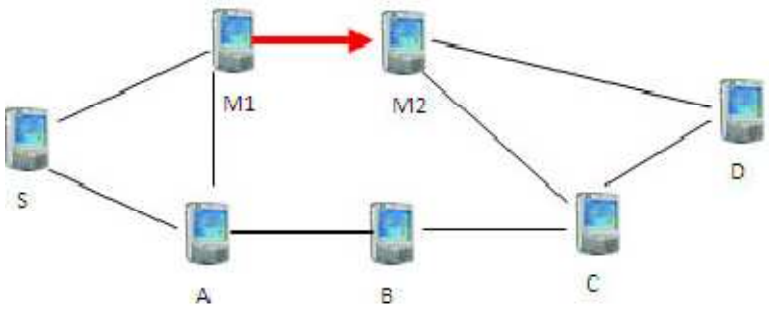


FIGURE 6: Attaque tunnel

Tableau récapitulatif des attaques Ad hoc:

Dans le tableau qui suit nous allons montrer les attaques dans les réseaux ad hoc

Attaque	Active	interne	Individuelle	Type visé
Sybille	Oui	oui	oui	Routage sans fil
Ecoute passive	non	oui(ou externe)	oui	Tous type de routage
Black hole	oui	oui	oui(ou distribué)	Routage et calcul distribué
Green hole	oui	non	oui	Routage et calcul distribué
Traffic jamming	non	non	oui	Routage sans fil
Wormhole	oui	oui	non(distribué)	Routage sans fil
spoofing	oui	oui	oui	Routage et calcule distrbué
Replay	oui	oui	oui	Tous type de routage
Dos	oui	oui	oui	Routage sans fil
Flood ruching	oui	non	oui	Routage sans fil

4.6 L'authentification et modèle de confiance dans les réseaux Ad hoc:

Authentifier c'est-à-dire permettre à un nœud d'assurer l'identité du nœud avec lequel il communique.Pour que deux nœuds puissent communiquer ensemble, la confiance mutuelle se révèle très importante en effet, la confiance doit être un pré requis nécessaire à l'établissement de la communication entre deux nœuds pour assurer en conséquence les services de confidentialité et de contrôle d'accès. Les services d'authentifications se révélé ainsi indispensable pour pouvoir faire communiquer deux nœuds appartenant à un réseau Ad hoc .Nous allons présenter les différentes approches d'authentifications [15]:

4.6.1 La technique de cryptographie à seuil :

Dans les réseaux filaires pour sécuriser la communication entre ces membres on utilisera une infrastructure à clé publique PKI (Public Key Infrastructure), donc chaque nœud détient une clé publique et une clé privée que l'autorité de certification CA (Certificat Authority) lui délivre, la disponibilité de CA dans le réseau est importante puisque les clés doivent être mises à jour périodiquement pour diminuer les risques d'attaques . Mais dans les réseaux Ad hoc avoir un seul CA présente un point de vulnérabilité, s'il n'est plus disponible alors les nœuds ne pourront plus prouver leur authenticité des clés publiques des autres nœuds et ne pourront plus mettre en place une communication sécurisée.

Et pour cela la cryptographie à seuil vient de remédier au problème d'absence de CA, le nouveau service de gestion de clé ayant la configuration (n, t+1) consiste à avoir n nœuds particuliers, qui sont des serveurs présents dans le réseau ad hoc et qui partagent la capacité de générer des certificats pour les autres membres du réseau. Chaque serveur a sa propre paire de clés, publique et privée et enregistre les clés publiques de tous les nœuds du réseau.

Dans [16], *Seung Yi et Robin Kravets* proposent de distribuer la confiance à des nœuds qui ont une forte sécurité physique et une bonne puissance de calcul par rapport aux autres et surtout dans un environnement hétérogène ces nœuds sont appelés MOCAs (MObile Certificate Authority). Dans la configuration (n, t+1), les n serveurs partagent la capacité de signer les certificats pour les autres nœuds du réseau. La clé privée k de tout le service est divisée en n secrets partagés (s1, s2, s3,..sn), un secret étant connu d'un seul serveur la figure 7 illustre cette configuration .

Chacun des serveurs génère une signature partielle du certificat du noeud et l'envoie à un Combinateur, qui a besoin de rassembler au moins t+1 signatures partielles pour générer la signature complète du noeud. Le combinateur peut aussi vérifier la validité d'une signature partielle et quand il s'aperçoit qu'elle est erronée, il la rejette et continuer la collecte de t+1 signatures valides.

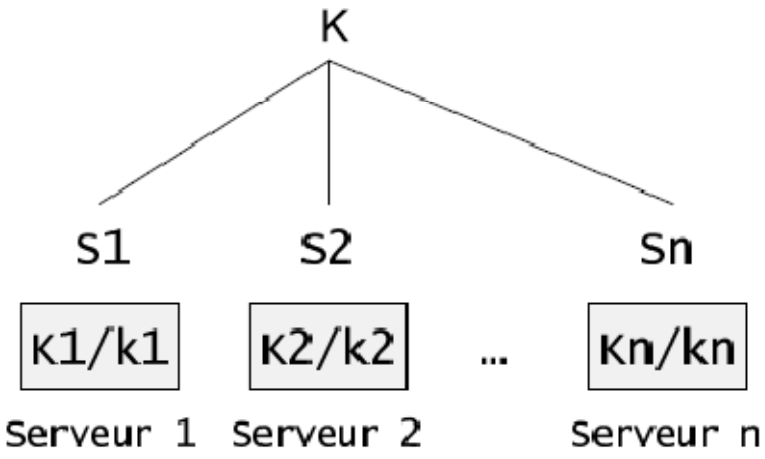


FIGURE 7: Configuration du service de gestion de clés

La figure 8 illustre bien l'opération de construction de signature. Etant donnée une configuration (3.2), le serveur 2 a été compromis et n'a pas pu envoyer sa signature partielle (SP), le combinateur va générer la signature du noeud m avec les signatures partielles s1 et s3 seulement.

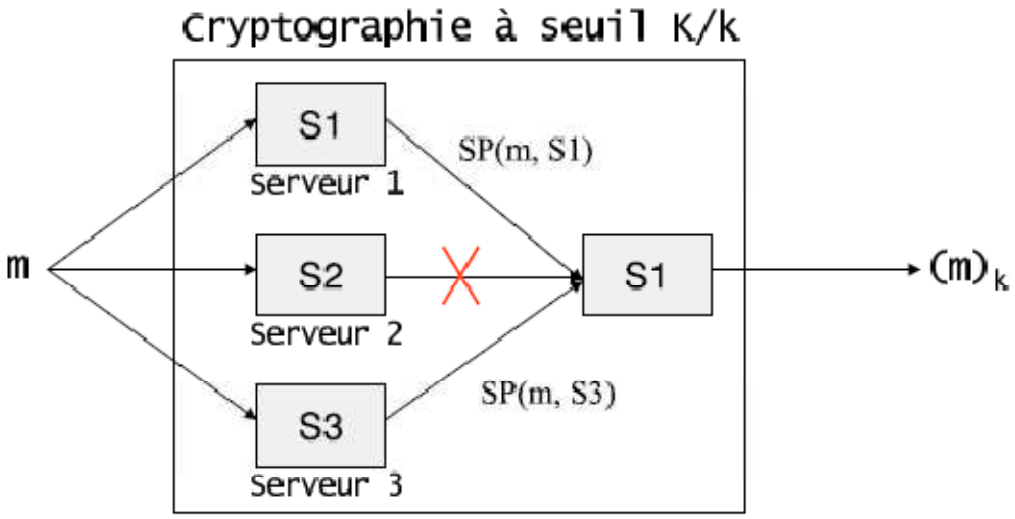


FIGURE 8: Cryptographie à seuil avec paramétrage (3,2).

Le combinateur est le responsable pour la génération des signatures pour les nœuds à son tour peut être attaqué et par conséquent devenir un point de vulnérabilité pour tous le système. *Seung Yi et Robin Kravets* présentent un protocole [16] de certification appelé MP (MOCACertification Protocol) qui n'utilise pas de combinateur et qui offrant un meilleur niveau de sécurité. Dans le protocole MP, les clients envoient des messages Send Request (SREQ) par inondation, chaque MOCA recevant ce message, répond par un message Certif Response (CREP) (protocole analogue au protocole de routage AODV) contenant une signature partielle. Quand le client collecte t CREP valides, il peut calculer sa signature.

4.6.2 L'infrastructure à clé publique auto-organisée :

Dans cette architecture proposée par [17], chaque nœud dans le réseau Ad hoc établit des certificats pour les nœuds en qu'il a fait confiance. Lorsque deux nœuds veulent communiquer sans connaissance préalable l'un de l'autre, ils s'échangent leurs listes de certificats et essayent de créer une chaîne de confiance entre eux. Par exemple, deux nœuds A et B veulent communiquer ensemble et font tous les deux confiances au nœud C, une chaine de confiance peut être établit entre A et B via C (les amis des amis sont des amis).

Les techniques de construction des bases de données locales contenants les certificats sont utilisés dans [17], d'une manière que deux nœuds du réseau ad hoc peuvent établir une chaîne

de confiance entre eux, même si la taille des bases de données locales est petite par rapport au nombre de nœuds dans le réseau. Donc le modèle de confiance peut être représenté par un graphe G(V, E), tel que V est l'ensemble des sommets (utilisateurs) et E représente l'ensemble des arcs(certificats) du graphe Ainsi, si entre deux sommets i et j existe un arc dans le graphe de confiance, c'est que le nœud i a généré un certificat pour le nœud j. L'existence d'une chaîne de confiance entre deux nœuds est ainsi représentée par un chemin direct entre les deux sommets correspondants à ces deux nœuds.

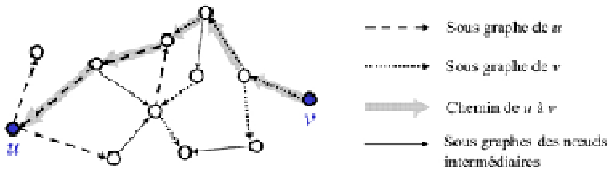


FIGURE 9: Graphe de confiance

Dans cette approche l'existence d'un chemin direct entre deux nœuds voulant communiquer n'est pas assurée, De plus, le stockage distribué des certificats engendrent un surcoût important, donc appliqué cette approche a large échelle n'est pas possible et encore des membres malicieux peuvent générer de faux certificats et les intégrer dans le graphe de confiance. la figure 9 illustre le process de l'etablissement d'une chaine de confiance entre de noeud u et v .

4.6.3 L'accord de clé (Key Agreement) dans les MANETs :

Dans cette approche un groupe de personnes dans une conférence se présentent dans une salle de réunion Ad hoc et faisant parti d'un système de chiffrement asymétrique.Ces personnes veulent s'échanger des données secrètement durant la réunion de sorte qu'aucune personne en dehors de la salle peut avoir accès à ces données. Le principe de ce protocole [18] consiste à partager un mot de passe faible (Multi-party version), à partir duquel une forte clé de session du groupe sera généré. Selon *Asokan et PHILIP Ginzboorg*, ce protocole doit avoir les propriétés suivantes:

Secret: seules les personnes qui connaissent le premier mot de passe doivent pouvoir déduire la clé de session résultante (Forward Secrecy).

Accord contribuant: la clé de session générée doit être composée des contributions des différents participants à la session.

La tolérance aux attaques d'insertion de messages dans le réseau.

Asokan et PHILIP Ginzboorg [18] présentent un protocole générique EKE (Ecrypted Key Exchange) Les intervenants sont deux nœuds A et B, qui se mettent d'accord sur une clé de session K. Le nœud A détient d'une paire de clés publique et privée (EA, DA) et il génère en plus deux valeurs aléatoires ChallengeA et SA, le nœud B génère trois valeurs aléatoires R, ChallengeB et SB. La clé de session entre A et B est K= f(S_A , S_B).

Sachant que f est une fonction à sens unique, h une fonction publique et P est le mot de passe faible de départ. Donc, pour calculer la clé K, les nœuds A et b doivent effectuer les échanges suivants:

A ——>B: A, P(E_A). Le noeud A envoie à B son identité et sa clé publique chiffré avec le mot de passe P.

B ——>A: P(E_A (R)). B envoie à A la valeur R chiffrée avec la clé publique de A, le tout chiffré avec le mot de passe P.

A——> B: R(ChallengeA , SA). A délivre à B sa contribution SA, ainsi que la valeur de ChallengeA, chiffrées par R.

B——> A: R(h(ChallengeA), ChallengeB , SB). A ce niveau, le noeud B envoie sa contribution SB à A, en plus des valeurs (h(ChallengeA)), ChallengeB) les deux entités A et B peuvent ainsi calculer la clé K= f(SA, SB).

A ——>B: R(h(ChallengeB)). Ce message permet à B de s'assurer que A détient le mot de passe faible P qu'il a utilisé pour déchiffrer R et par suite ChallengeB.

Dans [18] proposé une extension du protocole EKE pour qu'il soit un protocole multi-parties (Multi-party version), la seule contrainte est qu'un leader doit déclencher les opérations d'authentification et d'échange de messages. Ce leader ainsi constitue un point de vulnérabilité du réseau Ad hoc, mais dans cette présentation le protocole ne satisfait pas la contrainte d'accord contribuant car le leader calcule la clé de session et l'envoie à toutes les unités de réseau, ce protocole tel qu'il est présenté ne satisfait pas la contrainte de contribution des autres nœuds à la génération de la clé de session, ASOKAN ET AL ont modifie le protocole EKE pour donner un protocole multi-partie qui permet à tous les intervenants de contribuer à la génération de la clé de session K Le protocole DEFFIE-HELMANN peut être adapté pour assurer un secret partagé entre les membres du réseaux, connaissant le secret faible P, les échanges entres les membre (M1 ,M2 …Mn) pour calculer la clé secrète K= $g^{S_1S_2S_3\ldots\ldots\ldots S_n}$ sont le Si (Si est la contribution du membre Mi)

Mi ——> Mi+1 : $g^{s1s2s3\ldots\ldots sn}$ i = 1, . . . , n - 2. En séquence dans cette étape on a n-2 envois de message a la fin de cette étape Mn-1 aura reçu S1S2….Sn-1 donc il pourra calculer $\prod = g^{s1s2s3\ldots\ldots sn-1}$

Mn-1 ——> tous : $\prod = g^{s1s2s3\ldots\ldots sn-1}$ en broadcast

Mi——> Mn : P(ci), i = 1, . . . , n - 1,en parallèle, tel que $C_I = \prod^{\hat{} S_i / S_i}$ et $\hat{} S_i$ est un facteur (aveugle) aléatoirement choisi par le nœud Mi. Chaque Mi enlève de $\prod$ sa contribution par Mi et ajoute un facteur aveugle $\hat{} S_i$

Mn ——> Mi : $C_i^{\frac{S}{n}}$, en i = 1, . . . , n – 1,en parallèle, Mn décrypte (C_i), lui ajoute son S_n et l'envoie au Mi correspondant ,a ce stage ,tous les nœuds pourront calculer K= $g^{S_1S_2S_3\ldots\ldots\ldots S_n}$

Mi ——> tous : Mi, K(Mi, H(M1, M2, . . . , Mn)), pour i donné, en diffusion Ce message pour une vérification H(M1, M2, . . . , Mn)

4.6.4 Les identificateurs cryptographiques :

En utilisant des identificateurs cryptographiques [19] dans les réseaux Adhoc qui sont générés et détenus par les nœuds afin de prouver leur identités auprès des nœuds avec lesquels ils communiquent, Ces identificateurs sont uniques et on peut vérifer leur validité par des techniques cryptographiques, ce qui signifie qu'il est rarement de trouver deux entités dans le réseau aient le même identifiant. On peut citer comme identité cryptographique la CBID (Crypto Based Identifier), noté par :CBID = hmac_sha1_128(sha1(imprint), sha1(PK)). Où :

PK est la clé publique du créateur de l'identifiant.

imprint est une valeur aléatoire de 64 bits.

hmac et sha1 sont deux fonctions de hachage.

L'idée de base est qu'un nœud affirme son identité aux autres membres du réseau en prouvant qu'il détient la clé privée correspondant à sa clé publique utilisée lors de la génération de son CBID. Comme exemple, pour prouver son identité, le nœud A envoie à un nœud B le message suivant :

A————————> B: $clé-pub_A$, imprint, { $CBID_A$ } $clé-priv_A$

Ce message contient la clé publique de A " $clé-pub_A$ " et l'entier imprint utilisés pour la création de CBID de A, ainsi que le CBID ¿A de A chiffré par sa clé privée $clé-priv_A$. Pour s'assurer de l'identité de son correspondant A, le noeud B calcule le CBID du noeud A en utilisant sa clé publique (celle de A) et l'entier imprint, ensuite déchiffre, le CBIDA pré-calculée par A en utilisant la clé publique ' $clé-pub_A$ B affirme que le noeud A détient la clé privée correspondant à sa clé publique si les deux CBID sont conformes et par conséquent assure que CBIDA est la vraie identité cryptographique de A. CASTELLUCIA et AL proposent d'intégrer la propriété de CBID dans les adresses IPv6 [19] .Ces adresses de 128 bits sont construites en combinant :

Les premiers 64 bits du préfix du réseau

Les derniers 64 bits correspondent au CBID de l'hôte, noté CBA (crypto Based Address) Le CBA remplace le CBID

CBA=hmac_Sha1_64(Sha1(imprint),Sha1(PK)) L'identificateur de 128 bits est routable évitant toute autre information de routage dans les paquets, quand un nœud change de réseau il peut garder son CBA et changer son préfix de réseau.

4.6.5 La technique de Resurrecting duckling (Résurrection du caneton)

C'est une technique inspirée de la biologie,elle décrit le comportement d'un canard sortant de son œuf et reconnaissant comme sa mère le premier objet. Ce phénomène est appelé imprinting. De façon similaire, une unité reconnaît comme son propriétaire (son contrôleur) la première entité qui lui envoie une clé secrète, l'envoie est effectué directement (via un contact électrique) évitant toute opération cryptographique, mais cette technique est restreinte à un type bien spécifique d'applications et inadaptée à un large déploiement d'un réseau Ad hoc. Le contrôleur d'un équipement lui envoie de façon sécurisé via la clé secrète, toutes informations qui détermine son comportement avec les autres membres du réseau. L'application cible de cette technique détaillé dans [20] est une application médicale dans laquelle équipements sont par exemple des thermomètres détenus pas les patients et les contrôleurs sont les PDAs des médecins

4.6.6 Conclusion:

Le Réseau Ad hoc est un environnement hostile qui est soumis à des attaques de différentes types (active, passive …) et due à la nature du réseau et ses spécificités (lien sans fils, capacité limités …) la sécurité présente un véritable défis. Dans ce chapitre nous avons présenté les différentes attaques dans les réseaux Ad hoc

Nous avons aussi vu les mécanismes qui permettent d'établir la confiance mutuelle entre les membres du réseau. Pour sécuriser l'échange plusieurs tentatives montrent qu'il est difficile voir impossible de protéger un réseau ad hoc en utilisant des moyens matériels, la solution la plus efficace consiste à l'utilisation des approches cryptographiques nommées: "Clés de chiffrement de trafic" TEK, ces dernières permettent d'assurer à la fois la sécurité et la non-vulnérabilité des nœuds ainsi l'intégrité et la confidentialité des informations circulants dans le réseau, et c'est le sujet du chapitre suivant.

Deuxième partie
Problème d'échanges de clé dans les réseaux Ad hoc : Etat de l'art

La gestion des clés de chiffrement du trafic est la fonctionnalité fondamentale de toute architecture de sécurité, elle permet d'assurer le chiffrement du flux, la confidentialité de données l'authentification et le contrôle d'accès aux membres du réseau.une telle distribution doit être à la fois optimale pour éviter la surcharge du réseau, et efficace pour éviter les différentes attaques mais aussi comment calculé et distribue et mettre à jour ces clés de chiffrement de trafic.

5 La sécurité dans les communications de groupe

Chaque architecture de sécurisation de communication de groupe doit garantir des services de sécurités qui sont au nombre de cinq:

Confidentialité de données: Cette propriété signifie que seuls les membres adhérents au groupe peuvent accéder aux données émises par la source , et pour assurer cette confidentialité une clé symétrique est utilisé par la source pour chiffré les données et par la destination pour les déchiffrer. TEK (Trafic Encryption Key) clé de chiffrement de flux

Secrets futur et passé : Un membre ayant quitté le groupe ne doit plus être capable de déchiffrer les données après son départ (forward confidentiality), et de même un nouveau membre qui joint le groupe ne doit pas être capable d'accéder au données émises avant son arrivée (backward confidentiality) donc il est nécessaire de déclencher un processus de renouvellement de clé de chiffrement de trafic (TEK)ainsi que l'envoi de la clé (TEK) doit être sécurise grâce aux clés de chiffrement de clés KEKs (Key Encryption Keys)

Contrôle d'accès des membres du groupe : ce service de sécurité garantit que l'adhésion au groupe est assurée via une liste de contrôle d'accès (ACL)

Authentification de la source : cette propriété exige que les membres du groupe s'assurent de l'identité de la source à chaque fois qu'ils reçoivent des données, c'est une partie essentielle dans la confidentialité et l'intégrité des données

Authentification du groupe : les membres du groupe s'assurent que la source de données est bien adhérente au groupe,et ajoutant l'impossibilité de Collusion (Collusion freedom) : aucun ensemble d'anciens utilisateurs ne peuvent collaborer pour déduire la clé en cours d'usage.

Et pour répondre à ces besoins un processus de génération d'une nouvelle clé TEK doit être déclenché après chaque arrivée/départ d'un membre

6 Approches de distribution de clés TEKs

Plusieurs protocoles de gestion et de distributions de clés ont été proposés, ces protocoles ont été classifiés selon plusieurs critères: nombre de noeuds concernés par la clé(un groupe ou le réseau en entier) , avec ou sans contribution, mode de calcul de la clé (fonction mathématique)[20], [21] et [BCF]. Généralement, on distingue deux familles de protocoles de distribution de clés dans les réseaux ad hoc: distribution par réseau où une seule clé est partagée entre tous les noeuds, et distribution par groupe dont seulement les membres d'un groupe de noeuds peuvent partager la clé entre eux.

6.1 Distribution de clés par réseau

Dans cette approche "une seule clé par réseau" SSK (shared secret key), c'est à dire faire partager une seule clé de chiffrement de trafic par tous les membres du réseau ad hoc, cette clé est utilisée à la fois pour chiffrer les données à l'émission et pour déchiffrer ce flux au niveau de récepteurs. Dans ce type d'approche peu de protocoles ont été proposés, ces protocoles peuvent être classés en deux catégories :

protocoles sans contributions tel que *BELLOVIN ET STEVEN* [2].

protocoles avec contribution tel que *STEINER ET TSUDIK et Wainder* [1].

6.1.1 Clé sans contribution :

Le principe du protocole de clés sans contribution est qu'un participant génère une clé et la transfère aux autres noeuds du réseau ad hoc d'une manière sécurisée. Dans [2], *BELLOVIN ET STEVEN* introduisent une nouvelle combinaison de cryptosystèmes symétrique (TEK) et asymétrique (Public key), cette approche permet à deux éléments du réseau de partager un mot de passe (secret) commun pour échanger confidentiellement les données à travers un réseau non sécurisé.

L'idée de base est que deux participants Alice A et Bob B partagent un secret P, A doit choisir un secret fiable R, le chiffrer avec la clé P ,(par exemple DES) et l'envoyer (P(R)) à B, en utilisant la clé P, B déchiffre P(R) lorsqu'il l'a reçu, donc R est la clé de session que A et B vont utiliser. Cette approche est généralement fiable sauf si les secrets sont des longues séquences aléatoires.

Et pour améliorer le processus d'échange, *BELLOVIN ET STEVEN* [2] essaient d'augmenter le degré de sécurité en intégrant la cryptographie asymétrique.Donc les échanges de messages entre A et B seront les suivants:

1. A génère une paire de clé publique/privée, E_A et D_A , A chiffre la clé publique E_A (cryptage symétrique) en utilisant le mot de passe P pour produire P(E_A), A envoie à B (A,P(E_A)), ce message contient le nom de A en clair.

2. Connaissant le mot de passe P, B déchiffre ce message pour obtenir E_A , et génère un

secret aléatoire R, le chiffre par la clé E_A (cryptographie asymétrique) et rechiffre encore le résultat E_A (R) avec le mot de passe P, donc B va envoyer à A le message suivant P(E_A (R)).

3. A décrypte le message précédent P(E_A (R)) pour obtenir R, et génère un challenge unique $Challenge_A$ puis le chiffre avec R et envoie le résultat R($Challenge_A$) à B.

4. Afin d'obtenir $Challenge_A$, B décrypte le message envoyé par A avant de générer un autre challenge $Challenge_B$. Finalement, B chiffre les deux challenges avec R et envoie à A le résultat R($Challenge_A$, $Challenge_B$).

5. En décryptant le message R($Challenge_A$, $Challenge_B$), A va comparer le $Challenge_A$ avec le sien, si les deux challenges sont identiques, A envoie $Challenge_B$ chiffré par R à B.

6. Si les cinq étapes précédentes sont bien déroulées, la session va être établie avec succès, en plus la clé de session sera R.

Cette approche a des inconvénients, parce qu'elle nécessite une seule autorité d'authentification (dans ce cas c'est B) pour distribuer la clé, et le passage à l'échelle sera défficile car le nombre de messages échangés sera élevé, en plus la mise à jour de la clé est très fréquente due à la dynamique du réseau (phénomène "1 affecte n").

6.1.2 Clé avec contribution :

Les protocoles utilisent le principe de DH Multi-Party Version, où la clé de session est calculée à partir des contributions personnelles de tous les membres. En effet, tous les membres envoient leurs contributions à un noeud particulier qui s'appelle Leader, ce dernier, en temps qu'un responsable de la génération de clé, intègre toutes les contributions qu'il a reçu dans une fonction de calcul (mathématique) afin d'établir la clé de session, le calcul de la clé peut s'effectuer au niveau du Leader comme il peut être au niveau des membres [1]

$\forall$ ¿i , Mi génère une valeur aléatoire Si. (Mi les participants du réseau)

1. Tous les membres échangent leurs contributions Si entre eux, dont l'objectif est de calculer la clé de la session K telle que K= $g^{S1\,S2.\ldots\ldots Sn}$ et / g est une fonction mathématique.

2. Chaque noeud Mi , quand il reçoit le message $g^{S1\,S2\ldots Si-1}$ de son prédécesseur Mi-1, il ajoute sa contribution Si et envoie le résultat $g^{S1\,S2\ldots Si}$ à son successeur Mi+1 en séquence: Mi → Mi+1 : $g^{S1\,S2\ldots Si}$, i = 1, . . . , n - 2.

3. Le noeud Mn-1 va diffuser (Broadcast) le message Π suivant: Mn-1 → ALL : Π = $g^{S1\,S2\ldots Sn-1}$.

4. Parallèlement, chacun des noeuds Mi envoie le message P(ci) au noeud Mn: Mi → Mn : P(ci), i = 1, . . . , n - 1, tel que ci = $\prod$ $¿\hat{S}i/Si$ et $\hat{}S_i$ est un facteur (aveugle) aléatoirement choisi par le noeud Mi.

5. Le noeud Mn envoie parallèlement (diffusion) à chacun des noeuds Mi le message suivant : Mn → Mi : $\square ci^{S}_{n}$, i = 1, . . . , n - 1.

6. Finalement, Mi → ALL : Mi, K(Mi, H(M1, M2, . . . , Mn)), pour chacun des Mi.

L'avantage de ce protocole est que tous les membres participent à la génération de la clé de session , mais il ne peut pas être déployer à grande échelle parce que le nombre de messages échangés est encore important.

6.2 Distribution de clés par groupe

Dans cette approche on s'intéresse à un groupe du réseau et pas à tous le réseau, presque tous les protocoles s'inspirent du Deffie Hellman.Ces protocoles peuvent être classés en trois catégories:

- Approche sans clustérisation.
- Approche avec clustérisation.
- Approche distribue (GKA).

6.2.1 Approche sans clustérisation:

Cette approche aussi dite centralisée tous les membres de groupe partagent une clé TEK symétrique et unique. Les approches LKH [4] et OFT [22] appartiennent à cette famille et font recours à un unique serveur de clés, ce serveur est responsable de la génération, de la distribution et du renouvellement de la clé du groupe.

Dans l'approche LKH (Logical Key Hierarchy), les membres du groupe sont organisés en feuilles d'un arbre avec des noeuds internes (intermédiaires) logiques où la racine de cet arbre est la clé du groupe. Un groupe est défini par un triplet (U, K, R) dont: U représente les membres du groupe, K est constitué par l'ensemble de clés du groupe et R définit les relations entre U et K qui donnent l'ensemble de clés détenues par chaque membre.

Après un événement d'ajout ou de retrait d'un noeud dans le groupe, un processus de renouvellement de clés est déclenché, et consiste à renouveler toutes les clés allant du noeud à joindre ou à supprimer jusqu'au la racine. Plusieurs schémas de distribution de ces nouvelles clés peuvent être utilisés (orienté utilisateur, orienté clé,...), mais souffrent tous du phénomène "1 affecte n" car l'ajout ou le retrait d'un membre affecte tous les autres adhérents du groupe.
Ce principe est illustré dans la figure10 .

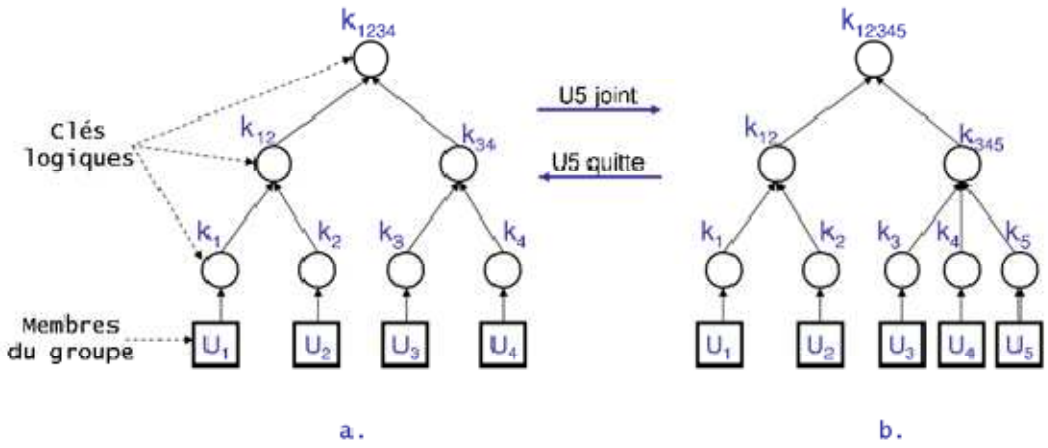


FIGURE 10: Arbre de distribution de clés dans LKH.

Plusieurs protocoles enrichissant cette classe, ont été proposés qui peuvent être divisé en deux catégories:

Protocoles avec et sans phase de pré-distribution de clés.

a)Protocoles avec phase de pré-distribution :

Dans ce type de protocole les noeuds sont configurés en pré-déployant un ensemble de clés ou KEKs (Key Encryption Key), qui lui permettent de déchiffrer le flux ou d'obtenir la TEK envoyée par la source lors d'un renouvellement de la clé. Parmi c'est protocoles nous citons le protocole GKMPAN [23, 24]. La distribution de la clé du groupe est assurée par le serveur de clés qui génère une nouvelle clé du groupe. Il la distribue ensuite, saut par saut, en utilisant les KEKs pré-déployées pour assurer le chiffrement de la clé du groupe. Ainsi, le serveur ne délivre la clé qu'à ses voisins directs (un seul saut), qui acheminent récursivement la clé à leurs voisins immédiats d'une manière sécurisée.

En l'absence d'une infrastructure fixe dans les réseaux ad hoc, qui rend difficile la disponibilité d'une entité centrale assurant l'initialisation de la gestion de clés, les protocoles de cette classe sont peu utilisés.

b)Protocoles sans phase de pré-distribution :

Cette famille de protocoles n'a pas de phase de pré-distribution de clés, parmi les protocoles qui ont été développés dans cette catégorie,le protocole de *Loukas Lazos Redha Poovendran* [25] .

Le principe de ce protocole est que les membres géographiquement proches peuvent potentiellement être atteints par un seul message de diffusion, ou utiliser le même chemin

pour accéder au flux multicast. Le réseau ad hoc est représenté par un espace à deux dimensions, en utilisant un algorithme de clustérisation K-means pour former des groupes de forte corrélation et ainsi déduire l'arbre de distribution de la clé du groupe.

Selon *Loukas Lazos Redha Poovendran* [25], la procédure de distribution de clé est composée de plusieurs étapes:

1. Affecter tous les membres du groupe à un seul cluster.
2. Diviser chaque cluster en deux sous-clusters via l'algorithme de K-means.
3. Utiliser une procédure de raffinement pour équilibrer le nombre de membres par cluster, c'est à dire affecter le même nombre de membres à chaque cluster.
4. Ré-exécuter itérativement les étapes 2 et 3 jusqu'au l'obtention des clusters contenant un ou deux membres seulement.
5. Fusionner les clusters qui contiennent un seul membre, si cela est possible, en utilisant l'algorithme de K-means.
6. Faire correspondre la hiérarchie des clusters en une hiérarchie logique (arbre) de distribution de clés LKH

La figure (Distribution de clés basée sur l'algorithme K-means) illustre l'éxecution de l'algorithme de K-means, les membres M4 et M6 sont géographiquement proches, ils sont donc frères dans l'arbre de distribution de clés LKH.

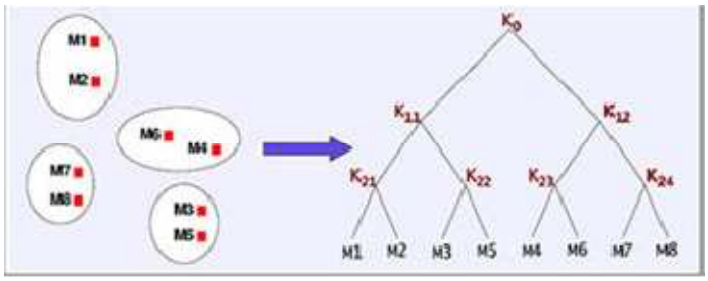


FIGURE 11: Distribution de clés basée sur l'algorithme K-means.

6.2.2 Approche avec clustérisation

L'approche avec clustérisation ou "décentralisée", divise le groupe de noeuds, désirant communiquer ensemble, en des sous-groupes ou clusters, où chacun de ces derniers est géré séparément par un contrôleur local. Les membres de chaque cluster partagent une clé de session locale (TEK locale) gérée et maintenue par leur contrôleur local.

Dans cette approche, en recevant le flux multicast envoyé par une source, les contrôleurs locaux le déchiffrent avec la clé appropriée et le rechiffrent avec les clés locales de leurs clusters et l'acheminent à leurs membres locaux. On distingue généralement deux types de clustérisation: statique et dynamique

Parmi les protocoles appartenant à cette classe on trouve celui de VARADHARAJAN ET AL [3]. Le protocole à TEKs locales de VARADHARAJAN ET AL fonctionne dans les réseaux NTDR (Near Term Digital Radio), ces derniers sont composés d'un ensemble de clusters contenant chacun un clusterhead (contrôleur local ou Leader). Un cluster est formé de noeuds à un seul niveau du clusterhead, et dont les communications inter-clusters sont restreintes aux clusterheads, qui partagent une clé (globale) symétrique de chiffrement CHGk (Clusterheads Group Key). Ainsi, tous les membres du réseau NTDR détiennent des certificats qu'ils ont reçu préalablement de la part d'une autorité de certification CA. La sélection d'un clusterhead, les fréquences et le mode de communications (intra et inter-cluster) sont bien expliqués dans [3].

La mobilité des noeuds est prise en compte lors de la formation des clusters et l'éléction des clusterheads. En effet, chaque noeud peut élire lui même en temps qu'un clusterhead s'il ne détecte aucun clusterhead dans son voisinage ou s'il reçoit deux identifiants différents pour une même partition. Afin de limiter le nombre de membres voulant devenir simultanément des clusterheads, des mécanismes de quantum de temps aléatoire sont utilisés; une fois que l'intervalle de temps aléatoire d'un noeud est expiré, il doit retester les deux conditions précédemment mentionnées avant de déclarer son nouvel état aux membres du cluster.

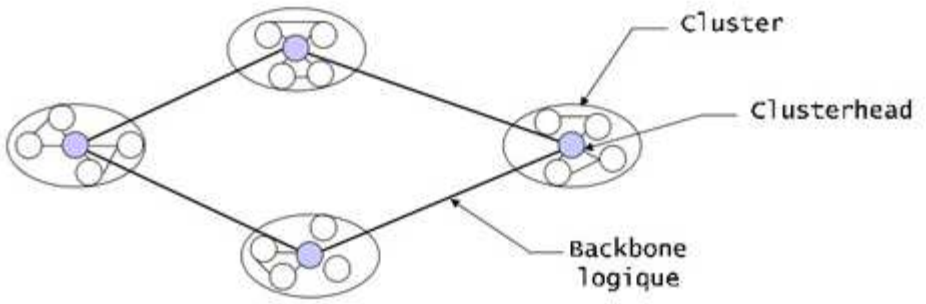


FIGURE 12: Architecture d'un réseau NTDR

Les principales fonctionnalités d'un clusterhead au sein de son cluster sont la maintenance de la liste de ses voisins, l'acceptation ou le refus de l'adhésion d'un noeud à son cluster (à travers son certificat) et l'acheminement des informations inter et intra-cluster.

La confidentialité des communications est réalisée via trois types de clés :

La clé symétrique globale CHG utilisée pour chiffrer les communications inter-clusters.

La clé locale CGK (Cluster's Group Key) du cluster, utilisée pour le chiffrement des données intra-cluster. Cette clé est une combinaison d'un secret partagé S et l'adresse IP du membre : CGK= f(S,@IP).

La clé de chiffrement de clé KEK partagée entre un clusterhead et un membre de son cluster. Le clusterhead chiffre la CGK par la KEK et l'envoie respectivement aux membres de son cluster.

Le changement de la clé locale d'un cluster n'affecte pas les autres clusters. Cependant les doubles opérations de déchiffrement et de rechiffrement constituent un inconvénient majeur.

6.2.3 Approche distribuée:

Comme son nom indique la clé de session dans cette aproche est calculée d'une manière distribuée on l'appelle aussi GKA (Group Key Agreement), tous les membres du groupe échangent entre eux des contributions personnelles, qui sont des secrets faibles. Chaque noeud doit obtenir les contributions de tous les autres membres du groupe pour calculer à son niveau la clé du groupe (secret partagé). Un leader de groupe, un noeud que les autres membres l'ont fait confiance est responsable de la mise à jour de cette clé en fonction de la dynamique du groupe (affiliation, résiliation des membres).

Les protocoles de cette famille sont dérivés à partir de protocole d'échange de clé "Two-party Diffie-Hellman Key Exchange", citant par exemple: [5], DMGSA [6].

Le protocole est composé de trois étapes:

1. Le noeud M_l , choisi comme leader de groupe, lance une requête initiale INIT avec son identité U_l et un nombre aléatoire N_l au groupe M.

2. Chaque membre M_i intéressé par l'affiliation au groupe M répond à la requête initiale INIT par un message IREPLY, contenant son identité U_i , un nombre N_i et un secret (aveugle) g^{ri} à M_l .

3. M_l collecte tous les secrets (aveugles) reçus, lève chacun d'entre eux à son secret r_l et les envoie tous avec les contributions originales au groupe, c'est à dire, il va envoyer un message IGROUP contenant { U_i , N_i , g^{ri} , g^{rirl} } pour chaque i ϵ M \ {l}. Chaque membre M_i vérifie si sa contribution était prise correctement,il obtient g^{rl} par le calcul de g^{rirl} .

Le protocole [5] utilise la procédure de signature pour vérifier qu'un message M_i , ainsi qu'un secret (N_i , g^{ri}) appartient réellement au membre désigné par l'identité U_i . En recevant une nouvelle IREPLY (nouveau membre) ou en détectant l'absence d'un ancien membre, le leader doit notifier les autres membres en envoyant un nouveau message IGROUP contenant en plus l'identité et le secret de ce nouveau membre si c'est une affiliation, ou bien sans l'identité de l'ancien membre si c'est une résiliation. Chaque membre doit recalculer immédiatement la clé après la réception d'un message IGROUP.

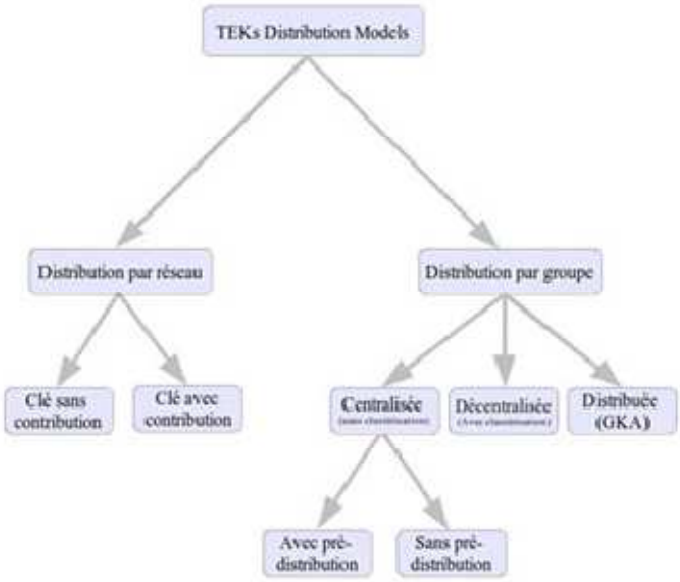


FIGURE 13: TEKs Distribution Models

6.3 Discussion :

Nous allons évaluer et comparer les protocoles de gestion de clés de groupe dans les réseaux ad hoc Et nous allons analyser leurs performances et leurs propriétés de sécurité. Les métriques de comparaison sont : les contraintes et les pré-requis des protocoles et leur application réelle, les propriétés de sécurité, le passage a l'échelle en terme du surcoût de calcul, du surcoût de stockage et de communications le tableau suivant résume cette comparaison [15d]:

			Contrainte et pré requis	Service de sécurité	Passage à l'échelle			Vulnérabilité
					Surcoût de calcul	Surcoût de stockage	Surcoût de communication	
Centralisé	Avec pré-distribution de clés	GKMPFAN [23,24]	-Pré-distribution de clés -Synchronisation	-Révocation des nœuds -Confidentialité des nœuds	-Chiffrement et déchiffrement de la TEK	N clés pré distribuée choisis parmi	Acheminement de la clé du groupe saut par saut	Serveur de clés
	Sans pré-distribution de clés	LAZOS [25]	Algorithmes K-means	-Confidentialité des nœuds	Algorithme de clustérisation k-means	Clés de l'arbre LKH	Initialisation du groupe N message de LKH	-source -Déséquilibre réseaux statiques
Décentralisé		YARAD-ARAJAN [3]	Algorithmes de clustérisation	-Confidentialité des nœuds	Données multicast déchiffré et re chiffré	Clés partagés entre clusterhead et tous les membres de son cluster	Acheminement de tous les messages via les clusterhead	Clusterhead
Distribué		DMGSA [6]	Algorithmes de clustérisation	-Confidentialité des nœuds	-Chiffrement et déchiffrement intermédiaire de la TEK -Algorithme de détection de voisinage	Clés partagées KEKS entre un GCKS et tout membre de son cluster	-Message d'initialisation de la clusterisation du groupe -message de maintenance du cluster	Clusterhead

Tableau 2 Evaluation des protocoles de gestion de clés

6.3.1 Contraintes et pré-requis :

Tous les protocoles proposés requièrent que tous les membres du groupe détiennent une clé publique, supposent la disponibilité d'une autorité de certification et de confiance dans le réseau ad hoc, capable de fournir la preuve de l'identité des membres. Cette supposition est très difficile à satisfaire dans un environnement sans infrastructure fixe.

6.3.2 Service de sécurité :

La confidentialité des données est assurée par tous les protocoles de gestion de clé par groupe présentés précédemment. Ce service est basé sur la distribution de la clé de groupe utilisée par la source du groupe pour chiffrer le flux multicast et le déchiffrer par les récepteurs. La révocation des nœuds malicieux est assurée par le processus de pré-distribution de clés de groupe

6.3.3 Passage à l'échelle :

Le passage à l'échelle est une métrique à prendre en compte lors de la conception d'un protocole de gestion de clés dans les réseaux ad hoc .Nous étudions dans ce qui suit les protocoles discutés selon trois éléments : le surcoût de calcul, le surcoût de stockage et le surcoût de communications.

6.3.4 Surcoût de calcul :

La métrique d'opérations intermédiaires de chiffrement de flux de données est très importante dans les réseaux ad hoc, par ce que la capacité de calcul est limitées des équipements du réseau. La solution de gestion de clés la plus appropriée ne doit pas utiliser des opérations intermédiaires. Les données multicast sécurisées sont déchiffrées seulement par les récepteurs finaux comme cela dans le protocole de LAZOS [25], l'inconvénient de ce protocole est qu'il est centralisé autour d'une seule entité du réseaux responsable de la génération et la distribution des clés de chiffrement ,cette centralisation autour d'un seul serveur ,accroît le phénomène 1affecte n ,qui consiste ce que le changement de l'état d'un membre affecte tous les autres . Pour atténuer ce phénomène des protocoles utilisant la clusterisation ont été proposé :

6.3.5 Surcoût de stockage :

Les réseaux ad hoc sont connu pas leur faibles capacité de mémoire et de stockage, le protocole proposé par VARADHARAJAN [3], requière un coût de stockage important dû aux opérations intermédiaires de chiffrement. Le stockage dans LAZOS [25] et LKH [4] concerne les clés de l'arbre LKH dont le nombre dépend du nombre total de membres N. Alors que GKMPAN [24] stockent les clés pré-distribuées pour chaque nœud.

6.3.6 Surcoût de communication :

Les protocoles centralisés sans pré-distribution de clés n'assurent pas le passage à l'échelle

(Phénomène 1 affecte n) Le protocole DMGSA [6] est contraint par le nombre de membres locaux par cluster, car chaque clusterhead partage avec tout membre de son sous groupe un secret partagé De plus, la maintenance distribuée des clusters requiert également des envois périodiques de messages, engendrent un surcoût de communication important.

6.3.7 Vulnérabilités et faiblesses :

Les protocoles centralisés se basent sur une seule entité du réseau ad hoc, qui gère les clés du groupe ou les certificats des membres. Cette entité centrale constitue un point de vulnérabilité en termes de sécurité. En plus cette entité peut être la cible d'attaque de type dénis de service

6.4 Conclusion:

D'après cette petite comparaison la solution la plus favorable pour assurer une communication de groupe sécurisée au sein d'un réseau ad hoc c'est l'établissement d'un protocole de gestion de clé de groupe. Ce protocole doit assurer la confidentialité des données en chiffrant le flux du coté de la source et le déchiffrer au moment de la réception avec la clé de groupe. Cependant la conception d'un protocole de gestion de clé doit prendre en considération toutes les caractéristiques du réseau ad hoc (La mobilité, ressource limitées en termes d'énergie, bande passante …ainsi l'absence d'une infrastructure fixe), Les services de sécurité assuré par un protocole de groupe dans le réseau sont lié à la nature de l'application à sécuriser. Dans cette partie nous avons vu les protocoles de gestion de clés dans les réseaux ad hoc, et nous avons vu les protocoles selon des métriques bien définies (services de sécurité assuré, contraintes et pré-requis, opération intermédiaires de chiffrement, coût de stockage, passage a l'échelle et les vulnérabilités). Nous avons vu que l'approche la plus appropriée des protocoles des gestions de clés et l'approche décentralisée, puisqu'elle ne souffre pas du phénomène 1 affecte n et ne se repose pas sur une seule entité vulnérable. Le passage à l'échelle c'est le critère le plus important pour assurer une application du protocole de gestion de groupe. Pour cela nous allons présenter une nouvelle approche décentralisé basé sur l'interpolation polynomiale, qui va assurer les services de sécurité de communication (confidentialité des données, authentification)

Troisième partie
Contribution : Une méthode d’échanges de clé pour les réseaux Ad hoc basé sur l’interpolation Polynomiale

Dans l'objectif de sécuriser les échanges de données à travers un réseau ad hoc, Particulièrement dans le protocole de routage ad hoc AOMDV qui est un protocole de routage multi chemins, et comme ce protocole ne détient pas de mécanisme permettant l'authentification et l'échanges de clés, nous proposons un modèle d'authentification et de distribution de clés basé sur l'interpolation Polynomiale à travers le protocole de routage multi chemins AOMDV, ce modèle utilise un système cryptographique symétrique et permet une authentification mutuelle des nœuds, ainsi une distribution à la volée des clés TEK. Le modèle d'authentification et de distribution de clés que nous allons proposer, sera présenté dans la section suivante, mais avant cela faisant un rappel sur le principe d'interpolation polynomiale:

7 Principe de l'interpolation polynomiale

Soit P un polynôme de degré n : $p(x)= a_0 + a_1x^1 + a_2x^2 + a_3x^3 + \ldots\ldots\ldots + a_nx^n = \sum_{i=0}^{n} a_ix^i$ (1).Résultat et théorème à exploiter dans le cadre de ce travail : Si l'on dispose de (n+1) points (x0,y0), (x1,y1),....(xn,yn), alors il existe un polynôme unique passant par tous ces points et dont on peut calculer les cœfficients de l'équation (1) (méthode itérative de Lagrange par exemple) [39]

On choisit n + 1 points distincts $x_0, x_1, \ldots, x_n$.

On calcule $y_0 = f(x_0), \ldots, y_n = f(x_n)$

On cherche un polynôme de degre n tel que $P_n(x_i) = y_i, i = 0, \ldots, n$.

Soient n+1 points distincts xi réels et n+1 réels yi, il existe un unique polynôme p $\in P_n$ tel que $p(x_i) = y_i$

pour i = 0 à n Construction de p : $p(x) = \sum_{i=0}^{n} y_i L_i(x)$

avec L_i polynôme de Lagrange $L_i(x) = \prod_{j=0, j\neq i}^{n} \frac{x - x_j}{x_i - x_j}$ / L est un polynôme d'ordre n

Propriétés de Li

· Li(xi)=1

· Li(xj)=0 (j $\neq$ i)

Exemple avec n=1

on connaît 2 points $(x_0, y_0) et (x_1, y1)$.

on cherche la droite $y= a x \square b$ (polynôme de degré 1) qui passe par les 2 points :

· $y_0 = ax_0 + b$

· $y_1 = ax_1 + b$

Tel que : $a = (y_0 - y_1) / (x_0 - x_1)$ $b = (x_0y_1 - x_1y_0) / (x_0 - x_1)$

$$y = \frac{y_0 - y_1}{x_0 - x_1} x + \frac{x_0y_1 - x_1y_0}{x_0 - Xx_1} .$$

$$y = y_0 \frac{x - x_1}{x_0 - x_1} - y_1 \frac{x - x_0}{x_0 - x_1} = y_0 L_0(x) + y_1 L_1(x) \ .$$

Exemple avec n=2

On connaît 3 points (0,1), (2,5) et (4,17)

polynômes de Lagrange associés :

$$L_0(x) = \frac{(x-2)(x-4)}{8} \ , L_1(x) = \frac{x(x-4)}{-4} \ , L_2(x) = \frac{x(x-2)}{8}$$

aprés le calcul,on trouve : $p(x) = x^2 + 1$

8 Description du nouveau protocole d'échange de clé

Le but de ce travail, est de proposer une méthode distribuée d'échange de clés public entre deux nœuds .

8.1 Principe de la méthode

Comme dans tous protocoles de communication sécurisé, avant que deux noeuds se mettent d'accord sur une session de communication entre eux, ils doivent s'authentifier l'un auprès de l'autre.

Un nœud A veut demander une clé publique à un nœud B (dans le but de lui envoyer plus tard un message chiffré par une méthode de chiffrement symétrique), et de plus A veut s'assurer que une fois la clé publique reçue, c'est bien le nœud B qui lui a envoyé cette clé (authentification de B).

8.1.1 Première phase : La phase de demande de clé de A auprès de B :

Le noeud A va envoyer de manière sécurisée, un « secret » de session au noeud B.

Le noeud A va chercher dans sa table de routage les routes vers le noeud B,

Le noeud A va choisir P nœuds d'une façon aléatoire pour assurer que les P noeuds ne peuvent pas former une coalition pour trouver l'échange de secret de session entre A et le noeud B.

On suppose que sur les p routes choisies par A vers ces différents nœuds, il y a t routes compromises (t<p). Une route est dite compromise s'il existe au moins un nœud sur cette route qui intercepte le message envoyé par A , de tel sorte que la «coalition» formée par un groupe de nœuds veulent essayer d'intercepter le « secret » se session que A va envoyer à B.

Comme il y a t routes compromises, le nœud A procédera comme suit :

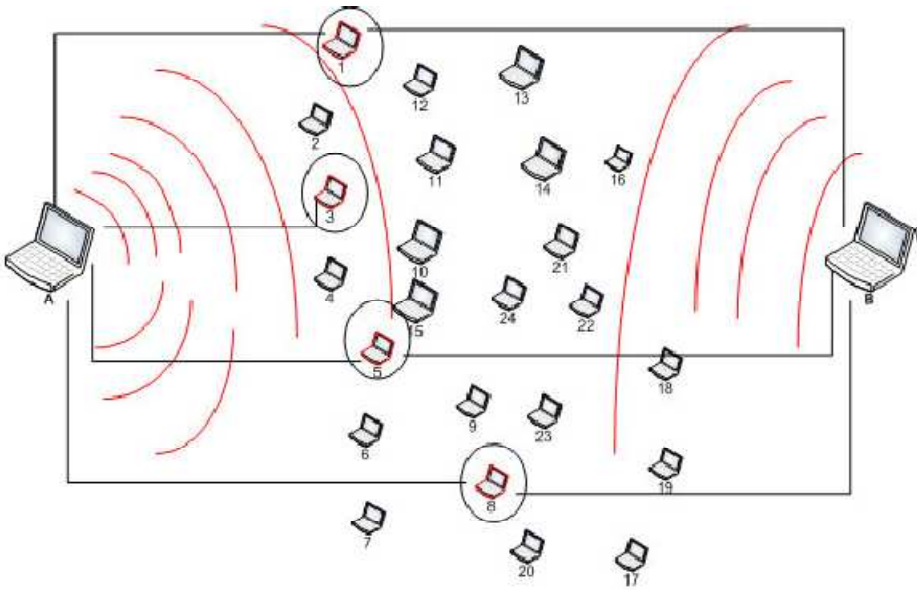
Il génère un polynôme P(x) de degré t, tel que P(0) est le « secret » de session que A va

envoyer à B, durant cette phase, il génère p points quelconques du polynôme P et les envoie le long des p routes de sa table de routage vers (N1, N2, …Np) (tel que : Ni (xi, yi)) sont des nœuds choisis aléatoirement. Les différents nœuds envoient ensuite à B ces différents points. Dès que le noeud B aura reçu (t+1) points, il reconstruira le polynôme P(x), puis calculera le «secret» de session en faisant : secret=P(0).

La génération du polynôme P(x) se fait par la methode iterative de Lagrange

8.1.2 Exemple d'une configuration

Le nœud A choisit de sa table de routage 4 routes vers B d'une façon aléatoire ,P=4 (les nœuds 1, 3, 5 et 8) Supposons qu'il ya 2 nœuds compromis donc t=2 =>le nœud A va procéder comme suit il va générer un polynôme de degré 2 donc il lui faut 3 points pour le générer (x0, y0) ,(x1, y1), (x2, y2)



B	1
4	10
B	3
11	14
B	5
B	4
B	8
7	17

Table de routage nœud A

FIGURE 14: Exemple configuration (2,4)

$p(x) = a_0 + a_1 x + a_2 x^2$

En appliquant la méthode itérative de Lagrange :

$p(x) = \Sigma_{i=0}^{n} y_i L_i(x)$ et L_i c'est le polynome de lagrange : $L_i(x) = \prod_{j=0, j\neq i}^{n} \frac{x_i - x_j}{x - x_j -}$

donc : $p(x) = y_0 L_0(x) + y_1 L_1(x) + y_{i2} L_2(x)$

Propriété de $L_i = \{L_i(x_i) = 1 \quad L_i(x_i) = 0 (i \neq j)$

Après avoir déterminer le polynôme, le secret de la session que le nœud A va transmettre vers B c'est P(0)

Le nœud A va générer des points quelconques du polynôme P et les envoie le long des p routes de sa table de routage .Ici dans l'exemple p=4 c à d (les nœuds 1, 3, 5 et 8)

Dès que le nœud B aura reçu (t+1) points (nous avons 2 nœud compromis t=2), il reconstruit le polynôme P(x), puis calcule le «secret» de session en faisant : secret=P(0),à ce niveau la le nœud B possèdera le secret de la session.

8.1.3 Deuxième phase : Envoi de la clé publique de B vers A avec authentification de B auprès de A

Le nœud B va envoyer sa clé public pubB, en utilisant une fonction de hachage (soit: MD4, MD5, SHA1).Pour envoyer une empreinte de la clé public pubB, donc le paquet chiffré construit par B sera le suivant : $E_{secret}(clépub_B, h(clépub_B))$, où :

E_{secret} est une fonction de chiffrement symétrique (DES, AES,..) paramétrée par le « secret » de session envoyé par le nœud A

$clépub_B$ est la clé publique générée par le nœud B

h est une fonction de hachage

Le nœud A déchiffre le message reçu grâce à son « secret » de session et authentifie la clé publique envoyée par B grâce à la fonction de hachage h, une fois le nœud A possèdera la clé du nœud B il peut lui envoyé un message chiffré par sa clé publique : $clépub_B$ (message).
Après un certain temps, les deux noeuds A et B doivent mettre à jour la clé de session ,En ré-établissant une nouvelle clé de session Key, les noeuds A et B ré-exécutent les étapes précédentes pour faire la mise à jour de la clé de session.

8.1.4 Les modifications apportées au protocole multi chemins AOMDV:

AOMDV permet de chercher plusieurs routes entre une source et une destination pendant une seule phase de découverte de routes, mais il n'utilise qu'une seule route pour la transmission de données :

Lorsque la source reçoit un ou plusieurs paquets RREP venant de plusieurs routes disjointes, elle décide :

Si aucun RREP n'est reçu, une nouvelle phase de découverte de route est déclenchée.

Si un seul RREP est reçu donc une seule route est reconnue entre la source et la destination, alors elle envoie les paquets de données sur cette route.

Sinon, si plusieurs RREP ont été reçu, la source choisit la meilleure route c'est-à-dire celle ayant le plus petit nombre de saut « hopcount ». Les autres routes restent en attente de l'arrivé d'un paquet RERR indiquant la rupture de la route principale, dans ce cas la meilleure route parmis les routes alternatives est sélectionnée pour retransmettre les données.

Nous avons apporté quelques modifications à cet algorithme pour que le protocole AOMDV puisse utiliser plusieurs routes pour la transmission de données entre une source et une destination

```
Si (aucune route n'est connue entre la source et la destination)
{
   initialiser une nouvelle phase de découverte de route;
}
Si (une seule route est connue)
{
   initialiser une nouvelle phase de découverte de route pour trouver plusieurs
routes ver la destination;
}
Sinon
   */ Si K routes sont connues entre la source et la destination*/
{
  Générer les paquets de données à transmettre sur les K routes afin d'assurer que t
paquets arrivent a la destination ;
```

Dans la partie qui suit nous allons donné quelques algorithmes du nouveau protocole AOMDV

8.1.5 Réception d'un message de contrôle :

Réception d'un message de contrôle (Packet_type) :

```
   /*Traitement a effectué*/
Type_packet ← Extraction du type de packet (packet_type)

   Switch (packet_type) :

         Cas 1 : AOMDV_RREQ : recvRREQ(packet);

         Cas 2 : AOMDV_RREP : : recv RREP (packet);

         Cas 3 : AOMDV_RERR : : recv RERR (packet);

         Cas4 : AOMDV_HELLO : : recv HELLO (packet);

         Cas 5 : AOMDV_SECURITY : : recvSecurity(packet);

  Default :Error invalid packet_type
```

8.1.6 Réception d'une requête (demande d'une route):

recvRREQ (Packet) :

```
/*Drop if
  -I recently heard this request
  -I'm the source
*/
Si (packet->dest=index)
Alors drop (packet)

/* If RREQ has already been received - drop it, else remember "RREQ id"   <src IP, bcast ID>. */
Vérifier_RREQ (packet)
If elle est déjà reçue
  Alors drop (packet)
Sinon
  /* création du Id de RREQ dans le cache *
  id_insert (packet->rq_src, packet->rq_bcast_id);

/* On va faire suivre REQUEST ou bien génère une réponse. On va insérer le chemin inverse dans la table de routage. */
rt0 = rtable.rt_lookup(packet ->rq_src);
if (rt0 == 0)
{
    /* Si elle n'est pas dans la table de routage, création d'une entrée */
        rt0 = rtable.rt_add(packet->rq_src);
}
Elseif (rt0->rt_seqno < rq->rq_src_seqno)
{
  path_insert()
}

/* Création /Mise à jour du chemin inverse */

rt0->path_insert(ih->saddr(), rq->rq_hop_count+1, CURRENT_TIME +
REV_ROUTE_LIFE, rq->rq_first_hop);

/* Si je suis la destination de RREQ, je vais envoyer une réponse */
if (Packet->rq_dst == index)
{
   sendReply ()
}
/* Si je ne suis pas  la destination de RREQ,  et j'ai pas une route vers la destination */
Forward(Packet)
```

8.1.7 Réception d'une réponse

```
recvRREP (Packet) :
   /*si je reçois  le message moi meme     */
    Si  (packet->dest=index)
    Alors drop (packet)

   /* Vérifier si j'ai pas une entré pour cette destination*/
   rt0 = rtable.rt_lookup (packet ->rp_dst);
    if  (rt0 == 0)
    {
        /* Si elle n'est pas dans la table de routage, création d'une entrée */
              rt0 = rtable.rt_add (packet->rp_dst);
    }
   /* Si le RREP  contient des nouvelles informations alors mise a jour de la table  */
   /* Insert forward path to RREP destination. */

    path_insert ()
/* Si je ne suis pas  la destination de RREP, et j'ai pas une route vers la destination */
Forward (Packet)
```

8.1.8 Réception des méssages des voisins

```
recvHELLO (Packet) :
   /*je cherche si mon voisin  est dans  ma table de routage*/
     Voisin = voisin_lookup (rp->rp_dst);
       if (Voisin == 0) {
             Voisin_insert (rp->rp_dst);
       }
       else {
             Voisin->mise_a_jour (experation_time)

       }
```

8.1.9 Réception des messages de sécurité

```
recvSecurity (Packet) :
  /*verifier si le packet est envoyé ou bien c'est un acusé de reception     */
   ret=packet->ret
   Si (ret=0)
   Alors /* Récupération des cordonnées X ET Y*/

    //la clé de Session
   session_key=polynome(x,y,n,VAL_POLY);
    /*polynome fonction pour calculer la clé de session */
  Le noued envoit une accusé de réception
Send(packet)
```

8.1.10 Envoit des messages de sécurité

```
SendSecurity (Packet) :
  /*preparer le packet a envoyé (Xi,Yi) sont les données */
   packet->type=AOMDV
   packet->X=Xi
   packet->Y=Yi
  packet->hach=h(Xi+Yi) /*pour vérifier l'intégrité des données*/
  Le nœud envoit une accusé de réception
Send(packet)
```

8.2 Attaques possibles

Nous mesurons l'efficacité de notre modèle, en examinant les différents types d'attaques, ainsi en évaluant les contre-mesures que nous avons pris en compte lors de la conception de notre modèle, nous conclons que:

8.2.1 Attaques actives

Pour les attaques actives, la récupération de la clé de session, donc la violation de l'identité des deux noeuds communiquant, est impossible car les $\square X_i, Y_i \square$ sont distribuées d'une manière intelligente, dont la fonction d'envoit permet d'affecter dynamiquement les $\square X_i, Y_i \square$, tel que au moins un des noeuds possédant une $\square X_i, Y_i \square$ n'appartient pas au voisinage d'un autre noeud.

Malgré l'effort, il existe deux cas où une telle attaque active sera possible:

Tous les noeuds voisins d'un noeud sont en coalition contre ce dernier.

Tous les noeuds du réseau ad hoc sont en coalition contre un noeud particulier

8.3 Conclusion:

Dans cette partie nous avons vu une nouvelle méthode de distribution de clés basée sur l'interpolation polynomiale qui à pour objectif de sécuriser les échanges de données à travers un réseau ad hoc, Particulièrement dans le protocole de routage ad hoc AOMDV.

Dans la partie suivante nous allons voir les résultats de simulation de notre protocole et l'analyse de la sécurité et la résistance de notre protocole à l'attaque Man in the middle .

Quatrième partie
Simulation du protocole et analyse de la sécurité.

Notre objectif dans cette partie c'est d'implémenter et d'interpréter les résultats des différentes simulations faites sur le protocole de routage à chemin unique AODV et un protocole de chemin multiple MSR (c'est une extension multichemin du protocole DSR) avec notre protocole AOMDV pour voir ces avantages. Cette analyse consiste à comparer et à souligner les diverses notifications apportées par le routage multi chemin du point de vue performance.

La simulation, par des simulateurs dédiés, est une méthode intéressante pour évaluer les performances d'une nouvelle approche avant l'implémentation sur une carte ou sur un autre équipement. Pour évaluer les approches nous avons choisi le simulateur de réseaux NS 2. Les simulations ont été effectuées avec la version de NS 2.34 sous le système d'exploitation Ubuntu 10.04 ,l'approche de TEK a été développée avec c++ et le langauge shell Otcl. Dans la partie qui suit, nous justifions le choix du protocole multi chemins AOMDV pour implementer notre approche de distribution de clé

9 Intérêt du routage multi chemins :

Les réseaux Ad hoc doivent être aptes à se créer et à s'organiser dynamiquement. Ces réseaux, dans lesquels les nœuds sont à la fois des terminaux et des routeurs, nécessitent la mise en œuvre de protocoles de routage spécifiques pour l'acheminement des messages [26]

Les protocoles de routage destinés à ce genre de réseaux doivent satisfaire essentiellement les critères suivants:

Opération distribuée : les protocoles doivent être entièrement distribués, fournissant la tolérance de faute et l'adaptabilité.

Pas de boucle infinie : si le chemin boucle, les paquets n'atteignent jamais leurs destinations. Pour un routage efficace, les protocoles doivent éviter les boucles infinies.

Minimum de charge de contrôle : les paquets de contrôles doivent être tenus aussi faibles que possibles, comme ils consomment la bande passante ils peuvent causer des collisions avec des paquets de données, réduisant ainsi le débit.

Adaptabilité : les protocoles doivent être capables de bien fonctionner dans un réseau de grand nombre de nœuds. Cela exige la minimisation du trafic de contrôle.

Minimisation de la charge du réseau : aucune entité n'est centralisée, les communications se font sur des multipoints fiables.

Conservation de ressources : les protocoles doivent optimiser l'utilisation des ressources rares comme la largeur de bande, la puissance de calcul, la mémoire et l'énergie.

Routes multiples : à cause des échecs de liaison fréquents surtout lorsque les noeuds sont très mobiles, les protocoles doivent fournir une multitude de routes pour garantir une

livraison maximale des paquets.

Qualité de service : les protocoles doivent être capables de fournir un certain niveau de qualité de service (QoS) surtout en terme du débit et du délai de transmission de données surtout lorsque le réseau est à forte charge comme exigé par les applications temps réel.

Répartition de la charge : les protocoles doivent être capables de répartir la charge sur tous les noeuds du réseau pour alléger le niveau de congestion.

Stabilité, robustesse et efficacité : les routes procurées doivent être stables, robustes et efficaces pour s'adapter à la forte mobilité des noeuds.

Temps de latence et nombre de saut minimum.

L'avantage du routage multi chemins et de permettre d'utiliser plusieurs routes pour la transmission des paquets de données entre une source et une destination, et de fournir d'une part une économie efficace de la bande passante, et d'autre part la livraison maximale de données en divisant le trafic de données à transmettre sur les différentes routes valables entre une paire de noeuds avec la contrainte des noeuds mobiles et de topologie dynamique.

9.1 Principe du routage multi-chemins:

Le principe du routage [27] à chemins multiples consiste à considérer plusieurs routes de la source vers une destination pour la transmission de données.

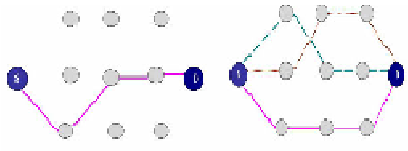


FIGURE 15: Routage multi chemins

Le routage multi-chemin est basé sur trois mécanismes : la découverte des routes, la maintenance des routes et la distribution du trafic.

La découverte des routes : le processus de recherche des routes multiples entre deux nœuds.

La maintenance des routes : le processus de recherche des routes en panne, les réparer et chercher une nouvelle route valide.

L'idée principale de la stratégie de l'allocation de trafic est de décider comment les données à transmettre vont être distribuées sur les routes disponibles entre une source et une destination dans le réseau.

9.2 Présentation du protocole AOMDV

AOMDV (Ad hoc On demand Multi-path Distance Vector), est un protocol de routage multi chemins, c'est une variante du protocole AODV dans lequel plusieurs routes disjointes sont recherchées.

9.3 Routage dans le protocole AOMDV

AOMDV se base sur le calcul des routes multiples dans la phase de découverte de route pour lutter contre les ruptures de liens[28]. En fait le but principal de la conception de ce protocole et de chercher plusieurs routes pendant une même phase de découverte de routes, mais on ne peut utiliser que la meilleure en terme de nombre de saut pour la transmission de données entre une source et une destination, les autres routes calculées ne seront utilisées que lorsque la route principale est rompue (la route devient invalide). Ce protocole est donc destiné pour les réseaux Ad Hoc où la mobilité des nœuds est importante et par conséquence la rupture des routes est fréquente.

Pour calculer les routes multiples, AOMDV utilise le maximum possible d'informations disponibles dans AODV en ajoutant un nombre supplémentaire de paquets de contrôles « overhead ». AOMDV est basé sur deux mécanismes essentiels:

Une règle de mise à jour des routes : pour maintenir des multiples routes sans les boucles de routage.

Un mécanisme distribué entre les différents noeuds du réseau: pour calculer des routes disjointes.

9.4 Modèle de constructions de routes multiples sans boucles de routage

Pour la construction des routes, AOMDV maintient la même règle de AODV . AOMDV est basé sur le principe du ‘‘advertised hopcount’’ [28]. L'advertised hopcount d'un noeud I pour une destination D représente le maximum hopcount (nombre de saut) des routes multiples disponibles pour I vers la destination D. Le maximum hopcount est considéré comme le nombre de sauts qui ne change jamais pour le même numéro de séquence.

Le protocole permet d'accepter seulement les routes alternatives ayant un hopcount inférieur à l'advertised hopcount. Cette condition est nécessaire pour garantir des routes sans boucles de routage. La figure (Structure des entrées des tables de routage de AMODV et AODV) représente la structure des tables de routage du protocole multi chemin AOMDV et celui à chemin unique AODV.

Destination		Destination
sequence number		sequence number
advertised_hop_count		hop_count
Route_list		next_hop1
next_hop1,hop_count1		experation-time
next_hop2,hop_count2		

FIGURE 16: Structure des entrées des tables de routage de AMODV et AODV

L'advertised_hopcount de AOMDV remplace le hopcount de AODV.

L'entrée route_list remplace nexthop et définit essentiellement les multiples sauts suivants des hopcounts correspondants

Dans cette partie, nous avons présenté la méthode avec laquelle AOMDV peut construire des routes multiples sans des boucles de routage. Maintenant, nous allons découvrir comment AOMDV procède-t-il pour que ces routes soient disjointes?

9.5 Recherche des routes disjointes

AOMDV permet de construire des routes multiples à liens disjoints, c'est-à-dire des routes multiples qui n'ont pas des liens en commun entre les différentes routes qui mènent d'une source vers une destination bien précise. Des modifications peuvent être mises en place dans le processus de découverte de route de AOMDV pour permettre la formation des routes à noeuds disjoints entre les noeuds intermédiaires et la source ou la destination.

AOMDV ajoute un nouveau champ appelé firsthop pour chaque paquet RREQ. Ce champ indique le premier saut (voisin de la source) qui l'a acheminé. En plus, chaque noeud maintient une liste, firsthop_list, pour chaque RREQ pour garder une trace de la liste des voisins de la source à partir desquels une copie de RREQ a été reçue. Dans les noeuds intermédiaires, les copies dupliquées de RREQ ne doivent pas être immédiatement supprimées comme dans le cas de AODV. Chaque copie est examinée pour voir si elle procure un nouveau chemin à noeuds disjoints vers la source, cette vérification est assurée grâce au champ firsthop du paquet RREQ et la liste firsthop_list du noeud. Si on s'assure que RREQ apporte un nouveau chemin, le AOMDV invoque la règle de mise à jour pour vérifier si la route inverse peut être mise en place ou non. Si la route inverse peut être mise en place et la route vers la destination est valide, le noeud intermédiaire envoie un RREP vers la source.

Comme pour les noeuds intermédiaires, la destination doit vérifier que les routes inverses peuvent être mises en place. Elle doit garantir que les liens sont disjoints, uniquement avec ces voisins. Au-delà du premier saut, le RREP suit les routes inverses qui ont été mises en place et qui sont déjà formées de noeuds disjoints. Chaque RREP arrivant à un noeud intermédiaire peut suivre plus qu'une route inverse lorsqu'une multitude de routes est déjà disponible. La destination répond à K copies de RREQ. Le paramètre k est utilisé pour

contrôler le nombre de RREP et pour éviter une RREP Storm.

10 Environnement de la simulation

Les simulations sont effectuées sous le simulateur de réseau (NS 2). Les scénarios ont été réalisés pour effectuer une utilisation d'essais, nous avons choisi comme paramètres : la mobilité des nœuds, le nombre de connexions actives dans le réseau…

10.1 Paramètres de la simulation

Dans nos simulations, l'environnement est un réseau de taille 1000 x 1000 m, dans lequel se trouvent généralement 30 à 100 nœuds, mais ce nombre est varié pour tester l'effet de la densité de nœuds sur les différents facteurs de performance à étudier le long de ce chapitre. Nous effectuons des simulations d'une durée de 300 secondes. La configuration de chaque nœud mobile doit être comme suit :

Paramètres	Valeurs
Type de la couche MAC	IEEE 802.11
Taille du buffer des noeuds	50 paquets
Modèle de propagation radio	Two Ray Ground
Couche physique	Bande passante de 2Mb/s

FIGURE 17: Paramètres de configuration d'un noeud mobile

Pour chaque scénario étudié, 4 topologies aléatoires sont générées. Elles se caractérisent par le même nombre de noeuds, le même nombre de connexions de trafic et le même scénario de mobilité. Les résultats utilisés dans l'analyse représentent la moyenne des résultats obtenus pour les 4 topologies étudiées.

De plus, il faut noter que NS 2 permet de générer aléatoirement le mouvement des noeuds, ainsi que le trafic entre ceux-ci. Pour chacune de nos simulations, nous avons donc créé des trafics et des mobilités spécifiques.

10.2 Mobilité des nœuds

Le mouvement des nœuds sur une simulation a un impact très important. En effet, la vitesse, la direction et la fréquence du mouvement ont un impact considérable sur la transmission d'information. Dans nos simulations, nous nous sommes basés sur le Random Waypoint Model, utilisé pour la première fois par Johnson et Maltz [33]. Ce modèle est devenu par la suite le standard dans la recherche sur les réseaux sans fil, il fournit des scénarios de simulation, où tous les nœuds mobiles se déplacent aléatoirement dans la surface de

simulation.

En fait, en variant les caractéristiques de la mobilité on s'attend à ce qu'ait un impact significatif sur les performances des protocoles de routage. Dans ce modèle, les noeuds choisissent une destination au hasard et s'y rendent à une vitesse comprise dans l'intervalle [0, 10],on peut aussi choisir le temps de pause qui représente le temps pendant lequel les noeuds restent immobiles, entre chaque mouvement. ET dans nos simulations, nous avons fixé la vitesse maximale à 10 m/s. Le temps de pause est quant à lui de 10 s.

Paramètre	Valeur
Temps de pause	10 s
Vitesse maximale	10 m/s

FIGURE 18: Paramètres de mobilité

10.3 Trafic entre les nœuds

Dans NS 2, il existe un générateur de trafic permettant de créer plusieurs types de trafic tel que : TCP (Transport Control Protocol), CBR (Constant Bit Rate)…

Le protocole TCP n'est pas approprié dans le cas des MANETs. TCP a été développé pour assurer la performance des réseaux fixes, pour lesquels la probabilité de perte de paquets est faible. De plus, dans ce type de réseau, la perte de paquets est principalement due à la congestion du réseau. Avec TCP, quand un paquet est perdu, le mécanisme de contrôle de congestion est initié, réduisant la vitesse d'envoi des paquets.

Le problème avec les réseaux mobiles est que les pertes de paquets sont essentiellement dues aux cassures de liens, et à la congestion. TCP interprète souvent les pertes de paquets causées par ces cassures comme la congestion, ce qui conduit à une réduction du débit et affecte ainsi les performances du réseau (en invoquant les mécanismes de contrôle de congestion quand cela n'est pas nécessaire). De nombreuses études sont faites afin d'améliorer le comportement de TCP pour les réseaux Ad Hoc. Pour cette raison, nous nous sommes basés sur le protocole CBR dont le fonctionnement est assez simple [34] : les paquets ont une taille fixe et sont envoyés à un rythme continu, l'intervalle d'envoi entre deux paquets est constant. De plus, la source d'un message n'essaie pas de savoir si son paquet a bien été reçu.

la taille d'un paquet est égale à 512 octets et la fréquence d'envoi est de 2 paquets/secondes. Le débit de chaque source est donc de : Débit = $2\times512\times8$ bit/sec $\approx$ 8 kbits/sec

Paramètre	Valeur
Type de trafic	CBR
Nombre de noeuds	30
Nombre de connexions	10
Fréquence d'envoi	2 paquets/s
la taille d'un paquet	512 octets

FIGURE 19: Paramètres de trafic.

10.4 Paramètres à évaluer

10.4.1 Taux de paquets livrés avec succès :

Ce paramètre représente le pourcentage des paquets livrés à leurs destinations par rapport aux paquets émis dans le réseau. Il se calcule de la façon suivante :

$$\text{Packet Delivery Ratio PDR} = 100 \times \frac{\sum PaquetsEmis}{\sum PaquetsRecu}$$

10.4.2 Trafic overhead

Ce paramètre nous informe sur la quantité des paquets de contrôles générés par le protocole en question pour la recherche, l'établissement et le maintien des routes:

Traffic Over Head : TOH = $\Sigma Paquets.de.controles$ $paquets$

10.4.3 Délai moyen de bout en bout

Les applications exigent une certain délai de transfert, il serai intéressant de calculer ce paramètre qui concrétise la durée moyenne pour transmettre un paquet de données de bout en bout, et il est donné par :

$$\text{Average end to end Packet Delay : APD} = \frac{\sum T_R(i) - T_S(i)}{\Sigma Paquets.Rec} \text{ (ms)}$$

Ou:

TR(i) : instant où le paquet de donnée i est reçu par l'agent de transport destination.

TS(i) : instant où le paquet de donnée i est émis par l'agent de transport source.

10.4.4 Débit total du réseau :

Certaines applications exigent un certain débit de transmission, il serai intéressant de calculer ce paramètre qui concrétise le débit du réseau total pour transmettre les paquet de données :

$$\text{Débit du réseau} = \frac{\sum Délai.de.Bout.en.Bout}{\sum Bits.Rec,us -} \quad \text{(Bits/s)}$$

11 Résultats et interprétations

Nous allons varier le temps de pause, le nombre de connexions, le nombre de nœuds et le temps de simulation afin d'évaluer leur effets sur les métriques précédemment développées.

11.1 Effet de la variation de la mobilité de nœuds dans le réseau

Le routage multi chemins offre une solution aux problèmes de rupture de liens, surtout lorsque les nœuds sont très mobiles. Il est alors nécessaire de varier le temps de pause afin d'exprimer l'effet de la mobilité sur les performances du réseau.

Débit total du réseau en fonction de la mobilité :

Dans la figure (influence de la mobilité de nœuds sur le débit total du réseau) montre une évolution croissante du débit total du réseau en fonction du temps de pause avec tous les protocoles de routage.Cette évolution est justifiée par l'augmentation des ruptures des liens à forte mobilité qui permet la dégradation des performances. Nous observons également sur cette figure que le protocole AOMDV offre le meilleur débit par rapport à tous les autres protocoles, ce protocole garantit un débit de 1700 bits/s à forte mobilité (temps de pause=0 s) et de 2700 bits/s à faible mobilité.

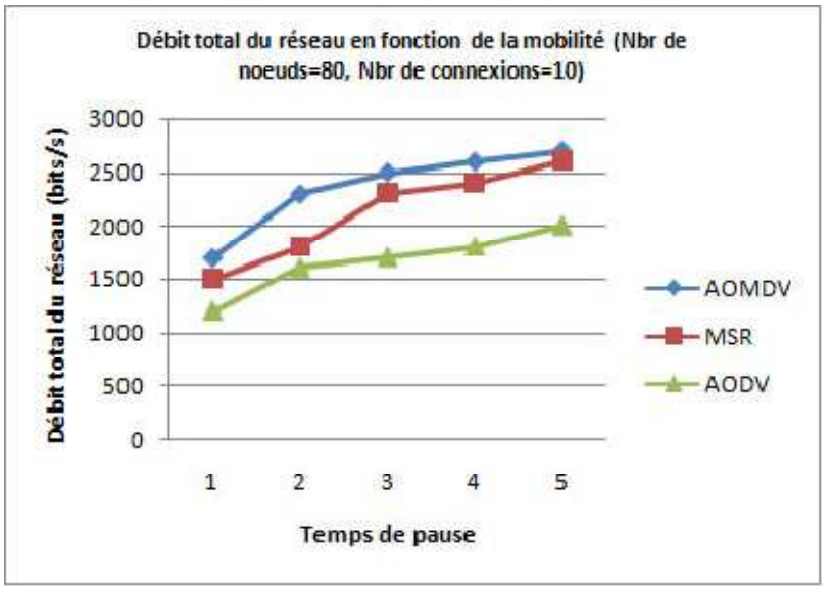


FIGURE 20: Influence de la mobilité de nœuds sur le débit total du réseau

La débit total réalisé par un protocole à chemin unique AODV et celui réalisé par le protocole multi chemins notamment AOMDV, nous aide à conclure que le routage multichemin permet d'augmenter considérablement le débit total du réseau surtout à forte mobilité grâce à l'utilisation de plusieurs routes pour l'acheminement des données entre une source et une

destination.

2.Délai moyen de bout en bout en fonction de la mobilité :

Dans la figure (influence de la mobilité sur le délai moyen de bout en bout), nous constatons une dégradation du délai moyen de bout en bout en fonction du temps de pause pour tous les protocoles de routage. Quant aux protocoles multi chemins AOMDV ou MSR, le délai moyen de bout en bout est toujours inférieur à celui réalisé par le protocole à chemin unique AODV. AOMDV est le protocole le plus performant en termes de délai moyen de bout en bout.

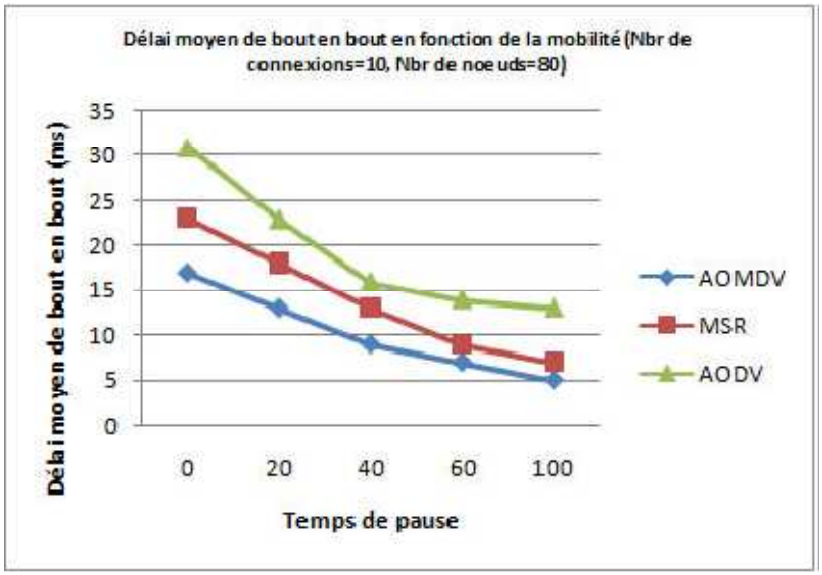


FIGURE 21: Influence de la mobilité sur le délai moyen de bout en bout

Le délai moyen de bout en bout réalisé par AOMDV est plus visible à forte mobilité du réseau, donc le protocole AOMDV est destiné essentiellement aux réseaux à forte mobilité.

3.Taux de paquets livrés avec succès en fonction de la mobilité :

Nous observons dans la figure (influence de la mobilité sur PDR) que le taux de paquets livrés avec succès pour chaque protocole augmente en fonction du temps de pause. Les protocoles de routage étudiés surtout ceux à chemin unique fonctionnent mieux à faible mobilité .(lorsque le temps de pause =100 s, le PDR atteint une valeur maximale de 97% pour AODV). Ceci est justifié par la mobilité des nœuds qui entraîne une augmentation de la fréquence des ruptures de liens, ce qui permet l'augmentation du taux de paquets perdus.

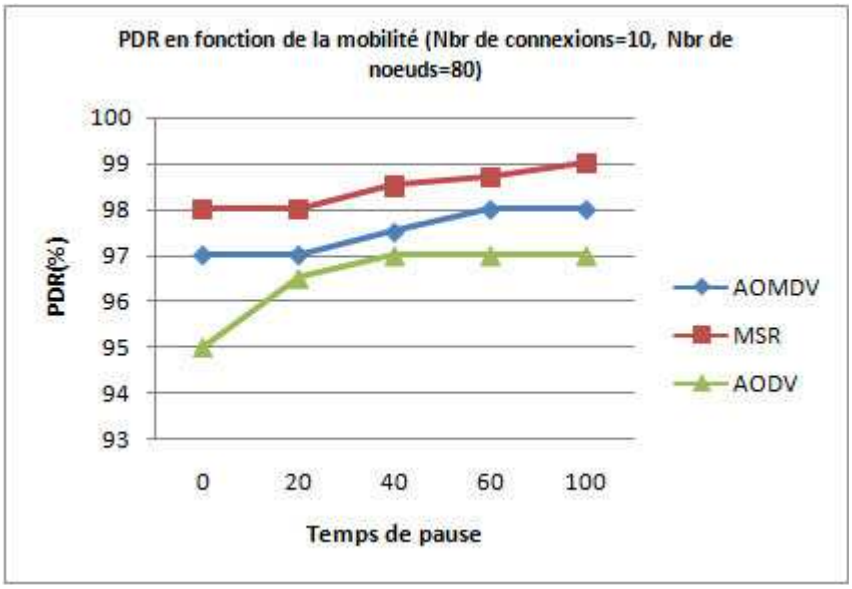


FIGURE 22: Influence de la mobilité sur PDR

Comme conclusion nous pouvons constater que les protocole AOMDV est plus performant en terme de taux de paquets livrés avec succès surtout à forte mobilité, donc le routage multi chemins offre une solution au problème de rupture de liens puisqu'il permet de diminuer le taux de paquets perdus dans le réseau. Cette diminution est expliquée par l'utilisation des routes disjointes pour la transmission des données entre une source et une destination, car une rupture d'un lien dans le réseau ne peut affecter qu'une seule route, donc seuls les paquets émisent le long du chemin défectueux peuvent être perdus.

4.Trafic Overhead en fonction de la mobilité

Dans la simulation (influence de la mobilité sur le TOH),on remarque une augmentation du trafic de contrôle en fonction de la mobilité. Pour tous les protocoles, le TOH atteint une valeur maximale à forte mobilité (temps de pause=0 s) car lorsque les nœuds sont très mobiles, le nombre de routes qui tombent en panne augmentent ce qui permet le déclenchement de plusieurs procédures de recherche de routes.

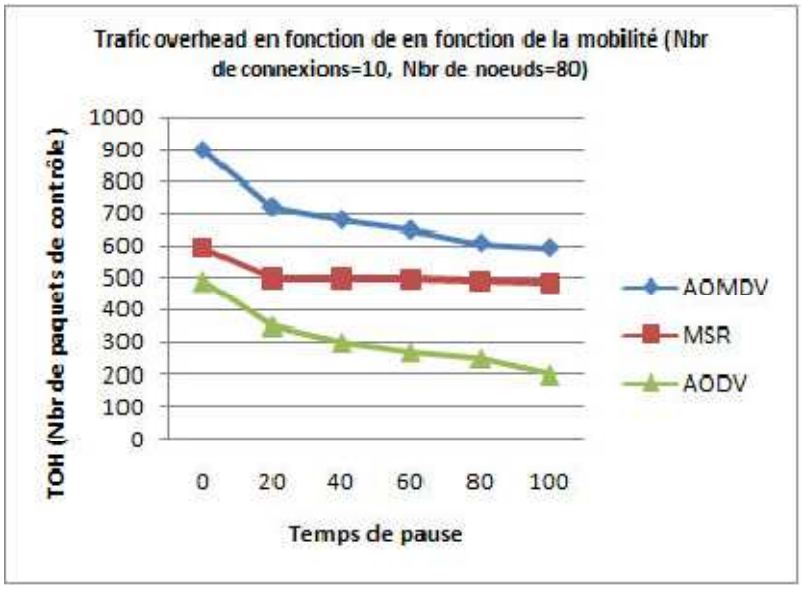


FIGURE 23: Influence de la mobilité sur le TOH

Les protocoles multi chemins génèrent plus de trafics de contrôles que le protocole à chemin unique AODV, car pour calculer les routes multiples et la quantité de trafic a acheminer dans chaque route, le nombre de paquets (RREQ et RREP) diffusés dans le réseau augmentent considérablement. AOMDV génère la quantité la plus importante de trafic overhead, ce qui permet une consommation importante de la bande passante.

11.1.1 Effet de la variation de la charge du réseau

1. Débit total du réseau en fonction de la charge du réseau :

Sur la figure (influence de la charge du réseau sur le débit total du réseau) nous observons que le débit total du réseau réalisé par tous les protocoles de routage diminu en fonction du nombre de connexions, cette diminution est expliquée par la congestion du réseau lors d'une forte charge du réseau. MSR et AOMDV améliorent considérablement le débit total du réseau par rapport au protocole AODV. AOMDV offre le meilleur débit total du réseau à forte charge du réseau. Lorsque la charge du réseau est de 20 connexions CBR, la différence entre le débit total du réseau réalisé par AOMDV et celui réalisé par chacun des autres protocoles est faible. Mais à forte charge du réseau (nombre de connexions=40), le débit offert par AOMDV est supérieur à celui réalisé par MSR,et énormément supérieur à celui réalisé par AODV. Le gain en terme du débit total du réseau offert par AOMDV est justifié par l'utilisation des routes les moins congestionnées pour la transmission des données.

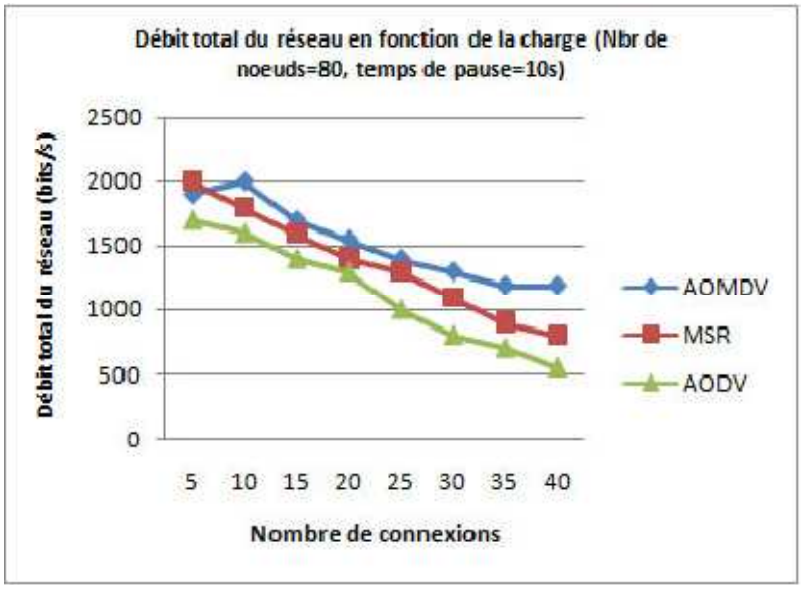


FIGURE 24: Influence de la charge du réseau sur le débit total du réseau

2. Délai moyen de bout en bout en fonction de la charge du réseau :

Le délai moyen d'un paquet de bout en bout augmente en fonction de la charge du réseau pour tous les protocoles. Nous constatons sur la figure (influence de la charge du réseau sur le délai moyen de bout en bout) que AOMDV reste le protocole le plus performant, car à forte charge du réseau (40 connexions),il offre une diminution en terme de délai moyen de bout en bout .

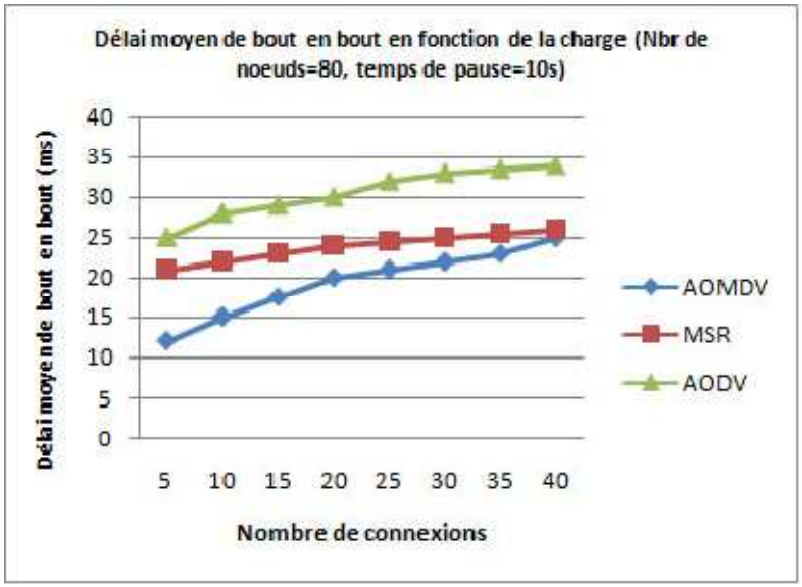


FIGURE 25: Influence de la charge du réseau sur le délai moyen de bout en bout

Le gain offert par AOMDV en terme de délai moyen de bout en bout est expliqué par l'utilisation de plusieurs routes pour la transmission de données entre une source et une

destination, et au fait que la sélection des routes dans le protocole AOMDV n'est pas basé uniquement sur le nombre de sauts comme dans les protocoles à chemin unique, mais il est basé aussi sur le niveau de congestion des nœuds intermédiaires, ce qui permet de répartir la charge sur tous les nœuds du réseau.

3. Taux de paquets livrés avec succès en fonction de la charge du réseau:

Dans la figure (influence de la charge du réseau sur le PDR) nous remarquons que le taux de paquets livrés avec succès avec tous les protocoles de routage diminue, en fonction du nombre de connexions. Cette dégradation du PDR est prévisible dans la mesure où le nombre de paquets émis est largement supérieur au nombre de paquets reçus surtout à forte charge. Or l'augmentation du taux de paquets perdus dans le réseau est due soit à la mobilité des nœuds qui a comme conséquence la rupture de routes ou bien à la congestion du réseau. Dans ce cas de figure les nœuds ne sont pas mobiles (temps de pause=10 s) alors que la charge du réseau est importante (40 connections) ce qui explique la diminution du taux de paquets livrés avec succès.

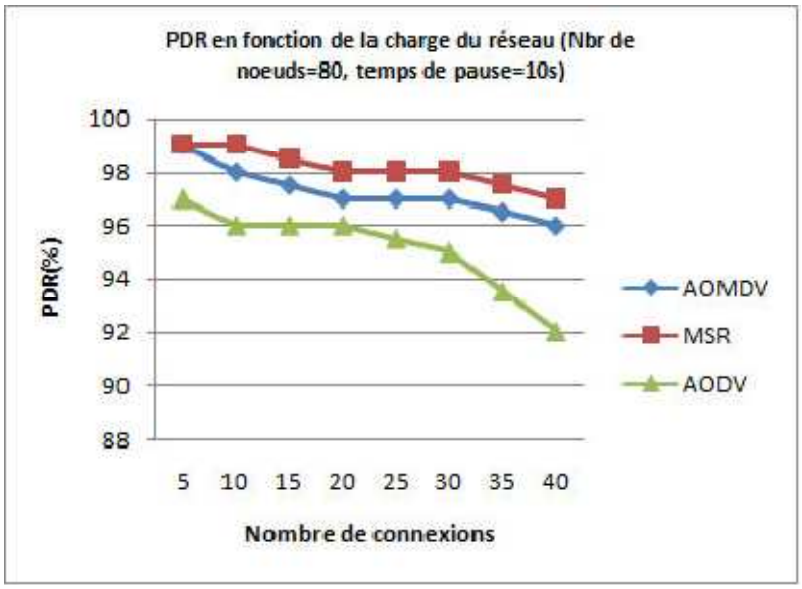


FIGURE 26: Influence de la charge du réseau sur le PDR

4. Trafic overhead en fonction de la charge du réseau :

La figure (influence de la charge du réseau sur le TOH) traduit l'évolution du trafic overhead en fonction de la charge du réseau, nous remarquons que le trafic de contrôle augmente énormément en fonction du nombre de connexions. Nous observons aussi que la différence du trafic de contrôle généré par les protocoles est trés faible lorsque le nombre de connexions vaut 5, mais cette différence augmente considérablement lorsque la charge du réseau augmente (40 connexions).Le protocole AODV est le plus performant en terme de taux de paquets de contrôles utilisés. Le trafic overhead généré par AOMDV est superieur à tous les autres protocoles ceci est justifié par l'utilisation d'un nombre plus important de paquets de

contrôles pour le calcul et le maintien des routes multiples avec un protocole multichemin.

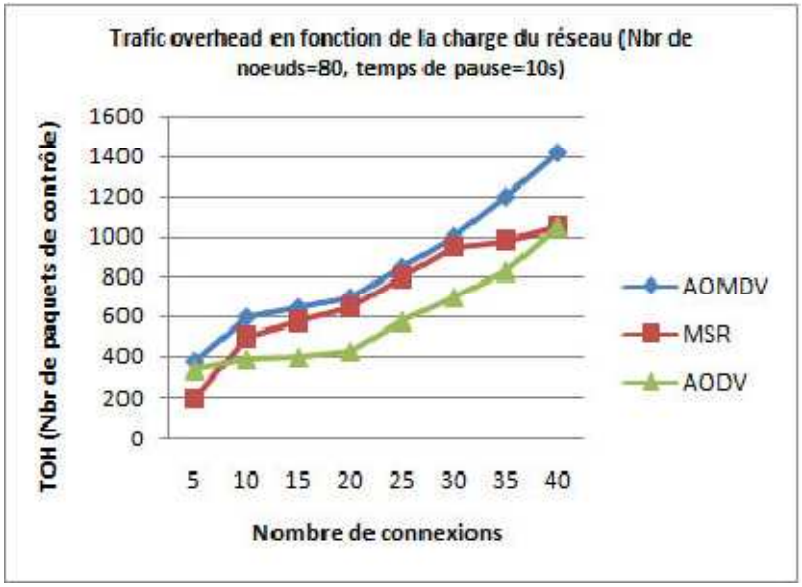


FIGURE 27: Influence de la charge du réseau sur le TOH

11.1.2 Taux de données livrés avec succès en fonction de la mobilité du réseau

Nous observons dans la figure suivante (calcule de la clé de session en fonction de temps de pause) que la clé de session est calculée à 93 %(lorsque le temps de pause =100 s) et à 80 % (lorsque le temps de pause =0 s). Ceci est justifié par la mobilité des nœuds qui entraîne une augmentation de la fréquence des ruptures de liens, ce qui permet l'augmentation du taux des données perdus.

Comme conclusion, nous pouvons constater que le protocole de gestion de clé est performant en termes de taux de paquets livrés avec succès, surtout à forte mobilité (temps de pause =0),le protocole réussi à calculé la clé de session de 80 %, donc le routage multi chemins offre une solution aux problèmes de rupture de liens, puisqu'il permet de diminuer le taux de paquets perdus dans le réseau. Cette diminution est expliquée par l'utilisation des routes disjointes pour la transmission des données entre une source et une destination.

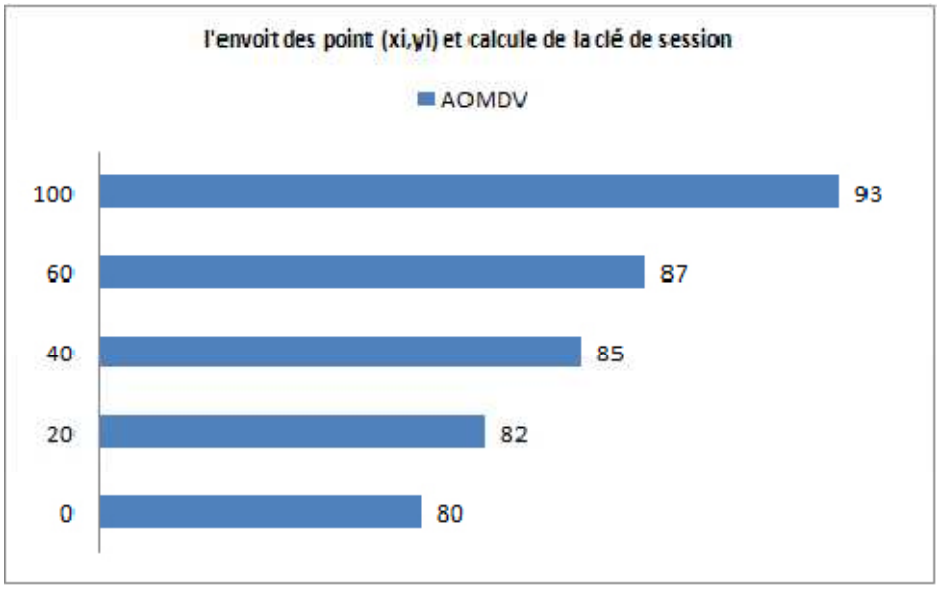


FIGURE 28: calcule de la clé de session en fonction de temps de pause

11.1.3 Taux de données livrés avec succès en fonction de la charge du réseau

Dans la figure (calcule de la clé de session en fonction de la charge du réseau), nous remarquons que le taux de paquets livrés avec succès avec la méthode déchange de clé, diminu en fonction de nombre de connexions. Cette dégradation de données livrée est prévisible dans la mesure où le nombre de paquets émis est largement supérieur au nombre de paquets reçus surtout à forte charge. Or l'augmentation de taux de paquets perdus dans le réseau est due soit à la mobilité des noeuds qui a comme conséquence la rupture de routes, ou bien à la congestion du réseau. A forte connexion 40 notre protocole arrive a calculé la clé de session à 75 %. Ce qui est un excellent résultat dans les réseau ad hoc .

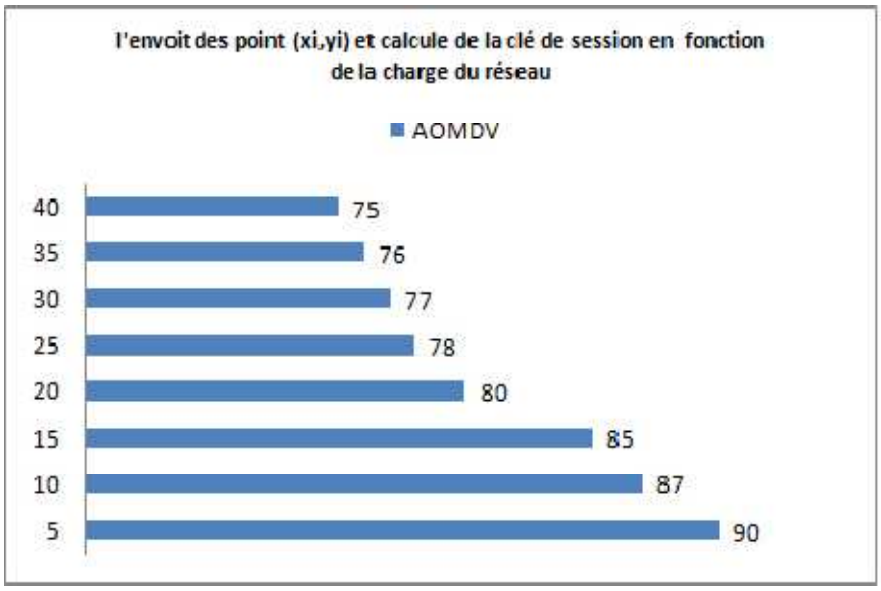


FIGURE 29: calcule de la clé de session en fonction de la charge du réseau

11.2 Résistance du protocole à l'attaque Man-In-The-Middle

11.2.1 Principe du protocole Diffie-Helman

Le protocole Deffie Hellman,(au nom de ses auteurs Whitfield Diffie et Martin Hellman) est une approche par laquelle deux noeuds du réseau peuvent se mettre d'accord sur un secret qu'ils peuvent utiliser comme clé pour chiffrer la communication et les données échangée (chiffrement symétrique) sans qu'une troisième personne puisse découvrir le secret, même en ayant écouté tous leurs échanges.

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.

a)Procédure du protocole :

Nous allons présenter les étapes du protocole [42] .Elle permet d'échanger des clés de manière sécurisée. A et B sont deux entités souhaitent envoyer un message crypte. Les données et les échanges sont les suivants :

Tout d'abord A et B se mettent d'accord sur un nombre premier p. Puis ils conviennent d'une racine primitive g

1. A choisi un nombre secret $0 \leq a \leq p - 1$.
2. A envoie la valeur $g^a mod p$ à B.
3. B choisi un nombre secret $0 \leq b \leq p - 1$
4. B envoie la valeur $g^b mod p$ à A.
5. A peut désormais calculer la clé secrète key = $(g^b mod p)^a mod p$.
6. B procède de manière analogue et obtient la même clé que A : key = $(g^a mod p)^b mod p$.

11.2.2 Analyse de la sécurité du protocole Diffie-Helman

Supposons qu'il existe un nœud E qui écoute les transmissions de A et B.Dans ce cas E n'a accés qu'aux p, g, $g^a mod p$, $g^b mod p$, on peut se demander pourquoi il n'est pas possible à E de calculer a ou b afin d'obtenir la clé secrète. Il peut, en apparence, parâitra 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 on $mod p$. Ce qui implique de calculer un logarithme discret. Or d''aprés la littérature, il n'existe pas à ce jour de solution rapide pour le calculer.

11.2.3 Principe de l'attaque Man-In-The-Middle

L'attaque de l'homme du milieu (HDM) ou man in the middle attack est une attaque qui a pour but d'intercepter les communications entre deux parties, sans que ni l'une ni l'autre ne puisse se douter que le canal de communication entre elles a été compromis. L'attaquant doit d'abord être capable d'observer et d'intercepter les messages d'une victime à l'autre. L'attaque « homme du milieu » est particulièrement applicable dans le protocole original d'échange de clés Diffie-Hellman, quand il est utilisé sans authentification.

Cette attaque repose sur l'interception de g^a et g^b , ce qui est facile puisqu'ils sont échangés en clair; l'élément g étant supposé connu par tous les attaquants. Pour retrouver les nombres a et b et ainsi casser complètement l'échange, il faudrait calculer le logarithme discret de g^a et g^b , ce qui est impossible en pratique.

Supposons que nous avons Alice et Bob veulent communiquer ensemble,mais un attaquant peut se placer entre les deux, intercepter la clé g^a envoyée par Alice et envoyer à Bob une autre clé $g^{'¿}$, se faisant passer pour Alice. De même, il peut remplacer la clé g^b envoyée par Bob à Alice par une clé $g^{'¿}$, se faisant passer pour Bob. L'attaquant communique ainsi avec Alice en utilisant la clé partagée $g^{ab'}$ et communique avec Bob en utilisant la clé partagée $g^{a'b}$, Alice et Bob croient communiquer directement. C'est ce que l'on appelle « attaque de l'homme du milieu ».

Alice et Bob croient ainsi avoir échangé une clé secrète alors qu'en réalité ils ont chacun échangé une clé secrète avec l'attaquant, l'homme du milieu

11.2.4 Notre nouveau Protocole d'échange de clé

Dans le cas de notre protocole, l'attaque de type Man-In-The-Middle ne peux réussir que si une « coalition » de nœuds se trouvent le long de tous les chemins par lesquels les points (Xi, Yi) ont été envoyés (dans ce cas p=t). Afin de renforcer la sécurité de notre protocole contre ce type de « coalition », nous proposons une extension de notre protocole qui va fonctionner de la façon suivante:

Deux nœuds A et B souhaitant s'échanger un « secret » de session », vont utiliser une méthode de type Diffie-Helman combinée à notre protocole d'échange de clé de session basé sur une interpolation polynomiale tel que décrite précédemment.

Principe du nouveau protocole:

1. Le nœud A envoie p_A points à p_A noeuds au hasard le long des p_A routes contenues dans sa table de routage.

2. Le nœud B reconstruit le polynôme et déduit le «demi-secret» envoyé par A

3. Le nœud B envoie p_B points à p_B neouds au hasard le long des p_B routes contenues dans sa table de routage.

4. Le nœud A reconstruit le polynôme et déduit le «demi-secret» envoyé par B

5. Les nœuds A et B peuvent alors construire un «secret» commun à l'aide des deux «demi-secret» échangés.

11.2.5 Analyse de la sécurité du nouveau protocole contre l'attaque Man-In-The Middle:

Dans cette partie nous allons étudier et estimer la probabilité de réussite de l'attaque de type Man-In-The-Middle, sur le protocole décrit plus haut.

Afin d'estimer cette probabilité, nous considérons ce qui suit:

Nous supposons qu'il existe une « coallition» (ou groupe de noeuds) pouvant jouer le rôle de Man-In-The-Middle (ces noeuds peuvent en effet s'échanger des informations interceptées lors des différents échanges entre les nœuds A et B). Cette «coallition» de nœuds peut se trouver le long de «tous les chemins » entre A et B (P_A routes) et entre B et A (P_B routes). On suppose que les routes communes entre les P_A routes et P_B routes sont de P_{AB} routes. On déduit donc que les nœuds en «coallition» peuvent se trouver sur $nbchemins = (P_A + P_B) - P_{AB}$ (on ne compte , en effet, qu'une seule fois les routes communes
Nous considérons aussi les paramètres suivants:

nbn: le nombre de noeuds total du réseau.

nbmoyen : le nombre de nœuds intermédiaire moyen contenus sur tout chemin reliant deux noeuds quelconques du réseau

nbchemins: le nombre de chemins moyen reliant deux nœuds quelconque du réseau.

nbm: le nombre de nœuds constituant la «coalition» de l'attaquant Man-In-The-Midde.

A l'aide de tous ces paramètres, nous déduisons que: La probabilité que pour tout chemin, aucun nœud ne soit compromis est donné par la formule suivante:

$$P = C_{(nbn-nbm,nbmoyen)} / C_{(nbn,nbmoyen)}$$ (C : calcul de combinaisons) (1)

Par conséquent, la probabilité qu'il existe au moins un noeud compromis le long de tous les chemins est de : P-1
L'attaque Man-In-The-Middle décrite plus haut pourra réussir s'il existe au moins un nœud compromis (appartenant à la «coalition) le long de tous les chemins reliant deux nœuds quelconques du réseau. On en déduite que la probabilité de réussite de l'attaque Man-In-The-Middle est donnée par: $P_{Man-In-The-Middle} = (1-P)^{nbchemins}$
Afin d'estimer la valeur de cette probabilité, nous considérons les données suivantes recueillies lors des différentes simulations réalisées sur notre protocole :

Nous avons réalisé des tests sur un réseau de nbn=80 nœuds . Nous supposons qu'il y a nbm=

23 nœuds malicieux formant une «coalition» dans le réseau (ceci correspond à un un réseau «pollué» à 28,75%, ce qui est considérable pour un réseau adhoc) . Nos tests ont permis de constater que le nombre moyen de nœuds intermédaires dans un chemin reliant deux noeuds quelconques est de nbmoyen=4 et le nombre de chemins moyens entre deux nœuds quelconques du réseau est de nbchemins=8.

En appliquant la formule de calcul donnée en (1), nous trouvons que:

$P_{Man-In-The-Muddle} = 0{,}100373.$

Ce résultat nous permet ainsi de vérifier que notre protocole résiste bien à l'attaque de type Man-In-The-Middle.

11.3 Conclusion Générale:

Les réseaux ad hoc représentent une solution partiellement exploitée dans le domaine des communications, du fait un certain nombre de contraintes rendant difficile la tâche de conception de protocoles de routage fiables (liberté totale de mouvement, absence d'infrastructure, vulnérabilité . . . etc).

Dans ce projet nous avons étudié globalement les problèmes de routages dans les réseaux Adhoc, plus précisément les problèmes liés à l'authentification et la distribution de clés TEKs .Nous avons présenté un protocole optimal pour le routage ad hoc, dans lequel on a intégré un modèle efficace d'authentification et de distribution de clés TEKs. En effet, le protocole que nous avons utilisé AOMDV est basé sur la stratégie de routage Proactif/Réactif permet aux noeuds du réseau ad hoc d'avoir une image globale et récente sur la topologie du réseau, ce qui aboutit à l'établissement des routes optimales et fiables. Du point de vue sécurité, l'authentification et l'échange de clés au sein de notre modèle sont à la charge de tout le réseau ad hoc et avec un minimum de messages échangés, donc il n' y a pas de noeuds particuliers (serveurs, leaders, ...) facilitants la tâche d'usurpation ou violation, de plus une telle récupération de la clé ou attaque ne pourrait avoir lieu.

A travers l'évaluation et la simulation, nous avons constaté que les protocoles de routage multi chemins AOMDV améliorent considérablement les performances du réseau (fiabilité, capacité du réseau et niveau de congestion) par rapport aux protocoles à chemin unique essentiellement dans un environnement à forte charge, forte densité de nœuds et à forte mobilité. Malgré les quantités importantes de trafics de contrôles générés.

Ce travail, ayant apporté des améliorations dans les domaines des réseaux ad hoc, envisage un certain nombre de perspectives tels que l'implémentation d'un système de gestion de confiance entre les membres du réseau.Ce système basé sur la notion de réputation permettra d'établir des niveaux de confiance entre les noeuds du réseau.

Références

[1] M. Steiner, G. Tsudik, and M. Waidner. Key agreement in dynamic peer groups. IEEE Transactions on Parallel and Distributed Systems, 2000.
[2] Steven M. Bellovin, Michael Merritt. Encrypted Key Exchange: Password-Based Protocols Secure Against Dictionary Attacks. AT&T Bell Laboratories Murray Hill, NJ 07974.
[3] Vijay Varadharajan, Michael Hitchens and Rajan Shankaran. Securing NTDR Ad Hoc Networks. In IASTED International Conference on Parallel and Distributed Computing and Systems 2001, Pages 593-598, Anahaim California, August 2001.
[4] C.Wong, M.Gouda, and S.Lam. Secure group communications Using Key Graphs. In ACM SIGCOMM, pages 68-79, 1998.
[5] R. Bhaskar, P. Muhlethaler, D. Augot, C. Adjih, S. Boudjit and A. Laouiti . Efficient and Dynamic Group Key Agreement in Ad hoc Networks. Rapport de Recherche n ¿ 5915 INRIA, May 2006.
[6] J. Kong, Y. Lee and M. Gerla; Distributed Multicast Group Security for Mobile Ad-hoc Networks. In IEEE Wireless communications and Networking Conference (WCNC), Las Vegas, Nevada, USA. April 2006
[7] IETF MANET Working Group. http://www.ietf.org/html.charters/manet-charter.html
[8] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. l. Cayirci; "A survey on sensor networks"; IEEE Communications Magazine, Vol. 40, No. 8, pp. 102-116, Août 2002.
[9] David B. Johnson "Routing in Ad Hoc Networks of Mobile Hosts" Computer Science Department, Carnegie Mellon University, Pittsburgh, PA 15213-3891 (Mansouri)
[10] M. Abolhasan , T. Wysocki , E. Dutkiewicz; "A review of routing protocols for mobile ad hoc networks"; Ad Hoc Networks 2, pp. 1-22, 2004
[11] V. Gayraud, L. Nuaymi, F. Dupont, S. Gombault, and B. Tharon. La sécurité dans les réseaux sans fil ad hoc. In Symposium sur la Sécurité des Téchnologies de l'Information et de la Communication SSTIC, Rennes France, June2003.
[12] Frank Stajano, Ross Anderson. «The Resurrecting Duckling:Security Issues for Ad- hoc Wireless Networks». Dans 7th International Workshop on Security Protocols, volume 1796 des Lecture Notes in Computer Science, pages 172–194. Springer-Verlag, 1999.
[13] Alexandre Poquet. «Les attaques sur le routage dans les réseaux ad hoc ». Présentation lors de la Journée scientifique de l'équipe ARMOR 2. IRISA, Rennes, le vendredi 09 février 2007.
[14] Julie Tomas « Détections de Malveillance et réaction dans les réseaux ad hoc ». Stage de fin d'étude à l'ENST Bretagne de Rennes, dans l'équipe SERES (Sécurité des réseaux et des applications réparties) du département RSM (Réseaux, Sécurité et Multimédia). France.2007
[15] Mohamed Salah BOUASSIDA. Sécurité des communications de groupe dans les réseaux ad hoc. LORIA Lorrain. décembre 2006.
[16] S. Yi , R. Kravets. Key Management for Heterogeneous Ad hoc Wireless

Networks. Report Research UIUCDCS-R-2002-2290, UILU-ENG-2002-1734, University of Illinois at Urbana- Champaign, Department of Computer Science, 1304 West Springfield Avenue, Urbana.IL 61801-2987 USA, Jully 2002.
[17] J. Hubaux, L. Buttyan, and S. Capkun. The Quest for Security in Mobile Ad hoc Networks. In ACM Symposium on mobile ad hoc networking and computing (MobiHOC), Long Beach, CA, 2001.
[18] N. Asokan and P. Ginzboorg; Key Agreement in Ad-hoc Networks. Computer Communication, 23(17): 1627-1637, February 2000.
[19] G. Montenegro and C. Castelluccia. Statistically Unique and Cryptographically Verifiable Identifiers and Address. In ISOC Network and Distributed System Security Symposium (NDSS), February 2002.
[20] F.Stajano and R Anderson.The Resurrecting Duckling;Security Issues for Ad-hoc Wireless Network .In Security Protocols,7th International Workshop Proceedins
[21] R. Bhaskar, P. Muhlethaler, D. Augot, C. Adjih, S. Boudjit and A. Laouiti . Efficient and Dynamic Group Key Agreement in Ad hoc Networks. Rapport de Recherche n ¿□ 5915 INRIA, May 2006.
[22] D.Mcgrew And A. Sherman. Key Establishment In Large Dynamic groups Using One-way Functions Trees, May 1998.
[23] S. Zhu, S. Setia, S. Xu and S. Jajodia. GKMPAN : An Efiicient Group Rekeying Scheme for Secure Multicast in Ad-Hoc Networks. In MobiQuitous, pages 41-52, 2004.
[24] S. Zhu, S. Setia, S. Xu and S. Jajodia. GKMPAN : An Efiicient Group Rekeying Scheme for Secure Multicast in Ad-Hoc Networks Technical report. February 2004.
[25] Loukas Lazos, Radha Poovendran. Energy-aware Secure Multicast Communication in Ad- hoc Networks Using Geographical Location Information. Network Security and Cryptography Laboratory , University of Washington, 2003.
[26] Hongbo Zhou, “A Survey on Routing Protocols in MANETs”, Technical Report: MSU-CSE-03-08, March 28,2003.
[27] Ignacy GAW, EDZKI, “Routage multichemin et QdS dans les réseaux Ad Hoc mobiles“, 1 mars – 31 juillet, 2004
[28] Marina,M.,Das,Samir, “On-demand multipath distance vector routing in Ad Hoc networks”. Proceedings of the International Conference for Network Procotols (ICNP) (2001)
[29] Erik Andersson, Magnus Andersson, Mattias Flodin, Peter Gardfjäll, Alexander Hellström, ”Understanding AOMDV routing in practice”, May 15, 2003
[30] D. Djenouri, L. Khelladi, and A. N. Badache. A survey of security issues in mobile ad hoc and sensor networks. Communications Surveys & Tutorials, IEEE, 7(4) :2– 28, 2005.
[31] Pietro Michiardi, Refik Molva. “Ad hoc networks security”. ST Journal of System Research, Volume 4, N ¿□ 1, Mars 2003.
[32] Alexandre Poquet. «Les attaques sur le routage dans les réseaux ad hoc ».

Présentation lors de la Journée scientifique de l'équipe ARMOR 2. IRISA, Rennes, le vendredi 09 février 2007.
[33] T. Camp, J. Boleng, and V. Davies, "A Survey of Mobility Models for Ad Hoc Network Research", Wireless Communication and Mobile Computing (WCMC), Special issue on Mobile Ad Hoc Networking Research Trends and Applications, vol. 2, no. 5, pp. 483-502, 2002.
[34] Kevin Fall &Kannan Varadhan The ns Manual March 14, 2008
[35] P. Anelli & E. Horlait : NS-2: Principes de conception et d'utilisation, Version 1.3
[36] Le routage à vecteur de distance. http://cisco.goffinet.org/s2/vecteur_distance
[38] NS par des exemples http://nile.wpi.edu/NS/
[39] Interpolation lagrangienne http://fr.wikipedia.org/wiki/Interpolation_lagrangienne
[40] NS-2 Tutorial Demokritos University of Thrace Depts of: Data Analysis & Computer Networks
[41] Mohammed Salah Bouassida ,"Sécurité des communications de groupe dans les réseaux ad hoc", Décembre 2006
[42] Le protocole Diffie-Hellman Jos R. Beuret e Gwenol Grandperrin Juin 2006
[43] Mubashir Husain Rehmani, Sidney Doria, and Mustapha Reda Senouci A Tutorial on the Implementation of Ad-hoc On Demand Distance Vector (AODV) Protocol in Network Simulator (NS-2) June 2009
[44] Valérie Gayraud, Loutfi Nuaymi, Francis Dupont , Sylvain Gombault et Bruno Tharon La Sécurité dans les Réseaux Sans Fil Ad Hoc
[45] Dr Nadjib BADACHE ,Tayeb LEMLOUMA Le Routage dans les Réseaux Mobiles Ad Hoc