REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université A. Mira de Bejaïa

Faculté des Sciences Exactes

Département d'Informatique

Mémoire de fin de cycle

En vue de l'obtention du diplôme de Master Recherche

En Informatique

Option :"Réseaux et Systèmes Distribués"

Thème

Gestion des clés publiques dans les réseaux mobiles ad hoc

Réalisé par :

M^{elle} .MAKHLOUFI Roza

M^{r}. BAOUCH Immad

Devant le jury composé de :

Président : L. KHENOUS

Encadreur : M. OMAR

Examinatrice : A. KOUICEM

Examinatrice : N. BOUSBA

Promotion : Juin 2012

Remerciements

Nous remercions tout d'abord Dieu le tout puissant pour nous avoir donné la santé, la force, le courage et l'intelligence nécessaires pour réaliser ce modeste travail.

Nous tenons à remercier notre promoteur ***Dr. Mawloud OMAR*** *d'avoir cru en nous dès le début et de lui exprimer notre immense gratitude pour son encadrement, ses conseils, son soutien constant, sa générosité, sa disponibilité et sa patience... nous avons beaucoup appris à vos côtés. Merci de nous avoir fait découvrir les plaisirs du travail bien fait, nous espérons pouvoir travailler avec vous dans le futur.*

Nos remerciements vont également à l'ensemble du jury pour avoir accepté de juger notre travail.

Nous tenons aussi à remercier tous nos professeurs et toute l'équipe pédagogique de l'université qui ont travaillé avec abnégation pour nous permettre de suivre ce cursus.

Nous remercions ainsi notre ami, l'e doctorant Mr SALHI Dahiaeldine. Qui nous a beaucoup aidés tout au long de la Simulation de notre travail.

De même nous remercions tous ceux qui ont contribué de près ou de loin à l'aboutissement de ce travail, soit avec leur support, leur amitié ou leur amour.

Dédicaces

Je dédie ce modeste travail :

A mes très chers parents pour leurs sacrifices, patience et soutien inconditionnels, nulle dédicace n'est susceptible de vous exprimer ma profonde affection.

A ma sœur Monia, mes frères Anis et Imsis.

A toute ma famille : mes tantes, mes oncles, mes cousins et cousines.

A tous mes amis(es), particulièrement Othman, Kiki, El guerra (ali), Deej Ed (idir), mounir, Soraya, Feriel, Melissa, Dida, sony, Karry, Cara, Fati, Nisset ainsi qu'A mon chéri FOUZI et toute sa famille

A tous ceux qui m'aiment.

A toute la famille du croissant rouge de Bejaia.

A tous ceux qui m'ont aidé de loin ou de près durant mes études.

ROSA

Dédicaces

Je dédie ce modeste travail :

A mes chers Parents qui m'ont soutenu et encouragés tout au long de ma scolarité.

A mes frères moh, abdou, sidou, moumen.

A toute ma famille.

A tous mes amis(es).

A tous ceux qui m'aiment.

Imad.

"Une personne qui n'a jamais commis d'erreurs n'a jamais tenté d'innover."

Albert Einstein.

Table des matières

Liste des figures

Liste des tableaux

Liste des abréviations

A

AODV: **A**d-hoc **O**n demand **D**istance **V**ectorrouting

AES: **A**dvanced **E**ncryption**S**tandars

B

BLR: **B**oucle **L**ocale **R**adio

C

CBRP: **C**luster **B**ased **R**outing **P**rotocol

CA: **C**ertification **A**uthority

CPU: **C**entral **P**rocessing **U**nit

CH: **C**luster **H**ead

CMN: **C**ertificate **M**anagement **N**ode

D

DARPA: **D**efense **A**dvanced **R**esearch **P**rojects **A**gency

DoS: **D**eny **O**f **S**ervice

DSR: **D**ynamic **S**ource **R**outing

DVB-S: **D**igital **V**ideo **B**roadcasting - **S**atellite

DES: **D**ata **E**ncryption **S**tandars

DICTATE: **D**istributed **C**erTification **A**uthority with probabilisticfr **E**shnessforad hoc networks

F

FSR: **F**isheye **S**tate **R**outing

G

GPS: **G**lobal **P**ositionning**S**ystem

GSM: **G**lobal **S**ystem for **M**obile **C**ommunications

GPRS: **G**eneral **P**acket **R**adio **S**ervice

H

HomeRF: **H**ome **R**adio **F**requency

I

IEEE: **I**nstitute of **E**lectrical and **E**lectronics **E**ngineers

IETF: **I**nternational **E**ngineering **T**ask **F**orce

IR: **I**nfra**R**ouge

M

MANET: **M**obile **A**d-hoc **Net**works

MAC: **M**essage **A**uthentification **C**ode

MD: **M**essage **D**igest

MOCA: **MO**bile**C**ertification **A**uthority

mCA: **m**other **C**ertification **A**uthority

dCA : **d**istributed **C**ertification **A**uthority

O

OLSR: **O**ptimized **L**ink **S**tate **R**adio

P

PRNet: **P**akets**R**adio **N**etwork

PDA: **P**ersonal **D**igital **A**ssistant

PKI: **P**ublic **K**ey **I**nfrastructure

R

RREQ: **R**oute **R**equest

RSA: **R**ivest**S**hamir **A**dleman

RIPEMD: **R**ACE **I**ntegrity **P**rimitives **E**valuation **M**essage **D**igest

S

SURAN: **S**urvivable **R**adio **N**etworks

T

TBRPF: **T**opology dissemination **B**ased on **R**everse **P**ath **F**orwarding

W

WiMAX: **W**orldwide **I**nteroperability for **M**icrowave **A**ccess

WLAN: **A**rea **N**etworks **W**ireless **L**ocal

WMAN: **W**ireless **M**étropolitains**A**rea **N**etworks

WPAN: **W**ireless **P**ersonal **A**rea **N**etwork

WWAN: **W**ireless **W**ide **A**rea **N**etworks

Z

ZRP: **Z**one **R**outing**P**rotocol

INTRODUCTION GÉNÉRALE

L'évolution rapide des communications sans fil a permis le développement de nouveaux réseaux dont l'organisation est fortement distribuée et dynamique. Un réseau mobile ad hoc [Per00, Gio01] est une collection de nœuds mobiles formant un réseau temporaire sans l'aide de toute infrastructure fixe, ni administration centralisée. Ces nœuds doivent donc collaborer pour organiser l'échange d'informations de contrôle et permettre l'acheminement du trafic via des liaisons sans fil, et qui sont parfois limités en termes de bande passante, de calcul, et de mémoire. Les réseaux ad hoc deviennent de plus en plus populaires et leurs utilisations augmentent de jour en jour, et ceci est dû à la facilité de leur déploiement. Initialement, utilisés dans les applications militaires, les réseaux ad hoc ont trouvé des champs d'utilisation dans de nombreuses nouvelles applications, telles que les opérations tactiques comme les secours et les missions d'exploration. Ces réseaux ont stimulé beaucoup de travaux de recherche ces dernières années, et ont donné naissance à plusieurs directions de recherche telle que le routage [QK04], sécurité [LJ04], qualité de service [PH02], etc.

Lors de la communication entre deux systèmes informatiques, il y a échange de paquets. Mais les données transmises sur un réseau public comme Internet peuvent facilement être interceptées et analysées pour en extraire des informations sensibles, ce qui rend Les réseaux mobiles ad hoc plus vulnérables aux attaques (passives et actives) que les réseaux filaires classiques. On peut s'attendre à des dégradations de la qualité des communications à partir du moment où des nœuds malveillants corrompent les opérations de base dont ils sont responsables. Par ailleurs, des mesures de sécurité sont mises en place, tel que les mécanismes de la sécurité traditionnels, dont la signature digitale et le chiffrement à clé publique qui ont resté toujours des outils essentiels pour garantir les besoins de sécurité dans les réseaux ad hoc, ces mécanismes nécessitent un service de gestion de clés afin de garantir l'authenticité des parties et l'intégrité des données transmises, et d'établir une confiance entre les nœuds du réseau.

Dans ce mémoire, on traite d'un axe majeur la sécurité des réseaux ad hoc, et en particulier la confiance. Et en effet, tout développement des services de sécurité doit s'appuyer sur un modèle de confiance. Dont la gestion de la confiance dans les réseaux ad hoc a fait l'objet de deux grandes catégories de modèles : les modèles de confiance à base de coopération [BB02, MM02, JFD09, HWK04, ZMH09, AF10, MD08], et les modèles de confiance à base de certification [ZH99, CBH02, CBH03, RTJ06, YK03, LHE05, GLG09, KAT05, KKD10].Dans la première catégorie de modèles, la confiance est basée sur la notion de réputation. Dans le cadre de notre rapport, nous nous intéressons à la deuxième catégorie : *les modèles de confiance à base de certification*. Un certificat est une structure de données dans laquelle une clé est liée à une identité (et éventuellement à certains autres attributs) qui est délivré par une tierce partie de confiance. Si cette dernière estime qu'un nœud donné est digne de confiance, elle lui délivre un certificat qui va lui permettre de prouver sa légitimité envers les autres nœuds du réseau.

Paradoxalement, la plupart des solutions existantes pour la sécurisation des réseaux ad hoc supposent l'existence de clés, donc d'une infrastructure de gestions de clés. Le service de gestion de clé est basé sur une entité digne de confiance appelé autorité de certification CA qui doit créer un certificat de clé publique à chacun des nœuds. En effet, la clé de l'autorité de certification joue un rôle essentiel pour l'établissement de la confiance dans le réseau où elle assure le mécanisme de signature des certificats. L'autorité de certification digne de confiance doit être en ligne pour traiter les cas de révocation et de renouvellement des certificats de clés publiques, Cependant il est dangereux d'installer un service de gestion de clés en utilisant une seule autorité de certification dans un réseau ad hoc Car si cette unique autorité de certification est compromise, la sécurité de tout le réseau est brisée.

Pour faire face à toutes ces contraintes, il faut mettre en œuvre une architecture d'un modèle de confiance à la fois robuste et hautement disponible dans un milieu ad hoc, ce qui représente l'objectif de notre solution, dont on a exploité l'avantage de cryptographie à seuil pour la délivrance des certificats ou on s'est basé sur le concept partagé du service de certification entre les utilisateurs.

Objectif du travail

Dans ce travail, nous nous intéressons à l'étude de certains protocoles de gestion de clés publiques dans les réseaux ad hoc mobiles. Notre objectif est d'exposer leurs spécificités, caractéristiques ainsi que leurs modes de fonctionnement, afin de mettre en œuvre une architecture d'un modèle de confiance à la fois robuste et hautement disponible dans un milieu ad hoc.

Plan et contenu du mémoire

Afin de présenter notre projet, nous organisons ce manuscrit en quatre chapitres. Dans le premier, nous introduisons les environnements sans fil et particulièrement les réseaux ad hoc et leurs caractéristiques. Dans le second, Nous présentons les concepts de base de la sécurité et la vulnérabilité des réseaux ad hoc. Nous mettons l'accent notamment sur la nécessité de mise en œuvre d'un modèle de confiance pour sécuriser les échanges entre les nœuds du réseau. Nous présentons, également, une vue d'ensemble sur les concepts liés a la cryptographie et l'infrastructure de gestion de clés publiques. Ensuite, dans le troisième chapitre, nous faisons un état de l'art sur les modèles de confiance à base de certification dans les réseaux ad hoc. Nous présentons une taxonomie détaillée de travaux qui englobe deux grandes catégories de modèles : les modèles autoritaires et les modèles anarchiques. Dans les modèles autoritaires, les nœuds sont supervisés par une autorité de certification qui assure la gestion des certificats. Dans les modèles anarchiques, la gestion des certificats est assurée par les nœuds eux-mêmes en utilisant le principe de la confiance transitive. Nous donnons pour chaque catégorie l'ensemble des solutions proposées dans la littérature avec leurs avantages et inconvénients. Nous donnons ensuite une étude comparative des travaux existants de chaque catégorie selon un ensemble de critères bien précis. Dans le quatrième chapitre nous proposons une architecture d'un modèle de confiance autoritaire à base de Certification à Seuil. A la fin, nous évaluons les performances de notre modèle à travers des simulations.

CHAPITRE I

Les environnements sans fil (Ad-hoc)

Introduction

Ces dernière années, les appareils mobiles ont connus un grand essor, car ils sont dotés d'une multitude de fonctionnalités leurs permettent d'assurer différents services. Comme exemple de service on a les services de connexion et les services de données (stockées) correspondants. Ces derniers représentent les services les plus demandés par les utilisateurs mobiles.

Mais, il y'a des situations où les besoins de connexion des utilisateurs ne peuvent pas être assurés par le réseau dans une zone géographique donnée (zone en territoire hostile, région qui a subi une catastrophe naturelle, etc.). Dans ce cas, fournir une connectivité relève du défi. Récemment, de nouvelles techniques pour fournir ces services ont été proposées. Elle repose sur le fait d'avoir des stations mobiles interconnectées entre elles à l'aide d'une configuration autonome, créant ainsi un réseau ad hoc flexible et performant. Conjointement, le réseau ad hoc peut être utilisé pour étendre un réseau filaire. Dans ce cas, les nœuds mobiles peuvent accéder à Internet à travers une passerelle, pour étendre les services internet au-delà de l'infrastructure filaire ou les zones couvertes par des points d'accès.

Historiquement, le premier développement des réseaux Ad-Hoc a été le résultat de la demande du milieu militaire pour le déploiement rapide d'infrastructures de télécommunication pouvant survivre aux pannes et aux attaques. Un réseau centralisé autour de stations de base n'est pas une bonne option dans ce milieu car elles doivent être déployées en premier lieu et le réseau est vulnérable dans le cas où une ou plusieurs de ces stations de base sont détruites. Face à ces limites, le département de la défense américaine, en particulier DARPA[RLS96], a sponsorisé le programme de recherche PRNet[JT87].

Ce projet traitait en particulier la problématique de routage et l'accès au média dans un réseau de communication multi-sauts par onde radio. Ce dernier a évolué vers le programme SURAN qui traitait en particulier la problématique de la sécurité, la gestion d'énergie et la capacité de traitement [FL01]. Et dès le début des années quatre-vingt-dix les ordinateurs portables ont été équipés de cartes sans fil et de ports infrarouges qui permettaient la communication directe et sans intermédiaire entre les ordinateurs portables. Ainsi, la technologie de PRNet était devenue accessible au grand public avec de réelles applications civiles. L'IEEE adoptait alors le terme "réseaux Ad-Hoc" pour le standard IEEE802.11 des réseaux locaux sans fil.

1.1. Classification des réseaux sans fil

Les Réseaux sans fil se différent par la fréquence d'émission utilisée, le débit, la portée des transmissions et le mode de fonctionnement. Plusieurs classifications peuvent être définies suivant ces caractéristiques. Nous présentons dans ce qui suit deux différentes classifications englobant tous les types de réseaux sans fil : une classification selon la distance séparant les unités mobiles et une classification selon l'infrastructure du réseau.

1.1.1. Classification selon la distance séparant les nœuds mobiles

On peut distinguer quatre grandes catégories de réseaux sans fil selon la distance qui séparent les nœuds mobiles. Différents technologies ont été mises au point pour répondre aux impératifs de chacune de ces catégories.

- **WPAN**

Les réseaux sans fil personnel ou *Wireless Personal Area Network*, son des réseaux sans fil à très faible portée, leur but est de faire communiqué deux ou plusieurs équipements présents sur une personne (une oreillette et un téléphone portable), ou a relier des équipements informatiques entre eux (par exemple pour relier une imprimante ou un PDA à un ordinateur de bureau).

Les principales techniques utilisées dans cette catégorie sont : Bluetooth (IEEE 802.15.1), Home Radio Frequency (HomeRF), InfraRouge (IR), ZigBee.

- **WLAN**

Les réseaux locaux sans fil ou *Wireless Local Area Networks,* sont généralement utilisés à l'intérieur des entreprises, universités, ou chez les particuliers. Ils permettent de relier des terminaux présents dans une portée d'environ une certaine de mètres.

Les principales technologies utilisées dans cette catégorie sont : HiperLan1, HiperLan2, IEEE 802.11.

- **WMAN**

Les réseaux métropolitains sans fil *Wireless Métropolitains Area Networks,* également appelés boucle locale radio (BLR) étaient à l'origine prévus pour interconnecter des zones géographiques difficiles d'accès a l'aide d'un réseau sans fil. Les WMAN sont basés sur la norme IEEE 802.16. Cette dernière offre un débit utile de 1 a 10 Mb/s pour une portée de 4 a 10 kilomètres, et pouvant atteindre 74 Mb/s pour IEEE802.16-2004 plus connue sous le nom WiMAX (*WorldwideInteroperability for Microwave Access*), sur un rayon de plusieurs kilomètres.

- **WWAN**

Les réseaux sans fil étendus ou *Wireless Wide Area Networks,* sont également connus sous le nom de Réseau cellulaire mobile. Leur principe consiste a partagé une zone géographique en sous zones appelées cellules qui sont organisées sous forme hexagonale par la suite une bonde différente de fréquence est affectée à chacune des cellules pour éviter les interférences des signaux ils regroupent notamment les différents réseaux téléphonique de première et de deuxième génération et également les réseaux satellitaires. Les réseaux cellulaires téléphoniques reposent sur des technologies comme : le GSM, GPRS, les réseaux satellitaires s'appuient quant à eux sur les nombres comme DVB-S pour transmettre l'information et proposent des débits élevés (de l'ordre de 40 Mb/s pour la norme DVB-S)

1.1.2. Classification selon l'infrastructure

Un réseau sans fil peut fonctionner selon deux modes : le mode infrastructure où les nœuds sont connectés via des points d'accès et le mode sans infrastructure (ad hoc) où les nœuds communiquent directement entre eux ou par l'intermédiaire d'autres nœuds.

1.2. Les réseaux ad hoc

1.2.1. Définition des réseaux ad hoc

Un réseau mobile ad hoc, appelé généralement MANET [CCN03](Mobile Ad hoc Network), est un réseau sans fil, spontané, temporaire, formé dynamiquement par un ensemble de nœuds mobiles sans l'utilisation d'une infrastructure réseau existante ou d'une administration centralisée. Les nœuds peuvent se déplacer aléatoirement et s'organiser en conséquent. De ce fait, la topologie peut varier de façon imprévisible.

Dans un réseau Ad hoc, le chemin entre un nœud source et un nœud destination peut impliquer plusieurs sauts sans fil, d'où l'appellation de « réseau sans fil multi sauts ». Un nœud peut communiquer directement avec un autre qui est à sa portée. Au-delà de cette portée des nœuds intermédiaires jouent le rôle du routeur. Donc, un nœud peut jouer à la fois le rôle d'un terminal et le rôle d'un routeur (on parle alors de réseau coopératif).

1.2.2. Caractéristique des réseaux ad hoc

Dans un réseau ad hoc, les hôtes mobiles forment le réseau, et ils doivent fournir en collaborant les fonctionnalités habituellement fournies par l'infrastructure réseau (ex. routeur, switchs, serveurs). Ce type de réseaux possède de nombreuses caractéristiques non connues dans les réseaux filaires, qui sont:

- ***Topologie dynamique***

Les unités mobiles du réseau se déplacent d'une façon libre et arbitraire. Ceci fait que la topologie du réseau peut changer aléatoirement et rapidement à des instants imprévisibles.

- ***Absence d'infrastructure***

Ce qui distingue les réseaux ad hoc des autres réseaux mobiles c'est l'absence d'infrastructure préexistante ou d'administration centralisées. Les hôtes mobiles se chargent d'établir et de maintenir la connectivité du réseau d'une manière continue.

- ***Bande passante limitée***

L'utilisation d'un médium de communication partagé est une caractéristique primordiale dans les réseaux sans fils. Ce partage fait que la bande passante réservée à un hôte soit modeste.

- ***Capacité des mobiles limitée***

Les réseaux ad hoc sont formés essentiellement par des unités mobiles portables (ex. téléphone mobiles, assistants numériques personnels, ordinateurs portables), dotées de moyens de communication sans fil et de faible quantités de ressources (mémoire et CPU), ce qui limite leurs capacités.

- ***Contrainte d'énergie***

Les hôtes mobiles sont alimentés par des sources d'énergie autonomes telles que les batteries, les cellules photoélectriques ou d'autres sources consommables, dont la durée de vie est limitée. Le paramètre d'énergie doit être pris en considération dans tout contrôle fait par le système (tels que l'émission et la réception des données, l'écoute du canal, etc).

- ***Une faible sécurité physique***

Les réseaux sans fil offrent de nouvelles failles aux pirates. De part la nature immatérielle du support physique, l'écoute clandestine sur un réseau sans fil est facile. Ces réseaux sont également vulnérables aux attaques actives comme la création, la modification et la destruction non autorisée des données. Il faut donc protéger l'accès aux ressources sans fil et aux informations qui circulent dans le réseau.

1.2.3. Domaines d'application des réseaux ad hoc

Les premières applications des réseaux ad hoc concernaient les communications et les opérations dans le domaine militaire. Cependant, avec le progrès des recherches dans le domaine des réseaux et l'émergence des technologies sans fil (tel que Bluetooth, IEEE 802.11 et Hiperlan); d'autres applications civiles sont apparues. On distingue:

- ***Les services d'urgence***

Opération de recherche et de secours des personnes, tremblement de terre, feux, inondation, dans le but de remplacer l'infrastructure filaire.

- ***Le travail collaboratif et les communications dans des entreprises***

Dans le cadre d'une réunion ou d'une conférence par exemple.

- ***Home Network***

Partage d'applications et communications des équipements mobiles.

- ***Applications commerciales***

Pour un paiement électronique distant (taxi) ou pour l'accès mobile à l'Internet.

- ***Réseaux en mouvement***

Informatique embarquée et véhicules communicants.

1.2.4. Modélisation et notation des réseaux ad hoc

Un réseau mobile ad hoc, consiste en une grande population, relativement dense, d'unités mobiles qui se déplacent dans un territoire quelconque et dont le seul moyen de communication est l'utilisation des interfaces sans fil, sans avoir à solliciter l'aide d'une infrastructure réseau existante, ou administration centralisée. Un réseau ad hoc peut être modélisé par un graphe non orienté $G = (V, E)$ tel que :

V : représente l'ensemble des nœuds du réseau (i.e. les unités mobiles)

E : modélise l'ensemble des connexions (des arêtes) qui existent entre ces nœuds.

S'il existe une arête : $e = (u, v)$, cela veut dire que les nœuds u et v sont à portée l'un de l'autre et qu'ils sont en mesure de communiquer directement. La figure 1.1 représente un réseau ad hoc de sept nœuds sous forme d'un graphe.

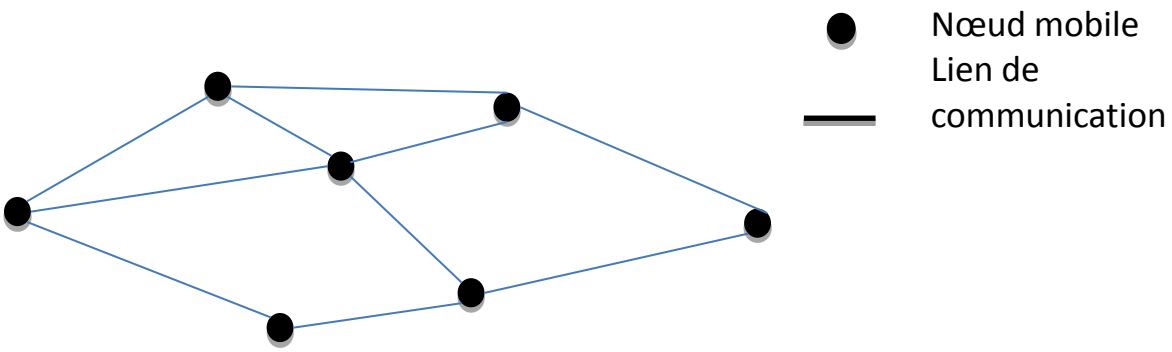


Figure 1.1 : Modélisation d'un réseau ad hoc

La topologie du réseau peut changer à tout moment du fait que les unités mobiles se déplacent, se connectent au réseau et se déconnectent aléatoirement. Elle est donc dynamique et imprévisible. Les liens entre les nœuds se créent et se coupent quand les nœuds se déplacent dans le réseau (voir Figure I.2).

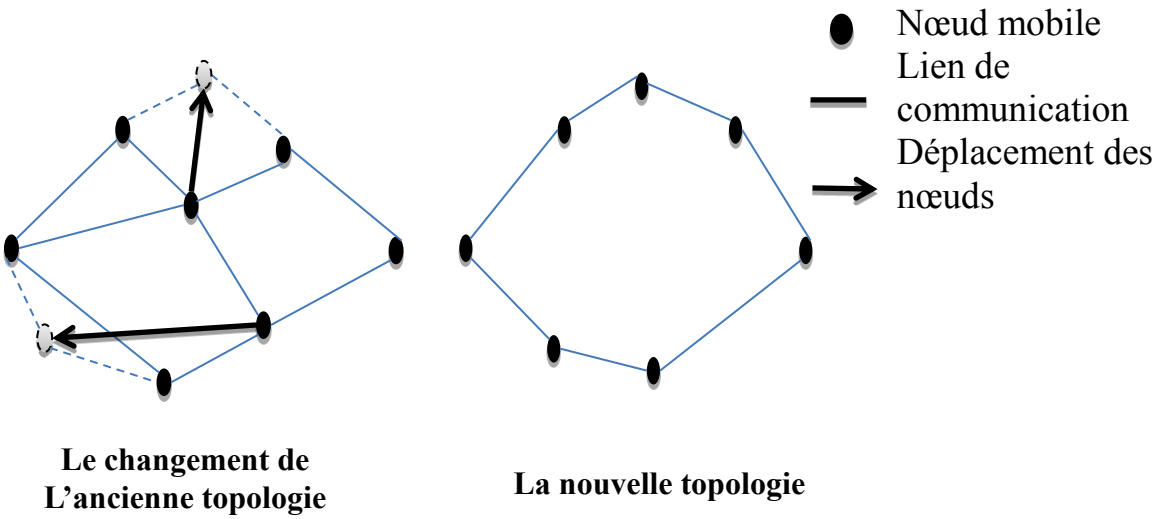


Figure 1.2 : Topologie dynamique d'un réseau ad hoc

1.2.5. Le routage dans les réseaux ad hoc

Le routage est en quelque sorte le mécanisme clé des réseaux ad hoc. Il peut arriver qu'une unité mobile veuille communiquer avec une autre qui n'est pas dans sa portée de communication directe. C'est grâce au mécanisme de routage que les unités mobiles formant le réseau vont pouvoir communiquer, même si elles ne sont pas à la portée directe de la communication. Les messages vont donc devoir être transmis de proche en proche jusqu'à la destination. Le routage est donc une méthode d'acheminement des

informations à la bonne destination à travers un réseau de connexion donné. Le problème de routage consiste à déterminer un chemin optimal des paquets à travers le réseau au sens d'un certain critère de performance (bande passante, délai, énergie, etc.). De nombreux protocoles de routage ont donc été proposés pour rendre les communications multi sauts plus efficaces que l'inondation.

Le groupe de travail MANET de l'IETF a développé et classifié les protocoles de routage pour les réseaux ad hoc en trois familles ou classes: proactif, réactif et hybride. Dans cette section, nous expliquons le principe de chacune de ces classes et nous citons quelques protocoles les plus connus dans chaque famille.

1.2.5.1. Les protocoles proactifs

Les protocoles de routage proactifs dans les réseaux ad hoc fonctionnent selon le même principe que les protocoles de routage utilisés dans les réseaux filaire. Ils font en sorte de maintenir à jour des tables de routage, de façon que lorsqu'une application désire d'envoyer un paquet à un autre mobile, une route soit immédiatement connue. Dans le contexte des réseaux Ad Hoc, les nœuds peuvent apparaître ou disparaître de manière aléatoire et la topologie même du réseau peut changer; cela signifie qu'il va falloir un échange continuel d'informations pour que chaque nœud ait une image à jour du réseau. Les tables sont donc maintenues grâce à des paquets de contrôle, et il est possible d'y trouver directement et à tout moment un chemin vers les destinations connues en fonction de divers critères. On peut par exemple privilégier les routes comportant peu de sauts, celles qui offrent la meilleure bande passante, ou encore celles où le délai est le plus faible.

Les protocoles de cette famille se différencient par la manière dont cette information de mise à jour est transmise à travers le réseau ainsi que par le nombre de tables de routage utilisées. Parmi les principaux protocoles proactifs, nous citons:

– **OLSR** (Optimized Link State Routing) [CJ03].
– **FSR** (Fisheye State Routing) [HGP02].
– **TBRPF** (Topology Dissemination Basedon Reverse-path Forwarding) [LFR04].

1.2.5.2. Les protocoles réactifs

Les protocoles de routage réactifs (dits aussi: protocoles de routage à la demande) représentent les protocoles les plus récents proposés dans le but d'assurer le service du routage dans les réseaux sans fil. Le principe des protocoles réactifs est de ne rien faire jusqu'à ce qu'une application demande explicitement d'envoyer un paquet vers un nœud distant. Lorsque le réseau a besoin d'une route, une procédure de découverte globale de routes est lancée, et cela dans le but d'obtenir une information spécifiée, inconnue au préalable. Plusieurs approches peuvent être appliquées dans la découverte des routes. La méthode classique de recherche de route consiste à inonder le réseau avec une requête de demande de route RREQ. Les nœuds voulant communiquer à travers le réseau lancent des requêtes à la recherche de routes permettant l'acheminement des paquets d'information. Parmi les principaux protocoles réactifs, nous citons:

– **AODV** (Ad-hoc On demand Distance Vector routing) [DBP03].
– **DSR** (Dynamic Source Routing) [HJM07].

1.2.5.3. Les protocoles hybrides

Les protocoles hybrides combinent les deux approches proactives et réactives. Ils utilisent un protocole proactif, pour apprendre le proche voisin (par exemple voisinage à deux ou trois sauts); ainsi ils disposent des routes immédiatement dans le voisinage. Au-delà de cette zone prédéfinie, le protocole hybride fait appel aux techniques des protocoles réactifs pour chercher des routes. Avec ce découpage, le réseau est partagé en plusieurs zones et la recherche de routes en mode réactif peut être améliorée. A la réception d'une requête de recherche réactive, un nœud peut indiquer immédiatement si la destination est dans le voisinage ou non, et par conséquent savoir s'il faut aiguiller la requête vers les autres zones sans déranger le reste de sa zone. Parmi les principaux protocoles hybrides, nous citons:

– **ZRP**(Zone Routing Protocol)[PZH02].
– **CBRP**(Cluster Based Routing Protocol)[LJT99].

Conclusion

Nous avons présenté dans ce chapitre, les réseaux ad hoc qui sont un type particulier de réseaux sans fil, ne nécessitant aucune infrastructure fixe pour se créer et s'organiser. Malgré les progrès réalisés dans ce domaine, pour atteindre les objectifs de ces réseaux, beaucoup de travail reste à faire. Les caractéristiques de ces réseaux constituent de réels défis. Le caractère fortement dynamique des réseaux ad hoc, nécessite l'implémentation de protocoles plus complexes que ceux des réseaux fixes ou des réseaux avec points d'accès. A cause de leurs vulnérabilités, les réseaux ad hoc sont sujets à de très nombreuses menaces. Pour faire face à la plupart de ces menaces, la cryptographie est un moyen très puissant. Mais, pour assurer une sécurité digne de ce nom, les systèmes cryptographiques doivent être associés à une gestion de clés efficace. Ces sujets seront abordés plus en détail dans les prochains chapitres.

CHAPITREII

La Sécurité des réseaux Ad hoc

Introduction

Un réseau ad hoc est une collection de nœuds mobiles formant un réseau temporaire sans l'aide d'une infrastructure de communication fixe, ni administration centralisée, qui comportent un ensemble de nœuds mobiles qui communiquent à travers des liaisons sans fil. Dans ce chapitre, nous citerons les concepts de base de leurs sécurités et nous mettons l'accent sur leurs vulnérabilités en motivant la nécessité de mise en œuvre d'un modèle de confiance pour sécuriser les échanges entre les nœuds du réseau. Nous rappellerons, également, des définitions de la cryptographie en général. Puis, en particulier, nous présenterons les notions de cryptographie à seuil, et l'infrastructure à clés publiques. Le rappel de ces notions facilitera la présentation des chapitres suivants.

2.1. Concepts de base sur la sécurité

2.1.1. Services de sécurité

La sécurité informatique se base sur des services qui permettent de faire face à certaines menaces telles que l'écoute, la création ou la modification des données et la répudiation. Ces services de sécurités ont des propriétés que certaines applications exigent et que les protocoles de sécurité utilisés dans ces applications devraient assurer [G09]. Parmi ces services:

- ***Authentification***

L'authentification d'un participant veut dire qu'un agent doit être sûr de l'identité de son correspondant. La propriété de l'authentification repose sur le fait d'assurer que l'origine est bel et bien l'entité déclarée. Différents mécanismes sont utilisés pour assurer l'authentification tel que : les signatures numériques.

- ***Confidentialité***

La confidentialité de données est la propriété par laquelle l'information n'est pas rendue disponible ou n'est pas révélée aux individus, aux entités, ou aux processus non autorisés. Celle-ci peut être garantie en utilisant le chiffrement.

La confidentialité des données dans une communication réseau fournit une protection contre l'analyse du trafic. Les données transportées ne peuvent pas être lues par un adversaire espionnant les communications.

- ***Intégrité***

L'intégrité des données est la propriété par laquelle on s'assure que des données n'ont pas été changées, détruite ou perdues d'une façon non autorisée ou accidentelle durant leur transit. Si une partie intermédiaire modifie les données d'origines envoyées par l'expéditeur, le destinataire final devrait être capable de détecter qu'il ya eu des changements.

- ***Contrôle d'accès***

Ce service limite et contrôle l'accès aux ressources et aux utilisateurs autorités. Pour réaliser ceci, l'utilisateur est d'abord identifie (authentifie) et après, on lui accorde des droits d'accès correspondants.

- ***Disponibilité***

Permet de maintenir le bon fonctionnement du système. En fournissant des services ou des ressources de réseaux aux utilisateurs légitimes. La disponibilité est une propriété difficile à gérer, à cause de la topologie dynamique, la limitation des ressources sur certain nœuds est la facilité de brouillage des communications.

- ***Non Répudiation***

La non répudiation est définie par l'impossibilité pour une des entités impliquées dans une communication de nier avoir participé à l'ensemble ou à une partie de la communication. Elle assure ainsi une protection contre le faux démenti d'une entité d'être impliquée dans une communication. Pour assurer la non répudiation on peut utiliser les MACs, les signatures digitales et les certificat.

- ***L'anonymat***

La propriété de l'anonymat procure à un participant la possibilité de faire les transactions anonymes qui ne peuvent pas être dépistées par un autre participant.

2.1.2. Vulnérabilité des réseaux ad hoc

Les réseaux mobiles ad hoc sont plus vulnérables que les réseaux traditionnels, à cause de la nature même de ces réseaux. La première vulnérabilité de ces réseaux est liée à la technologie sans fil sous-jacente. Quiconque possédant le récepteur adéquat peut potentiellement écouter ou perturber les messages échangés. Et ceci, même s'il se trouve dans un lieu public, à l'extérieur du bâtiment ou se déroulent les échanges.

Les nœuds eux-mêmes sont des points de vulnérabilités du réseau car un attaquant peut compromettre un élément laissé sans surveillance. L'absence d'infrastructure fixe pénalise l'ensemble du réseau dans la mesure où il faut faire abstraction de toute entité centrale de gestion pour l'accès aux ressources. Les mécanismes de routage sont d'autant plus critiques dans les réseaux ad hoc que chaque entité participe à l'acheminement des paquets à travers le réseau. De plus, les messages de routage transitent sur les ondes radio.

Enfin, ces réseaux héritent de toutes les vulnérabilités propres aux technologies sans fil WLAN et WPAN.

2.1.3. Les attaques

Parmi les problèmes liés à la sécurité des réseaux ad hoc, une liste des attaques les plus probables est ainsi dressée dans cette section

- **Les attaques de dénis de service (DoS)**

Les attaques DoS apparaissent comme les plus faciles à réaliser par un attaquant, mettant en péril la disponibilité des membres du groupe et plus particulièrement des entités qui jouent le rôle de serveurs ou de contrôleurs au sein du réseau ad hoc. Les attaques DoS les plus connues dans un MANET sont les suivantes:

- Brouillage du canal radio pour empêcher toute communication.
- Tentative de débordement des tables de routages des nœuds servant de relais.

- Non coopération d'un nœud au bon fonctionnement du réseau, dans le but de préserver son énergie. Cette attaque est connue sous le nom de "Selfishness" et peut être détectée grâce à des mécanismes de réputation et de détection de comportements égoïstes (Nulets[BH01], confidant[BB02], Core[MM02]).
- Tentative de gaspillage de l'énergie des nœuds ayant une autonomie de batterie faible. Cette attaque est connue sous le nom de "SleepDepriviation Torture [S02] et consiste à faire en sorte que le nœud cible soit obligé de rester en état d'activité et ainsi de lui faire consommer toute son énergie.
- Dispersion et suppression du trafic en attaquant les mécanismes de routage. L'attaque Wormhole[MM02] fait partie de ce type d'attaques qui est particulièrement difficile à détecter ou à prévenir. Elle consiste à ce qu'un nœud malicieux achemine tous les paquets du réseau via un tunnel privé partagé avec un autre attaquant, offrant à une source un meilleur chemin (erroné) pour atteindre sa destination et éliminer ainsi la possibilité de découverte des chemins fiables dans le réseau.
- Attaque des mécanismes de sécurité eux-mêmes

o Les attaques passives d'écoute et analyse de trafic

Ces attaques, appelées aussi "Sniffing", consiste à écouter le réseau dans lequel transitent des paquets de données et à récupérer à la volée et illégalement des données qui peuvent être confidentielles. Les attaques d'écoute et d'analyse de trafic sont d'autant plus dangereuses dans les réseaux sans fils tels que les MANETs. En effet, les ondes radio électrique sont intrinsèquement une grande capacité à se propager dans toutes les directions avec une portée relativement grande, facilitant ainsi à une personne non autorisée, d'écouter le réseau et d'analyser le trafic. Ces attaques constituent une menace certaine pour la confidentialité des données, ainsi que pour l'anonymat des utilisateurs.

o L'usurpation de l'identité d'un nœud

L'usurpation d'identité, appelée également mystification, consiste à se faire passer pour une entité connue et fiable auprès des autres nœuds, afin de récupérer illégalement des informations confidentielles ou d'injecter des messages dans le réseau. Ce type d'attaques met en péril l'authentification et le contrôle d'accès des membres des réseaux ad hoc.

- **Les attaques physiques d'un élément valide du réseau**

Ces attaques compromettent des nœuds valides du réseau (destruction, altération ou changement physique d'un composant) et s'avèrent être des attaques particulièrement dangereuses dans les réseaux ad hoc.

2.1.4. Concepts de base sur la cryptographie

Le mot cryptographie est un mot d'origine grecque composé de deux parties : « crypto » (kruptos) qui signifie caché et « graphie » (graphein) qui signifie écrire. D'une manière générale, la cryptographie consiste en une paire (e, d) d'opérations. La première opération e est le chiffrement (appelée aussi le cryptage). Elle permet de convertir un texte initial *M*, dit texte en claire, en un autre texte *C*, dit texte chiffré, supposé incompréhensible. La forme *C* dépend d'un paramètre *K* appelé clé de chiffrement. La deuxième opération d est le déchiffrement, (appelée aussi décryptage). Le déchiffrement permet de reconstruire le texte en clair à partir du texte chiffré. Cette reconstruction requiert une deuxième clé, K^{-1}, dépendante de la clé du chiffrement, dite clé de déchiffrement.

A l'opposé de la cryptographie, la **cryptanalyse** l'art de déchiffrer des messages sans connaitre les clés de déchiffrement. Les méthodes de cryptanalyse sont bien sûr très nombreuses et dépendent en grande partie du type d'algorithme auquel en est confronté. La science qui embrasse à la fois la cryptographie et la cryptanalyse s'appelle la **Cryptologie**.

La définition de la paire (e, d) constitue un système cryptographique. Les systèmes cryptographiques les plus utilisées peuvent être classés en deux types : système cryptographique a clé symétrique et système cryptographique a clé asymétrique (à clé publique).

2.1.4.1. Cryptographie à clé symétrique

Dans la cryptographie symétrique, appelée aussi cryptographie à clé secrète, une seule et même clé *K* est utilisée pour le chiffrement et le déchiffrement du message *M* (cf. figure 2.1).

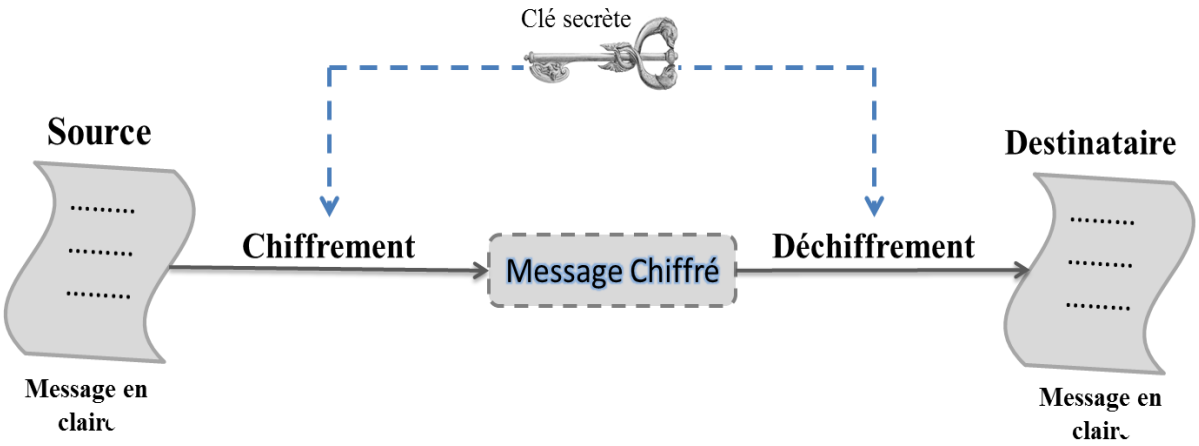


Figure 2.1 : Chiffrement symétrique

En générale, on utilise ce système pour assurer la confidentialité des données. Pour cela, chaque interlocuteur qui désire communiquer les données confidentielles doit partager une clé secrète avec son partenaire. Cette clé est utilisée par l'expéditeur pour chiffrer le message avant de l'envoyer et par le destinataire pour déchiffrer le message reçu.

Ce type de cryptographie a l'avantage d'être rapide car le nombre de clés ainsi que les calculs sont réduits. Mais le problème majeur de **Cryptographie symétrique reste la distribution de clé** et l'envoi de cette unique clé de chiffrement et de déchiffrement à tous les utilisateurs de façon sécurisée. Comme exemples d'algorithmes de chiffrement symétrique, on peut citer DES (*Data Encryption Standars*) et AES (*Advanced Encryption Standars*).

2.1.4.2. Cryptographie asymétrique

La cryptographie asymétrique appelée aussi la cryptographie à clé publique, est un procédé utilisant une paire de clé (clé privée et clé publique), la clé publique c'est pour chiffrer les messages à envoyer, et clé secrète (privée) pour déchiffrer les messages reçus

(cf. figure 2.2). La clé privée reste secrète et la clé publique peut être connue par les autres interlocuteurs.

La cryptographie asymétrique n'est pas seulement utilisée pour assurer la confidentialité, mais aussi pour assurer d'autres propriétés comme l'authentification qui est basée sur l'utilisation un mécanisme cryptographique appelé la signature numérique qui prouve l'origine des données.

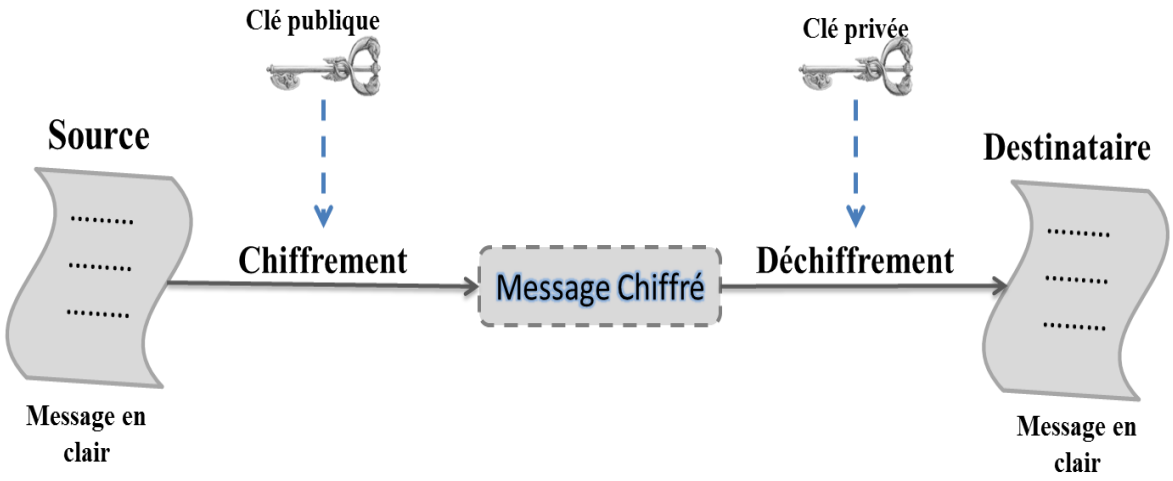


Figure 2.2 : Chiffrement asymétrique

Le principal avantage de la cryptographie asymétrique est de permettre un échange de données de manière sure sans aucun dispositif de sécurité. Les utilisateurs n'auront plus besoin d'échanger une clé secrète entre eux, les communications impliquent l'utilisation des clés publiques et aucune clé privée n'est transmise au partagée.

2.1.5. Cryptographie à seuil

Le partage de secret repose sur le concept de détention d'une portion d'une information secrète par plusieurs personnes, comme un coffre-fort bancaire dont l'ouverture est commandé par l'introduction simultanée de plusieurs clés. Le partage de secret est traditionnellement utilisé en informatique pour scinder les clés de chiffrement en plusieurs partie de sorte que chacune d'elles possède une portion de la clé. Le concept de la cryptographie a seuil est inventé par Shamir [sha79]. Il a proposé un mécanisme basé l'interpolation polynomiale. Il permet le calcule et le partage d'une valeur secrète S à

un ensemble de n serveurs, sans que chacun d'eux connaisse sa valeur. A partir d'au moins k serveurs on peut reconstruire le secret, si le nombre de serveurs est inferieur a k, aucune information n'est obtenus sur le secret S. cette technique de cryptographie a été combinée avec le système cryptographique asymétrique RSA pour avoir un système qui permet de partager le pouvoir de signature à un ensemble de serveur [ZH99]

2.1.5.1. Le protocole de partage du secret

Ce protocole permet de mettre en commun un secret S entre plusieurs serveurs (s_1,s_2,…s_n) de tel sorte qu'à partir de seulement de k parts on peut reconstruire le secret S. on crée un polynôme F(x) de degré k-1 avec des coefficients aléatoire en mettant a_0= S. on choisit ensuite publiquement n points distincts X_i , tel que $X_i \neq 0$, et on distribue secrètement à chaque serveur s_i une part privée (X_i, $F(x_i)$). Le point X_i pourrait être n'importe quelle valeur publique qui identifie les serveurs s_i d'une manière unique. Pour simplifie la notion, nous mettant X_i = i par conséquent les parts privées sont dénotées par F(1), F(2),…..,F(n).

2.1.5.2. Le protocole de la reconstruction du secret

Ce protocole permet de reconstruire le secret S à partir d'un sous-ensemble de k parts. Etant donné k paires de points distincts (i, F(i)), il existe un polynôme unique F(x) de degré k-1 passant par tous les points. Ce polynôme peut être calculé à partir des points (i, F(i)) en utilisant l'interpolation de LaGrange [Sha79]

2.1.6. Fonction de Hachage

Une fonction de hachage h, est par définition une fonction à sens unique. Elle associe à un message de longueur variable un résultat de longueur fixe appelé condensé ou digest. Connaissant le condensé, il est impossible de trouver le message original. En plus, une fonction de hachage doit vérifier les deux contraintes suivantes:

1. Pour un message m de taille quelconque en entrée, h(m) génère toujours la même longueur fixe en sortie. L'application d'une fonction de hachage n'est pas trop consommatrice en matière de temps de calcul.
2. La fonction h doit être définie de manière qu'étant donné un message m1, nous ne pouvons pas trouver un autre message m1 = m2, tel que h(m1) = h(m2).

Le condensé représente, donc, l'empreinte digitale du document. Les algorithmes de hachage les plus connus sont: MD2, MD4, MD5, SH1 et RIPEMD -160.

2.1.7. Signature numérique

Un des avantages majeurs de la cryptographie à clé publique est qu'elle procure une méthode permettant d'utiliser des signatures numériques. Le paradigme de signature numérique (appelé aussi signature électronique) est un procédé permettant à la personne qui reçoit une information de contrôler l'authenticité de son origine, et également de vérifier que l'information en question est intacte. Ainsi, les signatures numériques des systèmes à clé publique permettent l'authentification et le contrôle d'intégrité des données. Une signature numérique procure également la non-répudiation, ce qui signifie qu'elle empêche l'expéditeur de contester ultérieurement qu'il a bien émis cette information. Ces éléments sont au moins aussi importants que le chiffrement des données, sinon davantage.

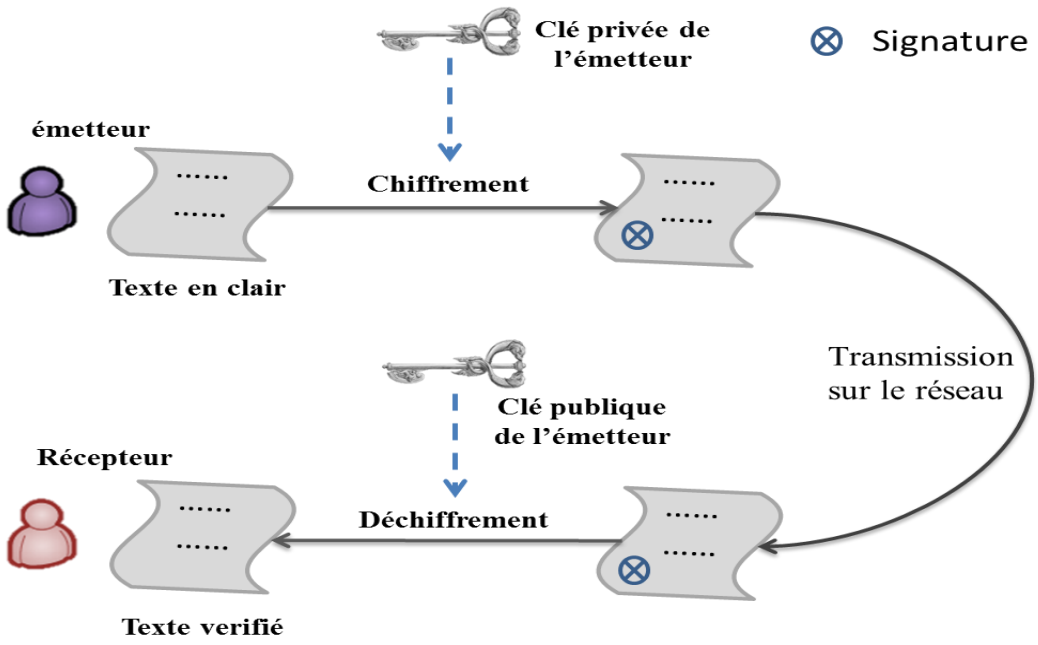


Figure 2.3 : Signature Numérique

2.2. PKI (Public Key Infrastructure)

Une infrastructure à clé publiques est un ensemble d'outils et de fonctions dédiés à la gestion de clé publique. Elle est composée d'un ensemble d'autorités de certification qui assurent la gestion des certificats. Une autorité de certification est un serveur particulier qui a le rôle de : générer, émettre, révoquer, et renouveler les clés publique des utilisateurs.

2.2.1. Les certificats

C'est un document électronique attestant qu'une clé publique est bien liée à une organisation, une personne physique, une machine ou une application. Il contient une clé publique, une identité du possesseur, et un certain nombre de propriétés le tous signé par une autorité de certification (cf. tableau 2.1). C'est la liaison de l'identité du possesseur et des propriétés à la clé publique, créée par la signature numérique qui constitue un certificat.

Propriété	Description
Version	La version du certificat
Numéro de Série	L'identifiant du certificat
Algorithme de signature	L'algorithme utilisé pour la signature du certificat
ID de signature	L'identité du signataire
Période de validité	La durée de vie du certificat
ID du porteur	L'identité du détenteur du certificat
Clé publique	La clé publique du détenteur du certificat
Attributs	Des informations sur les privilèges du certificat
Extensions	Extensions optionnelles
Signature	Signature numérique

Table 2.1 : Certificat à clé publique

2.2.2. Autorité de certification

C'est un tiers de confiance dont la responsabilité est essentiellement de certifier les clés publique des entités. La fonction d'une AC est analogue à celle d'un bureau chargé de l'émission des passeports dans un gouvernement. Un passeport est un document authentique, émis par une autorité appropriée, qui certifie que son détonateur est bien la personne qu'elle prétend être. C'est à toute fin pratique le document d'identité de la

personne. Tout pays qui a confiance en l'autorité d'un bureau de passeport d'un pays étranger honorera les passeports des ressortissants de ce pays. Ceci illustre bien ce qu'on entend par confiance entre les tiers. Tout comme un passeport, l'identité électronique de l'utilisateur d'un réseau, émise par une AC, est une preuve que cet utilisateur est connu de l'AC. Par conséquent, grâce au mécanisme de confiance entre les tiers, quiconque a confiance en l'AC, peut avoir confiance en l'identité du client. Les politiques de certification qui établissent l'AC sont d'une importance primordiale pour déterminer le degré de confiance qu'on peut avoir dans les AC

2.2.3. La confiance

Selon [ITU01], le terme confiance est défini comme suit : On dit qu'une entité fait confiance à une autre entité si et seulement si cette dernière se comporte exactement comme la première le prévoit. Ceci signifie qu'un nœud ne peut faire confiance à un autre nœud seulement si ce dernier se comporte d'une façon correcte. Cette définition a donné naissance à de deux grandes catégories de modèles de confiance : les modèles de confiance à base de coopération, et les modèles de confiance à base de certification. Dans la catégorie des modèles de confiance à base de coopération [MGF06], la mise en œuvre de la confiance est basée sur la notion de réputation. La réputation d'un nœud augmente quand il effectue correctement les tâches qui correspondent au bon fonctionnement du réseau, tel que le routage. Chaque nœud observe le comportement de ses voisins et déclare une accusation si l'un de ses voisins se comporte d'une manière incorrecte.

Un nœud peut faire confiance à un autre nœud si et seulement si ce dernier est certifié par un tiers nœud envers lequel le premier fait confiance. Ainsi, les nœuds utilisent la vérification des certificats pour établir des liens de confiance avec les autres nœuds. En effet, un certificat, est une structure de données dans laquelle une clé est liée à une identité (et éventuellement à certains autres attributs) délivrée par une tierce partie de confiance. Si cette dernière estime qu'un nœud donné est digne de confiance, elle lui délivre un certificat qui va lui permettre de prouver sa légitimité envers les autres nœuds du réseau. Dans cette catégorie de modèles de confiance, la relation de confiance entre les nœuds est gérée d'une manière transitive, telle que si A fait confiance à B, et B fait confiance à C, alors A peut faire confiance à C. Dans cette relation, l'intermédiaire B est la tierce partie de confiance. Cette dernière pourrait être une autorité centrale (par exemple une autorité de certification) ou un simple nœud intermédiaire. Dans la suite,

nous donnons une description des différentes techniques de cryptographie nécessaires pour la mise en œuvre des modèles de confiance à base de certification.

2.2.4. Organisation des autorités de certification

Les certificats générés par tous les utilisateurs ne peuvent pas être issus d'une même autorité de certification. Ainsi, il est nécessaire que le rôle soit réparti à travers plusieurs autorités organisées d'une manière particulière.

2.2.4.1. Le modèle hiérarchique

Le modèle hiérarchique est organisé comme suit. L'autorité de certification mère délivre un certificat spécifique, qui confère le rôle de certification à une ou plusieurs autres autorités, qui elles-mêmes à leurs tours délivrent des certificats à d'autres, et ainsi de suite (cf. figure 2.3). Au sommet, on trouve l'autorité racine dont le certificat est signé avec sa propre clé privée.

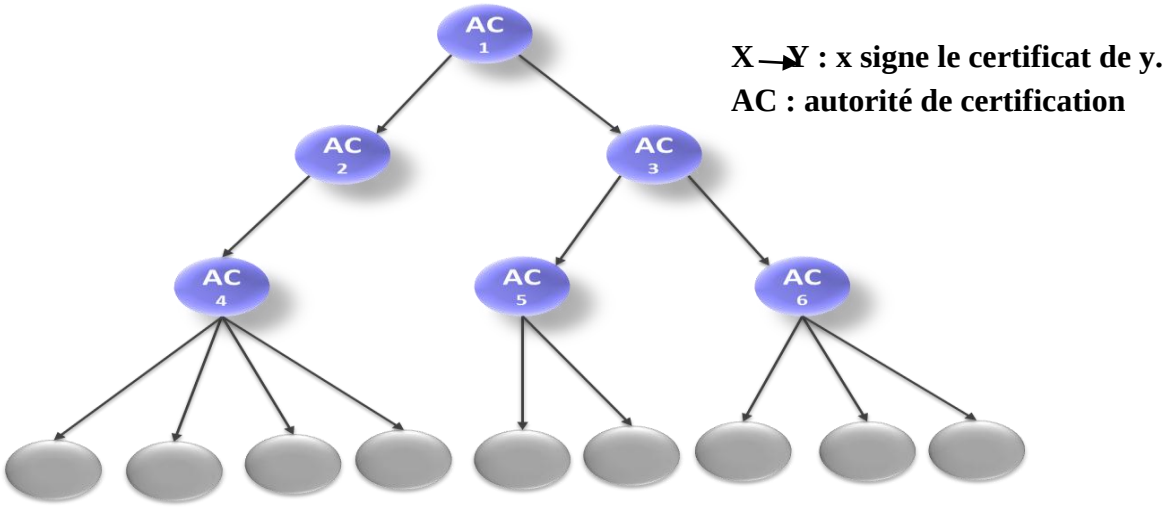
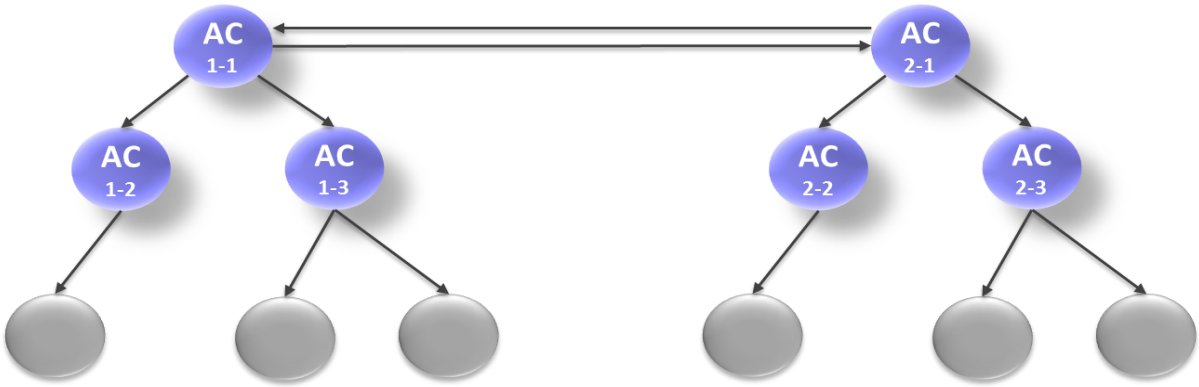


Figure 2.4 : le modèle hiérarchique

2.2.4.2. Le modèle croisé

Les relations croisées servent à relier deux hiérarchies d'autorité de certification de deux organismes différents. La racine de chaque hiérarchie signe le certificat à clé publique de l'autre pour former une passerelle (cf. figure 2.4). Ainsi n'importe quel

utilisateur de la première hiérarchie pourra vérifier la clé publique de n'importe quel utilisateur de l'autre hiérarchie.

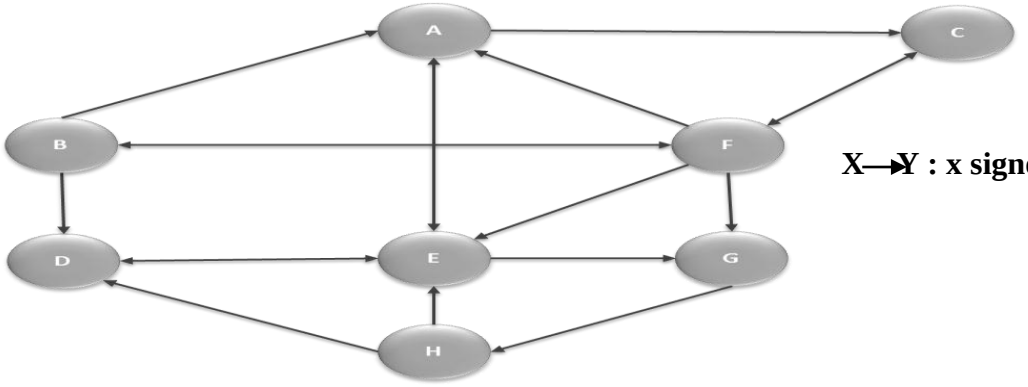


X⟶Y : x signe le certificat de y.
AC : autorité de certification

Figure 2.5 : Le modèle croisé

2.2.4.3. Le modèle anarchique

Ce type de modèle considère chaque utilisateur comme une autorité de certification ou il peut générer et signer des certificats pour les autres utilisateurs. Il utilise un graphe de confiance particulier, appelé web-of-trust, il consiste en l'établissement d'un réseau de gestion complétement distribuée de clés publique. Nous illustrons dans la figure 2.5 un exemple de graphe de confiance qui comporte un certain nombre d'utilisateurs. Chaque utilisateur délivre des certificats à l'ensemble des utilisateurs qu'il estime digne de confiance.



X⟶Y : x signe le certificat de y.

Figure 2.6 : Le modèle anarchique

Conclusion

Dans ce chapitre nous avons présenté les notions de base liées à la sécurité des réseaux ad hoc, ainsi qu'à la cryptographie et ces différents types qui feront objet du prochain chapitre. De plus nous avons décrit des infrastructures à clés publique PKI.

CHAPITRE III

Etat de l'art sur les modèles de confiance à base de certification dans les réseaux Ad hoc

Introduction

Dans le cadre des modèles de confiance à base de certification, la relation de confiance entre les utilisateurs est gérée d'une manière transitive, telle que si A fait confiance à B, et B fait confiance à C, alors A peut faire confiance à C. Dans cette relation, l'intermédiaire B est appelé tierce partie de confiance. Cette dernière pourrait être une autorité de certification ou un simple utilisateur intermédiaire. Ces deux points de vue ont donné naissance à deux catégories de modèles : (1) les modèles autoritaires, et (2) les modèles anarchiques. Dans ce chapitre, nous donnons un état de l'art sur les architectures des modèles de confiance à base de certification qui ont été proposés pour le cadre des réseaux ad hoc, en classifiant et comparant les modèles appartenant à chaque catégorie.

3.1. Quelques définitions préliminaires

Pour lever l'ambiguïté terminologique, nous donnons quelques définitions préliminaires :

- **Service de certification :** un ensemble de fonctionnalités liées à la certification qui sont assurées par des nœuds spéciaux dans le réseau et aux qui s'adressent les nœuds clients afin de procurer ou vérifier la validité d'un certificat.
- **Disponibilité du service de certification :** le degré d'accessibilité des nœuds qui assurent le service de certification dans le réseau.
- **Chaîne de certificats :** un ensemble ordonné des certificats nécessaires pour vérifier la filiation d'un certificat à un porteur.
- **Politique de certification :** un ensemble de règles et conditions à travers lesquelles on décide ou non à la délivrance d'un certificat à un nœud donné.

- **Autorités de certification homogènes :** deux autorités de certification dites homogènes si elles utilisent la même politique de certification.
- **Autorités de certification hétérogènes :** deux autorités de certification dites hétérogènes si elles n'utilisent pas la même politique de certification.
- **Modèle de certification autoritaire :** c'est un modèle de certification dans lequel les nœuds sont supervisés par une ou plusieurs autorités de certification.
- **Modèle de certification monopoliste :** c'est un modèle de certification dans lequel une seule autorité de certification prend le monopole du service de certification dans le réseau.
- **Modèle de certification oligopolistique :** c'est un modèle de certification dans lequel coexistent plusieurs autorités de certification hétérogènes où chacune d'elles supervise un sous-ensemble de nœuds dans le réseau.
- **Modèle de certification anarchique :** c'est un modèle de certification dans lequel les nœuds ne sont pas supervisés par une autorité de certification où le service de certification est assuré par les nœuds eux-mêmes avec la confiance transitive.

3.2. Critères d'évaluations des solutions existantes

Les problèmes liés à la distribution des clés publiques et à la gestion des certificats ont été largement étudiées dans le cadre des réseaux avec infrastructure. Cependant, la gestion de certificats dans les réseaux ad hoc présente des problèmes liés aux contraintes imposées par la nature du réseau : mobilité, limitation de ressources, absence totale ou partielle d'infrastructure, etc. Les problèmes étudies dans ce chapitre touchent les points suivants :

- **La disponibilité du service de certification :** dans les réseaux ad hoc, il est difficile de maintenir une autorité de confiance centrale fixe pour tout le réseau, à cause des déconnexions et ruptures de liens radio fréquentes dues principalement à la mobilité des nœuds et leurs ressources limitées. Par ailleurs, une telle autorité centrale serait un point d'échec singulier, ce qui entraverait la disponibilité des services de sécurité. Ainsi, l'une des exigences fondamentales est que le service de certification puisse assurer la disponibilité du service malgré les éventuelles déconnexions voire partitionnement du réseau.
- **La consommation de ressources :** les nœuds d'un réseau ad hoc peuvent être réduits à des systèmes embarqués mobiles disposant de peu de ressources d'énergie, de bande

passante, de stockage et de calcul. De ce fait, les protocoles du service de certification doivent être acceptablement optimisés en termes de calcul, communication, et stockage.

- **La scalabilité :** diverses applications dans les réseaux ad hoc impliquent un nombre important d'utilisateurs. Dans une telle condition, si le service de certification est assuré par une autorité centrale, cette dernière peut devenir surchargée à cause du nombre important de requêtes de certification. Autrement, si le service de certification est assuré d'une manière distribuée sur plusieurs nœuds du réseau, chaque participant doit maintenir un dépôt local, qui contient les certificats liés aux autres nœuds du réseau. Par conséquent, le stockage des certificats serait proportionnel à la taille du réseau, ce qui va compromettre les performances du service de certification à large échelle.

- **La gestion de l'hétérogénéité :** comme dans le cas des réseaux avec infrastructure, les autorités de certification pourraient être hétérogènes, de même dans les réseaux ad hoc. Ceci signifie que deux nœuds ou plus appartenant à différents domaines peuvent avoir le besoin de s'authentifier. Dans ce cas, il doit y avoir une certaine relation de confiance entre les deux domaines.

3.3. Classification

Dans la figure 3.1, nous proposons une classification des modèles de confiance à base de certification existants pour les réseaux ad hoc. Nous les avons organisés en deux catégories selon l'existence ou pas des autorités centrales :

1. Modèles autoritaires : dans cette catégorie, il existe une ou plusieurs autorités de certification dans le réseau. Selon le nombre d'autorités, nous organisons cette catégorie en modèles monopolistes et modèles oligopolistiques :

a. Modèles monopolistes : dans cette sous-catégorie, le service est assuré par une autorité de certification. Cette dernière est distribuée sur plusieurs serveurs qui assurent collectivement le service de certification à travers un schéma de cryptographie à seuil (k,n). Ceci signifie que la clé privée de l'autorité de certification est divisée en n parts privées délivrées à chacun des serveurs. Pour délivrer un certificat à un nœud donné, chaque serveur génère un certificat partiel (certificat signé en utilisant une part privée), de telle sorte que la combinaison de n'importe quel sous-ensemble de k certificats partiels permet de générer un certificat signé par la clé privée de l'autorité de certification. Ce

modèle peut s'appuyer sur une seule autorité centralisée ou bien sur plusieurs autorités organisées de manière hiérarchique. de certification.

i. **Autorité centralisée :** le service de certification dans le réseau tout entier est assuré par une unique autorité de certification qui est distribuée sur plusieurs serveurs.

ii. **Autorités hiérarchiques :** le service de certification est assuré par plusieurs autorités de certification homogènes organisées en hiérarchie. Chacune ou certaines autorités de certification sont distribuées sur plusieurs serveurs.

b. Modèles oligopolistiques : dans cette sous-catégorie, le service de certification se compose de plusieurs autorités de certification. Chaque autorité a sa propre politique de certification. Chacune ou certaines autorités de certification sont distribuées sur plusieurs serveurs.

2. Modèles anarchiques : dans cette catégorie de modèles, il n'y a aucune autorité centrale. Chaque utilisateur dans le système agit en tant qu'autorité de certification. La propagation de la confiance dans le réseau forme un graphe entre les utilisateurs, appelé graphe de confiance (ou *web-of-trust*), qui est géré par les utilisateurs eux-mêmes. Ce modèle décentralisé dans sa nature, est bien adapté aux réseaux mobiles ad hoc. Dans cette catégorie de modèles, deux opérations principales sont nécessaires : (1) la construction du graphe de confiance, et (2) la découverte des chaînes de certificats. Nous organisons cette sous-catégorie en modèles proactifs et modèles réactifs :

a. Modèles proactifs : dans cette sous-catégorie, le protocole d'échange de certificats est exécuté systématiquement entre les nœuds voisins. Ainsi, si un nœud client a besoin d'une chaîne de certificats, il les récupère directement à partir de son dépôt.

b. Modèles réactifs : dans cette sous-catégorie, le protocole de collection de certificats s'exécute à la demande. Quand un nœud à besoin de vérifier un certificat, à cet instant il collecte à travers un protocole distribué la chaîne de certificats appropriée.

Dans ce qui suit, nous donnons une présentation de quelques solutions de chaque catégorie. Nous discutons les avantages et inconvénients de chaque modèle. Ensuite, nous présentons une étude comparative des solutions existantes par rapport aux critères cités dans la section 3.2.

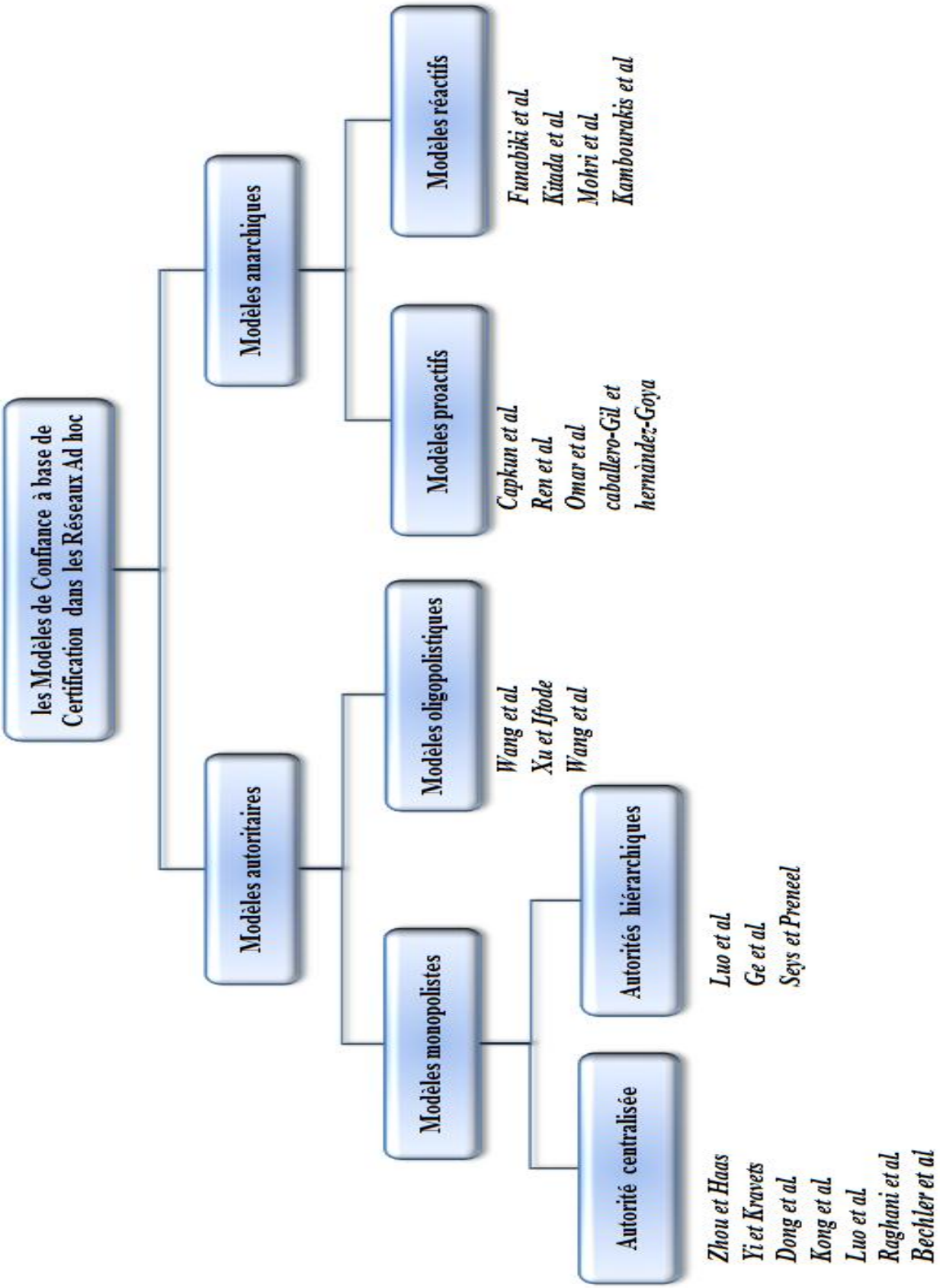


Figure 3.1 : Classification des modèles de confiance à base de certification dans les réseaux ad hoc

3.4. Modèles autoritaires

Dans cette section, nous présentons et discutons les modèles de confiance à base de certification appartenant à la catégorie des modèles autoritaires.

3.4.1. Modèles monopolistes

Dans cette classe de modèles, le service de certification est assuré par une ou plusieurs autorités de certification homogènes

3.4.1.1. Autorité centralisée

Dans cette sous-classe de modèles, le service de certification est assuré par une autorité de certification centrale, qui est distribuée sur plusieurs serveurs.

Solution de Zhou et Haas

Zhou et Haas [ZH99] sont les premiers à avoir exploré la conception d'un modèle de confiance à base de certification pour les réseaux ad hoc en se basant sur la cryptographie à seuil. Ils ont supposé l'existence d'une autorité de certification centrale distribuée sur un ensemble de serveurs suivant un schéma de cryptographie à seuil (k, n). Chaque serveur est capable de produire un certificat partiel en utilisant sa part privée, de telle sorte que la combinaison d'au minimum k certificats partiels produit un certificat valide signé par la clé privée de l'autorité de certification. Le système comporte trois types de nœuds : nœuds clients, serveurs, et combinateurs. Les nœuds clients sont les utilisateurs du système, tandis que les serveurs et combinateurs représentent l'autorité de certification. Les serveurs sont responsables de produire les certificats partiels pour les nœuds clients, et les combinateurs sont les responsables de les combiner pour générer un certificat valide et le transmettre au nœud client. Nous illustrons à travers la figure 3.2 le fonctionnement général de leur modèle. Tout d'abord, un nœud client sollicite un combinateur pour un certificat, ensuite ce dernier redirige la requête aux serveurs. Chaque serveur répond au combinateur par un certificat partiel afin de construire le certificat complet.

Avec ce modèle, même si un adversaire découvre les parts privées de moins que *k* serveurs, il ne pourra pas reconstruire la clé privée de l'autorité de certification, ce qui procure au système une certaine robustesse. Cependant, les auteurs n'ont pas explicité diverses opérations. Entre autre, comment un nœud exécute le protocole de certification avec *k* serveurs quand les serveurs sont dispersés dans un large réseau ? Comment

maintenir au moins *k* serveurs disponibles pour chaque nœud client ? Et comment les nœuds localisent les serveurs ? Gérer toutes ces opérations rend l'entretien du système complexe. Egalement, les auteurs n'ont pas discuté la valeur optimale du seuil *k*, qui influence fortement sur deux paramètres très importants : la disponibilité et la robustesse du service de certification. En effet, si la valeur de *k* est petite, ce sera relativement moins difficile de compromettre les parts privées du service de certification. Dans ce cas, la robustesse du système diminue, mais la disponibilité du service devient de plus en plus forte. Alors que, si la valeur de *k* est grande, l'adversaire devra compromettre un nombre important de serveurs pour pouvoir compromettre la clé privée de certification, ce qui rend le système beaucoup plus robuste. Mais dans ce cas, les nœuds clients devront solliciter un nombre important de serveurs pour pouvoir satisfaire leurs requêtes, et donc la disponibilité du service de certification s'affaiblit.

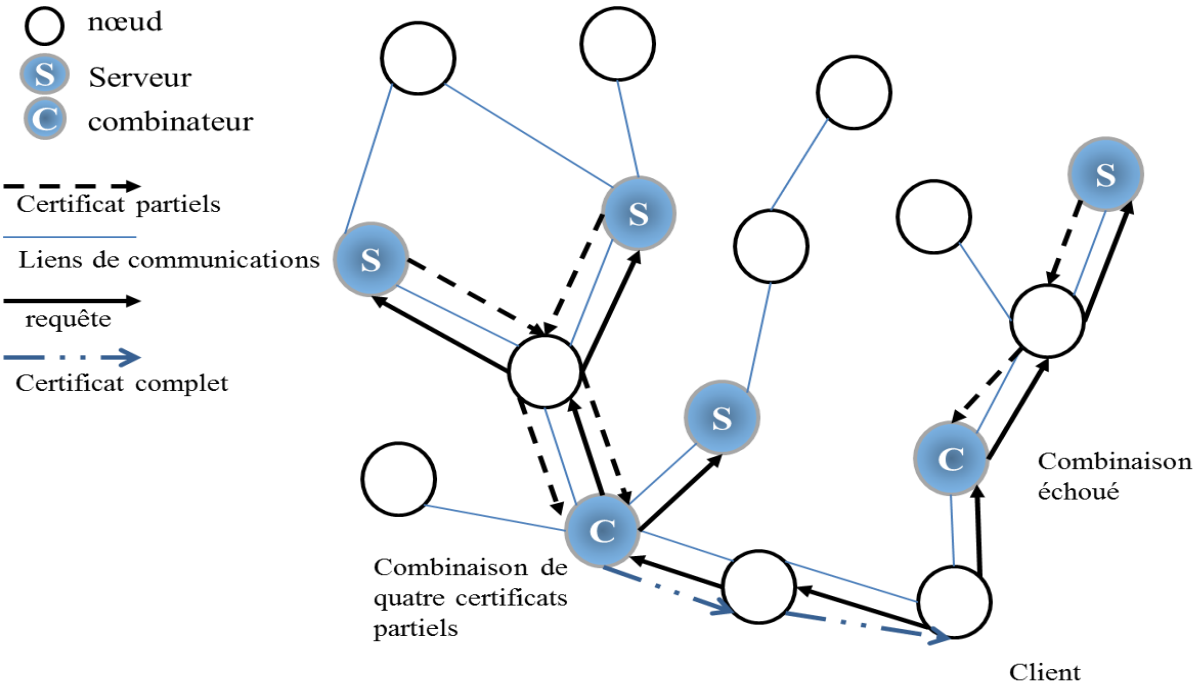


Figure 3.2: Zhou et Haas (*k* = 3, *n* = 6)

Solution de Yi et Kravets

Dans [YK03], Yi et Kravets ont proposé une architecture similaire à celle de Zhou et Haas. Le service de certification est assuré par un ensemble de nœuds spéciaux, choisis comme serveurs nommés MOCA (*MObile Certification Authority*). Ce modèleest destiné aux réseaux ad hoc dans lesquels les équipements sont hétérogènes. Deux principaux critères ont été considérés : la sélection et la localisation des serveurs. Les serveurs sont

choisis parmi les nœuds existants du réseau qui sont les plus performants en termes de traitement et/ou physiquement les plus sécurisés. Chaque nœud du réseau est préconfiguré par un protocole de certification MP (*MOCA certification Protocol*), qui permet la localisation des serveurs MOCA dans le réseau.

Comparant ce modèle à celui de Zhou et Haas, celui-ci limite le choix des serveurs qui assurent le service de certification. Le service est assigné exclusivement aux nœuds physiquement sécurisés et/ou ayant une puissance supérieure de traitement, ce qui rend le système robuste et performant en termes de délai de réponse. Cependant, les auteurs ont laissé ouvert une question importante : comment et qui juge le niveau de sécurité en choisissant les serveurs ?

Solution de Dong et al.

Dans [DSY07], Dong et al. ont proposé un modèle basé sur la clusterisation, et ils ont étudié en particulier le problème de localisation des serveurs de l'autorité de certification.

Le modèle organise le réseau en plusieurs clusters. Chaque leader de cluster (CH - *Cluster Head*) maintient une table d'informations concernant la localisation actuelle des serveurs de l'autorité de certification qui se trouvent dans son cluster local et/ou ceux qui se trouvent dans les autres clusters. L'opération de certification est exécutée comme suit. Quand un nœud client C_i nécessite un certificat, il envoie sa requête au leader de son cluster CH_i pour obtenir des informations sur les serveurs de l'autorité de certification dans son cluster local. Ensuite, il choisit *k* serveurs selon la réponse de *CHi*, et il leur envoie la requête de certification. Le processus de certification est maintenu par le leader de cluster où il reçoit l'ensemble des certificats partiels de *Ci* pour les combiner et lui renvoyer le certificat valide. Si le nombre de serveurs dans le cluster local n'est pas suffisant (moins de *k* serveurs), le *CHi* sollicite les serveurs de l'autorité de certification appartenant aux autres clusters. Ainsi, le *CHi* envoie la requête à l'ensemble de leaders des autres clusters pour avoir des informations de localisation, puis il sollicite les serveurs eux-mêmes pour les certificats partiels manquants. Ce modèle rend le service de certification flexible, vu que les informations liées aux serveurs de l'autorité de certification sont gérées par les leaders de clusters, ce qui réduit les délais de réponses. En effet, les auteurs ont donné plus d'intérêt au critère de localisation des serveurs dans le réseau. Cependant, ils ont laissé ouvert une question importante : s'il y a assez de

serveurs dans un cluster donné, parmi lesquels il y en a ceux qui sont compromis, comment les éviter ?

Solution de Kong et al.

Dans [KZL01], Kong et al. ont proposé un nouveau genre d'autorité de certification basée sur la cryptographie à seuil. Contrairement aux modèles précédents qui fixent un ensemble de serveurs, dans ce modèle, le service de certification est distribué sur tous les nœuds du réseau. Ainsi, chaque nœud détient une part privée du service de certification. Quand un nœud nécessite un certificat, il envoie sa requête à l'ensemble des nœuds de son voisinage à un seul saut. Si la coalition considère que le nœud demandeur est digne de confiance, ils lui délivrent des certificats partiels, qui seront combinés par le nœud demandeur lui-même (nous illustrons dans la figure 3.3, le fonctionnement général de ce modèle). Dans ce modèle, il est supposé que chaque nœud a au moins *k* nœuds voisins légitimes. De plus dans ce modèle, de nouveaux nœuds peuvent être intégrés dynamiquement au service de certification au moment de leurs adhésions au réseau. Pour réaliser cette adhésion, le nouveau nœud doit seulement envoyer une requête à un ensemble de *k* nœuds voisins, qui vont lui répondre chacun avec une part privée partielle. Ainsi, le nouveau nœud combine l'ensemble des parts privées partielles reçues pour avoir sa part privée, et à partir de là il pourra participer au service de certification.

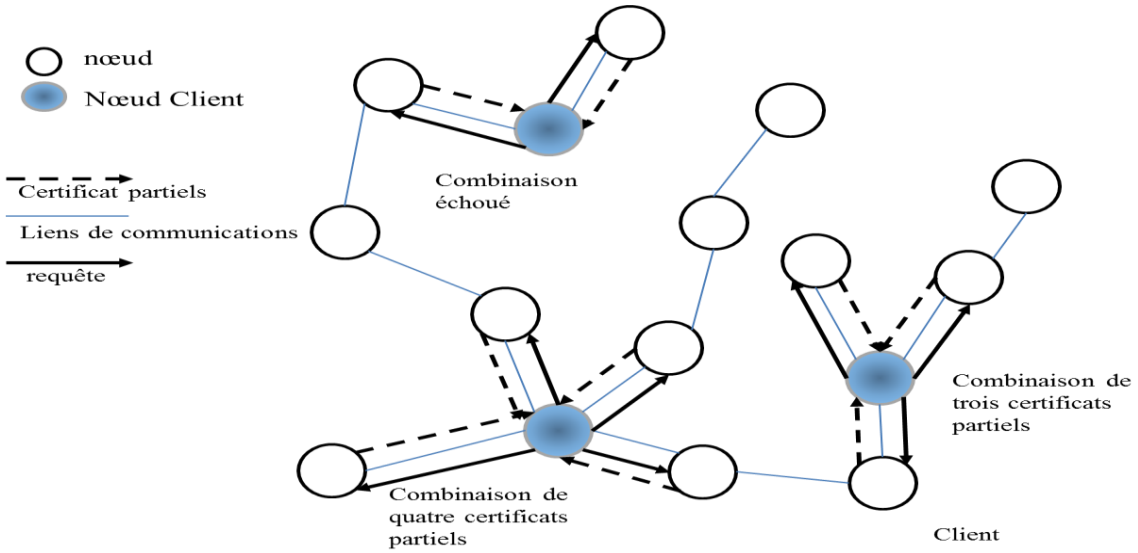


Figure 3.3: Kong et al. (*k* = 3, *n* = 16)

Contrairement aux modèles décrits précédemment, ce modèle est complétement distribué où le service de certification est distribué sur tous les nœuds du réseau. Ainsi, la

disponibilité du service de certification augmente. D'autre part, ce modèle réduit la complexité de localisation des serveurs de l'autorité de certification où le nœud sollicite directement son voisinage. L'une des faiblesses principales de ce modèle est qu'il est difficile de maintenir *k* voisins disponibles pour chaque nœud, vu que le seuil est un paramètre globalement fixe et préconfiguré pour chaque nœud du réseau. Comme ce modèle permet aux nouveaux nœuds d'obtenir leurs parts privées à partir des nœuds membres du réseau, un adversaire pourrait prendre plusieurs identités (*Sybil attack*[Dou02]) et pourra collecter assez de parts privées lui permettant de reconstruire la clé privée du service de certification.

Solution de Luo et al

Dans [LL00, LZK02], Luo et al. ont proposé l'intégration d'un service de révocation de certificats pour le modèle de Kong et al. Le mécanisme de révocation est fondé sur l'hypothèse que tous les nœuds surveillent le comportement de leurs voisins directs et maintiennent leurs propres listes de révocation de certificats. Si un nœud découvre que l'un de ses voisins se comporte d'une manière incorrecte, il rajoute son certificat à la liste de révocation et diffuse une déclaration d'une accusation contre ce nœud vers tous les autres. Tout nœud recevant l'accusation, il vérifie sur sa liste de révocation de certificats que l'accusation n'est pas parvenue d'un nœud dont le certificat a été révoqué. Si le certificat de l'accusateur a été révoqué, l'accusation sera ignorée. Autrement, le nœud accusé sera marqué suspect.

Solution de Raghani et al.

Dans [RTJ06], Raghani et al. ont proposé une autorité de certification complétement distribuée où le service de certification est distribué sur tous les nœuds du réseau. Cependant, afin d'assurer une forte disponibilité du service de certification, ils ont proposé de maintenir la valeur du seuil *k* dynamique, ajustable systématiquement selon le nombre moyen de voisins directs de chaque nœud du réseau. Chaque nœud exécute, périodiquement, un protocole de découverte de voisinage, calcule le nombre de ses voisins directs, et envoie cette information à un nœud spécial, nommé leader. Ce dernier, en recevant le nombre de voisins directs de chaque nœud, calcule le degré moyen *d* du réseau. S'il est inférieur à la valeur actuelle du seuil *k*, le leader recalcule une nouvelle valeur de *k*, tel que knew = max(2, *d*). Quand le leader décide de changer la valeur du

seuil, il diffuse cette dernière à tous les nœuds du réseau. Ensuite, les nœuds mettent à jour leurs parts privées correspondantes à la nouvelle valeur de *k*.

Comparant ce modèle à celui de Kong et al, la disponibilité du service de certification est améliorée grâce à la flexibilité de la valeur du seuil *k*. Cependant, cette solution provoque une charge importante de transmission. En effet, à chaque changement du degré moyen du réseau, ce qui est fréquent dans les réseaux ad hoc, une nouvelle valeur de *k* est diffusée. En plus, cette opération sera suivie par le recalcule des parts privées de chaque nœud, ce qui implique également une charge importante et périodique de traitement. Il reste à signaler qu'il existe un autre cas très important qui nécessite le changement de la valeur du seuil, qui est lié à la robustesse du service de certification.

Par exemple, quand le nombre de nœuds malveillants augmente dans le réseau, la valeur du seuil doit être augmentée, afin de rendre le système résistible face aux faux certificats partiels générés. Une telle mesure n'a pas été envisagée dans ce modèle.

Solution de Bechler et al.

Dans [BHK04], Bechler et Cie ont proposé un travail basé sur le concept de groupage pour distribuer l'autorité de certification. Ce modèle est d'une architecture qui utilise le schéma de la cryptographie à seuil (k, n) pour distribuer l'autorité de certification. L'idée consiste à distribuer la clé privée de autorité de certification sur les chefs de groupe appelés (cluster-Head :CHs). Chaque CH possède un fragment de la clé privée de *CA*. Si un nœud visiteur veut certifier sa clé publique et obtenir son certificat, il doit avoir au moins un certain nombre (w) de certificats générés par les nœuds garants. Une fois que les certificats des garants sont rassemblés, le nœud visiteur doit demander au moins k autres certificats aux CHs pour avoir le certificat final du réseau. Le nombre k représente le paramètre de la cryptographie à seuil. Cependant, le nombre w est le seuil de garanti nécessaire pour la génération d'un certificat par le nœud chef de groupe (*AC*). Les inconvénients de cette architecture sont les suivants : premièrement, cette approche n'est pas réaliste, car les nœuds garants n'ont pas d'information sur les nœuds visiteurs qui sont généralement de nouveaux arrivants dans le réseau (pas d'historique sur ces nœuds). Deuxièmement, même si le nœud visiteur a réussi à rassembler les w certificats que lui apporte le garant, il ne peut pas avoir son certificat final car il lui faut k autres certificats partiels des CHs pour obtenir son certificat final. Troisièmement, le trafic réseau généré

par chaque nœud qui veut obtenir son nouveau certificat est au moins égal à 2 * (*w* + *k*) paquets. Un autre inconvénient se présente dans le cas de la fusion de plusieurs réseaux pour déterminer la clé secrète de la nouvelle *CA* du réseau fusionné. Or, le mélange des clés secrètes n'est pas possible, donc une seule clé est considérée comme la clé secrète du nouveau réseau et est appelée « la clé dominante ». Cette clé est sélectionnée en fonction du nombre de groupes (clusters) dans le réseau. La clé de *CA* du réseau qui possède le plus grand nombre de groupes devient la nouvelle clé de *CA* de tous les réseaux fusionnés. Cette procédure présente une vulnérabilité, car dans cette architecture, n'importe quel nœud peut former son propre cluster. Par conséquent, un ensemble de nœuds malicieux peuvent former un réseau avec le nombre maximum de groupes (clusters) dans le but de prendre le contrôle de *CA* une fois qu'ils auront fusionné avec le réseau victime.

3.4.1.2. Autorités hiérarchiques

Dans cette sous-classe de modèles, le service de certification est assuré par une hiérarchie de plusieurs autorités de certification homogènes.

Solution de Luo et al.

Dans [LHE05], Luo et al. ont proposé une autorité de certification hiérarchique pour les Roseau mixtes 2, nommée DICTATE (*DIstributed CerTification Authority with probabilisticfrEshness for ad hoc networks*). Ce modèle inclut deux niveaux d'autorités de certification : une autorité de certification mère (mCA - *mother Certification Authority*) qui se trouve dans la partie du réseau avec infrastructure, et une autorité de certification distribuée sur un ensemble de serveurs (dCA - *distributed Certification Authority*) qui se trouvent dans la partie du réseau ad hoc, et qui partagent la clé privée de l'autorité de certification en utilisant la cryptographie à seuil. Les nœuds dans la partie du réseau ad hoc peuvent être collectivement isolés de l'autorité mCA, mais ayants toujours le besoin du service de certification. Pour cette raison, l'autorité mCA délègue son rôle à l'ensemble des serveurs dCA pendant la période d'isolement afin d'assurer la disponibilité du service de certification. Quand la partie du réseau ad hoc est détachée du réseau, les nœuds clients soumettent leurs requêtes de certification aux serveurs dCA. Nous illustrons dans la figure 3.4 le fonctionnement général de ce modèle. En comparant ce modèle aux modèles précédents, le nombre de serveurs qui assurent le service de certification est flexible où l'autorité de certification mère a le pouvoir de

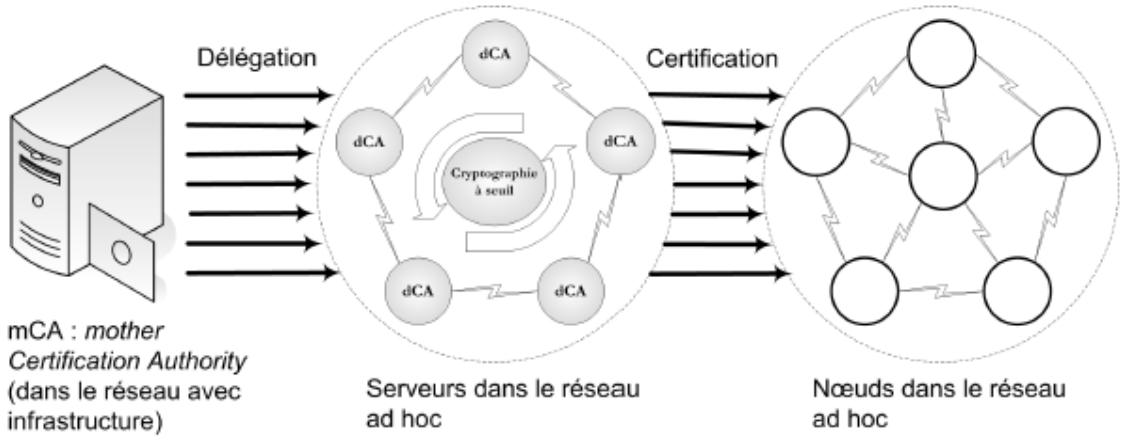


Figure 3.4: DICTATE, Luo et al.

déléguer son autorité à de nouveaux serveurs. Cependant, ce modèle n'est pas purement adapté aux réseaux ad hoc, à cause de la nécessité d'une administration centralisée. En plus, le point sensible dans ce modèle est l'autorité centrale mCA, qui assure la grande partie des services de sécurité : admission des nœuds, délégation du pouvoir de certification aux serveurs dCA, distribution des parts privées, etc. Les auteurs ont bien discuté la robustesse du service de certification contre les serveurs compromis, cependant le cas de la compromission de l'autorité de certification mère n'a pas été étudié

Solution de Ge et al

Dans [GLG09], Ge et al. ont proposé une amélioration du service de délégation indépendamment du réseau avec infrastructure. Le service de certification inclut un groupe de serveurs distribués qui ont le pouvoir de déléguer le service de certification à d'autres serveurs dans le réseau. En cas de besoin, quelques nœuds ordinaires seront sélectionnés et convertis en serveurs auxiliaires par les serveurs originaux qui seront activés plus tard. Quand un certificat spécifique doit être délivre dans un segment isolé du réseau où il n'y a pas assez de serveurs, quelques serveurs auxiliaires seront activés. Une fois activés, les serveurs auxiliaires se comportent exactement comme des serveurs ordinaires en termes de capacité de certification. Quand le service d'un serveur auxiliaire n'est plus nécessaire, il sera ensuite désactivé.

Solution de Seys et Preneel

Dans [SP03], Seys et Preneel ont proposé une autorité de certification hiérarchique de plusieurs niveaux. Chaque niveau est assuré par un ensemble de nœuds qui partagent le

service de certification à travers un schéma de cryptographie à seuil (*k, n*). Ainsi, dans chaque niveau *i* de l'hiérarchie, une clé privée *Ki*est partagée entre *n* nœuds. Dans le niveau supérieur de la hiérarchie (niveau 0), une clé privée est employée pour délivrer des certificats aux nœuds du niveau 1. De même, les nœuds appartenant au niveau 1 partagent la clé privée de certification du niveau 2, et ainsi de suite. Si un nœud à un certain niveau nécessite un certificat, il doit solliciter l'ensemble des nœuds du niveau supérieur pour collecter et combiner *k* certificats partiels.

En comparant aux modèles précédents, ce modèle assure une certaine robustesse, vu que chaque niveau dans l'hiérarchie est assuré par un service de certification à base de cryptographie à seuil. Cependant, afin de vérifier une chaîne de certificats selon une hiérarchie comportant *h* niveaux, le nœud doit combiner *k* certificats partiels pour chaque niveau, et en totalité $k \times h$ certificats partiels, ce qui implique une charge importante de calcul.

3.4.2 Modèles oligopolistiques

Dans cette classe de modèles, le service de certification est composé de plusieurs autorités de certification hétérogènes où chacune d'elles a sa propre politique de certification.

Solution de Wang et al.

Dans [WZL03], Wang et al. ont proposé un modèle comportant plusieurs autorités de certification hétérogènes. Chaque autorité de certification est distribuée sur un ensemble de serveurs. Chaque nœud du réseau maintient une liste de serveurs envers lesquels il fait confiance. Quand un nœud nécessite d'authentifier un autre nœud dans le réseau, les deux nœuds s'échangent leurs listes de serveurs. Ensuite, ils comparent les listes en essayant de trouver *k* serveurs en commun. Si l'opération échoue, ils essayent de continuer la recherche auprès de leurs voisins d'un et deux sauts afin de trouver d'éventuelles intersections.

En comparant aux modèles présentés, ce modèle permet la coexistence de plusieurs autorités de certification hétérogènes dans le réseau. Cependant, les auteurs n'ont pas justifié le choix de solliciter exclusivement les nœuds qui se trouvent à un et deux sauts si l'authentification échoue. Si le nombre de nœuds à solliciter est important, le nœud client augmente ses chances de trouver autant de serveurs communs entre les deux nœuds.

Solution de Xu et Iftode

Dans [XI04], Xu et Iftode ont proposé un modèle comportant plusieurs autorités de certification hétérogènes. Chaque autorité de certification est distribuée sur un ensemble de serveurs à base de cryptographie à seuil. Selon ce modèle, le réseau ad hoc est considéré comme un ensemble de sous-réseaux. Le service de certification de chaque sous-réseau est assuré par une autorité de certification. Chaque autorité de certification de chaque sous-réseau comporte un nœud spécial, nommé *leader*, qui assure la distribution des parts privées du service de certification. Egalement, il est supposé l'existence de certaines relations de confiance entre les autorités de certification de chaque sous réseau en formant un graphe de confiance. Ces relations sont introduites pour permettre la vérification des certificats délivrés par les autres autorités de certification.

Comme le modèle de Wang et al., l'avantage majeur de ce modèle est la prise en charge de l'hétérogénéité des autorités de certification. L'inconvénient de ce modèle est la charge de calcul induite pour la vérification des chaînes de certificats. En effet, pour vérifier un certificat d'un nœud appartenant à une autre autorité de certification, le nœud doit collecter la chaîne des certificats générés à partir de son autorité vers l'autorité de certification du nœud en question, en combinant pour chaque certificat, *k* certificats partiels.

3.4.3 Etude comparative

Dans le tableau 3.1, nous donnons une comparaison des différents modèles autoritaires par rapport au critère de la disponibilité du service de certification. Pour chaque modèle, nous estimons le degré de disponibilité du service de certification par rapport à la coalition des serveurs adoptée et le choix de la valeur du seuil *k*. Cette comparaison nous a permis de classifier les modèles de cette catégorie en quatre classes :

- **Classe A :** A cette classe appartient le modèle de Raghani et al. [RTJ06], qui assure un niveau très élevé de disponibilité du service de certification. Dans ce modèle, le rôle de l'autorité de certification est assuré par les nœuds eux-mêmes. En plus, ce modèle maintient la valeur de *k* dynamique, ajustée périodiquement par rapport au degré moyen *d* du réseau, afin que la disponibilité du service de certification reste toujours élevée.

Modèle	Serveurs	Seuil	Disponibilité du Service de Certification
Zhou et Haas	***n est statique.***	**Statique.**	**Moyenne**
MOCA	***n est*** **statique où les serveurs sont présélectionnés.**	**Statique.**	**Moyenne**
Dong et al.	***n est statique.***	**Statique.**	**Moyenne**
Kong et al.	***n est*** **dynamique, tel que chaque nœud est un serveur.**	**Statique.**	**Elevée**
Raghani etal.	***n est*** **dynamique, tel que chaque nœud est un serveur.**	**Dynamique.**	**Elevée**
Bechler et Cie	***n*** **est dynamique ou chaque nœud peut faire partie de AC**	**Statique.**	**moyenne**
DICTATE	***n est*** **dynamique, tel que des serveurs auxiliaires dCA peuvent être intégrés dynamiquement par l'autorité mCA.**	**Statique.**	**Elevée**
Ge et al.	***n est*** **dynamique, tel que des serveurs auxiliaires peuvent être intégrés dynamiquement par les serveurs originaux.**	**Statique.**	**Elevée**
Seys et Preneel	***n est*** **statique à chaque niveau de l'hiérarchie.**	**Statique à chaque niveau de l'hiérarchie.**	**Faible**
Wang et al.	***n est statique.***	**Statique, tel que chaquepair de nœuds doiventavoir en commun *k* serveurs.**	**Faible**
Xu et Iftode	***n est*** **statique où les serveurs sont présélectionnés.**	**Statique.**	**Moyenne**

Table 3.1 : Comparaison par rapport à la disponibilité du service de certification (*k* est la valeur du seuil, et *n* est le nombre de serveurs)

- **Classe B :** Dans cette classe, nous trouvons au sommet, le modèle de Kong et al. [KZL01], qui assure un niveau élevé de disponibilité du service de certification. Ceci est lié au fait que l'autorité de certification est distribuée sur tous les nœuds du réseau. Nous trouvons également dans cette classe, le modèle de Ge et al. [GLG09], dans lequel l'autorité de certification est assurée par *n* serveurs où des serveurs auxiliaires pourraient être dynamiquement intégrés dans les portions isolées du réseau. Nous trouvons également dans cette classe, le modèle DICTATE [LHE05]. Cependant, dans ce dernier les nouveaux serveurs à intégrer doivent être accessibles au serveur délégateur qui se trouve dans la partie du réseau avec infrastructure.

- **Classe C :** Dans cette classe, nous trouvons les modèles qui assurent un niveau moyen de disponibilité du service de certification. Dans cette classe, le nombre de serveurs *n* est statique. Les modèles : Zhou et Haas [ZH99], MOCA [YK03], Xu et Iftode [XI04], Bechler et Cie[BHK04]et Dong et al. [DSY07] appartiennent à cette classe.

- **Classe D :** Dans cette classe, nous trouvons les modèles qui assurent un faible niveau de disponibilité du service de certification due au nombre fixe de serveurs et à la complexité du protocole de certification. Le modèle de Seys et Preneel [SP03] inclut une hiérarchie d'autorités de certification de *h* niveaux où chaque niveau utilise un schéma de cryptographie à seuil. Ainsi, pour accomplir le processus de certification, $h \times k$ certificats partiels doivent être combinés. Par conséquent, le nœud client doit solliciter un nombre important de serveurs. Le modèle de Wang et al. [WZL03] appartient également à cette classe. Dans ce modèle, le système comporte plusieurs autorités de certification hétérogènes où chaque nœud fait confiance à un sous-ensemble de serveurs. Dans ce cas, une autre contrainte apparait selon ce modèle. Afin d'assurer un processus de certification significatif, le nœud doit collecter les certificats partiels auprès d'au minimum *k* serveurs en qu'il fait confiance, et en qui tous les autres nœuds font confiance. Cette condition réduit considérablement la disponibilité du service de certification.

Dans le **tableau 3.2**, nous donnons une comparaison globale des modèles de confiance à base de certification appartenant à cette catégorie. Nous récapitulons les principales leçons de cette comparaison dans les points suivants :

- La disponibilité du service de certification dépend de la capacité des nœuds d'atteindre les serveurs de l'autorité de certification. Cette propriété est strictement liée

à : (1) comment choisir les serveurs ? (2) combien de serveurs *n* ? et (3) comment choisir la valeur de *k* du mécanisme cryptographique à seuil utilisé ? Si le choix des serveurs est strict, par exemple en termes de stockage ou/et capacités de calculs des nœuds, le nombre de serveurs sera limité, et ainsi le service de certification sera moins disponible. Par contre, si la valeur de *n* est grande, les nœuds auront plus de chances d'avoir à leurs disponibilités *k* serveurs. Egalement, le choix de la valeur du seuil *k* influence fortement la disponibilité du service de certification. Si la valeur de *k* est réduite ou dynamiquement réglable selon la capacité d'accès des nœuds aux serveurs, le système peut achever une disponibilité très forte du service de certification. Autrement, si la valeur de *k* est grande, les nœuds seront obligés de solliciter un nombre important de serveurs, ce qui réduit par conséquent la disponibilité du service de certification.

- Dans cette catégorie des modèles autoritaires, la ressource la plus consommée est le traitement. Ce dernier est le résultat de la complexité du calcul induit par les algorithmes de chiffrement à clés publiques et la cryptographie à seuil. Par ailleurs, la charge de stockage est considérable afin de maintenir les dépôts de certificats à chaque serveur (ou à chaque nœud). La charge de communication est également importante, et ceci à cause du nombre important de certificats transmis dans le réseau. En effet, pour délivrer un certificat, *k* certificats partiels doivent être transmis.
- La scalabilité du service de certification dépend fortement du nombre d'autorités de certification dans le système. Ce critère est important dans le sens où le service de certification doit être assuré par plusieurs autorités de certification, au lieu de surcharger une seule à large échelle. En plus, le service de certification doit garder la possibilité d'intégrer de nouvelles autorités de certification auxiliaires en cas de besoin. Une solution alternative consiste à la distribution du rôle de l'autorité de certification à tous les nœuds du réseau afin d'éviter la centralisation du service. Dans ce cas, il est possible d'achever un bon niveau de scalabilité si les nœuds ne sont pas obligés de stocker tous les certificats générés dans le réseau.
- Dans cette catégorie, le problème d'hétérogénéité a été traité seulement dans quelques modèles. L'objectif principal est de mettre en place une passerelle de confiance entre les différentes autorités de certification.

Modèle	Disponibilité	Ressources	Scalabilité	Pris en considération de l'Hétérogénéité
Zhou et Haas	**Moyenne**	**Les serveurs stockent tous les certificats générés.**	**Non**	**Non**
MOCA	**Moyenne**	**Les serveurs sont puissants en termes de capacité de traitement. Le reste des nœuds maintient seulement une table de routes vers les serveurs.**	**Non**	**Non**
Dong et al.	**Moyenne**	**Chaque leader de cluster maintien des informations sur tous les serveurs.**	**Non**	**Non**
Kong et al.	**Elevée**	**Un nombre important de certificats stockés au niveau de chaque nœud.**	**Oui**	**Non**
Raghani et al.	**Elevée**	**Un nombre de messages important transmis périodiquement pour les valeurs du seuil, suivi par une mise à jour des parts privées des nœuds impliquant une charge importante de calculs.**	**Oui**	**Non**
DICTATE	**Elevée**	**Les serveurs stockent tous les certificats générés.**	**Oui**	**Non**
Ge et al.	**Elevée**	**Le calcul est expansif en générant les parts privées des serveurs auxiliaires.**	**Oui**	**Non**
Seys et Preneel	**Faible**	**Le calcul est expansif; la vérification des certificats implique la combinaison d'un grand nombre de certificats partiels.**	**Oui**	**Non**

Wangetal.	**Faible**	**Chaque nœud maintient une liste d'autorités de certification en qu'il fait confiance.**	**Oui**	**Oui**
Xu et Iftode	**Moyenne**	**Chaque autorité de certification maintient une table des relations de confiance avec les autres autorités.**	**Oui**	**Oui**
Bechler et al.	**Moyenne**	**Chaque leader de cluster maintien des informations sur tous les nœuds.**	**Non**	**Non**

Table 3.2: Comparaison globale

3.5. Modèles anarchiques

Dans cette section, nous présentons et discutons les modelés de confiance à base de certification appartenant à la catégorie des modèles anarchiques.

3.5.1. Modèles proactifs

Dans cette sous-catégorie, le protocole de collection de certificats est exécuté entre les nœuds voisins périodiquement. Si un nœud aura besoin de vérifier une chaîne de certificats, il les récupère directement à partir de son dépôt local.

Solution de Capkun et al.

Dans [CBH02, CBH03], Capkun et al. ont proposé un service de certification distribué sur tous les nœuds du réseau. Dans ce modèle, il est supposé qu'il existe des relations de confiance sociales entre les différents utilisateurs. Chaque utilisateur génère sa propre paire de clés (privée, publique) et sollicite un ensemble d'utilisateurs afin de lui délivrer des certificats correspondants à sa clé publique. Si un utilisateur *u* croit que la clé publique K*v* appartient à l'utilisateur *v*, il lui délivre un certificat signé par sa propre clé privée. De ce fait, un graphe de confiance sera établi entre tous les utilisateurs du réseau. Egalement, chaque nœud du réseau maintient un dépôt local de certificats, qui est mis à jour périodiquement à travers un protocole d'échange de certificats entre les nœuds voisins. Quand un utilisateur *u* nécessite d'authentifier la clé publique d'un utilisateur *v*,

les deux nœuds fusionnent leurs dépôts locaux et essayent de trouver une chaîne de certificats à partir de l'utilisateur *u* vers l'utilisateur *v* dans le dépôt fusionné (cf. figure 3.5). Si une telle chaîne n'est pas trouvée, *u* peut solliciter les *helper nodes*, qui sont les nœuds voisins à un ou deux sauts.

L'avantage majeur de ce modèle est l'autonomie du service de certification où aucune autorité n'est employée pour assurer le service de certification. Cependant, le graphe de confiance établit entre les utilisateurs peut ne pas être fortement connecté, ce qui peut empêcher l'authentification de certains utilisateurs du réseau. Un autre problème lié à ce modèle est la consommation de ressources. En effet, ce modèle provoque une charge importante de calcul induite par la vérification des chaînes de certificats à chaque authentification. Egalement, chaque utilisateur est requis de maintenir un dépôt local de certificats, qui est enrichi systématiquement par le protocole d'échange de certificats, ce qui provoque une charge importante de stockage et de transmission.

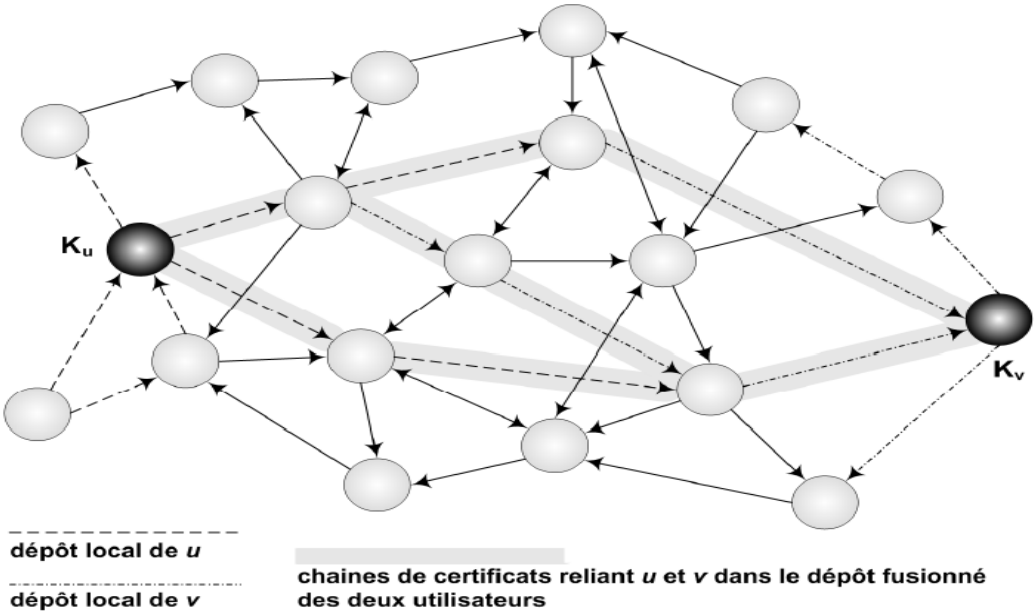


Figure 3.5: Capkun et al.

Solution de Ren et al.

Dans [RLW04], Ren et al. Ont proposé une version modifiée du modèle de Capkun et al. En se basant sur un serveur central pour initialiser le service de certification.

Ce serveur initialise le système en distribuant pour chaque nœud une liste contenant un ensemble d'identificateurs de quelques utilisateurs et leurs clés publiques. Ensuite, chaque nœud génère les certificats correspondants. Ainsi, un graphe de confiance sera établi entre les utilisateurs du réseau, et le système continuera son fonctionnement indépendamment du serveur central où l'authentification des nœuds sera assurée à travers des chaînes de certificats.

Comme le modèle de Capkun et al, l'avantage majeur de ce modèle est l'autonomie du service de certification qui ne dépend pas d'une autorité de certification.

Cependant, le service de certification est d´dépendant toujours d'un serveur central à l'étape d'initialisation. En plus, l'efficacité du service de certification de ce modèle est fortement liée à la longueur des listes délivrées par le serveur central. Plus la longueur est grande, plus chaque nœud génère un nombre important de certificats, ce qui augmente la disponibilité du service de certification. Mais dans ce cas, la charge de stockage sera importante.

Solution de Omar et al.

Dans [OCB09]. Omar et al. ont proposé un nouveau style de toile de confiance afin de produire un modèle de confiance entièrement distribuée pour les réseaux mobiles ad hoc. Le régime permet aux nœuds de générer, stocker et distribuer leurs certificats de clé publique sans serveur central ou tiers de confiance. Comme les régimes précédents, dans celui-ci, les clés publiques ou privés des utilisateurs sont créés par eux-mêmes, et l'authentification par clé est effectuée par l'intermédiaire des chaînes de certificats de clé publique, et au lieu de stockage des certificats dans des dépôts de certificats centralisés, ils sont stockés et distribués par les nœuds eux-mêmes. L'idée principale de la solution proposée est l'inclusion d'un système à seuil dans la toile de confiance. Lors de l'initialisation du réseau, les nœuds partagent le système de clé privée, et chaque nœud est titulaire d'une part privée. Au lieu d'utiliser les clés privées pour la signature des certificats, les nœuds utilisent leurs parts privées. Chaque nœud dans le réseau maintient une vue partielle de la toile de confiance, qui est mis à jour systématiquement.

L'authentification à clé publique parmi les nœuds est effectuée via la combinaison de la chaîne de certificats partielle. Quand un nœud client u a besoin d'authentifier une clé publique d'un autre nœud v, les deux nœuds fusionnent et valident leurs certificats partiels. Le processus de validation est effectué via la combinaison de toutes les signatures de certificats partiels. Si la vérification réussit pour un nœud donné, tous les certificats partiels émis de ce nœud sont marquées comme étant de confiance. Sinon, si la combinaison de signatures échoue, ils seront marqués comme non fiable. Enfin, le nœud u essaie de trouver une chaîne de confiance des certificats à partir du nœud u vers le nœud v Si la chaîne est trouvée donc l'authentification est réalisée.

Avantages et inconvénients: L'avantage de ce système c'est qu'il est capable de découvrir et d'isoler un pourcentage élevé de nœuds malveillants par rapport aux régimes précédents. L'inconvénient c'est que les certificats partiels et leur combinaison de chaque authentification sont stockés dans chaque nœud, et cela consomme à la fois de la mémoire et beaucoup du temps.

Solution de caballero-Gil et hernàndez-Goya

Dans [CH10], p.caballero-Gil and C.hernàndez-Goya ont proposé un service de certification basé sur la méthode MPR (multipoint relay) ou chaque membre du réseau a des taches à développer dans le schéma de gestion de certificat. Dans ce modèle, il est supposé qu'il existe des relations de confiance entre les nœuds. Chaque utilisateur génère sa propre clé publique et il échange le certificat lié à sa clé publique avec d'autres utilisateurs et à partir de ça il construit son propre répertoire de certificat, chaque nœud doit mettre à jours son répertoire de certificat pour la vérification d'une clé publique l'utilisateur doit d'abord trouver une chaine de certification après il va la vérifier.

L'avantage de ce modèle est l'autonomie du service de certification ou aucune autorité n'est employée pour assurer le service de certification. En effet, ce modèle provoque une charge importante de calcul induite par la vérification des chaînes de certificats à chaque authentification. Egalement, chaque utilisateur est requis de maintenir un dépôt local de certificats, qui est enrichi systématiquement par le protocole d'échange de certificats, ce qui provoque une charge importante de stockage et de transmission.

3.5.2. Modèles réactifs

Dans cette sous-catégorie, le protocole de collection de certificats est exécuté à la demande, le moment où un nœud aura besoin de vérifier une chaîne de certificats donnée. De ce fait, il doit solliciter l'ensemble de nœuds intermédiaires concernés par cette chaîne.

Solution de Funabiki et al.

Dans [FIK06], Funabiki et al. ont proposé un service de certification complétement distribué basé sur la clusterisation. L'établissement des certificats dans le réseau est assuré par les nœuds eux-mêmes, qui seront stockés par un nœud particulier appelé CMN (*Certificate Management Node*) pour chaque cluster. Tous les nœuds doivent solliciter le nœud CMN du même cluster pour collecter les chaînes de certificats à chaque authentification.

Ce modèle est intéressant dans le sens que les utilisateurs sont libérés de l'opération de stockage des certificats. Mais, avec cette manière, ce système converge vers les modèles centralisés où les certificats sont stockés par un ensemble de nœuds spéciaux, ce qui met en cause la disponibilité du service de certification.

Solution de Kitada et al.

Dans [KAT05, KWT05], Kitada et al. Ont supposé l'existence d'un graphe de confiance établi entre les différents utilisateurs dans un réseau ad hoc, et ils ont proposé un mécanisme de collection de certificats sans avoir besoin de dépôts locaux pour les stocker. Quand un nœud client *u* nécessite de vérifier le certificat d'un nœud *v*, il doit premièrement collecter une chaîne de certificats de *u* vers *v*. La recherche d'une telle chaîne suivant ce modèle est fait comme suit. *u* envoie une requête *p* à l'ensemble des nœuds pour lesquels *u* a déjà signé des certificats. Si un nœud *w* reçoit la requête *p*, il rajoute à *p* son propre certificat, et il la redirige vers l'ensemble des nœuds pour lesquels il a déjà signé des certificats. Si *w* est le nœud destinataire (*v*), il rajoute à *p* son propre certificat et renvoie directement le résultat au nœud source *u*. A la fin de ce processus, *u* reçoit *p* qui contient une chaîne de certificats de *u* vers *v*, et à partir de là il peut procéder à la vérification des signatures.

L'avantage principal du modèle de Kitada et al. Est que les nœuds ne maintiennent pas des dépôts de certificats. Cependant, ce système provoque une charge importante de

transmission, vu que la recherche des chaînes de certificats est faite à travers la diffusion d'un ensemble de certificats qui ne s'arrêtera qu'à l'aboutissement du nœud destinataire. Si le destinataire n'est pas abouti, le processus de recherche pourra s'exécuter indéfiniment. En plus, à la fin d'exécution du protocole le nœud source recevra plusieurs chaînes de certificats, alors qu'une seule chaîne suffit pour accomplir l'authentification.

Solution de Mohri et al.

Dans [MYT07], Mohri et al. ont proposé une version améliorée du modèle de Kitada et al. Ils ont proposé de diviser le processus de collection des chaînes de certificats en deux phases : (1) collection des identificateurs des nœuds concernées par les chaînes de certificats, et (2) s´élections d'une chaîne et collection des certificats concernées. La première phase consiste à exécuter le même protocole proposé par Kitada et al, mais au lieu de rajouter les certificats dans la requête *p*, les nœuds intermédiaires rajoutent leurs identificateurs. Dans la deuxième phase, quand le nœud client reçoit les différentes chaînes d'identificateurs, il choisit une chaîne et exécute la deuxième phase. Dans cette phase, le nœud client sollicite l'ensemble des nœuds intermédiaires concerné par la chaîne choisie pour collecter les certificats. En comparaison avec le modèle de Kitada et al, celui-ci minimise la charge de transmission. En plus, la deuxième phase permet au nœud client de choisir une chaîne de certificats selon une politique de s´élections particulière. Par exemple, il peut choisir la chaîne la plus courte afin d'optimiser la procédure de vérification des signatures, ou choisir une chaîne qui comporte des nœuds intermédiaires avec un degré élevé de confiance.

Solution de Kambourakis et al.

Kambourakis et al. [KKD10] ont étudié le même problème, et ils ont proposé que le graphe de confiance suit la structure d'un arbre binaire. De ce fait, chaque nœud du réseau (1) est certifié par seulement un de ses nœuds voisins, et (2) il certifie, au maximum, deux de ses nœuds voisins. Cette approche est efficace en termes de simplicité pour la collection des chaînes de certificats. Cependant, elle sera ingérable si les nœuds du réseau ont une forte mobilité, ce qui rend le système complexe et difficile à maintenir la structure d'arbre binaire pour le graphe de confiance.

Modèle	Dépôt de Certificats	Découverte des Chaînes	Disponibilité du Service de Certification
Capkun et al. Ren et al.	Créé, géré et mis à jour par les nœuds eux-mêmes.	Le nœud client collecte directement la chaîne de certificats à partir de son dépôt local.	Elevée
Omar et al	Créé, géré et mis à jour par les nœuds eux-mêmes.	Le nœud client collecte directement la chaîne de certificats à partir de son dépôt.	Elevée
Funabiki et al.	Géré par des nœuds spéciaux dans le réseau(CMNs).	Le nœud client doit solliciter les CMNs.	Moyenne
Kitada et al. Mohri et al.	Ne contient que les certificats générés par le nœud lui-même.	Le nœud client diffuse la requête de découverte à tous les nœuds intermédiaires concernés par une chaîne de certificats donnée.	Faible
Kambourakis et al.	Ne contient que les certificats générés par le nœud lui-même.	La requête du nœud va suivre la structure de l'arbre binaire.	Faible
Caballero-Gil et Hernàndez-Goya	Créé, géré et mis à jour par les nœuds eux-mêmes.	Le nœud client collecte directement la chaîne de certificats à partir de son dépôt local.	Elevée

Table 3.3: Comparaison globale par rapport à la disponibilité du service de certification

3.5.3. Etude comparative

Dans le tableau 3.3, nous donnons une comparaison des différents modèles anarchiques par rapport au critère de la disponibilité du service de certification. Pour chaque modèle nous expliquons comment les dépôts de certificats sont gérés et la manière avec laquelle les chaînes de certificats sont collectées (nous notons α, la longueur des chaînes de certificats). Ces deux critères influencent fortement la capacité des nœuds à collecter les chaînes de certificats et indirectement le taux de réussite du service de certification. En effet, ce dernier d´dépend du choix et du nombre (noté β) de

collaborateurs de certification. Cette comparaison nous a permis de classifier les modèles de cette catégorie en trois classes :

- **Classe A :** Dans cette classe, nous trouvons au sommet, les modèles de Capkun et al. [CBH02, CBH03] et Ren et al. [RLW04], qui assurent un niveau élevé de disponibilité du service de certification. En effet, dans cette classe, chaque nœud maintient un dépôt local, qui est mis à jour périodiquement à chaque arrivée d'un nœud voisin en utilisant un protocole d'échange de certificats. De ce fait, la collection des certificats se fait localement à l'instant de l'exécution du processus d'authentification, ce qui augmente la disponibilité du service de certification.

Quand un nœud client *Ci* nécessite la vérification de la clé publique d'un autre nœud *Cj*, les deux communicants fusionnent leurs dépôts de certificats et essayent de trouver une chaîne de certificats à partir du dépôt fusionné. Si une telle chaîne n'est pas trouvée, *ci* peut solliciter d'autres nœuds (*helper nodes*) qui se trouvent à un ou deux sauts de son voisinage. Ceci garde le nombre de collaborateurs de certification réduit. On retrouve aussi dans cette classe le modèle de confiance de Omar et al mais nécessitant plus de certificats, ce qui est le multiple de la valeur à seuil k.

- **Classe B :** Dans cette classe, nous trouvons le modèle de Funabiki et al. [FIK06], qui assure un niveau moyen de disponibilité du service de certification. Ceci est dû à la gestion centralisée des certificats qui sont stockés dans des nœuds spéciaux (CMNs). A l'exécution du processus d'authentification, le nœud client doit solliciter les nœuds CMNs afin de collecter une chaîne de certificats appropriée. De ce fait, la disponibilité du service de certification est fortement liée à la disponibilité des nœuds CMNs, ce qui diminue la disponibilité du service de certification.

- **Classe C :** Dans cette classe, nous trouvons les modèles qui assurent un faible niveau de disponibilité du service de certification. Dans les modèles de Kitada et al. [KAT05, KWT05] et Mohri et al. [MYT07], les chaînes de certificats sont collectées à la demande ; c'est-à-dire à l'instant même de l'exécution du processus d'authentification auprès des nœuds intermédiaires qui sont concernées par cette chaîne. De ce fait, la disponibilité du service de certification dépend fortement de la disponibilité des nœuds intermédiaires. Ainsi, le nœud client doit à chaque fois solliciter $\beta = \alpha - 1$ collaborateurs de certification (nœuds intermédiaires) afin de collecter la chaîne de certificats, ce qui rend cette classe

fortement liée à la longueur des chaînes, contrairement aux autres classes. Nous trouvons, aussi dans cette classe, le modèle de Kambourakis et al. [KKD10], qui utilise le même mécanisme, et par la structure de l'arbre binaire, il limite la valeur maximale de αà ln(m) où m représente la taille du réseau.

Modèle	Disponibilité	Ressources	Scalabilité	Pris en considération de l'hétérogénéité
Capkun et al. Ren et al.	**Elevée**	**Chaque nœud maintient un dépôt contenant un nombre important de certificats.**	**Non**	**Oui**
Omar et al	**Elevée**	**Chaque nœud maintient un nombre important de certificats.**	**Non**	**Oui**
Funabiki et al.	**Moyenne**	**Chaque CMN maintient un nombre important de certificats concernant tous les nœuds de son cluster.**	**Non**	**Oui**
Kitada et al.	**Faible**	**Une charge de transmission élevée due au nombre de certificats transmis lors de la découverte d'une chaîne de certificats.**	**Oui**	**Oui**
Mohri et al.	**Faible**	**Une charge de transmission élevée due au nombre de messages diffusés lors de la découverte d'une chaîne de certificats.**	**Oui**	**Oui**
Kambourakis et al.	**Faible**	**Une charge importante de transmissions et de calculs afin de maintenir la structure de l'arbre binaire.**	**Oui**	**Oui**
Caballero-Gil et Hernànd ez-Goya	**moyenne**	**Chaque nœud maintient un dépôt Contenant un nombre important de certificats.**	**Non**	**Oui**

Table 3.4: Comparaison globale

Dans le tableau 3.4, nous donnons une comparaison globale des modèles de confiance à base de certification appartenant à cette catégorie. Nous récapitulons les principales leçons de cette comparaison dans les points suivants :

- La disponibilité du service de certification dépend de la capacité des nœuds à collecter n'importe quelle chaîne de certificats concernant n'importe quel nœud du réseau pour l'authentifier. Cette propriété est fortement liée à la manière de gérer les dépôts de certificats dans le système. Si le service de certification fonctionne de telle sorte que chaque nœud maintient un dépôt local, mis à jour systématiquement par un protocole d'échange de certificats, le système peut achever un niveau élevé de disponibilité, puisque la collection des certificats sera faite localement par le nœud lui-même. Autrement, si le système se base sur un dépôt central de certificats, le niveau de la disponibilité du service diminue. En effet, dans ce cas la disponibilité du service de certification est liée à l'accessibilité du dépôt local dans le réseau.

- Comme le cas de la première catégorie, dans celle-ci, nous trouvons également une charge importante de calcul, qui est liée aux algorithmes de chiffrement à clés publiques. Une autre charge de traitement est impliquée dans cette catégorie liée à la vérification des signatures numériques des chaînes de certificats, qui est exécutée à chaque authentification d'un nœud. Egalement, nous trouvons dans cette catégorie une charge liée au stockage et à la communication. Cette charge est induite par le maintien et l'échange des certificats dans le réseau.

- La scalabilité du service de certification dépend fortement du nombre de certificats stockés par chaque nœud. Le service de certification sera moins scalable si le nombre de certificats à stocker dans chaque nœud est linéaire à la taille du réseau. Un autre problème de scalabilité peut concerner les solutions qui se basent sur un dépôt central de certificats, qui pourra être surchargé à large échelle.

- Dans cette catégorie, le problème d'hétérogénéité est remédié par le mécanisme du graphe de confiance où chaque nœud génère les certificats en utilisant sa propre politique de certification.

Conclusion

Dans ce chapitre, nous avons donné une vue d'ensemble sur les objectifs concernant la gestion des certificats dans les réseaux ad hoc : la disponibilité, la consommation de ressources, le scalabilité, et la prise en considération de l'hétérogénéité. Nous avons classifié les solutions existantes selon une taxonomie détaillée. Nous avons également présenté des comparaisons et discussions de l'ensemble des solutions.

CHAPITREIV

Un Modèle de Confiance à base de Certification à Seuil

Introduction

Pour sécuriser les réseaux Ad hoc, nous envisageons une architecture hiérarchique 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 sa durée de vie. Cette architecture est composée d'un modèle de confiance qui est basée sur la sélection des chefs du groupe (leaders). Pour atteindre cet objectif nous proposons l'utilisation du concept de la clusterisation qui consiste à diviser le réseau sous forme de groupes, avec un nœud chef (leadre) 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.

Dans ce qui suit, on propose une architecture d'un modèle de confiance à base de certification à cryptographie à seuil qui permet la coexistence de plusieurs autorités de certification hétérogènes. La solution proposée est partiellement distribuée, où le rôle de chaque autorité de certification est assuré par un ensemble de serveurs. Chaque autorité de certification supervise un ensemble d'utilisateurs appartenant à son cluster. Le passage d'un cluster à un autre se fait à l'aide d'un graphe de confiance établi entre les différentes autorités de certification.

Notre architecture a pour but de développer les systèmes dynamiques de gestion de clés adaptés aux caractéristiques du réseau Ad hoc. Nous proposons un modèle de confiance probabiliste basé sur le principe de la réputation pour définir la connectivité entre les nœuds de confiance, afin de mettre en place un bon modèle de gestion de la confiance a pour objectif d'évaluer la robustesse de notre nouvelle architecture dans le but de sécuriser les réseaux Ad hoc.

IV.1. Modélisation du réseau

Notre système est modélisé par un graphe connexe G = (V, E); où V est l'ensemble des nœuds, E ⊆ V² l'ensemble des arcs reflétant les communications directes possibles entre les nœuds. Les couples appartenant à E dépendent de la position des nœuds et de leur portée de transmission.

Pour chaque nœud *u*, nous attribuons une valeur unique qui le caractérise, appelée identifiant et notée ID(u). Nous supposons que tous les liens dans le réseau sont bidirectionnels, c'est-à-dire que si *u* est un voisin de *v* alors *v* est un voisin de *u*, l'ensemble des voisins d'un nœud v ∈ V est noté N. Chaque nœud u du réseau peut communiquer avec un sous ensemble N(u) ⊆ V. On définit la distance entre deux nœuds u et v dans le graphe G comme le nombre d'arêtes minimal le long du chemin entre u et v. La distance entre deux nœuds *u* et *v* est exprimée en nombre de sauts.

L'ensemble des voisins a un saut N1(u) d'un nœud *u* est défini par l'équation suivante *:*

$$\mathbf{N_1(u) = \{v \in V \mid v =/= u \wedge (u, v) \in E\}}$$

L'ensemble des voisins à deux sauts N2(u) d'un nœud *u* représente l'ensemble des nœuds qui sont les voisins des voisins du nœud *u* et qui ne sont pas les voisins de *u*. Il est défini comme suit :

$$\mathbf{N_2(u) = \{w \in V \mid (v,w) \in E : w =/= u \wedge w /\in N_1(u) \wedge v \in N_1(u)\}}$$

La réunion des ensembles N1(u) et N2(u) représente l'ensemble de tous les nœuds présents à une distance inférieure ou égale à deux sauts de *u*. Elle est notée N12 (u, v) et définie comme suit :

$$\mathbf{N_{12}(u) = N_1(u) \cup N_2(u) = \{v \in V \mid v =/= u \wedge d(u, v) \leq 2\}}$$

IV.2. Formation des clusters

On suppose que notre modèle est répartit en plusieurs clusters qui sont représentés par des graphes connexes hiérarchique dont la hauteur est inférieur ou égale à 2, pour cela on doit déclencher le processus d'élection du cluster Head et affectation des nœuds dans un unique cluster.

Après la formation des clusters chaque nœud possède:

ID : identifiant unique

succ : critère de priorité qui participe à l'élection du successeur du cluster Head tel quesucc = {vrai ou faux}

Cl-id : c'est l'identifiant du cluster auquel appartient le nœud

Statut∈ {CH, NS, NU}

- **CH** (Cluster Head) :Statut(u) = CH ssi u =max (v / v ∈ N(u) ∧ cl-id(u) = cl-id(v))
- **NS** (nœud serveur) : Statut(u) = NS ssi (∀ v ∈ N1(u))
- **NU** (nœud utilisateur): Statut(u) = NU ssi (∀ v ∈ N2(u))

Supposons que le réseau est partitionné en clusters dont chacun a un cluster Head qui effectue des tâches particulières. La construction des clusters se fait par échange périodique de message hello entre tous les nœuds du réseau. Chaque message hello transmis par un nœud contient son **id**, son successeur **succ** (qui est initialisés à faux), son **statut** et son **cl-id**. Ce message sert également à chaque nœud pour annoncer sa présence. Pendant une certaine durée, chaque nœud enregistre les messages reçus. A l'expiration de cette durée, chacun va comparer les identités des expéditeurs et le nœud dont l'identité est plus grande au sein d'un même cluster sera cluster Head et prendra le statut CH. De plus chaque message de type hello contient le Statut de l'expéditeur. A la réception de ce message, tous les voisins à un saut deviennent alors des nœuds de statut NS (nœuds serveurs), et ceux a deux sauts deviennent de statut NU (nœuds utilisateurs), A la fin du processus le nœud cluster Head élira son successeur parmi l'ensemble de ses voisins a un saut et lui envoie une requête de validation ainsi qu'une requête de refus au reste des nœuds serveurs, ce qui leurs permettra de maitre à jour la variable **succ** selon la requête reçu par le CH, en suite chaque nœud sera dans un des trois états suivants : cluster Head, nœud serveur, nœud utilisateur.

IV.3. la description du modèle visé par notre approche

Notre modèle peut être illustré dans plusieurs domaines tels que les entreprises, les hôpitaux et le milieu militaire ainsi qu'au niveau des universités, et bien dans le but d'organisation d'un réseau informatique sécurisé au sien de l'administration de l'université de Bejaïa, et pour une bonne sécurité dans l'échange des documents ainsi que des informations, on propose une modélisation a ce réseau qui sera répartit en plusieurs facultés de l'enseignement supérieur ainsi que le rectorat de l'université, dont chaque faculté est représenté par un groupe (cluster) du réseau, sachant que chacune d'elles est constitué d'un doyen qui effectue des tâches particulières, des chefs de département ainsi

que des enseignants. Dont chacun possède une machine mobile tel qu'un ordinateur portable ou même un téléphone portable tel que l'IPhone ou le BlackBerry, sachons que chacun est représenté par :

ID : chaque individus appartenant à ce réseau administratif possède un identifiant unique

succ: critère de priorité qui participe à l'élection du successeur du doyen tel que niveau d'étude ainsi que l'expérience qui sera attribué au vice doyen

Fac : c'est l'identifiant de la faculté à laquelle appartient chaque individu

Statut∈ {D, VD, CD, E}

- D : le doyen de la faculté et c'est le serveur délégué de l'autorité de certification (Clusterhead)
- CD : chef de département qui représente un serveur de l'autorité de certification et qui communique directement avec le doyen (nœud serveur) et les enseignants.
- VD : représente le vice doyen et c'est lui le successeur du doyen.
- E : c'est un enseignant qui est représenté par un utilisateur et qui doit passer par le chef de son département pour communiquer avec le doyen (nœud utilisateur).

La figure 4.1 représente la schématisation de notre modèle sur le réseau administrative des facultés de l'université de A/mira, et pour la clarté de la figure voilà la représentation de chaque nœud du réseau :

Rectorat :

- Vice-rectorat (VC)
- Secrétariat général (SG)
- Service des ressources humaines (RH)
- Service comptabilité (SC)
- Service pédagogique (SP)
- Service travailleurs (ST)
- service culturel (CL)

Faculté science de la nature et de la vie (Fac SNV) :

- Département tronc commun (TC)
- Département de biologie physico-chimie (B)
- Département de biologie organisme et population (OP)
- Département de microbiologie (MB)
- Département de science alimentaire (SA)

La faculté de Droit (Fac droit) :

- Département des Sciences Juridiques et Administratives (SJ)
- Département de droit LMD (D)

La faculté des lettres et sciences humaines :

- Département d'arabe (Ar)
- Département de français (Fr)
- Département d'anglais (An)
- Département de langue et culture Amazigh (Am)

La faculté Technologie (Fac Tech) :

- Architecture (Arc)
- Génie électrique (GE)
- Génie civil (GC)
- Génie Mécanique (GM)
- Génie des Procédés (GP)
- Hydraulique (H)
- Mines (M)
- Sciences techniques T (ST)

La faculté SEGC :

- Science Economique (SE)
- Science de Gestion (SG)
- Science Commerciale (SC)
- FSEGC LMD (TC)

La faculté des Sciences Exacte (Fac SE) :

- Chimie (C)
- Informatique (I)
- Mathématiques Informatique (MI)
- Physique (Ph)
- Recherche Opérationnelle (RO)

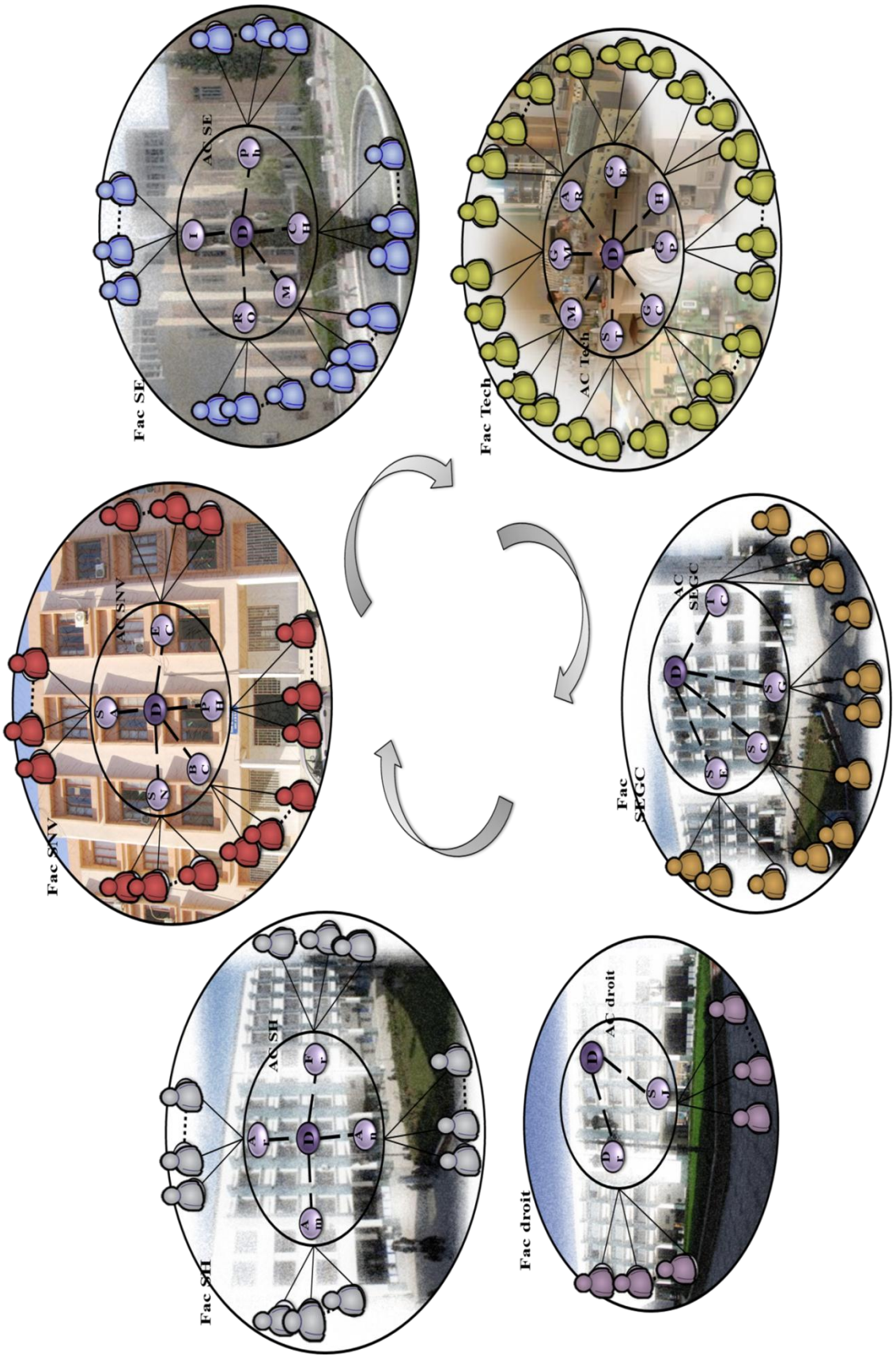


Figure 4.1 : Le réseau administrative des facultés de l'université de A/mira

IV.4. L'architecture du modèle de certification

Pour garantir un niveau suffisant de sécurité de la falsification des signatures et du trafic des documents ainsi que de l'information, on propose un service de certification qui permet à tous les serveurs que ce soit le doyen, le recteur ainsi que tous les chefs de département de signer électroniquement les accords en utilisant sa part de clé privée même tout on étant en déplacement avec un appareil mobile tel qu'un PC ou un téléphone portable.

IV.4.1. Description du modèle de certification

A travers le concept de la clusterisation, on suppose l'existence de plusieurs clusters dont chacun est supervisé par une autorité de certification qui exécute sa propre politique de certification pour la délivrance des certificats aux utilisateurs appartenant à son cluster. Ainsi l'authentification des utilisateurs du même cluster se fait directement à travers l'autorité de certification locale.

On profitant des caractéristiques intrinsèques des réseaux Ad hoc, on propose la conception d'une nouvelle approche de gestion des certificats. Notre système de certification de clés dont l'autorité n'est pas confié à une seule entité fixe mais qui est au contraire distribuée entre plusieurs nœuds du réseau. Notre modèle comporte plusieurs autorités de certifications hétérogènes, chaque autorité de certification est distribuée sur un ensemble de serveurs qui délivrent des certificats pour les utilisateurs en utilisant la cryptographie à seuil. Ainsi, le service de certification obtenu revient à définir une autorité de certification distribuée disposant d'une paire de clés publique/privée. La clé publique est connue de chaque nœud du réseau, ce qui leur permet de vérifier en confiance tout certificat signé avec cette clé privée. La clé privée est partiellement distribuée sur l'ensemble des nœuds de l'autorité de certification. Ainsi, un nœud utilisateur qui souhaite obtenir les clés publiques des autres utilisateurs ou lancer des mises à jour pour changer sa propre clé publique, émet une requête vers le service de certification. Chaque cluster comporte un cluster Head qui prend le rôle du serveur délégué. Ce serveur doit être choisi par accord à travers l'ensemble des autres serveurs. Une fois élu, l'ensemble des serveurs (y compris lui-même) lui délivre un certificat de délégation.

Pour garantir un niveau suffisant de sécurité même dans un contexte distribué, le service de certification repose sur **la cryptographie à seuil**. Notre schéma de cryptographie à seuil *(n, t+1)* est conçu de telle manière que parmi les *n* nœuds qui se partagent la gestion des clés, *t+1* auront la possibilité de procéder aux opérations de chiffrement, tandis que *t* nœuds seuls en seront incapables, même en coalition, Ainsi, lorsque le service doit signer un certificat, chaque nœud serveur génère une signature partielle en utilisant sa clé privée, et transmet le résultat au serveur délégué qui sera chargé d'assembler les portions de signature des *t* nœuds. Lorsque ce serveur a reçu *t+1* signatures partielles correctes, il est capable de calculer la signature finale du certificat. Dans le cas où la vérification échoue, le serveur délégué se doit de désigner un autre ensemble de *t+1* signatures partielles. Cette procédure continue jusqu'à ce qu'il parvienne à générer une signature correcte.

Dans ce modèle, on suppose la préexistence de certaines relations de confiance entre les clusters, ces relations de confiance sont nécessaires pour permettre à deux utilisateurs appartenant à deux cluster différents de s'authentifier.

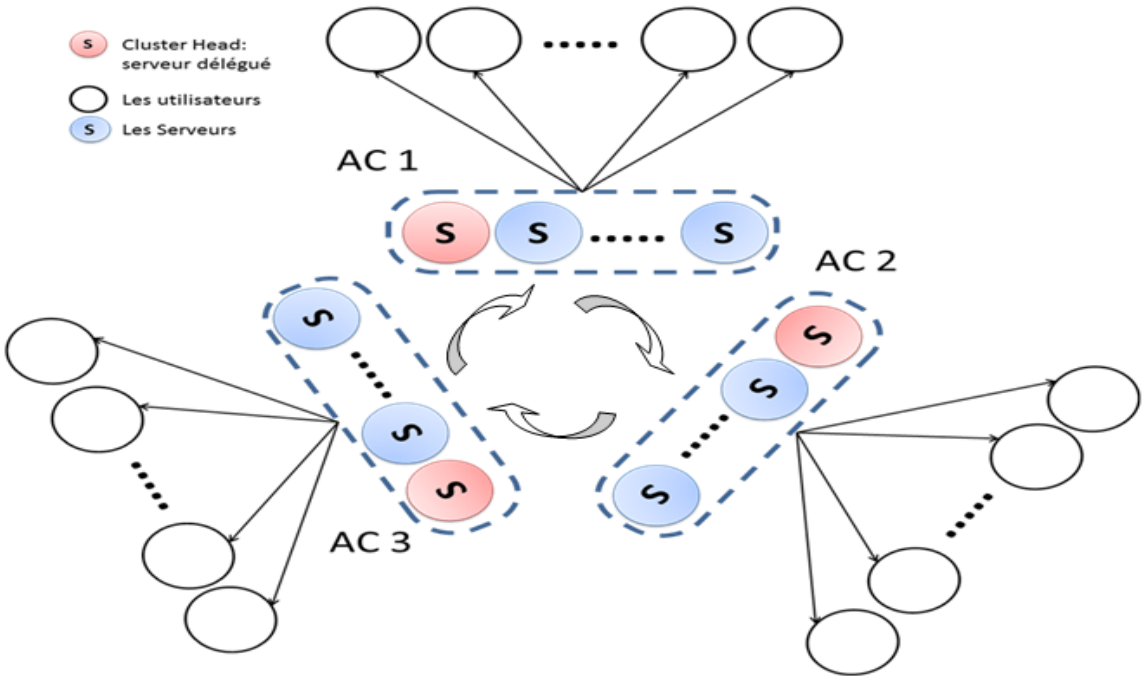


Figure 4.2 : L'architecture de notre modèle

IV.4.2. La distribution des rôles des ACs

On applique une architecture hiérarchique pour distribuer le rôle de l'autorité de certification (CA) sur le doyen et l'ensemble des chefs de département, sachons qu'à l'élection du doyen, ce dernier signe un certificat de rôle pour tous les chefs de départements justifiants leurs appartenances a la faculté et leurs génère des paires de clé. Ainsi l'ensemble des chefs de département lui signe collectivement un certificat de délégation. De ce fait, chaque chef de département peut prouver son rôle en tant qu'une partie de l'autorité de certification de la faculté auprès des enseignants. Et pour cela on suppose que chaque faculté est supervisé par une autorité de certification qui exécute sa propre politique de certification pour la délivrance des certificats aux enseignants appartenant à sa faculté en utilisant la cryptographie à seuil.

VI.4.3. Gestion des pairs de clés

En premier le doyen définit une clé secrète S que seul lui détient qui sera divisé en n sous-partage (s_1, s_2, s_3...s_n) qui seront attribuer secrètement à chaque chef de département avec une clé publique K.

De plus, le doyen de chaque faculté tient à jour une liste des chefs de département membre de l'autorité de certification et enregistre une clé publique k_i pour chaque nœud i.

Ainsi de la même manière l'ensemble de l'autorité de certification génère une paire de clé pour les enseignants de la sa faculté, sachons que chaque individus de la faculté détient une table de clé publique.

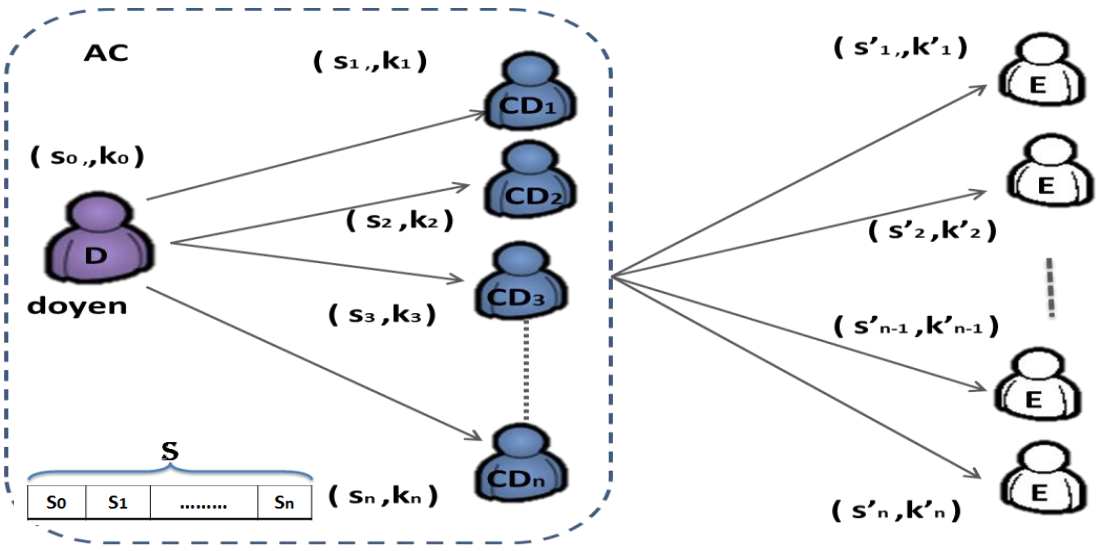


Figure 4.3. Gestion des paires de clés

IV.4.4. Délivrance des certificats

Un doyen est le représentant de l'autorité de certification de sa faculté à l'intérieur et à l'extérieur de la faculté. Ceci ne lui attribue pas de droit de prendre les décisions de certification des accords. Il est considéré comme une interface de négociation, ce qui permet d'éviter la diffusion des documents à la totalité des serveurs.

S'il s'agit d'une requête de certification interne comme dans le cas d'un enseignant de la même faculté qui voudrait avoir un certificat pour avoir accès aux matériaux de la faculté ainsi qu'aux salles par exemple , le doyen va jouer le rôle du coordinateur en déclenchant le protocole de certification pour délivrer le certificat à l'enseignant. Ce dernier peut vérifier la validité du certificat seulement à travers les identités des serveurs.

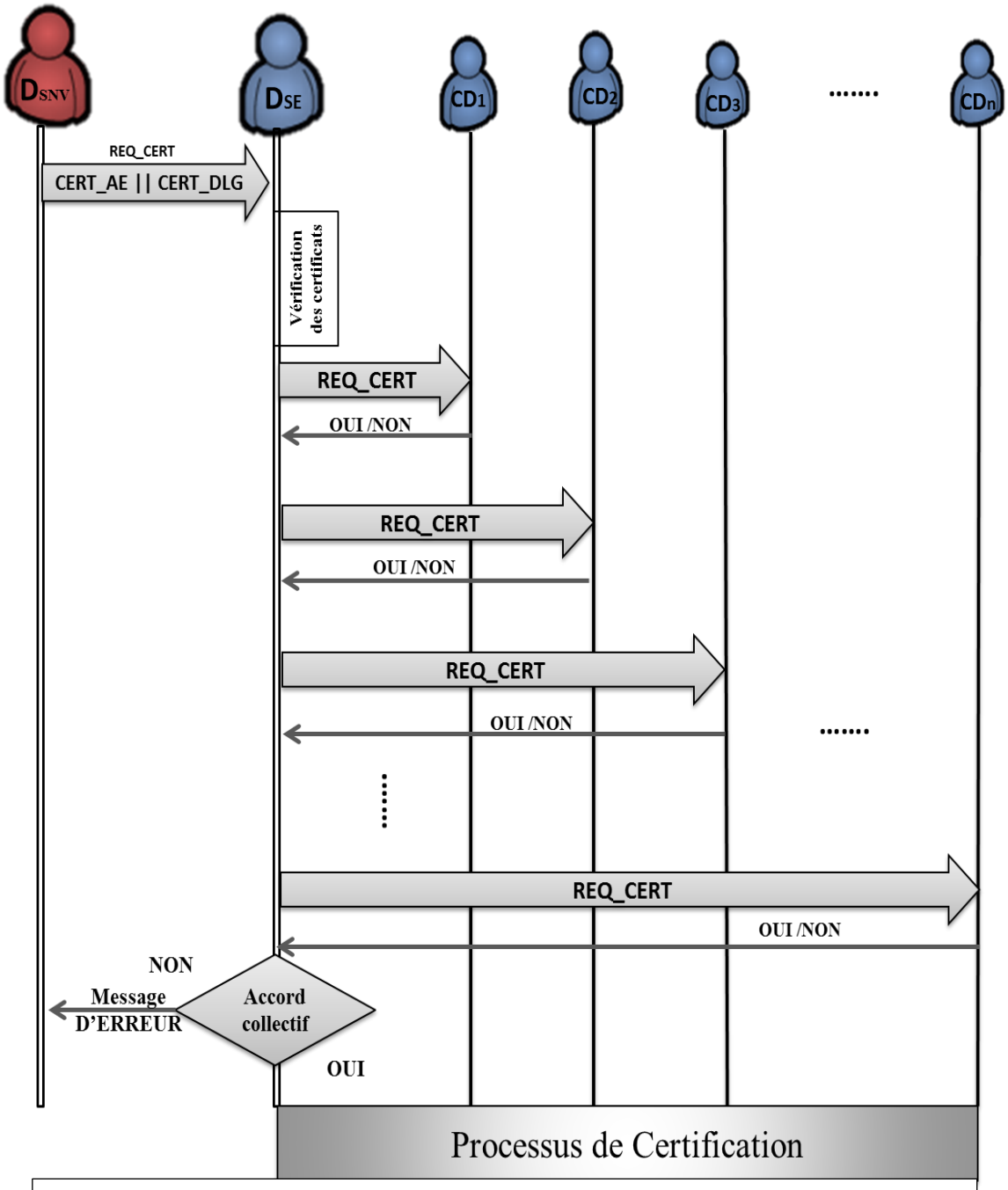
Dans le cas d'une requête de certification externe (inter-clusters, une autorité de certification d'une faculté externe souhaite être certifiée par l'autorité de certification), comme dans le cas où un enseignant X de la faculté **SNV** a obtenue des résultats d'une certaine étude établit en laboratoire pour laquelle il voudrait faire une étude statistique et une évaluation de ses performance ainsi qu'une simulation et bien pour cela il demanderait le matériels nécessaire au doyen et de l'information à des enseignants de la faculté **SE** donc l'autorité de certification **SNV** doit être certifiée par l'autorité de certification **SE** on lui envoyons une requête. Cette dernière déclenche le protocole de certification qui s'exécute en deux phases : (1) demande d'accord, et (2) la phase de certification.

Phase 1 : Demande d'accord

Le déroulement de ce protocole est illustré sur la figure 4.4. La requête de certification inter-clusters contient: (1) le certificat de rôle CERT_AE de l'ensemble des serveurs appartenant à l'autorité externe, (2) le certificat de délégation CERT_DLG du serveur délégué.

En recevant la requête, le serveur délégué vérifie la validité du certificat CERT_CA en utilisant sa clé publique K. Egalement, il vérifie le certificat de délégation vis-à-vis les identités des serveurs figurés dans le certificat de rôle. Ensuite, il diffuse la requête à l'ensemble des serveurs pour déclencher la procédure de négociation. Chaque serveur en recevant la requête, s'il estime que l'autorité externe est digne de confiance, il envoie son

accord au serveur délégué. Si ce dernier, reçoit un accord collectif, il prépare la procédure de certification en générant un certificat pour l'autorité de certification externe. Ce certificat sera diffusé à l'ensemble des serveurs pour l'étape de la certification à cryptographie à seuil. Si l'accord est défavorable, un message d'erreur lui sera transmis.



REQ_CERT : Requête de certification
CERT_AE : Le Certificat qui prouve le Rôle des serveurs de l'autorité externe
CERT_DLG : Le Certificat de Délégation du serveur délégué

Figure 4.4. Phase 1 demande d'accord

Phase 2 : certification

Le déroulement illustré sur la Figure 4.5 du protocole de certification qui est basé sur la cryptographie a seuil. Dans ce cas, n serveurs peuvent signer le certificat de confiance. Pour assurer que le service tolère t serveurs corrompus, on utilise également le schéma a seuil (n, t+1) et on divise la clé privée S en n sous partages (qui appelé le partage (n, t+1) de S), et on associe à chaque serveur i un sous-partage privé s_i. On crée un polynôme F(x) de degré t-1 avec des coefficients arbitraires en mettant a_0 = S. on choisit ensuite publiquement n points distincts k_i, et on distribue secrètement à chaque serveurs une part privée ($F(x_i)$, k_i).

Dans notre protocole pour signé un certificat de confiance, une demande est envoyée du serveur délégué aux serveurs de l'autorité de certification dont chaque serveur génère une signature partielle au serveur délégué, lorsque ce dernier obtient t+1 signatures correctes, il devient capable de générer un certificat de confiance signé par la clé secrète S de l'autorité de certification.

Les certificats sont représentés par des documents administratif, des fiches de présence ou même par des accords collectifs qui doivent être signés électroniquement par un seuil de serveurs (doyen, vice doyen et chefs de département) selon les serveurs visés dans le certificat.

Un enseignant X de la faculté **SNV** qui voudrait communiquer avec un autre enseignants Y de la faculté **SE**. Lorsque le doyen de la fac **SE** reçoit un accord collectif par un seuil de chefs de département, l'enseignant X de l'autorité de certification **SNV** doit être certifiée par l'autorité de certification **SE,** à ce moment le doyen prépare la procédure de certification en générant un certificat pour l'autorité de certification SE. Ce certificat sera diffusé à l'ensemble des chefs de département concernés pour l'étape de la certification à cryptographie à seuil. Si l'accord est défavorable, un message d'erreur lui sera transmis. Chaque chef de département génère une signature partielle on utilisant son sous partage de clé et la transmet au doyen, lorsque ce dernier obtient t+1 signatures correctes (sachons que t+1 représente le nombre d'individus concerné par l'accord), il devient capable de générer un certificat de confiance signé par la clé secrète S de l'autorité de certification. Ce qui permettra de crée un lien de confiance entre la faculté SNV et la faculté SE.

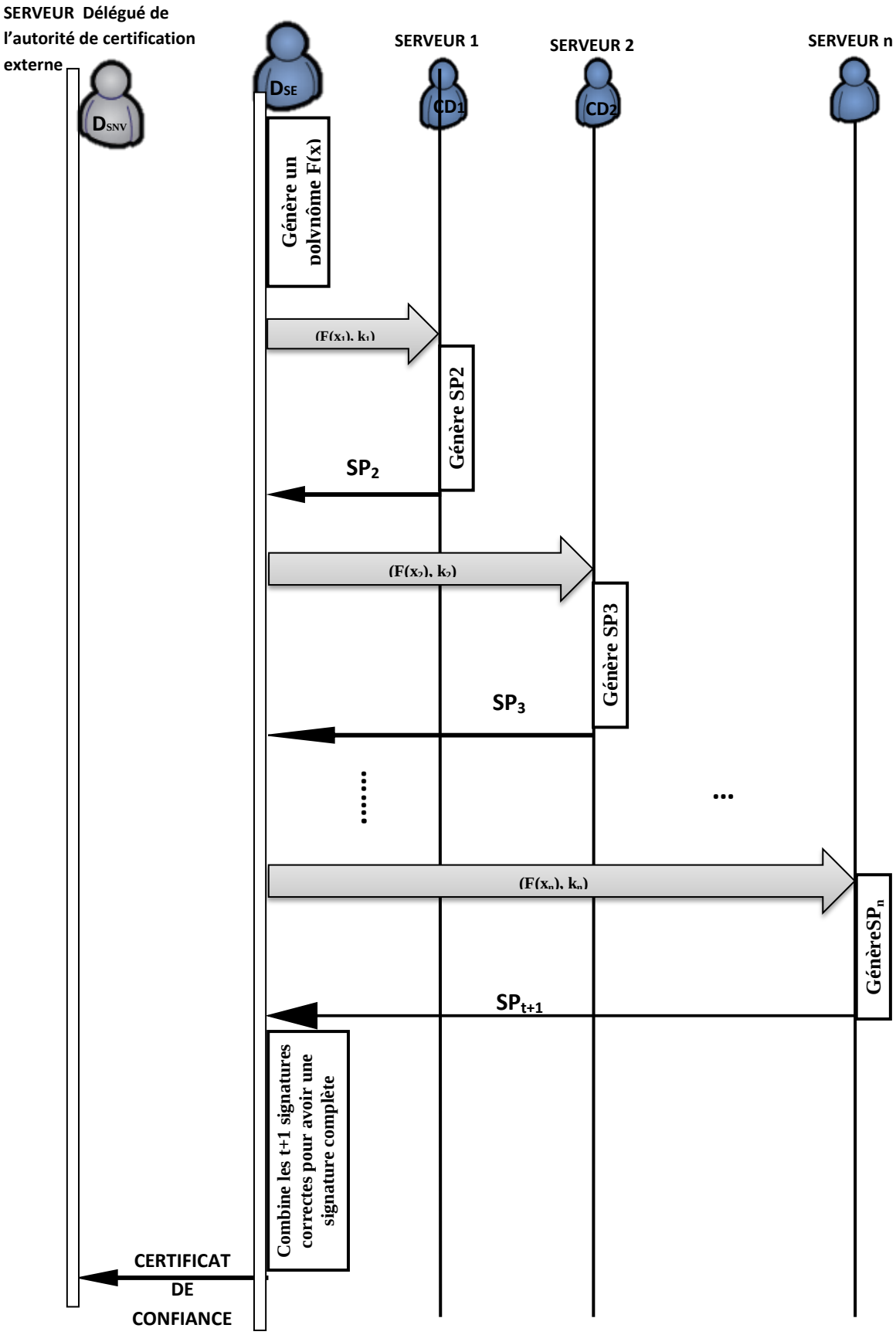


Figure 4.5. Phase 2: Certification

IV.4.5. Authentification des clés publique

La propriété de l'authentification repose sur le fait d'assurer que l'origine est bel et bien l'entité déclarée et l'Authentification des clés publique s'établit à travers la vérification de la validité des certificats. S'il s'agit d'une communication interne, chaque utilisateur vérifie le certificat de son interlocuteur vu qu'ils sont supervisés par la même autorité de certification. Si les deux utilisateurs appartiennent à deux clusters différents, les deux utilisateurs doivent vérifier l'existence d'une relation de confiance mutuelle entre leurs autorités de certification auxquelles ils appartiennent. De ce fait, chaque utilisateur essaye de trouver une chaîne de certificats inter-clusters qui relie son autorité de certification avec celle de son interlocuteur. S'il existe au moins une chaîne de certificats valide, l'authentification est établie.

IV.5. La maintenance de la topologie du réseau

Dans notre réseau, la topologie du réseau change fréquemment due aux individus qui possèdent une machine mobile telle qu'un ordinateur portable ou même un téléphone portable tel que l'IPhone ou le BlackBerry. Nous devons donc gérer :

1) Les nœuds qui apparaissent

Quand une nouvelle personne arrive dans le réseau, elle diffuse, à un intervalle de temps régulier un message de type hello (qui contient son CV). A l'expiration de son time-out, s'il n'a reçu aucun message de type hello (accusé de réception) de la part d'un doyen ou un chef de département, la personne rediffuse le même message hello jusqu'à l'obtention d'une réponse. Si cet individu a reçu un message hello (avis favorable) de la part d'un doyen, cette personne devient alors un nœud utilisateur (enseignant). Dans le cas où il reçoit des messages de plus d'un doyen, le nœud choisi la faculté qui lui offre le meilleur poste qui veut dire que c'est celui qui le relie a un saut du cluster Head pour prendre le statut NS au lieu de NU.

2) Les nœuds qui disparaissent

Une personne qui veut quitter la faculté elle envoie une requête pour informer l'ensemble de des serveurs, et si ce dernier est un doyen son succ le vice doyen le remplacera et ce dernier élira un nouveau succ parmi l'ensemble des chefs de

département. Dans le cas où ils disparaissent sans prévenir après l'échange des messages les voisins se rendent compte de cette disparition.

3) Les nœuds qui se déplacent

Chaque personne qui veut se déplacer, demande un avis favorable qui veut dire un certificat signer par l'ensemble de l'autorité de certification de la faculté qui va l'accueillir. Après la réception d'un certificat elle envoie une requête de déplacement à son doyen actuel pour l'informer de sa disparition.

IV.6. Les avantages de notre approche

Sachons que notre modèle peut être illustré dans plusieurs domaines tels que les entreprises, les hôpitaux et le milieu militaire ainsi qu'au niveau des universités, et cela et bien dans le but d'organisation d'un réseau informatique sécurisé dont on a utilisé la gestion de clés distribuées qui implique la désignation d'un ensemble de nœuds confidentiels qui partagent la clé secrète d'une autorité de certification. Chaque nœud de confiance garde un dossier de toutes les clés publiques dans le réseau. Le nombre de nœuds nécessaires pour générer une signature valide d'une autorité de certification peut être inférieur à celui des nœuds confidentiels, Par conséquent, même si un attaquant compromet quelques nœuds, une signature valide peut encore être générée grâce à la cryptographie à seuil.

L'avantage est bien dans le fait que *t* nœuds malveillants complices ne peuvent créer de certificat valide puisque *t+1* signatures partielles valides sont nécessaires, Bien entendu, nous ne sommes pas à l'abri d'un attaquant qui génère systématiquement de fausses signatures, en vue de conduire à la création d'un certificat invalide. Toutefois, Le serveur délégué a toujours la possibilité de vérifier la validité d'une signature en utilisant la clé publique du service.

IV.7. Résultats de simulations

Afin de valider nos propositions, Nous avons mené une série de simulations afin d'évaluer les performances du mécanisme de certification proposé. Nous avons utilisé pour cela le langage Java, dans lequel nous avons implémenté notre approche. Dans cette section, nous présentons quelques résultats liés à la certification dans un réseau mobile ah doc.

IV.7.1. Environnement et paramètres de simulations

Nous avons opté pour une durée de simulation de 500 *s*. Les requêtes des utilisateurs arrivent aux autorités de certification selon une loi de Poisson avec une durée moyenne de 10 *s*. Le simulateur estime si un lien radio existe entre deux nœuds quelconques en fonction de la distance qui les sépare. Les nœuds se déplacent sur une surface rectangulaire de 10000 m^2. Les nœuds ont les mêmes caractéristiques matérielles et la même puissance de traitement. Le graphe de confiance inter-domaines est fixé par le simulateur d'une manière aléatoire. Pour chaque requête, nous tirons deux nœuds appartenant à deux domaines différents d'une manière aléatoire, à travers laquelle les deux nœuds tentent de récolter une chaîne de certificats reliant les deux domaines. Le critère évalué, à travers cette simulation, est le taux moyen de certificats réussis. Les impacts étudiés sont respectivement : le nombre d'autorités de certification et nombre de serveurs de certification.

IV.7.2. Impact du nombre d'autorités de certification

Dans cette sous-section, nous étudions l'impact du nombre d'autorités de certification (noté n) sur le taux moyen de certificats réussis. Nous avons effectué cette simulation pour une variation de valeurs de $n = 1$ à $n = 10$ autorités de certification pour trois cas de portées de communication : $p = 50$ *m*, $p = 100$ *m*, et $p = 150$ *m*. Chaque autorité de certification comporte $k = 5$ serveurs qui supervisent un certain nombre de nœuds. Les résultats de cette simulation sont illustrés sur la figure 4.6 ,Nous constatons que le nombre de certificats réussis augmentent énormément durant la variation de *n*. Cela veut dire, que le nombre d'autorités de certification a un impact très fort sur le taux moyen de certificats réussis. En effet, quand nous augmentons *n*, le réseau subit un partitionnement de plusieurs clusters de certification. Si les deux nœuds appartiennent au même cluster, il

suffit seulement de vérifier les certificats de l'un de l'autre vu que l'autorité de certification est commune pour les deux nœuds. Malgré ça, cette vérification peut s'échouer si les serveurs de certification ne sont pas accessibles lors de la requête. Si les deux nœuds appartiennent à deux clusters différents, la réussite de certification dépend de l'existence d'une chaîne de certificats reliant les deux nœuds. Pour un nombre important d'autorités de certification, la probabilité que les nœuds soient des serveurs est élevée contrairement au cas d'un nombre réduit d'autorités de certification ceci permet la délivrance de plus de certificat. Ce qui interprète l'augmentation du taux moyen de certificats réussit.

IV.7.3. Impact du nombre de serveurs k

Dans cette sous-section, nous étudions l'impact du paramètre *k* : le nombre de serveurs impliqués dans chaque autorité de certification. Nous avons effectué cette simulation pour une variation de valeurs de $k = 5$ à $k = 15$ serveurs pour trois cas de portées de communication : $p = 50$ *m*, $p = 100$ *m*, $p = 150$ *m* et un nombre de $n = 5$ autorités de certification. Les résultats de cette simulation sont illustrés sur la figure 4.7. Nous constatons que le nombre de certificats réussis augmentent considérablement durant la variation de *k*. En effet, à chaque requête de certification, l'existence de plus de serveurs dans chaque autorité augmente la probabilité qu'il existe un chemin pour certifier un utilisateur d'un autre cluster, la totalité des *k* serveurs est sollicité, ce qui fait que le taux moyen de certificats réussis est légèrement sensible à la grandeur de *k*. Si ce dernier est grand, et dans le cas normal ceci rend difficile d'avoir tous les *k* serveurs accessibles à un moment donné vu la nature du réseau ad hoc mobile mais nous avons pris en considération tous les certificats partiels généré par chaque serveur. D'où la légère augmentation du taux moyen de certificats réussit.

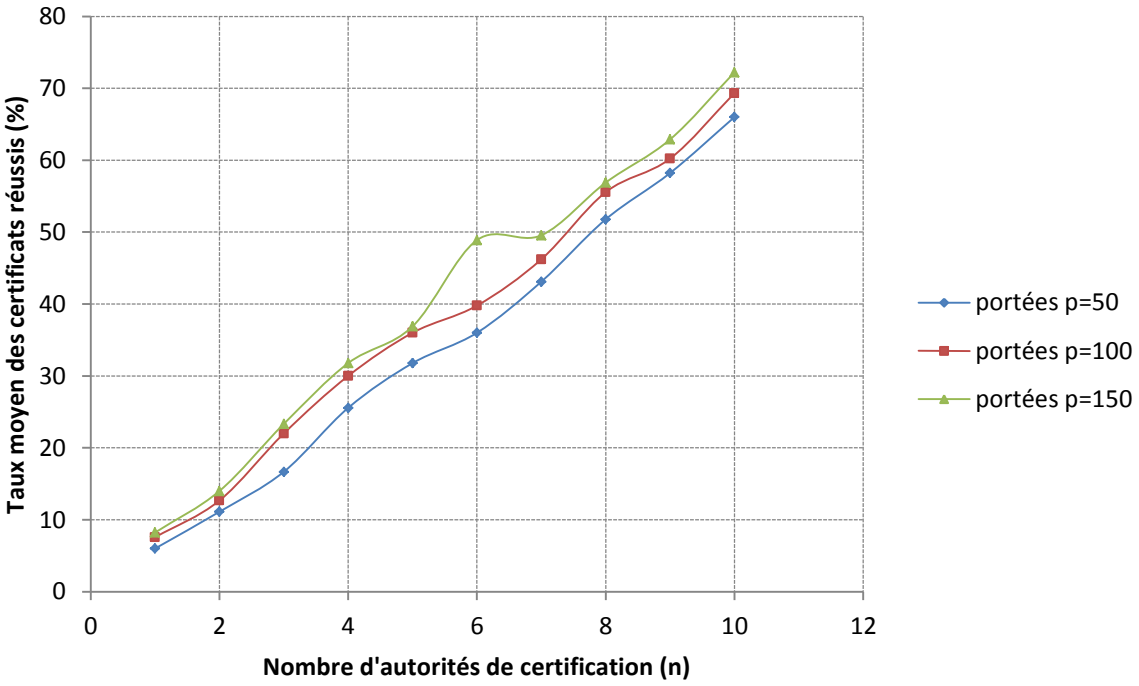


Figure 4.6. Impact du nombre d'autorités de certification sur le taux moyen de certificats réussis. *k*= 5 serveurs

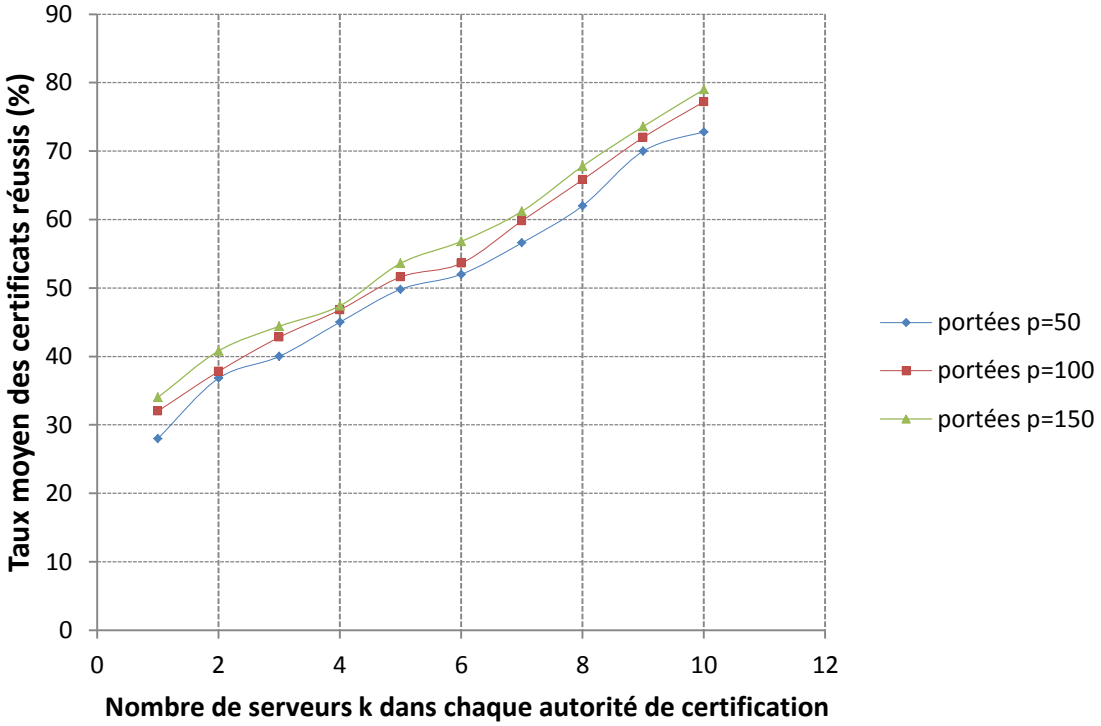


Figure 4.7. Impact du nombre de serveurs *k*. *n* = 5 autorités de certification

Conclusion

Dans ce chapitre, nous avons proposé un modèle de confiance à base de certification qui prend en compte l'hétérogénéité des autorités de certification. Le service de certification dans chaque domaine repose sur l'aspect collaboratif avec une gestion partiellement distribuée des certificats à travers un ensemble de serveurs. Notre modèle fait l'objet d'une solution anarchique inter-cluster. Cette solution élimine la centralisation du service de certification ce qui la rend une solution très pratique.

Pour mettre en valeur les qualités de performance de notre modèle, nous avons effectué des simulations, où ces dernières ont montré que notre modèle est légèrement influencé par le nombre d'autorité de certification dans un milieu ad hoc. Cependant, il est nécessaire d'augmenté le nombre de serveurs impliqués dans chaque autorité de certification pour la délivrance des certificats partiels et de les réduire pour la délivrance des certificats complet et cela afin d'aboutir à un taux acceptable de certificat réussis.

CONCLUSION GÉNÉRALE ET PERSPECTIVES

La sécurité des réseaux mobiles et spécialement des réseaux mobiles ad hoc n`a jamais cessé de susciter des préoccupations du fait qu'ils sont exposés à des menaces supplémentaires par rapport aux réseaux filaires. En général, ces menaces viennent du fait que les communications sans fil sont transmises par ondes radios et peuvent être écoutées par des personnes non autorisées.Les techniques de chiffrement les plus utilisées dans les systèmes filaires ne conviennent pas toujours aux systèmes sans fils vu leurs caractéristiques limitées (puissance de calcul, capacité de stockage, bande passante).

Dans ce travail, on s'est intéressé à l'étude de certains protocoles de gestion de clé publique dans les réseaux ad hoc mobiles. On a exposé leurs spécificités, caractéristiques ainsi que leurs modes de fonctionnement, afin de mettre en œuvre une architecture d'un modèle de confiance à la fois robuste et hautement disponible dans un milieu ad hoc.

Dans une première partie, nous avons donné une vue d'ensemble sur les environnements sans fil et particulièrement les réseaux ad hoc et leurs caractéristiques. En second on a présentés les concepts de base de la sécurité et la vulnérabilité des réseaux ad hoc. On a notamment met l'accent sur la nécessité de mise en œuvre d'un modèle de confiance pour sécuriser les échanges entre les nœuds du réseau. Nous avons, également, une vue d'ensemble sur les concepts liés à la cryptographie et l'infrastructure de gestion de clés publiques. Et objectifs concernant la gestion des certificats dans les réseaux ad hoc à savoir la disponibilité, la consommation de ressources, la scalabilité, et la prise en considération de l'hétérogénéité de certification. Nous avons classifié les solutions existantes en deux catégories : (1) les modèles autoritaires, et (2) les modèles anarchiques. Dans les modèles autoritaires, le service de certification est assuré par une ou plusieurs autorités de certification. Afin de prendre en considération les contraintes liées à la nature du réseau ad hoc, dans cette catégorie le service de certification est distribué sur un ensemble de nœuds spéciaux dans le réseau en utilisant la cryptographie à seuil.

Dans les modèles anarchiques, chaque nœud dans le réseau se considère comme une autorité de certification et établit ses propres rapports de confiance selon des règles qui peuvent exiger la coopération des autres nœuds dans le réseau.

Nous avons donné pour chaque catégorie l'ensemble des solutions proposées dans la littérature avec leurs avantages et inconvénients. Et ensuite, on a proposé une étude comparative des travaux existants de chaque catégorie selon un ensemble de critères bien précis.

Enfin on a met en œuvre une architecture d'un modèle de confiance à la fois robuste et hautement disponible dans un milieu ad hoc, un modèle de confiance qui repose sur des autorités de certification particulières qui assure la gestion des certificats, dont AC assurent le service de certification en utilisant la cryptographie à seuil. Le proposé décentralisé et partiellement distribué supporte la mobilité des nœuds et la défaillance jusqu'à n-k+1 et pour mettre en valeur les privilèges du modèle, nous avons effectué des simulations pour montrés ces qualités.

En guise de perspectives, ce travail peut être enrichi dans un premier temps par une application de signature adaptée aux différents appareils mobile tel que les IPhone, les BlackBerry et les IPad, afin de permettre la mobilité de chaque individu du réseau.

BIBLIOGRAPHIE

[AF10] E. Ayday, F. Fekri. A protocol for data availability in mobile ad-hoc networksin the presence of insider attacks. Ad Hoc Networks, 2010.

[BB02] S. Buchegger and J. Le Boudec. Proceedings of the tenth euromicro workshop on parallel, distributed and network-based processing. In Nodes Bearing Grudges : Towards Routing Security, Fairness, and Robustness in Mobile Ad Hoc Networks, pages 403–410, Canary Islands, Spain, 2002. IEEE Computer Society.

[BH01] L. Buttyan and J.Hubaux. Nuglets : a virtual currency to simulate cooperation in self organized ad hoc networks. Technical Report Technical Report DSC/2001/001, Swiss Federal Institute of Technology, Lausanne, 2001.

[BHK04] M. Bechler, H.-J. Hof, D. Kraft, F. Pahlke, et L. Wolf, ACluster-Based Security Architecture for Ad Hoc Networks. Dans les actes de IEEEINFOCOM'2004.

[CBH02] S. Capkun, L. Buttyan, J. Hubaux. Small worlds in security systems – ananalysis of the PGP certificate graph. In Proceedings of New Security ParadigmsWorkshop (ACM), 2002.

[CBH03] S. Capkun, L. Buttyan, J. Hubaux. Self-organized public key management formobile ad hoc networks.IEEE Transactions on Mobile Computing, 2003.

[CCN03] M. Conti I.Chlamtac and J.J.-N.liu. Mobile ad hoc networking : imperative and challenges. Elsevier, 2003.

[CJ03] T. Clausen and P. Jaquet. Optimized Link State Routing Protocol (OLSR).RFC 3626 (Experimental), 2003.

[DBP03]S. Das E. Belding-Royer and C. Perkins. Ad-Hoc On demand Distance Vector routing (AODV). IETF RFC 3561(Experimental), 2003.

[Dou02] J. Douceur. The sybil attack. In Proceedings of the International Workshop of Peer-to-Peer Systems, 2002.

[DSY07] Y. Dong, A. Sui, S. Yiu, V. Li, L. Hui. Providing distributed certificate authorityservice in cluster-based mobile ad hoc networks. Elsevier, Computer Communications,2007.

[FIK06] S. Funabiki, T. Isohara, Y. Kitada, K. Takemori, I. Sasase. Public key managementscheme with certificate management node for wireless ad hoc networks.In Proceedings of the International Multiconference on Computer Science andInformation Technology, 2006.

[FL01] J. A Freebersyser and B. Leiner. A DoD perspective on mobile Ad hoc networks. Ad hoc networking. Addison-Wesley Longman Publishing Co., Inc, 2001.

[G09] S. GHAROUT. Sécurité des communications dans les groupes dynamiques. PhDthesis, Université de Technologie de Compiègne, 2009.

[Gio01] S. Giordano. Mobile ad-hoc networks. Handbook of Wireless Networks and Mobile Computing, John Wiley and Sons, 2001.

[GLG09] M. Ge, K.Y. Lam, D. Gollmann, S.L. Chung, C.C. Chang, J.B. Li. A robustcertification service for highly dynamic MANET in emergency tasks. WileyInterScience : International Journal of Communication Systems, 2009.

[HGP02]X. Hong M. Gerla and G. Pei.Fisheye State Routing Protocol (FSR) for Ad Hoc Networks. IETF Internet Draft : draft-ietf-manet-fsr-03.txt, 2002.

[HJM07]Y. Hu D. Johnson and D. Maltz.The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4.IETF RFC 4728, 2007.

[HWK04] Q. He, D. Wu, P. Khosla. SORI : a secure and objective reputation-based incentive scheme for ad-hoc networks. In Proceedings of IEEE WCNC'04,2004.

[ITU01] ITU-T Recommendation X509/ISO/IEC 9594-8. Public-key and attribute certificate frameworks. 4th edition, 2001.

[JFD09] H. Janzadeh, K. Fayazbakhsh, M. Dehghan, M.S. Fallah. A secure creditbased cooperation stimulating mechanism for MANETs using hash chains.FutureGeneration Computer Systems, 2009.

[JT87] J. Jubin and J. D. Tornow. The darpa packet radio network protocols. Proceedings of the IEEE, 75(1) :21–32, 1987.

[KAT05] Y. Kitada, Y. Arakawa, K. Takemori, A. Watanabe, I. Sasase. On demanddistributed public key management using routing information for wireless ad hocnetworks. IEICE Transactions on information and Systems, 2005.

[KKD10] G. Kambourakis, E. Konstantinou, A. Douma, M. Anagnostopoulos, G. Fotiadis.Efficientcertification path discovery for MANET. EURASIP Journal onWireless Communications and Networking, 2010.

[KWT05] Y. Kitada, A. Watanabe, K. Takemori, I. Sasase. On demand distributedpublic key management without considering routing tables for wireless ad hocnetworks. Asia Pacific Symposium on Information Technology (APSITT), 2005.

[KZL01] J. Kong, P. Zerfos, H. Luo, S. Lu, L. Zhang. Providing robust and ubiquitous security support for mobile ad-hoc networks.In Proceedings of InternationalConference on Network Protocols, IEEE Computer Society, 2001.

[LFR04]G. Lewis Mark L. Fred Templin and G. Richard Ogier. Topology dissemination based on Reverse- Path Forwarding (TBRPF), 2004.

[LHE05] J. Luo, J. Hubaux, P. Eugster. DICTATE - distributed certification authoritywith probabilistic freshness for ad hoc networks. IEEE Transactions on Dependableand Secure Computing, 2005.

[LJ04] W. Li, A. Joshi. Security issues in mobile ad hoc networks - a survey. Department of Computer Science and Electrical Engineering University of Maryland, Technical Report, 2004.

[LJT99]J. Li M. Jiang and Y.C. Tay. Cluster Based Routing Protocol (CBRP). IETF Internet Draft, 1999.

[LL00] H. Luo, S. Lu. Ubiquitous and robust authentication services for ad hoc wirelessnetworks. Technical Report, UCLA Computer Science, 2000.

[LZK02] H. Luo, P. Zerfos, J. Kong, S. Lu, L. Zhang. Self-securing ad hoc wirelessnetworks. In Proceedings of the Seventh International Symposium on Computersand Communications (ISCC'02), 2002.

[MD08] N. Marchanga, R. Dattab. Collaborative techniques for intrusion detection inmobile ad-hoc networks.Ad Hoc Networks, 2008.

[MGF06] G.F. Marias, P. Georgiadis, D. Flitzanis, K. Mandalas. Cooperation enforcement schemes for MANETs : a survey. In wireless communication and mobile computing, 2006.

[MM02] P. Michiardi and R. Molva. Proceedings of ifip tc6/tc11 sixth joint working conference on communications and multimedia security. In Core : a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks, pages 107–121, Deventer, The Netherlands, 2002.

[MM02]P. Michiardi and R. Molva. Ad hoc networks security. ST Journal of System Research4(1), 2003.

[MYT07] H. Mohri, I. Yasuda, Y. Takata, H. Seki. Certificate chain discovery in Webof trust for ad hoc networks.In Proceedings of the 21st International Conference on Advanced Information Networking and ApplicationsWorkshops (AINAW'07), 2007.

[OCB09] M.Omar, Y. Challal, and A. Bouabdallah.Reliable and fully distributed trust model for mobile ad hoc networks.Computers and Security Journal (Elsevier), 2009.

[Per00] C. Perkins. Ad hoc networking. Addison-Wesley Professional, 2000.

[PH02] D. Perkins, H. Hughes. A survey on quality-of-service support for mobile ad-hoc networks. Wireless Communications and Mobile Computing, 2002.

[PZH02]R. Marc Pearlman and Zygmunt J. Haas.The Zone Routing Protocol (ZRP) for Ad Hoc Networks.IETF Internet Draft, 2002.

[QK04] L. Qin, T. Kunz. Survey on mobile ad hoc network routing protocols and crosslayer design. Systems and Computer Engineering, Technical Report, 2004.

[RLS96] R. J. Ruther B. M Leiner and A. R. Sastry. Goals and challenges of the darpaglomo program [global mobile information systems]. IEEE Personal Communications, 3(6) :34–43, 1996.

[RLW04] K. Ren, T. Li, Z. Wan, F. Bao, R. Deng, K. Kim. Highly reliable trust establishmentscheme in ad hoc networks.Elsevier, Computer Networks, 2004.

[RTJ06] S. Raghani, D. Toshniwal, R. Joshi. Dynamic support for distributed certificationauthority in mobilead hoc networks.In Proceedings International Conferenceon Hybrid Information Technology (IEEE), 2006.

[S02] F. Stajano. Security for Ubiquitous Computing. John wiley and sons edition, 2002.

[SP03] S. Seys, B. Preneel. Authenticated and efficient key management for wirelessad hoc networks.In Proceedings of the 24th Symposium on Information Theoryin the Benelux, 2003.

[WZL03] W. Wang, Y. Zhu, B. Li. Self-managed heterogeneous certification in mobilead hoc networks. In the Proceedings of the Vehicular Technology Conference(IEEE), 2003.

[XI04] G. Xu, L. Iftode. Locality driven key management architecture for mobile ad-hocnetworks. In Proceedings of the First IEEE International Conference on Mobileand Sensor Networks (MASS'04), 2004

[YK03] S. Yi, R. Kravets. MOCA - mobile certificate authority for wireless ad hocnetworks.In Proceedings of the Second Annual PKI Research Workshop, 2003.

[ZH99] L. Zhou, Z. Haas. Securing ad hoc networks. IEEE Networks, 1999

[ZMH09] C. Zouridaki, B.L. Mark, M. Hejmo, R.K. Thomas. E-Hermes : a robust cooperative trust establishment scheme for mobile ad hoc networks. Ad Hoc Networks,2009.