Architecture hiérarchique distribuée pour sécuriser les réseaux ad hoc mobiles

Abderrezak Rachedi, Abderrahim Benslimane

Architecture Hiérarchique Distribuée pour Sécuriser les Réseaux Ad hoc Mobiles

Abderrezak Rachedi et Abderrahim Benslimane
LIA/CERI, Univesité d'Avignon
Agroparc, BP 1228
84911, Avignon, France
Email : {abderrezak.rachedi, abderrahim.benslimane}@univ-avignon.fr

***Résumé*— Dans cet article, nous présentons une nouvelle architecture distribuée pour sécuriser les réseaux ad hoc mobiles. Cette architecture est basée sur un modèle de confiance, ce dernier permet d'attribuer un niveau de confiance aux nœuds selon leurs comportements dans le réseau. Cette architecture consiste à diviser le réseau sous forme de groupes interconnectés entre eux. Chaque groupe contient au moins deux nœuds de confiance dont un seul joue le rôle du chef de groupe. L'idée principale est d'établir une infrastructure distribuée à clé publique (PKI) dans chaque groupe. La certification des clés publiques des nœuds est assurée par le chef de groupe. Pour éviter des attaques de type déni de service (DoS) au niveau des chefs de groupe, nous avons introduit un nouveau concept Dynamique Dimilitarized Zone(DDMZ). La DDMZ est formée par des nœuds sacrifiables qui possèdent un niveau de confiance élevé. Ces nœuds jouent le rôle d'intermédiaires entre le chef de groupe et les nœuds ayant un faible niveau de confiance. Les résultats de simulations montrent que cette architecture est sécurisée, stable et permet le passage à l'échelle.**

I. Introduction

Les réseaux ad hoc mobiles sont formés par deux ou plusieurs nœuds mobiles capables de communiquer entre eux via des liens sans fil et sans une infrastructure pré-deployée et aussi sans aucune unité de contrôle centralisé. Ainsi, elle constitue une topologie dynamique. Ces caractéristiques rendent les réseaux ad hoc mobiles sophistiqués et capables d'opérer dans des conditions difficiles, mais aussi vulnérables aux différents problèmes de sécurité, comme la gestion des clés de chiffrage, distribution des certificats, gestion de confiance entre les nœuds, la coopération, ... etc

Les solutions de sécurité doivent proposer certains services de base comme : l'authentification, le contrôle de l'intégrité, la confidentialité, la disponibilité et la non-répudiation. La majorité des solutions de sécurité proposées dans la littérature sont basées sur la cryptographie symétrique ou asymétrique. Mais le problème majeur de ces solutions dans l'environnement des réseaux ad hoc mobiles est la gestion et la distribution des clés de chiffrage. Proposer une seule autorité de certification (AC) pour tout le réseau n'est pas un solution souhaitable car cette conception est vulnérable aux attaques de type déni de service (DoS) sur l'AC. Le protocole ARAN [14] utilise une seul AC pour tout le réseau ; si le nœud d'AC est compromis, tout le réseau sera compromis. Cette solution n'est seulement pas souhaitable, mais n'est en plus pas adaptée à la dynamique de la topologie du réseau.

Dans cet article, nous proposons une architecture pour distribuer le rôle de l'autorité de certification sur les nœuds qui bénéficient d'un certain niveau de confiance pour la sécurité, et d'une certaine stabilité, pour optimiser la charge du réseau et augmenter la durée de vie du réseau. Pour atteindre cet objectif, nous proposons un modèle de confiance distribué pour fixer des niveaux de confiance pour chaque rôle dans le réseau. Ainsi nous proposons un algorithme d'élection distribué qui consiste à diviser le réseau sous forme des groupes, avec un nœud chef de groupe pour chaque cluster (groupe). Le rôle de l'autorité de certification est affecté au nœud chef de groupe qui doit disposer d'un certain niveau de confiance et une meilleure stabilité par rapport à ses nœuds voisins. Pour sécuriser l'autorité de certification dans chaque cluster, nous avons introduit un nouveau concept, appelé Dynamique Demilitarized Zone (DDMZ). Ce concept consiste à sélectionner des nœuds sacrifiables qui possèdent un niveau de confiance élevé et qui doivent être voisin de l'autorité de certification. La DDMZ est une zone intermédiaire entre l'AC et les nœuds qui disposent d'un faible niveau de confiance.

Le reste de l'article est organisé comme suit : Dans le deuxième paragraphe, nous discutons les solutions proposées dans la littérature, qui traitent de la distribution et de la gestion des clés dans l'environnement des réseaux ad hoc mobiles. Ensuite dans le troisième paragraphe, nous décrivons notre architecture avec le modèle de confiance et l'algorithme distribué d'élection. Puis le quatrième paragraphe consiste à évaluer l'architecture et présenter les résultats obtenus lors des simulations. Enfin, le dernière paragraphe est réservé à la conclusion et aux perspectives.

II. Positionnement bibliographique

Plusieurs travaux dans la littérature proposent des solutions aux problèmes de la sécurité dans les réseaux ad hoc mobiles. Nous nous intéressons spécialement aux modèles de confiance distribués et à la distribution du rôle de CA dans un environnement mobile.

A. Modèle de confiance distribué

Le modèle de confiance proposé par le système de cryptographie à clé publique PGP (Pretty Good Privacy) [1] est basé sur le schéma de confiance Peer-to-peer "les amis de mes amis sont mes amis". D'autres approches similaires au

système PGP, adaptées aux réseaux ad hoc mobiles existent, nous citons le système auto-organisant pour gérer les clés du chiffrage [2].

B. Distribution de l'autorité de certification

Un réseau ad hoc mobile peut être représenté par un ensemble de groupe des nœuds. Chaque groupe est représenté par un chef de groupe (leader) et des nœuds passerelles (GW) qui gèrent la communication avec des groupes voisins. Parmi plusieurs solutions de sécurité qui se base sur ce principe, nous étudions une architecture proposée par Becheler et autres [13]. Cette architecture utilise la cryptographie à seuil (thershold cryptography) avec le schéma (n,k) [5] pour distribuer le rôle de l'autorité de certification. L'idée est de distribuer la clé privée de l'autorité de certification sur les chefs de groupe. Chaque chef de groupe doit posséder un fragment de la clé privée de l'AC. L'association de (k) fragments de clé permet de générer la clé privée de l'AC. Pour certifier un nœud visiteur, ce dernier doit avoir un certain nombre de certificats (W) délivrés par des nœuds qui ont le statut de garant. Une fois les (W) certificats de garantie rassemblés, le nœud visiteur peut faire sa demande auprès d'au moins (k) chefs de groupe qui possèdent les fragments de la clé privée de l'autorité de certification. Si les (k) certificats sont réunis, alors le certificat du réseau peut être généré.

Les inconvénients de cette approche sont les suivants :

- La procédure d'admission au réseau ; cette approche n'est pas applicable car en réalité, les nœuds garants ne peuvent pas garantir des nœuds qui ne les connaissent pas, donc ils doivent avoir un minimum d'informations sur le nœud pour qu'ils puissent lui délivrer un certificat de garantie.
- La disponibilité, si le nœud visiteur réussit à avoir les (W) certificats de garantie et pour des raisons de mobilité ou de disponibilité ne peut pas avoir les (k) certificats, alors il ne peut pas être certifié.
- La charge du trafic réseau, le trafic généré par chaque nouveau nœud est au moins $2 * (W + k)$ paquets.
- Processus de fusion de plusieurs réseaux en un seul, comme chaque réseau possède sa propre clé privée de certification, mixer plusieurs clés réseau n'est pas possible. Cela nécessite de retenir une seule clé et les autres clés doivent être ignorées. Le critère de sélection de la clé dominante dépend du nombre de groupe dans chaque réseau. La clé dominante retenue est celle du réseau ayant le maximum de groupe. Ce point rend l'architecture vulnérable, car chaque nœud peut monter son propre groupe, donc si on a un ensemble de nœuds malicieux, ils peuvent former leur réseau avec un maximum de clusters, puis ils attaquent les réseaux voisins dans le but de prendre le contrôle de l'autorité de certification.
- Les critères de sélection des chefs de groupe ne sont pas pris en compte par cette architecture.
- Le renouvellement de la clé du réseau nécessite l'intervention d'un tiers de confiance pour distribuer les fragments de la clé privée sur les chefs de groupe.
- La mobilité des chefs de groupe (leaders) n'est pas prise en compte, ce qui rend l'architecture non adaptée à l'environnement des réseaux ad hoc mobiles.

Pour remédier à tous ces inconvénients, nous proposons un nouveau modèle de confiance sur lequel l'architecture doit se baser.

III. Architecture distribuée

Dans ce paragraphe, nous décrivons notre architecture distribuée. Cette dernière est composée d'un modèle de confiance sur lequel la sélection des chefs du groupe (leaders) est basée. Nous présentons aussi l'algorithme distribué d'élection des chefs de groupe et de formation des groupes.

A. Trust Model

Nous supposons qu'il existe une relation sociale entre les nœuds dans le but d'établir des relations de confiance. Aussi chaque nœud possède une paire de clés privées/publiques. Initialement, les nœuds de confiance se connaissent entre eux (l'identité et la clé publique) et ils sont considérés comme des nœuds honnêtes qui ne doivent pas générer des faux certificats. Dans notre modèle de confiance, nous définissons une métrique de confiance (Tm) dans l'intervalle $[0..1]$. Un nœud (i) possède une métrique de confiance plus élevée $(Tm(i) = 1)$, s'il est connu par d'autres nœuds de confiance et a échangé les clés via un canal sécurisé (rencontre physique) [9][2] avec un ou plusieurs nœuds de confiance. Une métrique de confiance très élevée, existe aussi si le nœud a prouvé sa coopération et son bon comportement. Si un nouveau nœud est ajouté à la liste des nœuds de confiance par un ou plusieurs nœuds de confiance, les autres nœuds doivent mettre à jour leurs listes des nœuds de confiance. Chaque nouveau nœud inconnu doit commencer par une faible métrique de confiance $(Tm = 0.1)$.

Nous définissons cinq rôles différents dans chaque cluster. Chaque rôle nécessite une valeur de métrique de confiance particulière.

1) CA_k : C'est l'autorité de certification du groupe k qui certifie les clés publiques des nœuds appartenant au même groupe. Le CA_k a une métrique de confiance plus élevé $(Tm(k) = 1)$.
2) $RA_{i,k}$: C'est l'autorité d'enregistrement du groupe k assurée par le nœud de confiance (i) avec $Tm(i) = 1$. Le but principal du RA est de protéger le CA contre les différents types d'attaques.
3) $GW_{i,j}$: C'est le nœud passerelle (g) qui assure la connection entre deux différents groupes (i) et (j). Les nœuds passerelles doivent être certifiés par au moins deux CA et leur $Tm(g) \in [0.7 - 1.0]$.
4) $MN_{i,k}$: C'est le nœud membre (i) qui appartient au groupe (k), il possède une métrique de confiance moyenne $Tm(i) \in [0.5 - 0.7]$. Le nœud (i) peut communiquer dans le groupe (k) et à l'extérieur du groupe.
5) $VN_{i,k}$: C'est le nœud visiteur (i) qui appartient au groupe (k), qui possède la plus faible métrique de

confiance. Le nœud (i) ne peut pas communiquer à l'extérieur du groupe (k).

La figure 1 ci-dessous montre le diagramme de transition d'un status à un autre. La transition entres les différents statuts est basée sur le comportement des nœuds et leur coopération, sauf pour le statut CA qui nécessite une élection entre les nœuds de confiance.

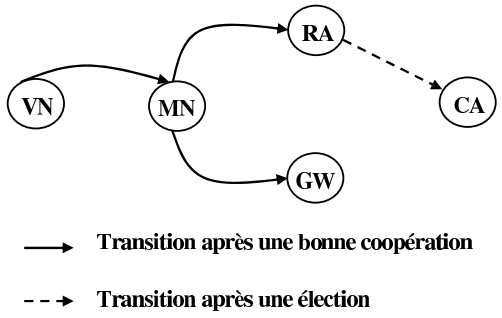

Fig. 1. Diagramme de transition d'état

1) Monitoring: Pour assurer les transitions entre les statuts, un processus de monitoring est ajouté pour superviser le comportement des nœuds. Chaque nœud avec une certaine métrique de confiance peut surveiller ses voisins qui disposent de métriques de confiance inférieures à la sienne. La figure 2 montre la possibilité des nœuds qui possèdent un certain statut de surveiller les nœuds ayant d'autres statuts inférieurs. Le

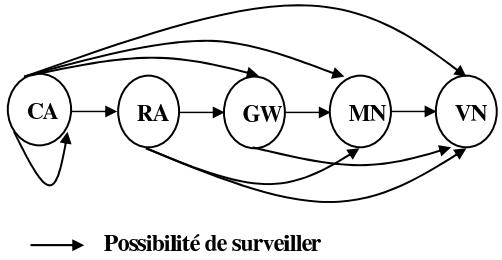

Fig. 2. Schéma de monitoring

nœud avec le statut CA peut surveiller tous les autres nœuds et même les nœuds qui possèdent le même statut que lui (CA). Les nœuds qui possèdent le statut RA peuvent surveiller les nœuds {GW, MN, VN} ; de même les nœuds avec le statut GW peuvent surveiller les statuts {MN, VN}. Enfin, les nœuds avec le statut MN ne peuvent surveiller que le statut VN mais les nœuds dont le statut est VN ne peuvent surveiller aucun statut.

Dans le modèle de confiance que nous proposons, la relation de confiance entre les clusters est assurée par les nœuds qui possèdent le statut CA. Un nœud CA peut recommander à un autre CA un noeud qui appartient à son cluster et qui possède un certain niveau de confiance. Les nœuds avec le statut RA peuvent recommander des nœuds au CA.

2) Chemin de confiance: La confiance d'un chemin dans le réseau dépend de la chaîne de confiance qui forme le chemin. Par exemple, la communication entre les clusters est basée sur l'évaluation du chemin de confiance entre les nœuds CA. L'évaluation de la confiance entre deux nœuds consiste a prendre la plus petite valeur parmi les deux métriques de confiance (ex. la confiance entre RA et GW est $min(1, w)$ telle que $w \in [0.7 - 0.9]$). La figure 3 montre deux exemples (a) et (b) pour évaluer les chaînes de confiance (TC). Nous remarquons que $TC(a) > TC(b)$, car la taille de chemin dans l'exemple (b) est plus petite que celle de (a), et aussi le fait d'avoir un nœud avec une faible métrique de confiance dans le chemin diminue le niveau de confiance du chemin.

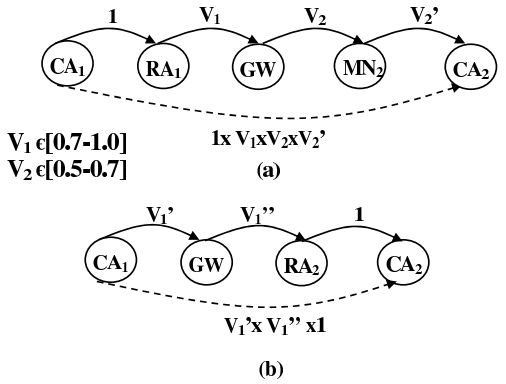

Fig. 3. Exemples des chaînes de confiance

3) DDMZ (Dynamique Demilitarized Zone): La DDMZ est définie comme une zone à un saut du nœud CA. Elle est formée par au moins un nœud de confiance et plus précisément avec un statut RA. Le but de la DDMZ est de filtrer les communications entre le nœud CA et les autres nœuds dont la métrique de confiance est faible. Tous les nœuds visiteurs doivent passer par la DDMZ pour demander leur certificat.
La figure 4 montre un exemple de DDMZ composé de deux nœuds RA {2,4}, dans un cluster de taille 2 (2 sauts). Le nœud (3) dont le statut est visiteur ne peut pas communiquer directement avec le nœud CA (1), il doit passer par la DDMZ plus exactement par le nœud (4).

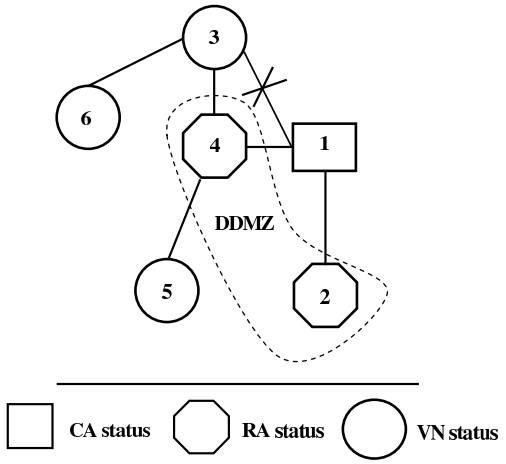

Fig. 4. Exemples d'un cluster avec la DDMZ

B. Algorithme distribué d'élection sécurisée (ADES)

L'élection des nœuds CA est assurée par l'algorithme sécurisé d'élection dont les règles principales sont :

1) Seulement les nœuds de confiance (Tm(i)=1) qui peuvent être candidats au statut CA.
2) Chaque chef de groupe est le CA d'un seul groupe.
3) Tous les nœuds de confiance voisins au nœud CA peuvent devenir RA dans le groupe.
4) Les nœuds qui appartiennent au groupe doivent être à (d) sauts du nœud CA tels que (d) est la taille du groupe à définir.

Notre algorithme est basé sur l'émission périodique des paquets balise par les nœuds de confiance vers leurs voisins à chaque période de temps pré-définie. Chaque paquet balise contient les informations nécessaires pour l'élection d'un nœud CA. La sélection d'un nœud CA est basée sur deux critères principaux, la sécurité et la stabilité.

Le paramètre de la sécurité dépend de la métrique de confiance, uniquement les nœuds (i) avec $Tm(i) = 1$ et au moins un nœud de confiance comme voisin direct qui peuvent se présenter comme candidats pour devenir un CA dans un groupe. Cette condition est nécessaire pour la formation des groupes. Pour renforcer la sécurité et augmenter la disponibilité de la DDMZ du groupe, l'algorithme sélectionne le candidat avec un nombre maximum de voisins de confiance, cela indique aussi le degré de confiance dans le groupe.

Le paramètre de la stabilité est très important pour la formation des groupes, ce paramètre est défini comme la durée de vie d'un groupe. Plusieurs stratégies sont utilisées par des algorithmes proposés dans la littérature, comme : Lowesr-ID [6], l'idée consiste à sélectionner le nœud dont l'identité est la plus petite. Connectivity maximal (Max-connectivity) permet de sélectionner le nœud dont le degré de connection est le plus élevé [10]. Dans notre algorithme, nous avons adopté la métrique de mobilité comme paramètre de stabilité [3], car cette métrique donne des bons résultats comparée à Lowest-ID et Max-connectivity, jusqu'a 33% de réduction du nombre de changements des chefs de groupe.

La métrique de mobilité est basée sur le niveau de puissance du signal à la réception sur chaque nœud (RxPr), c'est un indicatif de distance relative entre les nœuds émetteurs et récepteurs. Le ratio RxPr entre les transmissions de deux paquets successifs, donne une connaissance sur la mobilité relative entre deux nœuds voisins X et Y [3].

$$RM_Y^{\mathrm{rel}}(X) = 10\log_{10}\frac{RxPr_{X\to Y}^{\mathrm{new}}}{RxPr_{X\to Y}^{\mathrm{old}}} \qquad (1)$$

Le calcul de la mobilité relative d'un nœud Y par rapport à ses (m) voisins, consiste à calculer la variance de l'ensemble de mobilité relative RM_Y^{rel} de ses voisins X_i

$$RM_Y = \mathrm{var}(RM_Y^{\mathrm{rel}}(X_1), RM_Y^{\mathrm{rel}}(X_2), \cdots, RM_Y^{\mathrm{rel}}(X_m)) \qquad (2)$$

Une faible valeur de RM_Y indique que Y est moins mobile par rapport à ses voisins. Par contre, une grand valeur de RM_Y montre que le nœud Y est très mobile par rapport à ses voisins.

Chaque nœud de confiance candidat à l'élection pour le rôle de CA, transmet son paquet balise d'élection qui contient les informations suivantes :

- ID du candidat : l'identité du nœud candidat au rôle de CA.
- Hop-Count : nombre de sauts vers le nœud CA.
- DTN : Degré de confiance, c'est le nombre de nœuds de confiance voisins au nœud candidat.
- RM : la mobilité relative, pour indiquer la stabilité du nœud candidat par rapport à ses voisins.
- ID-num : c'est le numéro de séquence du beacon qui est incrémenté par un à chaque nouveau paquet balise transmis par le candidat.
- MAC (Message Authenticated Code) : pour authentifier le paquet balise et aussi pour vérifier l'intégrité de ses informations. Le nœud candidat doit utiliser sa clé privée pour générer le MAC du paquet balise.

$$(MAC_{K-}[CA, Hopcount, DTN, RM, ID-num])$$

Initialement, chaque nœud de confiance envoie deux paquets "hello" successivement pour calculer la mobilité relative RM. Puis, il annonce sa candidature au rôle de CA, cela par la génération de son propre paquet balise d'élection. Quand les nœuds de confiance reçoivent des paquets balise de la part de leurs voisins, ils effectuent notre algorithme d'élection et de formation de groupe pour définir leur statut : CA (leader de groupe), RA ou juste membre du groupe. La décision dépend des paramètres de sécurité et de stabilité. Lorsqu'il y a compétition entre deux candidats, le nœud avec le nombre de nœuds de confiance voisins le plus petit et avec la mobilité relative la plus élevée perd la compétition et devient soit RA soit un MN, cela dépend du nombre de sauts par rapport au nœud qui a gagné l'élection. Si c'est un nœud situé entre deux groupes adjacents alors il peut devenir passerelle (GW).
L'algorithme 1 ci-dessous est exécuté par chaque nœud de confiance (Tm=1) à la réception d'un paquet balise dont le nombre de sauts est inférieur à (d) (taille du cluster).

Dans le but de détecter le changement de topologie, nous proposons l'algorithme 2. Le déplacement du nœud CA est détecté par ses voisins de confiance, si les nœuds RA ne reçoivent pas les paquets balise pendant un temps pré-défini, cela implique que le nœud CA n'est plus disponible. Aussi, les nœuds du groupe peuvent détecter la mobilité des nœuds RA, cela par la non réception des paquet balise en provenance de ces nœuds. La mobilité des nœuds CA et RA est très importante pour la durée de vie du groupe et sa stabilité.
Chaque nœud appartenant au groupe avec un statut autre que RA ou CA doit recevoir les paquet balise venant du nœud CA à chaque période de temps pré-définie. Il doit vérifier l'authentification et l'intégrité de l'information du paquet balise par l'utilisation de la clé publique du CA ($K_{CA}+$). Si la vérification est réussie alors le nœud récepteur met à jour les changements à propos du nombre de saut vers le CA (hop-count) ou un nouveau RA.

La figure 5 montre le résultat d'un exemple d'élection et de division d'un réseau sous forme de groupes dont la taille est

Algorithm 1: Algorithme d'élection

Quand le nœud (j) reçoit un paquet balise du nœud (i);
begin
  Authentication do **if** *($Tm(i)! = 1$)* **then**
  $\mathbf{RejectBeacon}()$; $\mathbf{Goto}(\mathbf{end})$;
  else if *($HopCount >= d$)* **then**
    $\mathbf{No-Competition}$; $\mathbf{Goto}(\mathbf{end})$;
  else if *($RM_i < RM_j$) OR (($RM_i == RM_j$) AND ($DTN_j < DTN_i$))* **then**
    Accepter le nœud (i) comme CA;
    if *($HopCount == 1$)* **then**
      $Status(j) = RA$;
      $HopCount(i) = 1$;
    else
      $HopCount(i) = HopCount + 1$;
      $Status(j) = MN$;
  else if *($RM_j < RM_i$) OR ($DTN_j > DTN_i$)* **then**
    Le nœud (j) rest candidat au CA;
  else if *($RM_i == RM_j$) AND ($DTN_j == DTN_i$)* **then**
    Exécuter Lowest-ID;
end

Algorithm 2: Algorithme executé par le nœud si ses RA ou CA ne sont plus disponible

Si le nœud (i) ne reçoit pas de paquet balise de CA après certain temps pré-définie;
begin
  if *Il peut atteindre CA avec un autre RA* **then**
  Garder le CA actuel;
  Mettre à jour le nœud RA et $Hopcount$;
  else if *Il peut trouver un autre CA* **then**
    Joindre le nouveau CA;
    if *($Tm(i) == 1$)* **then**
      if *($HopCount == 1$)* **then**
        $Status(i) = RA\_NODE$;
        $HopCount(newCA) = 1$;
      else
        $Status(i) = MN$;
        $HopCount(newCA) = HopCount + 1$;
    else
      Demande de certification au nœud RA;
end

de deux sauts. Les nœuds 2, 4, 5, 7, 8 et 3 sont des nœuds d'un groupe dont le chef est le nœud 1 qui joue le rôle d'un CA. Les nœuds 3,6,10 et 11 sont des nœuds d'un groupe dont le CA est le nœud 9. Les nœuds 2 et 4 sont des nœuds de confiance avec le statut RA, et ils sont à un seul saut de CA. C'est aussi le cas du nœud 6. Le nœud 3 appartient au deux clusters, il peut devenir un nœud passerelle s'il possède une certaine métrique de confiance et aussi, il doit être certifié par les deux CA. Les nœuds visiteurs comme 5, 8, 7, 10 et 11 ne peuvent pas communiquer directement avec le CA, malgré le fait que les nœuds 5 et 10 sont des voisins de CA.

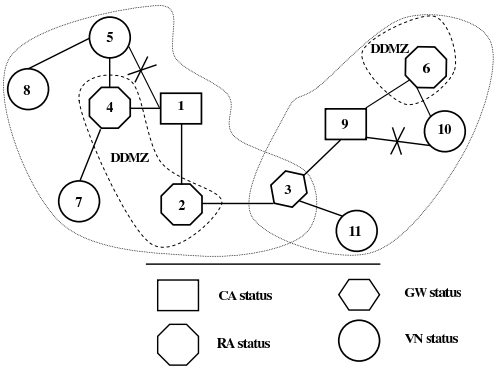

Fig. 5. Un exemple de formation des clusters à deux sauts

IV. Evaluation de performance

Nous avons intégré nos algorithmes décrits précédemment dans le simulateur réseau (NS-2) [15]. Pour générer le modèle de mobilité, nous avons utilisé CMU pour simuler nos algorithmes. Les scénarios de simulation sont générés avec les paramètres cités dans le tableau I. Le déplacement des nœuds est généré aléatoirement (Random waypoint) et de manière continue pendant les simulations.

TABLE I
Paramètres de simulation

Parameter	Valeurs
Nombre des nœuds (N)	50
Taille de la surface (mxn)	$670x670m2$
Vitesse de mobilité	20 m/sec
Portée de transmission	10 m - 125 m
Interval de diffusion (BI)	0.75-1.25 s
Interval de découverte	10*BI s
Période de contention	3.0 s
Temps de simulation	300 s

La figure 6 montre la comparaison entre notre algorithme et deux autres algorithmes : MOBIC [3] et Lowest-ID [6]. Nous remarquons une grande différence au niveau de la portée de transmission à 50 m, cela est dû à notre condition de formation de groupe (clusters), un nœud de confiance tout seul ne peut pas former son propre groupe, il doit avoir au moins un nœud voisin de confiance. Dans cette simulation, le nombre de groupes formés ne doit pas dépasser 25 groupes. Cependant, avec la portée de transmission entre 50 et 125 m, le nombre de groupes diminue et lorsque la porte de transmission dépasse les 150 m, le réseau devient plus stable et le nombre de groupes devient plus ou moins stable. Avec des groupes de taille égale à 2 sauts, nous obtenons moins de groupes que dans le cas de MOBIC [3] et Lowest-ID [6].

La figure 7 montre le nombre moyen de différents statuts des

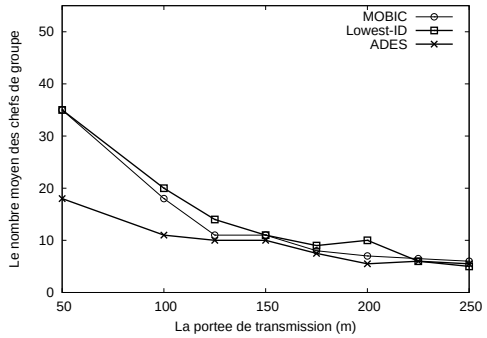

Fig. 6. Comparaison entre les algorithmes de formation des clusters

nœuds dans le réseau. Les nœuds isolés sont des nœuds qui ne peuvent joindre aucun groupe. Nous remarquons que le nombre moyen des nœuds isolés diminue lorsque la portée de transmission diminue. Les autres statuts de nœuds augmentent quand nous augmentons la portée de transmission. Pour avoir une meilleure configuration afin de sécuriser le réseau, nous devons diminuer le nombre de nœuds isolés, car plus le nombre des nœuds isolés est grand, moins le réseau est sécurisé.

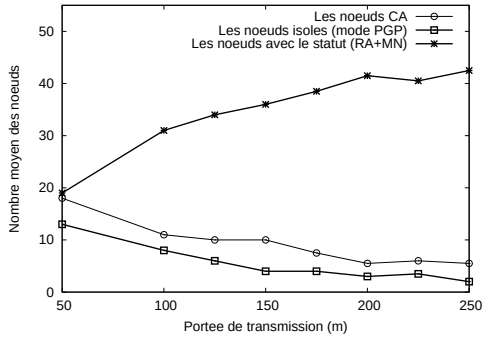

Fig. 7. Nombre moyen de différents statuts des nœuds

V. Discussion et analyse

La sécurité de l'architecture qu'on propose dépend principalement du modèle de confiance. La présence d'un grand nombre de nœuds de confiance augmente le niveau de sécurité du réseau. Les nœuds avec une faible métrique de confiance ne peuvent pas participer à l'élection du nœud CA. Seuls les nœuds de confiance peuvent être candidats au rôle de CA. Si un nœud malicieux tente de s'introduire dans le processus d'élection, cela soit par l'annonce de sa candidature soit par la manipulation non autorisée de l'information des paquets balise d'élection, les nœuds de confiance le détectent au niveau de la phase d'authentification dans l'algorithme 1. Supposons que les nœuds malicieux ont réussi à former leurs groupes et qu'ils tentent de communiquer avec d'autres groupes, les nœuds CA des groupes de destination authentifient le nœud CA du groupe source, enfin, selon le résultat de l'authentification et aprés l'évaluation de la métrique de confiance, les nœuds CA décident d'accepter ou de rejecter la communication.

L'attaque de type déni de service (DoS) sur le nœud CA est évitée par la DDMZ. Cette dernière consiste à filter toutes les requêtes venant des nœuds de faible niveau de sécurité. La robustesse de la DDMZ dépend du nombre de nœuds RA qui collaborent entre eux dans le but de protéger le nœud CA. Si un nœud malicieux tente d'usurper l'identité des nœuds CA ou RA, il sera détecté et isolé par le processus de monitoring. Un nœud malicieux peut usurper l'identité d'un nœud de confiance légitime, s'il réussit à avoir sa clé privée. Pour qu'un nœud malicieux réussisse à compromettre tout le réseau, il doit compromettre tous les nœuds CA.

Le nombre de groupes formés par notre approche est en relation avec le nombre de nœuds de confiance ainsi que leur mobilité. Si nous avons K nœuds de confiance, le nombre maximum de groupes sera $K/2$ si K est un nombre pair et $(K-1)/2$ si K est impair. La taille de groupe doit être adaptée au nombre de nœuds de confiance pour mieux sécuriser le nœud CA. La présence de deux nœuds de confiance est la configuration minimale pour la formation d'un groupe, mais cette configuration doit suivre le nombre de nœuds de confiance et les autres.

Avec notre architecture, nous pouvons utiliser la cryptographie à seuil dans chaque groupe après l'élection du nœud CA. Ce dernier divise sa clé privée à (n) fragments de clé et l'association de (k) clés génere la clé privée du CA.

L'approche de notre architecture oblige les nœuds à collaborer et à adopter un bon comportement pour l'obtention d'un niveau de confiance plus élevé. Chaque nœud inconnu doit commencer avec le statut visiteur dont le niveau de confiance est le plus bas.

Dans le but de calculer le niveau de confiance de la procédure d'authentification entre les groupes, nous calculons la qualité d'authentification (QoA). Pour cela, nous appliquons un facteur d'atténuation à la chaîne de confiance [7] [4]. Ce facteur est $(1-p)^{d-1}$, avec (p) probabilité d'existence de nœud compromis ou malicieux et (d) longueur de la chaîne de confiance.

$$QoA(V_1 - V_2) = TC(V_1 - V_2) * (1-p)^{(d-1)} \qquad (3)$$

Plus la chaîne de confiance est longue, plus le risque d'être compromis est important. Donc, la taille du groupe doit être choisie avec prudence.

La QoA entre deux groupes dépend de la chaîne de confiance (TC) qui relie les deux CAs des groupes et aussi le pourcentage de présence des nœuds malicieux dans le réseau. La communication entre les nœuds CAs doit passer par les chaînes de confiance dont le niveau de confiance est plus élevé.

La figure 8 ci-dessous décrit la qualité d'authentification (QoA) en fonction de la probabilité des nœuds malicieux. Nous avons dessiné les courbes dans le cas d'un groupe de taille 1 ou 2 saut(s) avec un maximum et minimum de valeurs de chaîne de confiance 1 et 0.49 (0.7*0.7) respectivement. Nous remarquons, dans le cas d'un groupe à 1 saut, que la QoA décroît linéairement lorsque la probabilité de présence de nœuds malicieux est importante. Dans le cas d'un groupe à 2 sauts, nous remarquons que la QoA diminue rapidement

comparée au cas d'un groupe à 1 saut. La figure 9 montre

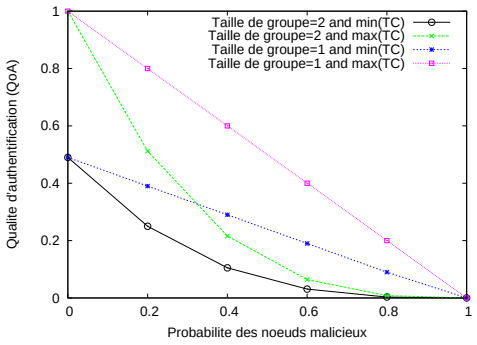

Fig. 8. QoA vs. probabilité des nœuds malicieux

le cas général de la QoA avec différentes valeurs de TC et la probabilité des nœuds malicieux. Nous comparons les trois cas de taille de groupe 1, 2 et 3 sauts. Nous remarquons que la meilleure valeur de la QoA est dans le cas d'un groupe avec un seul saut et avec une faible probabilité des nœuds malicieux avec une chaîne de confiance (TC) plus élevée.

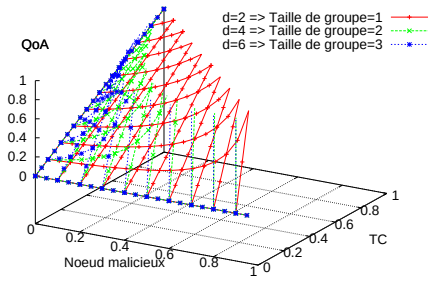

Fig. 9. QoA vs. probabilité des nœuds malicieux et TC

D'après les deux dernières figures 8 et 9, nous pouvons conclure que, plus la taille du groupe est grande, plus le risque d'avoir un QoA est faible.

VI. Conclusion

Dans cet article, nous avons proposé une nouvelle architecture distribuée basée sur un modèle de confiance et un algorithme d'élection et de formation de groupes, dans le but de distribuer l'autorité de certification (CA).

L'algorithme d'élection de formation des groupes et d'élection de CA est basé sur deux paramètres : la sécurité et la stabilité. La sécurité est un paramètre lié au modèle de confiance, seuls les nœuds de confiance qui peuvent jouer le rôle de CAs. La stabilité est un facteur basé sur la métrique de mobilité pour assurer la stabilité des groupes. Dans notre approche, le modèle de confiance est évalué par le processus de surveillance (monitoring), qui permet aux nœuds avec un niveau de confiance plus élevé de surveiller les nœuds dont le niveau de confiance est moins élevé. En plus, nous avons proposé un nouveau mécanisme DDMZ pour protéger les nœuds CAs contre les attaques de types déni de service. Ce mécanisme augmente la robustesse de sécurité dans les groupes.

Notre architecture assure la sécurité et la disponibilité de l'authentification dans chaque groupe. Cette architecture est adaptée au changement dynamique de topologie du réseau.

Les résultats de la simulation montrent que l'algorithme que nous avons proposé pour la formation des groupes est mieux que les algorithmes proposés dans MOBIC [3] et Lowest-ID [6]. Nous avons aussi remarqué que la disponibilité et la robustesse de la DDMZ dépend de la portée de transmission et du nombre de nœuds de confiance, ainsi que de leur mobilité. La stabilité des groupes permet de conserver l'énergie et d'augmenter la durée de vie du réseau.

Pour les perspectives de ce travail, nous allons étudier et analyser notre architecture dans différents modèles de mobilité et aussi évaluer la résistance de la DDMZ face aux différents types d'attaques comme le déni de service (DoS).

References

[1] Philip R. Zimmermann : The official PGP user's guide.MIT Press Cambridge. "MA, USA. (1995)

[2] S. Capkun and L. Buttyan and J. Hubaux : Self-Organized Public-Key Management for Mobile Ad Hoc Networks. ACM International Workshop on Wireless Security, WiSe. **2** (2002) 52–64

[3] P. Basu and N. Khan and T. Little : A mobility based metric for clustering in mobile ad hoc networks. In Proceedings of Distributed Computing Systems Workshop. (2001) 43–51

[4] A. Rachedi and A. Benslimane : A Hiearchical Distributed Architecture to Secure Ad-Hoc Networks. MSN Int. Conf., 13-15 Dec., Honk-Kong, China (2006).

[5] A. Shamir *How to share a secret*, ACM Comm. Vol. 22, No. 11 1979

[6] M. Gerla and J. T.-C. Tsai : SMulticluster, Mobile Multimedia Radio Networks. Wireless Networks. (1995) 255–256

[7] S. Yi and R. Kravets : Quality of Authentication in Ad Hoc Networks. ACM, MobiCom2004. (2004)

[8] I. Inn Er and Winston K.G. seah. Mobility-based d-hop Clustering Algorithm for Mobile Ad Hoc Networks. (2004)

[9] S. Capkun and J. P. Hubaux and L. Buttyan : Mobility Helps Peer-to-Peer Security. IEEE Transactions on Mobile Computing. **5** (2006) 48–60

[10] C. Chiang and H. Wu and W. Liu and M. Gerla : Routing in Clustered Multihop Mobile Wireless Networks with Fading Channel. IEEE Proceedings of SICON'97. (1997) 197–211

[11] Alfarez Abdul-Rahman and Stephen Hailes : A Distributed Trust Model. New Paradigms Workshop 1997, ACM. (1997)

[12] Lidong Zhou and Zygmunt J. Haas : Securing Ad Hoc Networks. IEEE Network. **13** (1999) 24 –30

[13] Marc Bechler and Hans-Joachim Hof and Daniel Kraft and Frank Pahlke and Lars Wolf : A Cluster-Based Security Architecture for Ad Hoc Networks. INFOCOM2004. (2004)

[14] Kimaya Sanzgiri and Bridget Dahill and Daniel LaFlamme and Brian N. Levine and Clay Shields and Elizabeth M. Belding-Royer : An Authenticated Routing Protocol for Secure Ad Hoc Networks. Selected Areas in Communication (JSAC). **23** (2005) 598–610

[15] UC Berkeley and USC ISI : The network simulator ns-2. Part of the VINT project. Available from http ://www.isi.edu/nsnam/ns. (1998)