République Algérienne Démocratique et Populaire
Ministère de L'enseignement Supérieur et de la Recherche Scientifique
Université A/Mira de Béjaia
Faculté des Sciences Exactes
Département Informatique

Mémoire de Fin de cycle

Master en informatique
Spécialité : Réseaux et Systèmes Distribués

Thème

Gestion des clés dans l'internet des objets

Réalisé par :

M[lle] CHALAL Lina
M. SIROUAKNE Slimane

Encadré par :

Encadreur : M. AISSANI Sofiane

Devant le jury composé de :

Président : M. SAADI Mustapha
Examinateur[1] : M. KHANOUCHE Mohamed Essaid
Examinateur[2] : M. ABBACHE Bournane

Promotion 2016 - 2017

Remerciements

Nous remercions en premier lieu ALLAH de nous avoir donné non seulement le courage mais aussi la force et la volonté nécessaire pour la réalisation de ce modeste travail.

Nos vifs remerciements s'adressent à nos parents, nos frères et sœurs pour leur soutien moral et leur encouragement.

Nous remercions nos chers ami(e)s qui sont toujours présent et fidèles.

Nous tenons à exprimer nos profondes gratitudes et nos sincère remerciements à notre encadreur ***M. AISSANI Sofiane*** *pour la haute qualité de son encadrement, son suivi, sa disponibilité et ses conseils. Sans vous, la réalisation de ce mémoire n'aurait pas eu lieu. Encore une fois, merci beaucoup.*

Nous adressons nos remerciements aux membres de jury qui ont fait l'honneur d'évaluer, examiner et enrichir notre modeste travail.

Notre reconnaissance va particulièrement à l'ensemble des enseignants du département Informatique l'université ABDRRAHMANE MIRA DE BEJAIA tout ce qui nous a été transmis tout au long de notre formation.

Enfin on remercie tous ceux qui ont contribué de loin ou de près à la réalisation de ce travail.

Dédicaces

C'est avec profonde gratitude et sincère mots, que je dédie ce modeste travail

A mes chers parents qui depuis mon jeune âge ont toujours fait leur maximum et qui ont sacrifié leur vie pour ma réussite.

*A ma très chère sœur « **Anais** » pour avoir contribué a la réussite de ce travail d'une manière indirecte, et pour tout le soutien moral.*

A ma famille et à tous mes proches grands et petits.

A mes amis fidèles et à tous ceux qui nous sont chers.

A tous mes enseignants du département informatique.

Que dieu les protèges tous. . ...

Lina

Dédicaces

À mes chers parents qui m'ont toujours entouré avec tout leur sacrifices,

et permis d'achever mes études en tant que je suis actuellement.

À mes sœurs, et mes frères,

pour leur soutien moral,

et leur intérêt envers notre travail,

À mon cher ami ***« M. KHAIROUNE Mohamed Amine »****,*

qu'était toujours la pour m'aider avec ses conseils précieux,

et pour tout le soutien moral.

À toute ma famille.

À tous mes amis et collègues.

À tous ceux qui m'ont aidé.

À tous ceux qui me sont chers.

À tous ceux qui j'avais omis.

Je dédie cet humble travail.

Slimane

Table des matières

Table des figures

Liste des tableaux

Liste des abréviations

AC	**A**utorité de **C**ertification
API	**A**pplication **P**rogram **I**nterface
GSM	**G**lobal **S**ystem for **M**obile communication
IBSG	**I**nternet **B**usiness **S**olutions Group
IdO	**I**nternet **d**es **O**bjets
IoT	**I**nternet **o**f **T**hings
LTE	**L**ong **T**erm **E**volution
M2M	**M**achine **t**o **M**achine
NFC	**N**ear **F**ield **C**ommunication
RCSF	**R**éseau **C**apteurs **S**ans **F**il
RFID	**R**adio **F**requency **I**dentification
UMTS	**U**niversal **M**obile **T**elecommunications **S**ervice
WAN	**W**ide **A**rea **N**etwork

Introduction générale

Internet est un réseau informatique mondial, qui se transforme progressivement en un réseau étendu dit Internet des Objets (IdO), reliant des milliards d'êtres humains et des dizaines de milliards d'objets.

Internet des objets désigne l'omniprésence autour de nous de diverses technologies sans fil telles que les étiquettes, les capteurs, les actionneurs, les téléphones mobiles et les RFID qui, à travers des schémas d'adressage uniques, ces objets interagissent les uns avec les autres et coopèrent pour atteindre les objectifs communs .

Cependant, l'IdO pose plusieurs problèmes de l'hétérogénéité, routage et d'identification et de sécurité. Ces caractéristiques rendent le mécanisme de gestion des clés dans l'IdO le problème le plus délicat de la cryptographie.

A ce fait, les techniques classiques de gestion des clés sont inadéquates aux environnements de l'Ido. Plusieurs recherches ont été réalisées durant ces dernières années pour la satisfaction de cette contrainte, mais reste un défis à relever. Nous classifions les travaux étudiés en 3 catégories selon leur architecture ,le but principale de ces approches est de sécuriser le transfère des données , cependant, chaque solution présente un point faible qui limite sa performance. Notre contribution résulte, en une amélioration d'un modèle de gestion des clés dans le but de fournir plus de sécurité.

Organisation du mémoire

Ce mémoire est organisé comme suit :

Dans le premier chapitre nous présentons les concepts généraux relatifs au domaine d'IdO et de la sécurité informatique en général. Dans le deuxième chapitre, nous faisons une étude sur quelques articles sur la gestion des clés dans l'IdO où nous étudions sur les protocoles de gestion des clés dans l'IdO. Après cela, nous proposons une amélioration à l'un des protocoles, ensuite nous présentons les résultats de simulation. Enfin nous finissons par une conclusion générale et quelques perspectives.

Chapitre 1

Internet des objets

1.1 Introduction

Internet des objets est un réseau mondial d'objets qui repose sur l'idée que tous les objets peuvent être connectés un jour à Internet, ces objets sont adressables de manière unique. Tout objet, y compris (des ordinateurs, des capteurs, des RFID et des téléphones mobiles) seront en mesure d'émettre de l'information et éventuellement de recevoir des commandes. IdO ouvre la voie vers une multitude de scénarios basés sur l'interconnexion entre le monde physique et le monde virtuel , Cependant, comme d'autres concepts , celui-ci fait face à un nombre de problématiques qui nécessitent d'être étudiées pour permettre à l'Internet des objets d'atteindre son plein potentiel [13].

Dans ce chapitre, nous présentons d'abord l'IdO, son architecture, ainsi que les vulnérabilités et les Problématiques posées par l'Internet des objets. Nous consacrons par la suite le reste du chapitre à la définition de quelques notions utilisées dans le domaine de la sécurité et enfin, nous allons finir par une conclusion.

1.2 Définition de l'internet des objets

L'internet des objets (IdO) est une infrastructure dynamique d'un réseau global. Qui permet d'interconnecter des objets (physiques ou virtuels) grâce aux technologies de l'information et de la communication interopérables.

D'un point de vue conceptuel, l'Internet des objets affecte, à chaque objet une identification unique sous forme d'une étiquette lisible par des dispositifs mobiles sans fil, afin de pouvoir de communiquer les uns avec les autres. Ce réseau crée une passerelle entre le monde physique et le monde virtuel.

D'un point de vue technique, l'IdO consiste l'identification numérique directe et normalisée (adresse IP, protocole http...) d'un objet physique grâce à un système de communication sans fil (puce RFID, Bluetooth ou WiFi) [14].

FIGURE 1.1 – L'IdO connecte des objets en utilisant des capteurs et Internet [1].

1.3 L'évolution de l'écosystème de l'internet des objets

Les premiers objets connectés n'apparaissent que dans les années 1990. Il s'agit de grille-pain, machines à café ou autres objets du quotidien. En 2000, le fabricant coréen LG est le premier industriel à parler sérieusement d'un appareil électroménager relié à internet, Les années 2000 verront les premières expérimentations d'appareils connectés à Internet. Ils l'utilisent notamment pour consulter des informations de manière automatique.

En 2003, la population mondiale s'élevait à environ 6,3 milliards d'individus et 500 millions d'appareils étaient connectés à Internet. Le résultat de la division du nombre d'appareils par la population mondiale (0,08) montre qu'il y avait moins d'appareil connecté par personne. Selon la définition de Cisco IBSG, l'IdO n'existait pas encore en 2003 car le nombre d'objets connectés était faible.

En raison de l'explosion des Smartphones et des tablettes, le nombre d'appareils connectés à Internet a atteint 12,5 milliards en 2010, alors que la population mondiale était de 6,8 milliards.

C'est ainsi que le nombre d'appareils connectés par personne est devenu supérieur à 1 (1,84 pour être exact) pour la première fois de l'histoire.

En affinant ces chiffres, Cisco IBSG a situé l'apparition de l'IdO entre 2008 et 2009 (voir Figure 1.2).

En ce qui concerne l'avenir, Cisco IBSG estime que 50 milliards d'appareils seront connectés à Internet d'ici à 2020. Il est important de noter que ces estimations ne tiennent pas compte des progrès rapides d'Internet ni des avancées technologiques, mais reposent sur les faits avérés à l'heure actuelle [2]

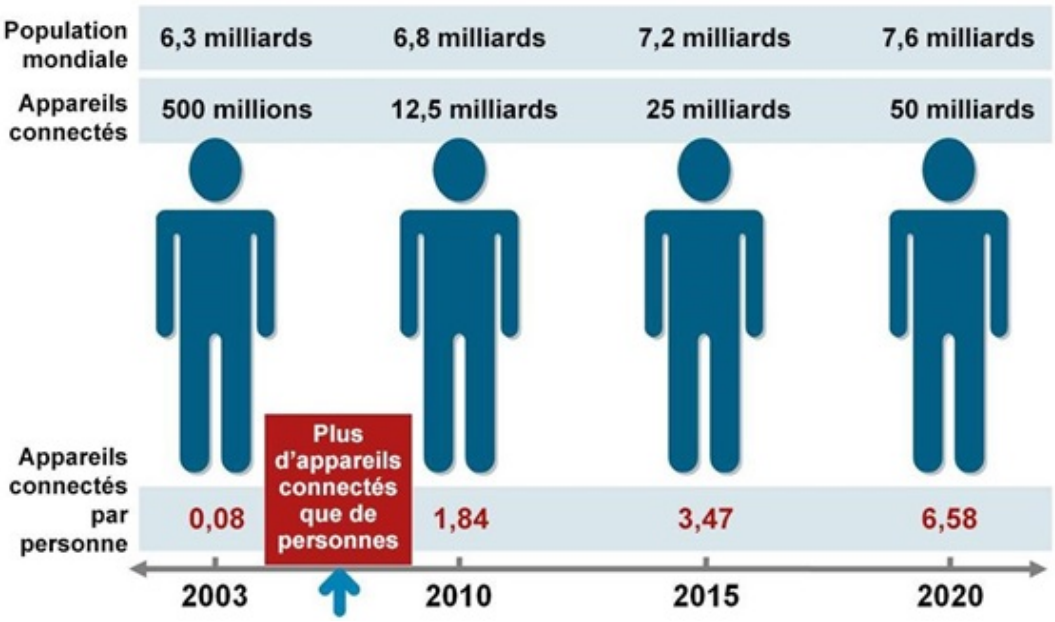

FIGURE 1.2 – L'évolution d'IdO entre 2003 et 2020 [2].

1.4 Technologie Machine-To-Machine M2M

Le concept de machine to machine, abrégé par M2M fonctionne sur un espace plus réduit. Les composants d'un tel réseau répondent généralement à une tâche particulière. Par exemple, au sein d'une usine des capteurs servent à récupérer des données sur le taux d'humidité et la chaleur.

Ces données sont transmises d'un capteur jusqu'à un serveur puis traitées via une application. Les domaines d'utilisation sont aussi larges que l'IdO : domotique, l'automobile, la santé, l'entreprise, etc. Cette technologie repose sur le hardware, sur l'ensemble du matériel et des coûts de connexion nécessaires à sa mise en place, tandis que l'IdO ne répond pas à des matériaux en particulier [15].

1.5 Internet des objets en tant que réseau de réseaux

L'IdO se compose d'un ensemble de réseaux hétérogène. Prenons l'exemple des véhicules. Plusieurs réseaux permettent de renforcer la sécurité routière, économiser du temps, etc. Les bâtiments commerciaux et résidentiels sont également équipés de différents systèmes pour avoir le contrôle global des différents équipements dédiés à la sécurité, l'éclairage, chauffage et aération, etc. Grâce à l'évolution de l'IdO, ces réseaux seront connectés à des fonctions évoluées de sécurité et de gestion (voir Figure 1.3). C'est ainsi que l'IdO atteindra son plein potentiel [2].

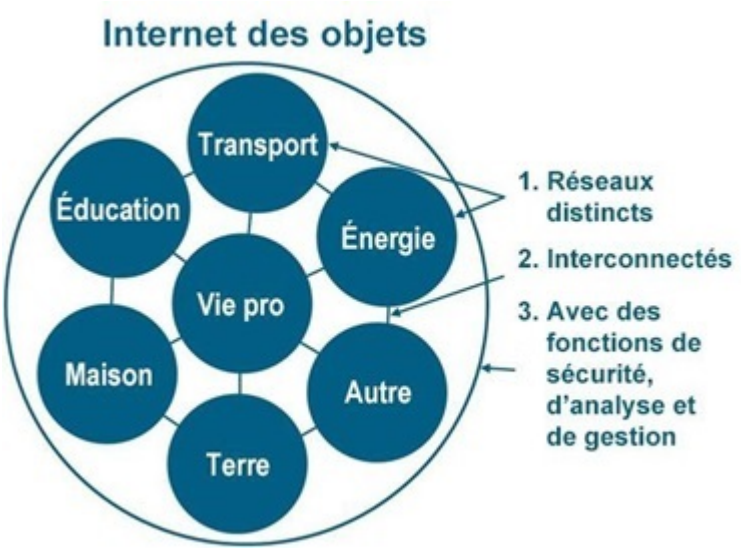

FIGURE 1.3 – L'IdO est en quelque sorte un réseau de réseaux [2].

1.6 Architecture de l'internet des objets

Les objets de l'environnement de l'internet des objets permettent de collecter, stocker et transmettre des données issues du monde physique. Ce sont des sources de données, qui possède au minimum un identifiant unique attaché à une identité ayant un lien direct ou indirect avec Internet.

On distingue deux types d'objet :

Les objets passifs : ils utilisent généralement un tag (puce RFID, code barre 2D). Ils ont une capacité de stockage faible (de l'ordre du kilooctet) et permettent de jouer le rôle d'identification. Ils peuvent parfois, dans le cas d'une puce RFID, d'embarquer un capteur (température, humidité) et être réinscriptibles.

Les objets actifs : ils peuvent être équipés de plusieurs capteurs, avec une grande capacité de stockage, capables d'accomplir des calculs et être en mesure de communiquer sur un réseau.

Précisons le rôle des différents processus présentés sur cette figure 1.4 :

- **Capter** permet de transformer une grandeur physique analogique en un signal numérique.
- **Concentrer** permet d'interfacer un réseau spécialisé d'objet à un réseau IP standard.
- **Stocker** permet de rassembler des données brutes, produites en temps réel, arrivant de façon non prévue.
- **Présenter** permet de collecter les informations de façon compréhensible par l'Homme, en lui offrant un moyen d'agir et/ou d'interagir.

Deux autres processus qui n'apparaissent pas sur le schéma :

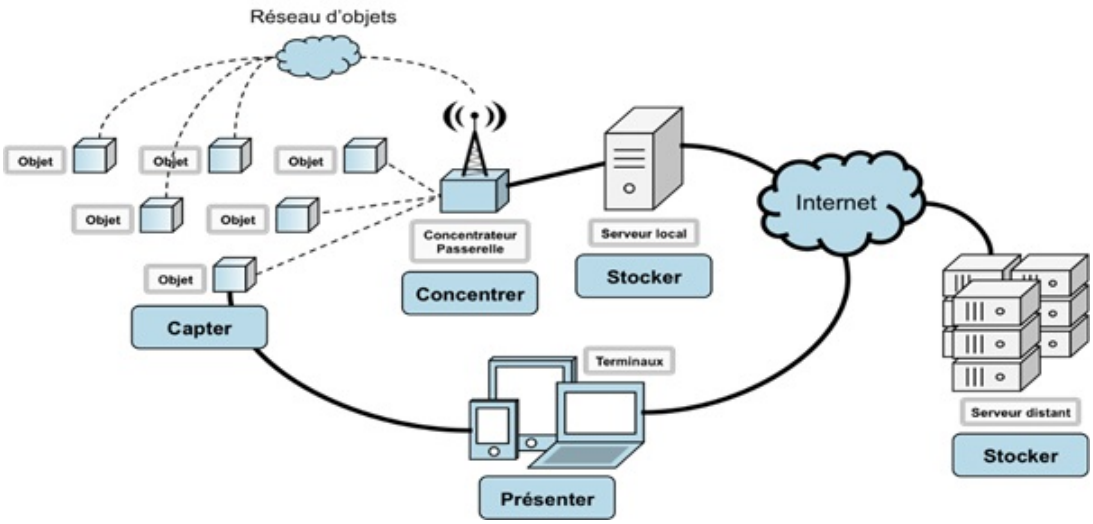

FIGURE 1.4 – Architecture d'IdO [3].

Le traitement des données : est un processus qui peut intervenir à tous les niveaux de la chaîne, depuis la capture de l'information jusqu'à sa restitution. Une stratégie pertinente, et commune quand on parle d'Internet des objets, consiste à stocker l'information. Cette stratégie est possible aujourd'hui grâce à des architectures distribuées, capables d'emmagasiner de grandes quantités d'information tout en offrant la possibilité de réaliser des traitements complexes en leur sein.

La transmission des données : est un processus qui intervient à tous les niveaux de la chaîne. Deux réseaux supportent des transmissions le réseau local de concentration (utilise ANT ,ZigBee et Zwave..) et le réseau WAN (WiFi, réseaux cellulaires. . . ..) [3].

1.7 Domaines d'application

L'IdO permettra le développement de plusieurs applications intelligentes à l'avenir qui toucheront essentiellement : la domotique, les villes, le transport, la santé et l'industrie [16]. Dans ce qui suit, nous citons brièvement des exemples du domaine d'applications de l'IdO.

1.7.1 La domotique en milieux urbains

Concerne la mise des dispositifs domestiques sur réseau. Cela permet de contrôler les différents équipements d'une maison depuis une même interface (une tablette ou un téléphone par exemple), mais aussi, il offre la possibilité de contrôler à distance ces équipements via la mise à disposition d'API sur le web.

Le champ d'application de l'IdO s'étale pour toucher les villes (smart cities), l'IdO permettra une meilleure gestion de tous les réseaux qui alimentent ces villes intelligentes (gaz, eau, électricité,

etc.). Des capteurs peuvent être utilisés pour améliorer la gestion des parkings et diminuer les embouteillages [16].

1.7.2 L'énergie

L'IdO propose des possibilités de gestion en temps réel pour une distribution et ine gestion efficaces de l'énergie, comme les réseaux électriques intelligents (smart grid). Cela permet d'avoir le contrôle de la consommation d'énergie et la détection des fraudes [16].

1.7.3 Le transport

Des voitures connectées aux systèmes de transport/logistique intelligents, l'IdO peut sauver des vies, réduire le trafic, minimiser l'impact des véhicules sur l'environnement et renforcer la sécurité routière [16].

1.7.4 La santé

Ce domaine de l'IdO assurera le suivi des signes cliniques des patients par la mise en place des réseaux personnels de surveillance, ces réseaux seront constitués de biocapteurs posés sur le corps des patients ou dans leurs lieux d'hospitalisation. Cela facilitera la télésurveillance des patients et apportera des solutions pour l'autonomie des personnes à mobilité réduite [16].

1.7.5 L'industrie

La technologie IdO permet aux usines d'améliorer l'efficacité de ses opérations, d'optimiser la production, d'améliorer la sécurité des employés, facilite la lutte contre la contrefaçon, la fraude et assure un suivi total des produits [16].

1.7.6 L'agriculture

L'IdO permettra une meilleure aide à la décision en agriculture. L'IdO servira non seulement à optimiser l'eau d'irrigation, mais aussi, cette technologie peut être utilisée pour lutter contre la pollution (l'air et les eaux) et améliorer la qualité de l'environnement en général [16].

1.8 Problématiques posées par l'Internet des objets

L'Internet des objets est l'évolution d'un réseau d'ordinateurs interconnectés vers un réseau d'objets interconnectés. Pour permettre a ce réseau d'atteindre son potentiel, plusieurs aspects doivent être étudiés et nécessitent de résoudre un certain nombre de problématiques : grande échelle, hétérogénéité, impact du monde physique, sécurité et vie privée des personnes [17].

1.8.1 Échelle de l'Internet des objets

Les différents problèmes liés à l'échelle de l'Internet des objets sont :

Adressage et nommage

L'accroissement du nombre d'objets nécessite un espace d'adressage très grand qui augmente avec la quantité d'informations que les serveurs de noms doivent stocker pour assurer leur rôle d'association entre les noms d'objet et leurs adresses. Cependant on l'espère le résoudre au moyen de technologies existantes comme par exemple le protocole IPv6 pour l'adressage des objets [17].

Découverte

Le processus de découverte permet aux machines de prendre connaissance de l'existence d'autres machines, soit par une découverte locale, soit en contactant un annuaire externe. Découvrir les objets est fondamental pour la réalisation de quelques scénarios de l'Internet des objets [17].

1.8.2 Hétérogénéité de l'Internet des objets

L'impact de l'hétérogénéité sur l'internet des objets sont les suivant :

Hétérogénéité fonctionnelle

Les objets sont pas égaux, les propriétés et les capacités des objets varient (statique ou mobile, alimenté par une batterie, ressources matérielles, capteurs, etc.) et à chacune d'elle correspond des contraintes particulières (durée de vie, tâches réalisables, etc.). Différentes approches et techniques doivent être considérées pour gérer les objets en fonction de leurs contraintes [17].

Hétérogénéité technique

L'Internet des objets est affecté par la diversité des composants matériels et logiciels utilisés pour construire les objets. Ils n'utilisent pas les mêmes systèmes d'exploitation et ne possèdent pas les mêmes interfaces de communication, ce qui conduit à une hétérogénéité technique significative [17].

1.8.3 Influence du monde physique sur l'Internet des objets

L'impact du monde physique sur internet des objets sont :

Variabilité

Le monde physique est un environnement qui évolue naturellement au cours du temps. Les objets d'un tel environnement doivent être en mesure de s'adapter dynamiquement aux changements qui surviennent au cours du temps [17].

Flux de Données

Les capteurs produisent des informations qui sont liées au temps, sous la forme de flux de mesures ou d'évènements. De manière générale, n'importe quelle information évoluant au cours du temps peut être représentée comme un flux .Contrairement aux ensembles finis de données classiques que l'on rencontre dans les bases de données ou sur le Web, qui nécessitent une réflexion différente en ce qui concerne la représentation des données et leur traitement [17].

1.8.4 Sécurité et vie privée

Les problèmes de sécurité et de vie privée dans l'Internet des objets sont les suivant :

Vulnérabilité

Les technologies modernes de sécurité (chiffrement, authentification, échange de clé, signature, etc.) sont inadéquates aux environnements de l'IdO ceci rend les objets vulnérables aux attaques informatiques cela, représente un danger pour les biens et les personnes [17].

Surveillance de masse

La simple présence d'un individu dans l'environnement entraîne implicitement la collecte d'informations, De manière générale, il n'est plus possible de savoir où, quand, pourquoi et par qui les données sont collectées. En effet, même en l'absence d'intentions malveillantes de la part des propriétaires d'objets, l'utilisation massive des réseaux sans fil facilite les écoutes clandestines. Enfin, l'identification unique des objets, rend possible la construction des profils possible. Ces profils uniques peuvent ensuite être utilisés pour suivre les individus en temps réel et à très grande échelle, en se basant sur la détection des objets et sur les différents capteurs placés dans l'environnement [17].

1.9 La sécurité dans Internet des Objets

1.9.1 Définition

La sécurité informatique est l'ensemble des moyens techniques qui visent à empêcher l'utilisation non-autorisée. On peut dire aussi que la sécurité informatique est un ensemble des moyens mis en œuvres pour réduire la vulnérabilité d'un système contre les menaces accidentelles ou intentionnelles [18].

1.9.2 Objectifs de la sécurité

La sécurité Informatique d'une manière générale vise à assurer plusieurs objectifs, dont les cinq principaux sont : l'authentification, la confidentialité, l'intégrité, la disponibilité et la non-répudiation [18].

A) Authentification

L'authentification peut être définie comme le processus de prouver une identité revendiquée. Lorsqu'il existe une seule preuve de l'identité (mot de passe) en parle d'authentification simple lorsque cette dernière nécessite plusieurs facteurs on parle alors d'authentification forte.

B) Confidentialité

La confidentialité est le fait de s'assurer que l'information n'est accessible qu'à ceux dont l'accès est autorisé, permet de garder la communication des données privée entre un émetteur et un destinataire. Le chiffrement des données est la seule solution pour assurer la confidentialité des données.

C) Intégrité

L'intégrité permet de garantir que les données sont bien celles que l'on croit être, donc permet de garantir la protection des données contre les modifications et les altérations non autorisées.

D) Disponibilité

La disponibilité est un service réseau qui permet de donner une assurance aux entités autorisées d'accéder aux ressources réseaux. L'objectif est d'évité les attaques de type Denis de service.

E) Non-répudiation

Non-répudiation permet de garantir qu'une transaction ne peut être niée et qu'un message a bien été envoyé par un émetteur et reçu par un destinataire aucun des deux ne pourra nier l'envoi ou la réception du message.

1.9.3 Outils de cryptographie

Pour réaliser la sécurité dans n'importe quel modèle de communication, il est important de chiffrer les messages transmis aux nœuds selon un arrangement De gestion des clés connues [19].

Chiffrement est un système cryptographique assurant les confidentialités. Pour cela, il utilise des clés. Selon cette utilisation, on distingue deux classes : symétrique et asymétrique [19].

- **Chiffrement symétrique** quand il utilise la même clé pour chiffrer et déchiffrer.
- **Chiffrement asymétrique** quand il utilise des clés différentes : une paire composée d'une clé publique, servant au chiffrement, et d'une clé privée, servant à déchiffrer la clé publique.

Gestion des clés La gestion des clés et l'un des aspects les plus difficiles de la configuration d'un système cryptographique de sécurité. Pour qu'un tel système fonctionne il est important que, chacun des utilisateurs doivent disposer d'un ensemble de clés secrète ou de pair de clés. Cela implique de générer les clés et de les distribuer de manière sécurisée. Chaque utilisateur doit

aussi pouvoir gérer et enregistrer ses clés de manière sûre [20]. Ainsi la figure 1.5 [4], illustre les différentes fonctions de la gestion des clés.

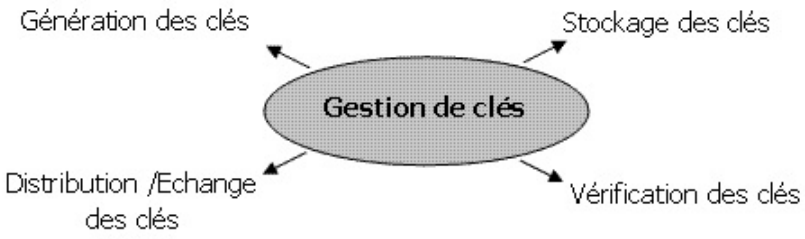

FIGURE 1.5 – Les différentes fonctions de la gestion des clés [4].

1.10 Conclusion

Dans ce chapitre nous avons survolé d'une façon générale la technologie Internet des Objets. Nous avons défini c'est quoi l'IdO. Ensuite, nous avons présenté son évolution, un modèle d'architecture, et enfin nous avons abordé les problématiques posées par l'internet des objets.

Ce chapitre a été consacré à la présentation de l'IdO, ses domaines d'application ainsi que son importance. Aussi, nous avons décrit en détail les problèmes relatifs à son déploiement. Puis nous avons mis l'accent sur quelques concepts liés à la sécurité.

Le chapitre suivant sera consacré à l'étude de quelques travaux récents réalisés dans le contexte de gestion des clés dans l'internet des objets.

Chapitre 2

Quelques articles sur la gestion des clés dans l'internet des objets

2.1 Introduction

Il est impossible d'aborder les défis à relever pour l'Internet des Objets sans parler de la sécurité. Elle prévoit de nouveaux défis de sécurité qui appellent à une révision substantielle des solutions de sécurité existantes ou le développement de nouvelles approches [16].

Ce chapitre sera consacré à la présentation et l'étude critique de quelques solutions proposées sur l'axe de gestion des clés dans l'IdO.

Pour cela, nous commençons d'abords par une problématique puis déterminer nos critères d'analyse, suivis par une classification. Par la suite, nous présentons notre description et une discussion de ces dernières et enfin, une comparaison des travaux analysés pour clôturer le chapitre.

2.2 Problématique

La gestion des clés dans l'IdO est le problème le plus délicat de la cryptographie, car elle représente un mélange d'objets (les objets sont pas égaux), les propriétés et les capacités des objets varient (statique ou mobile, alimenté par une batterie, ressources matérielles, capteurs, etc.) et à chacune d'elle correspond à des contraintes particulières (durée de vie, tâches réalisables, etc.) ceci rend Les technologies modernes de sécurité inadéquates aux environnements de l'IdO et le mécanisme de gestion des clés plus complexe .

2.3 Critères de comparaison des solutions

Afin de bien évaluer les articles et les travaux que nous avons lus et en se basant sur les objectifs, et les besoins de l'Internet des Objets. Nous avons établi une liste de critères d'évaluation. Notre liste comprend les éléments suivants :

2.3.1 Résistance aux attaques (résilience)

Cette propriété indique la capacité de faire face aux informations d'identification volées. La résilience d'un système de gestion de clé est faible si un attaquant qui extrait des informations d'un seul périphérique (par exemple, hôte Internet, nœud de capteur) est capable d'accéder à tous les flux d'informations sécurisés. En revanche, la résilience est élevée si un attaquant ne peut imiter l'identité de l'appareil qui a été attaqué. Sans surprise, une forte résilience est souhaitable pour les appareils IdO [21].

2.3.2 Scalabilité

La scalabilité est l'une des caractéristiques essentielles d'un modèle de gestion de clé dans l'Internet des Objets, elle permet d'assurer son bon fonctionnement lors des changements dynamiques de la taille du réseau d'objets [22].

2.3.3 Consommation d'énergie

Cette propriété spécifie à la fois la communication et les frais généraux de calcul de l'exécution du processus de négociation. Cette propriété se référera principalement aux frais généraux imposés dans les nœuds des capteurs, car ils sont beaucoup plus restreints aux ressources que les autres hôtes Internet et l'énergie consommée par l'envoi et la réception d'informations via le canal sans fil est assez élevée et la plupart des nœuds capteurs sont alimentés par batterie [8].

2.3.4 Classification des protocoles de gestion des clés

Les protocoles de gestion des clés sont traditionnellement classés dans la littérature en trois grandes catégories (Figure 2.1) : centralisée, décentralisé et distribuée [23].

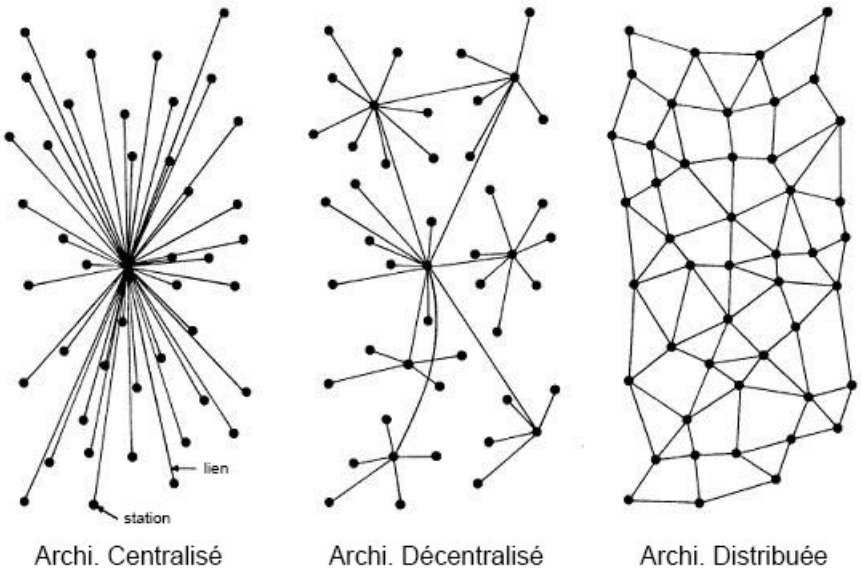

FIGURE 2.1 – Différentes architectures de l'IdO.

- **Architecture centralisé** Un système centralisé c'est un système où tout le monde dépend d'une même autorité, un serveur à priori dans le cas informatique.

- **Architecture décentralisé** Système décentralisé est un système de communication, où toute entité (individu, association, organisation, ... etc.) puisse être une partie d'un réseau qui n'a pas d'autorité principale, et que ces autorités puissent parler entre elles.

- **Architecture distribuée** Système distribué est un système qui s'oppose à celui d'architecture centralisée, toutes les ressources ne se trouvent pas au même endroit ou sur la même machine. Internet est un exemple de réseau distribué puisqu'il ne possède aucun nœud central. C'est un ensemble d'ordinateurs indépendants connectés en réseau, et communiquant via ce réseau [24].

Nous classifions les travaux que nous avons analysés comme suit (Figure 2.2)

2.4 Protocoles de gestion des clés

2.4.1 Architecture centralisé

A Cooperative End to End Key Management Scheme for E-health Applications in the context of Internet of Things

Dans cet article [5], MR.Abdmeziem et D.Tandjaoui ont proposés un nouveau système de gestion des clés basé sur la collaboration pour établir un canal de communication sécurisé entre

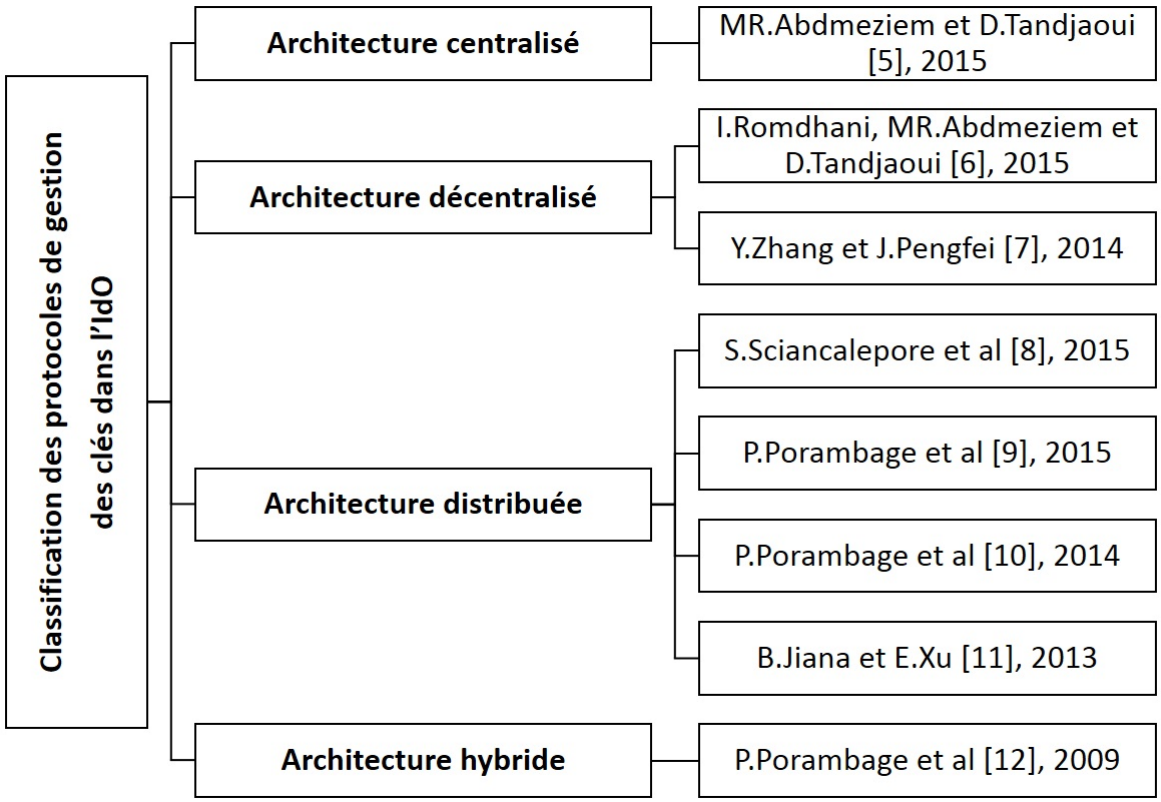

FIGURE 2.2 – Le schéma de classification des protocoles de gestions de clés.

un nœud à ressources limitées et une entité distante (serveur). Le canal sécurisé permet au nœud CN de transmettre les données tout en assurant la confidentialité et l'authentification.

La solution est basée sur le déchargement des primitives cryptographiques vers les tiers qui représentent un élément clé du protocole.

Après la psehase d'initialisation où chaque CN est préchargé avec un ensemble d'identités de tiers (TP_i) avec des clés pré-partagées (K_{CN,TP_i}), le protocole se déroule avec des phases successives. La Figure 2.4 illustre les phases successives de protocole.

Phase 1 : *Echange initial.* Le nœud CN initie l'échange en envoyant un message CN_Hello (A) à UN. Le message informe l'UN sur les politiques de sécurité et le processus d'établissement de la clé coopérative qu'il prend en charge. Si le nœud UN est d'accord, il sélectionne l'une des politiques de sécurité proposées et répond avec un message d' UN_Hello (B). Les nonces sont inclus dans les messages échangés pour empêcher les attaques de répétition.

Phase 2 : *Sécuriser la connexion entre les entités.* Cette phase suit la connexion réussie entre le CN et l'UN. Il vise à établir un canal sécurisé soit entre CN et TP_i, soit entre TP_i et UN.

Dans le message (C), le CN informe les tiers sur l'identité de l'UN. Le message comprend un code d'authentification des messages (MAC) et est crypté à l'aide de K_{CN,TP_i} . Les tiers expriment

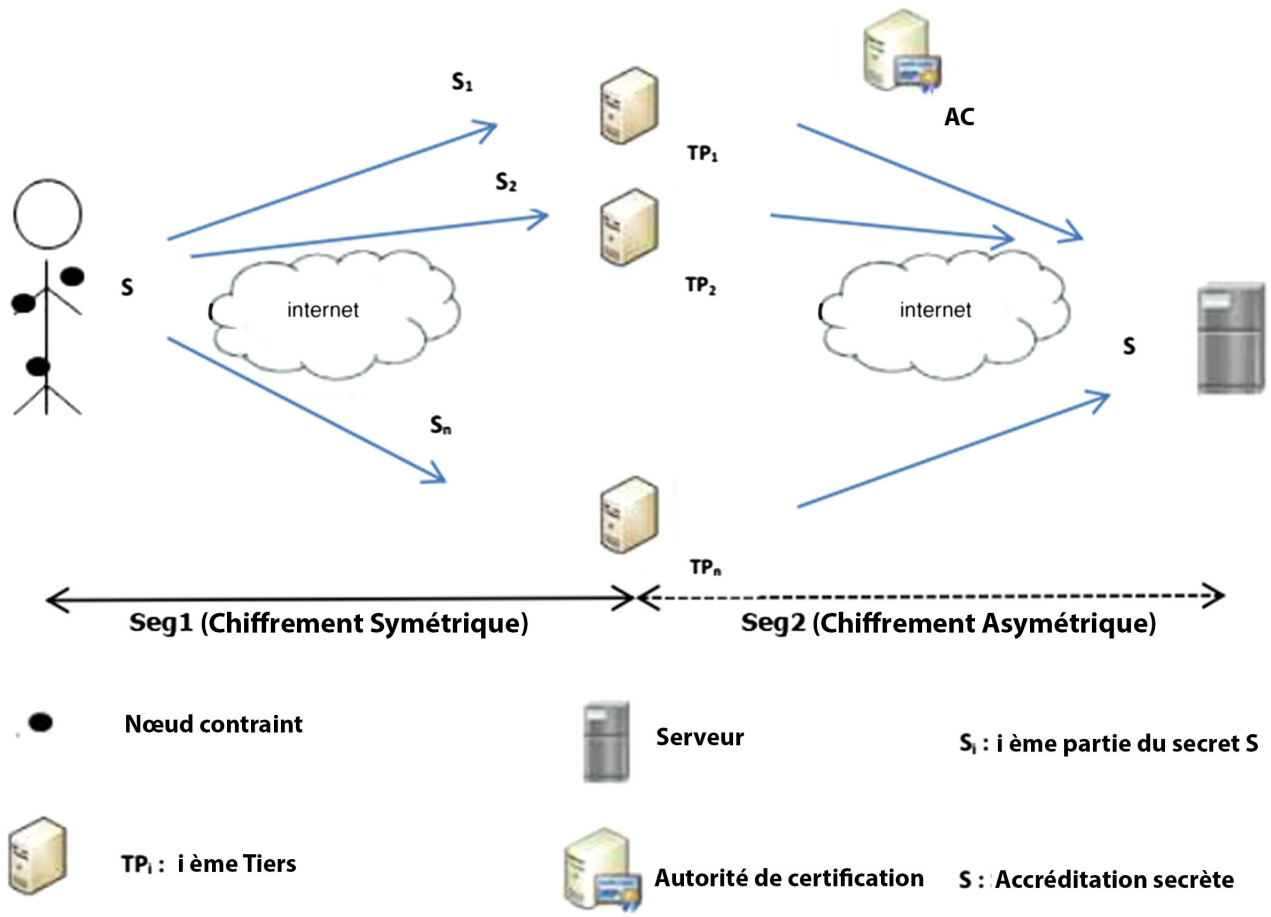

FIGURE 2.3 – Modèle physique du réseau [5].

leur volonté de faire partie du protocole clé d'échange par le message (D). Il convient de noter que tous les tiers demandés ne répondent pas avec le message D en raison de l'épuisement éventuel des ressources ou de toute autre raison. Par conséquent, Ils considèrent que seul m TP_i ($m <= n$) répond avec le message (D) exprimant sa volonté de participer au processus d'échange de clés.

Dans le message (E), chaque TP_i fournit aux UN son certificat contenant sa clé publique (délivré par AC) et demande à l'UN son propre certificat. L'UN vérifie que le tiers a fourni une clé publique valide. Il répond donc avec le message F qui contient le certificat demandé. Nous soulignons que tous les messages contiennent des nonces contre les attaques de répétition.

Phase 3 : *Prouver la représentativité des tiers du CN à l'UN*. Cette phase vise à prouver la représentativité du CN par les tiers à l'UN. L'authentification est réalisée en utilisant les clés pré-partagées entre CN et TP_i. Dans le message (G), l'UN demande les clés par paires partagées avec le TP_i. Le CN applique une fonction de hachage sur chaque touche pour la garder confidentielle et l'envoyer à l'UN par l'intermédiaire du message (H). L'authentification se produira plus tard après avoir reçu le message (J) des troisièmes entités contenant les hachages de la clé.

Phase 4 : *Génération et livraison secrètes*. Lors de la préparation réussie des entités im-

pliquées, le CN génère un secret S utilisé plus tard entre CN et UN. CN applique un schéma de redondance d'erreur au secret original S. L'objectif est de permettre à l'UN de récupérer le secret sans nécessiter la réception de tous les paquets, dans le cas où certains d'entre eux sont modifiés pendant le processus de transmission. Dans cette solution, ils ont choisi le code Reed-Solomon [25].

Le secret est divisé en m parties $S_1, S_2, \ldots S_m$. Chaque partie est envoyée au TP_i approprié dans le message (I). La communication est sécurisée à l'aide de la clé symétrique K_{CN,TP_i} . En recevant le message (I), chaque TP_i utilise la clé publique de l'UN pour chiffrer le message (J) qui contient la partie secrète S_i, K_{CN,TP_i} 's hash et la signature de TP_i qui couvre tous les champs du message. Après son décryptage, l'UN vérifie l'authenticité de chaque message à l'aide de la clé publique de TP_i. Si les messages sont authentifiés, l'UN vérifie la représentativité de TP_i du CN. La vérification se fait par la comparaison des hachages reçus dans le message (H) et ceux reçus dans le message (J). Si les hachages correspondent, le TP_i agit au nom du CN comme prétendent.

L'UN reconstruit alors le secret S après avoir reçu suffisamment de paquets. Le secret est utilisé pour dériver d'autres informations clés avec les non échangés pendant les messages précédents. Les messages (I) et (J) contiennent des nonces pour éviter les attaques de répétition.

Phase 5 : *Phase de terminaison.* Cette phase conclut les échanges par le message (K) en prouvant au CN la connaissance du secret S.

La secret S, ainsi que les nonces échangés, sont utilisés par l'UN et le CN pour dériver une clé principale. Le processus de dérivation est assuré par une fonction de hachage convenue lors de la première phase. Les deux parties peuvent ensuite dériver des clés de connexion d'état pour le cryptage et l'authentification des données échangées. Par conséquent, un canal de bout en bout sécurisé est créé entre des capteurs et des serveurs à distance.

Discussion et critiques

Les auteurs de [5], ont proposés une solution basée sur le déchargement de primitives cryptographiques lourdes à des tiers afin d'autoriser les nœuds à ressources limitées à établir un secret partagé de bout en bout avec n'importe quel serveur distant. Limiter les demandes de calcul pour les nœuds CNs diminue leur consommation d'énergie et augmente ainsi leur durée de vie de la batterie. Le critère de scalabilité est respecté vu que le modèle permet d'intégrer de nouveaux capteurs. Le nouveau capteur doit passer par une phase d'initialisation, Lors d'une phase d'initialisation réussie, le nouveau capteur peut établir un canal sécurisé de bout en bout avec une entité distante. Il n'est pas résistant aux attaques parce qu'un attaquant peut compromettre le secret S échangé, s'il rassemble toutes les clés d'échanges entre les CNs et les $TPis$ car les clés $K_{CN,TPis}$ sont fixe.

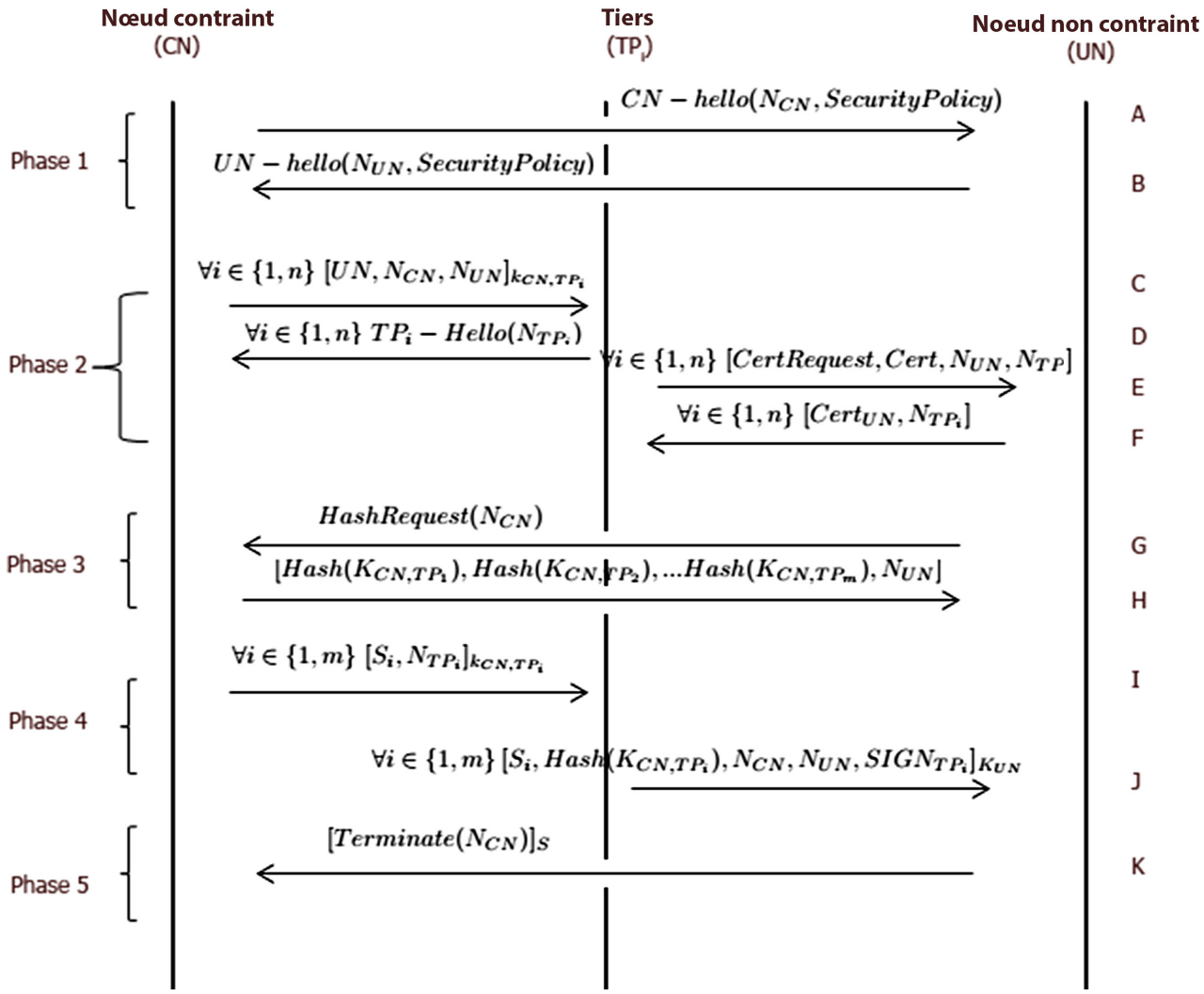

FIGURE 2.4 – Illustration des différentes phases et échanges de messages de schéma [5].

2.4.2 Architecture décentralisé

A) A Decentralized Batch-based Group Key Management Protocol for Mobile Internet of Things (DBGK)

Dans cet article [6], I.Romdhani, MR.Abdmeziem et D.Tandjaouiont ont proposé un nouveau protocole de gestion de clé de groupe décentralisé par lot appelé DBGK pour sécuriser les communications multicast dans le contexte d'IdO.

Les applications multidiffusion peuvent être classées en trois catégories : un à plusieurs, beaucoup à plusieurs, et beaucoup à un. Pour sécuriser ces types de communications de groupe, les protocoles de gestion de clé de groupe sont impliqués pour générer, distribuer et maintenir une clé partagée.

Dans ce contexte, le protocole est conçu pour tenir compte de la pénurie des ressources et de la mobilité des appareils IdO. Pour atténuer le problème du point unique d'échec, ils optent pour une architecture décentralisée (Comme KEK Oi, GKMS est partagé entre Oi et GKMS, ce

dernier sera en mesure de gérer les demandes de la zone concernée jusqu'à sa restauration. GKMS servira de back-up AKMS, ce qui atténuera le problème du point unique d'échec). En outre, pour réduire le phénomène 1-affects-n, ils considèrent que chaque sous-groupe du réseau est sécurisé avec une clé de groupe différente. Ils utilisent une approche axée sur le temps où une clé de groupe est utilisée dans chaque intervalle de temps.

Le protocole ne concerne que les membres actifs dans le processus de réécriture. Cela permet aux autres membres de rester en mode veille, ce qui permet d'économiser de l'énergie. Ce dernier s'appuie sur un modèle de réseau divisé en plusieurs domaines. Chaque zone couvre un certain nombre d'objets et elle est gérée par un serveur de gestion de clé de la zone (AKMS) (voir la Figure 2.5). L'AKMS établit une clé de chiffrement de trafic (TEK) pour chaque objet dans la zone. Chaque zone possède sa propre TEK qui est différente de la TEK d'autres zones. À l'aide de TEK, les objets sécurisent leurs communications dans la zone. En cas d'événement, le AKMS est responsable de la mise à jour de la TEK. L'événement peut être déclenché par un nouvel objet reliant la zone, un objet existant quittant la zone, ou un objet en mouvement entre les différentes zones. L'AKMS conserve une liste appelée Active Object List (AOL), qui stocke les informations d'identification livrées aux objets pour chaque intervalle de temps. Ces informations d'identification sont utilisées pendant le processus de réécriture. Un General Key Management Server (GKMS) gère les différents AKMS et définit la politique de sécurité pour l'ensemble du groupe. En particulier, il assure la politique de contrôle d'accès appropriée de chaque domaine. Le modèle de réseau est positionné dans la catégorie des architectures décentralisées avec un TEK différent pour chaque sous-groupe. Le réseau est hétérogène et contient deux types d'entités dotées de différentes capacités en termes de puissance informatique et de ressources énergétiques.

Dans le protocole, les clés de chiffrement de trafic sont générées à l'aide d'une fonction à sens unique qui a comme entrée une clé à long terme SK et un ticket valide Ti, t. Par définition, une fonction à sens unique garantit que les données utilisées comme entrée ne peuvent pas être récupérées à partir de la sortie résultante (dans ce cas TEKi, t). Ainsi, la divulgation d'une clé ne donne aucune information supplémentaire à un attaquant pour récupérer des clés antérieures. De plus le protocole réduit le nombre d'entités fiables au minimum. En effet, chaque zone est gérée par une seule gestion de clé de zone approuvée, et l'ensemble du groupe est géré par une clé de confiance serveur de gestion. La perte d'un membre en raison d'un échec ou d'une attaque n'affecte pas les autres membres. En outre, ce protocole s'appuie sur le serveur de gestion des clés générales pour gérer la perte d'un serveur de gestion de la clé de zone afin de garder la zone disponible.

Discussion et critiques

La consommation d'énergie du protocole augmente avec l'augmentation des nœuds répondants. Ceci est causé par le calcul et les frais généraux de communication causés par l'échange d'un nombre important de messages. Il convient de noter que la consommation d'énergie du protocole

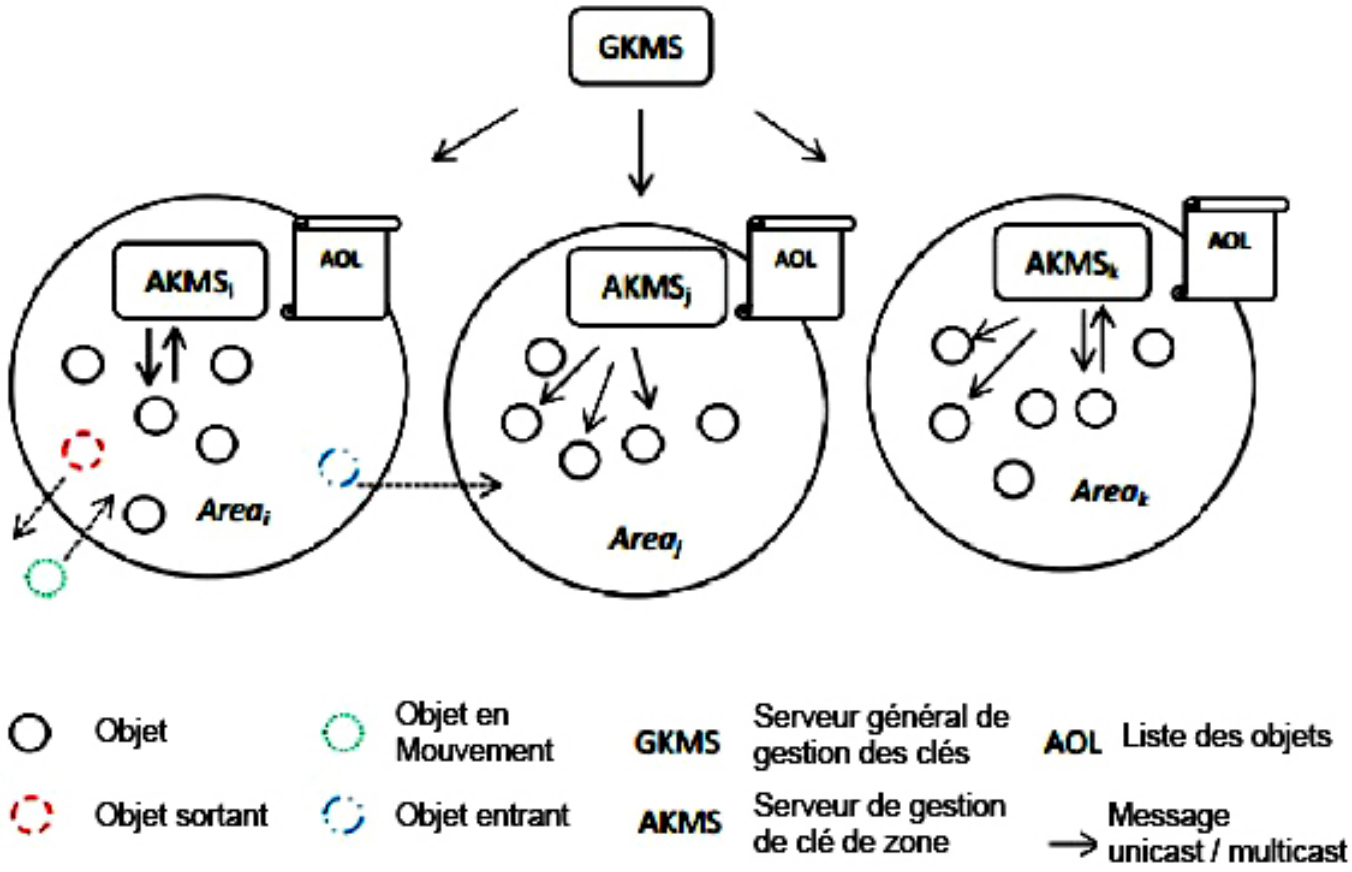

FIGURE 2.5 – Modèle de réseau : une architecture décentralisée basée sur une clé de groupe indépendante par zone [6].

est inférieure à près d'une proportion de 50 % des nœuds en possession d'un ticket valide au moment de l'événement de départ. En s'appuyant sur les résultats d'évaluation de la sécurité et des performances, ils peuvent prouver que le protocole est plus adapté aux groupes avec un grand nombre de membres qui peuvent rejoindre le réseau pendant une longue période (sans être nécessairement actifs) et quitter le réseau de manière inattendue. Donc le protocole leur permet de rester en mode veille sans être interrompu par les opérations de réenclenchement.

La mobilité est traitée avec moins d'opérations en particulier dans le cas d'un nœud mobile sans tickets valide. En outre, le secret avant est assuré pour les événements de mobilité sans aucune hypothèse sur la zone source. Considérer l'adoption du protocole pour les applications IdO contraintes et hautement dynamiques.

B) An efficient and hybrid key management for heterogeneous WSN in context of IoT

Dans cet article [7], les auteurs tentent d'améliorer la gestion de clés dans les réseaux de capteurs sans fil hétérogènes. Ils proposent une méthode appelée EHKM, où il s'agit de combiner la cryptographie sur les courbes elliptiques avec la cryptographie symétrique pour améliorer la

sécurité du réseau tout en prenant en considération la limitation de ressource des capteurs sans fil.

La méthode opte pour un modèle de réseau hiérarchique et hétérogène (voir la figure 2.6), de ce fait ils distinguent :

Base Station (BS) : dotée d'une grande capacité de calcul, les différentes informations récoltées sont traitées à son niveau, du fait qu'elle ait suffisamment d'énergie et que tous les nœuds lui font confiance.

High-end sensors (H-sensor) : dotés d'une grande capacité énergétique, d'une largeur de bande passante, de capacités de calcul et d'espace de stockage assez importants ; de plus ils sont équipés de matériels inviolables qui empêchent toute donnée d'être extraite si le nœud est capturé, contrairement aux L-Sensors.Ils sont, par défaut, Cluster-Heads.

Low-end sensors (L-sensor) : ont une capacité énergétique inferieure à celle des Hsensor. Tous les nœuds sont statiques et connaissent leurs propres coordonnées.

La sécurisation de la communication entre les nœuds suit un système hybride, combinant des clés asymétriques et des clés symétriques.

La communication entre la station de base et les H-sensors se fait soit directement soit par l'intermédiaire d'autres H-sensors.

La communication entre les deux entités se fait en utilisant les paires de clés générées avec ECC avant le déploiement des nœuds. Les H-sensors et les L-sensors utilisent l'algorithme de l'échange de clés de Diffie-Hellman pour établir une clé partagée entre les deux nœuds.

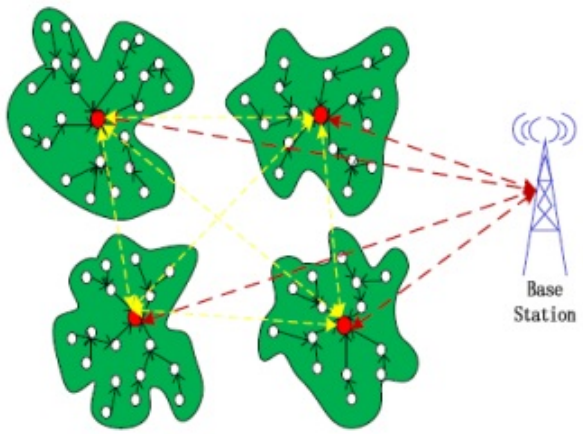

FIGURE 2.6 – Hiérarchie de la méthode EHKM [7].

Discussion et critiques

L'analyse des performances de ce système démontrent une amélioration de la gestion des clés en comparaison avec la méthode E-G. EHKM est donc un moyen efficace et hybride pour la gestion des clés dans les réseaux de capteurs sans fil hétérogènes car permet un gain considérable en matière d'énergie et les résultats d'analyse de cette méthode prouvent sa capacité à améliorer la sécurité du réseau, sa résilience et à réduire l'espace de stockage des clés.

L'approche assure la scalabilité car si un nœud veut rejoindre le réseau après avoir été chargé avec sa clé privée, la clé publique des H-sensors et une fonction de hachage, la station de base informe le H-sensor de l'arrivée d'un nouveau nœud. Le H-sensor lui envoie des informations sur le routage ainsi que le nombre r, qui sera utilisé par le nouveau nœud pour établir des clés de session avec les autres nœuds du cluster.

2.4.3 Architecture distribuée

A) Key Management Protocol with Implicit Certificates for IoT systems

Dans cet article [8] S.Sciancalepore et al, décrivent un protocole de gestion des clés pour mobile et les systèmes industriels Internet des Objets, KMP (Key Management Protocol) intégré à la couche 2 de la pile protocolaire 802.15.4 dans le but de fournir des services de sécurité pour les différents scénarios d'IdO. Le protocole proposé s'appuie sur un échange "fixe" ECDH (Elliptic Curve DiffieHellman), avec des coefficients publics implicitement certifiés à l'aide de ECQV (Elliptic Curve Qu-Vanstone) et il a été compléter par l'échange des nonces ainsi l'authentification des séquences des messages échangés, afin de garantir l'authentification mutuelle et la fraîcheur dans la dérivation de la clé.

L'algorithme KMP Proposé :

Le schéma KMP développé repose sur l'échange de quatre messages logiques différents. Les deux premiers messages portent les matériaux clés (le certificat implicite de ECQV et un nonce). Les certificats implicites ECQV offrent des services d'authentification et d'accord clés dans le sens où chaque nœud est capable de calculer, via un mécanisme ECDH fixe, un secret partagé à partir d'une clé publique authentifiée. Les deux derniers messages, par contre, sont échangés pour finaliser l'authentification mutuelle. Pour éviter la répétition des attaques sur la deuxième partie du protocole, Le champ d'authentification stocké dans ces messages est calculé en tenant compte des nonces échangés initialement. Toutes les opérations KMP sont traitées à la couche 2 de la pile protocolaire en fonction de la technologie IEEE 802.15.4. En outre, le secret négocié pendant la procédure sera adopté pour générer des matériaux clés à utiliser avec l'algorithme CCM *, qui représente la primitive de cryptographie de la norme IEEE 802.15.4. Comme indiqué dans la Figure 2.7.

Discussion et critiques

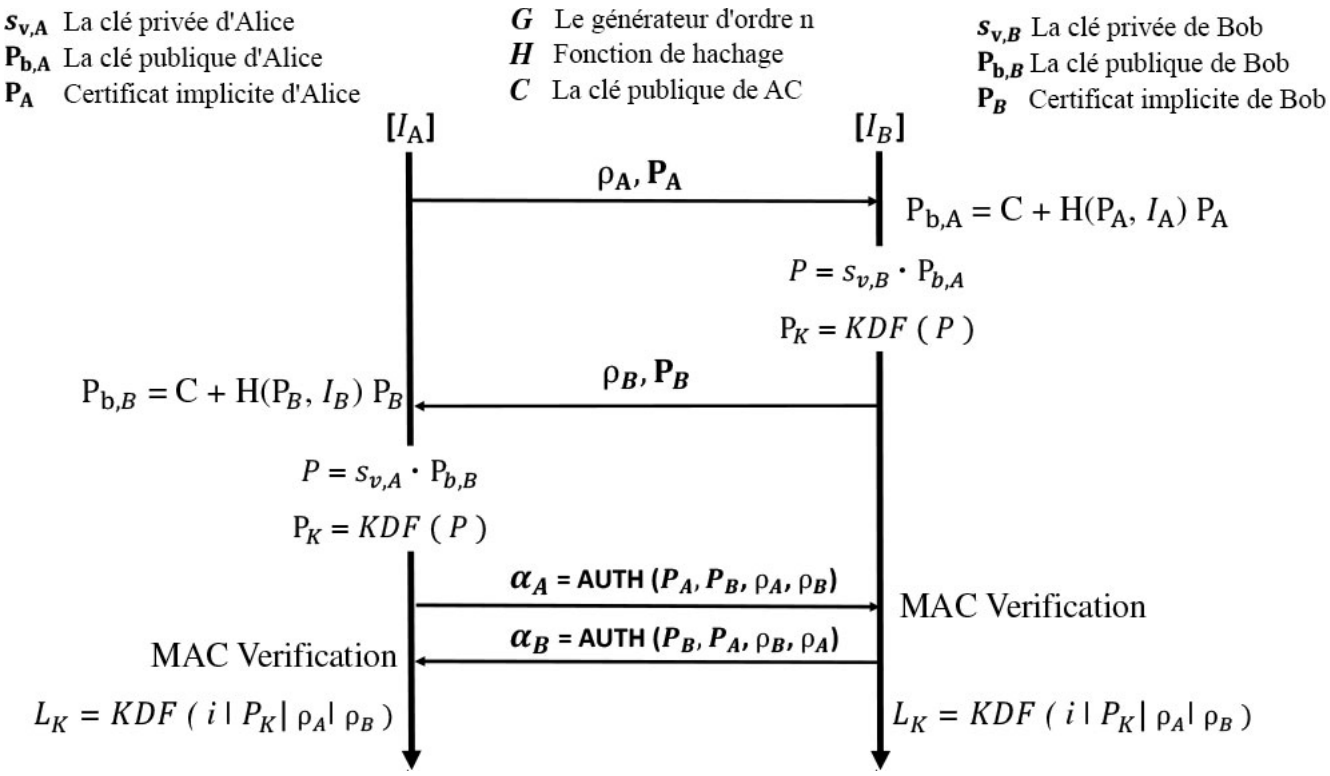

FIGURE 2.7 – Protocol de négociation des clés [8].

Le protocole de gestion de clé proposé dans [8], cible une négociation de clé, une authentification de nœud, une reconfiguration rapide et une protection contre les attaques de répétition. L'utilisation de certificats implicites en conjonction avec un échange de messages optimisé donne des gains impressionnants en termes de consommation. Une évaluation préliminaire de performance a été réalisée pour déterminer son efficacité dans des scenarios simples mais significatifs, cependant la solution n'est pas proposée dans des situations d'IdO complexe.

B) Proxy-based End-to-End Key Establishment Protocol for the Internet of Things

Dans cet article [9], P.Porambage et al, ont proposé un protocole d'établissement de clé basé sur proxy pour l'Internet des objets (IdO), qui permet à deux dispositifs inconnus à haute ressource (Initiateur A et Répondeur B) d'initier une communication sécurisée bout en bout (end-to-end E2E). Les dispositifs fortement puissant devraient maintenir des connexions sécurisées avec les périphériques voisins a ressource limitée dans les réseaux locaux dans lesquels ils sont déployés. Les dispositifs a ressource limitée fonctionnent en tant que proxys et défendent en collaboration les opérations cryptographiques coûteuses pendant le calcul de la clé de session.

Selon le flux de messages du protocole d'établissement de clé basé sur proxy (Figure 2.8), l'initiateur et le répondeur délèguent les opérations de consommation de ressources aux proxys. N'importe quel proxy de l'initiateur ou de répondeur n'a pas une connaissance totale du sous-ensemble P des proxys participant. Par conséquent, bien que chaque proxy contribue à calculer une partie de la clé secrète finale K_{AB} et aucun des proxys ne peut collaborer avec d'autres pour

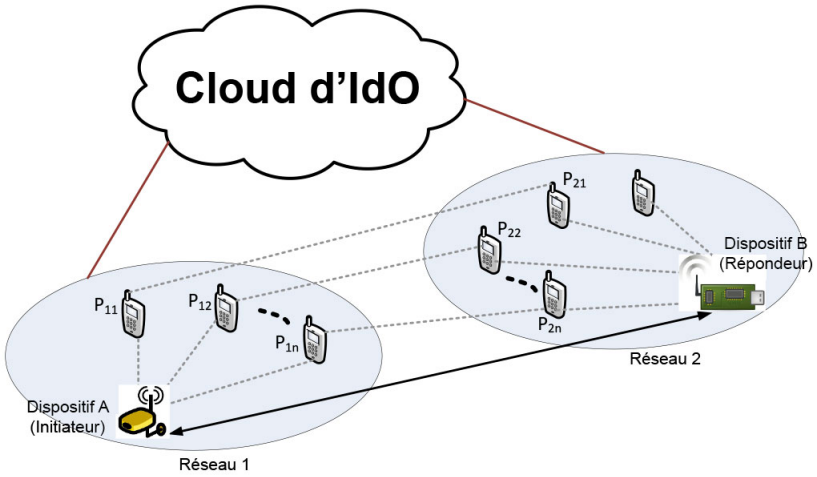

FIGURE 2.8 – Modèle physique du réseau [9].

dériver la clé finale.

Discussion et critiques

Le protocole proposé génère beaucoup moins de frais de calcul et de communication sur les noeuds à ressources limitées que le protocole d'échange de clé DH. En raison du profil d'énergie extrêmement faible et de la cohérence avec les caractéristiques du réseau IdO. Pour le critère de scalabilité est respecté vu que le modèle proposé est distribué. Pour la résistance aux attaques, le protocole est bien securisé contre plusieurs attaques selon sa politique. Ce protocole d'établissement clé peut être facilement intégré avec de nombreux systèmes de sécurité largement adoptés dans l'environnement IdO.

C) Two-phase Authentication Protocol for Wireless Sensor Networks in Distributed IdO Applications

Dans cet article [10] P.Porambage et al, proposent un mécanisme d'authentification implicite basé sur un certificat pour les RCSFs dans les applications d'IdO distribuées. Le protocole d'authentification développé permet aux nœuds capteurs et aux utilisateurs finaux de s'authentifier et d'établir des connexions sécurisées.

La Figure 2.9, est une illustration d'architecture de réseau pour un schéma d'authentification proposé, où les utilisateurs finaux peuvent collaborer avec différents périphériques de bord afin d'obtenir une information ou un service particulier. Les réseaux de bord peuvent inclure des dispositifs hétérogènes et les utilisateurs finaux peuvent être des êtres humains ou des entités virtuelles.

Sur la base de la Figure 2.9 l'authentification est envisagée pour trois scénarios de communi-

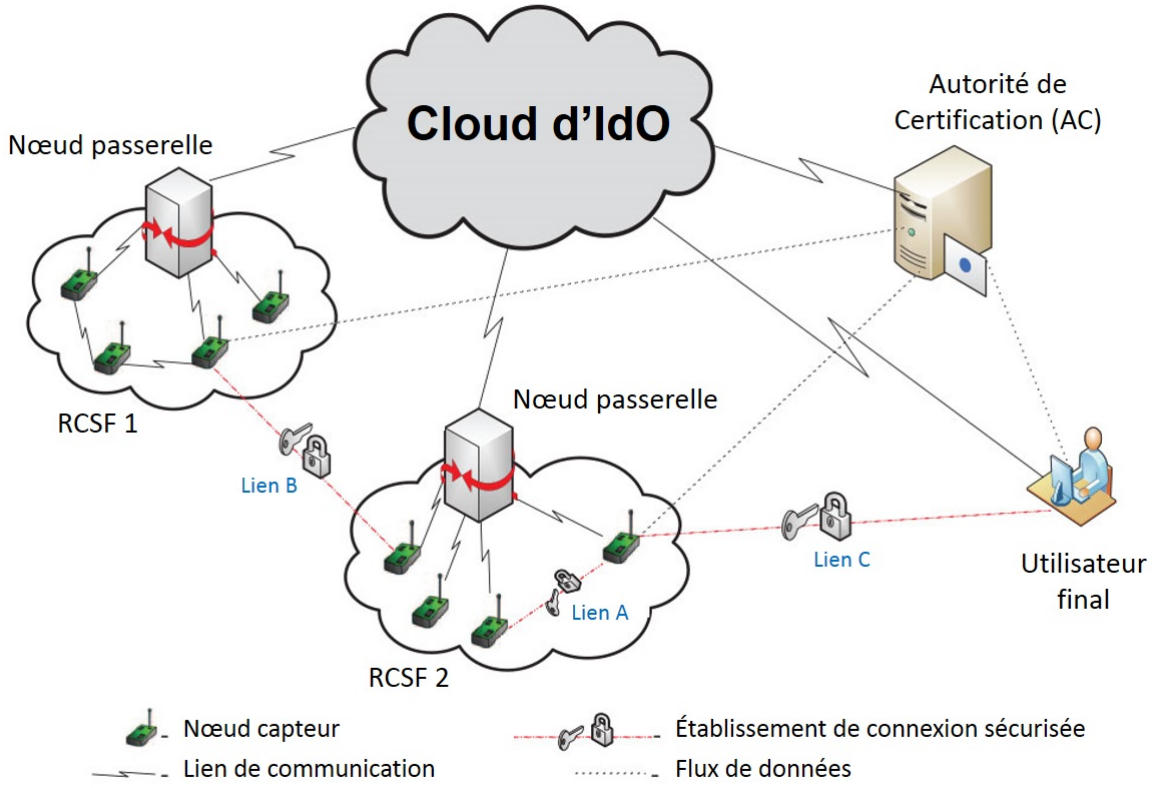

FIGURE 2.9 – Architecture physique du réseau [10].

cation :

1. Deux nœuds capteurs dans le même RCSF (lien A).
2. Deux nœuds capteurs provenant de RCSF distinctifs (lien B).
3. Un utilisateur final et un nœud de capteur (lien C).

Avant de démarrer le protocole d'authentification réel entre deux entités de réseau, il est nécessaire d'effectuer un processus d'enregistrement par chaque partie de la communication afin de récupérer les informations cryptographiques utilisées pour la phase d'authentification.

Solution d'authentification à deux phase :

Le schéma d'authentification proposé pour les applications RCSFs dans l'IdO distribué basée sur ECC (Elliptic Curve Cryptography), englobe deux phases. La première phase, appelée phase d'enregistrement, consiste à obtenir des informations d'identification de sécurité, La deuxième phase, appelée phase d'authentification, consiste à démarrer une communication mutuellement entre deux entités réseau, et La conception de la solution proposée est inspirée de la technique ECQV (Elliptic Curve Qu-Vanstone) pour générer des certificats "implicites" et s'appuie sur un mécanisme d'échange de clé ECDH (Elliptic Curve Diffie Hellman).

Discussion et critiques

Le schéma d'authentification proposé peut être facilement déployé dans les dispositifs à ressources limitées, avec une sécurité raisonnablement élevée. Les certificats sont de petite taille, consomment moins de mémoire pour chaque nœud capteur. Étant donné que le protocole est basé sur des opérations ECC standard, qui sont favorables à tous les nœuds capteurs indépendamment de leur fabricant et peuvent être effectués chez les utilisateurs finaux, il est faisable pour déployer dans les RCSFs. En outre, le schéma d'authentification proposé prend en charge l'addition du nouveau nœud (ou de l'utilisateur final) et la mobilité des nœuds et des utilisateurs finaux, de plus il assure l'intégrité des données. Cependant, la résilience contre les attaques de capture de nœuds n'a pas été abordée dans la conception de la solution proposée.

D) An Energy-efficient Security Node-based Key Management Protocol for WSN

Dans cet article [11], B.Jiana et E.Xu proposent un schéma de gestion de clé basé sur la sécurité des nœuds pour les clusters dans les RCSFs (SNKM). Ce schéma utilise plusieurs genres de clés. Ainsi, les nœuds peuvent choisir différentes clés pour le chiffrage et l'authentification selon les différents types de paquets de données. Ce protocole a été proposé afin d'améliorer le degré de sécurité des clusters Head et réduire l'énergie consommé pour l'établissement des clusters.

Selon les différents niveaux de sécurité des nœuds, les auteurs de SNKM adoptent différents schémas de sécurité et différents types de clés. Le cluster Head joue un rôle important dans les RCSFs, ainsi sa sécurité doit être assurée. Si un comportement anormal du cluster Head est détecté, il doit être remplacé immédiatement.

Les nœuds de sécurité : Selon une fonction aléatoire, le nœud calcule un nombre aléatoire. Si ce nombre est plus grand que T, ce nœud peut être un nœud de sécurité. Dès qu'un nœud détecte un comportement anormal de ses voisins, il envoie un rapport ua nœud de sécurité.

La clé par-paire : Avant le déploiement, les nœuds ne connaissent pas leurs voisins. Ainsi, lorsqu'un nouveau nœud joint le réseau, il essaye de découvrir ses voisins en diffusant un message avec son ID et se met en attente de réponse de ses voisins. Quand un voisin reçoit le message, il lui répond par un ACK et chiffre l'information avec la clé publique. Ensuite dès que le nouveau nœud reçoit l'ACK, il calcule alors la clé par-paire entre eux.

La clé du cluster : Cette clé est négociée par les nœuds de sécurité. Au début, chaque nœud de sécurité produit une clé aléatoire et l'envoie vers tous les autres nœuds de sécurité avec une estampille de temps. Les nœuds de sécurité comparent les estampilles de temps et prennent la clé aléatoire avec une estampille de temps minimale comme la nouvelle clé du cluster. Le cluster Head récupère cette clé puis la chiffre avec une la clé par-paire pour informer tous les autres nœuds.

La clé publique : Cette clé est utilisée pour chiffrer les informations de diffusion (Broadcast) et doit être mis à jour régulièrement. Le protocole utilise une fonction aléatoire unidirectionnelle pour générer la chaîne de clé d'authentification. En outre, dans ce protocole de gestion des clés, les informations de diffusion ne sont traitées que par les nœuds de sécurité et non par tous les nœuds. Dans certains cas, les nœuds peuvent recevoir les informations de diffusion et ne communiquent qu'avec le cluster Head et les nœuds de sécurité.

Discussion et critiques

L'analyse et la simulation de performance montrent que le protocole de gestion de clé proposé consomme moins d'énergie et que le délai de génération de clé est court. Le protocole peut fournir une plus grande sécurité d'authentification collaborative pour la clé. Il a une forte résilience contre la capture des nœuds, et peut supporter un réseau à grande échelle

2.4.4 Architecture hybride

Group Key Establishment for Enabling Secure Multicast Communication in Wireless Sensor Networks Deployed for IoT Applications

Multidiffusion groupe est un terme qui représente un groupe particulier de nœuds, qui sont intéressés ou qui ont le droit de recevoir un ensemble commun d'informations ou d'instructions. Le nombre de nœuds considérés dans le réseau de multidiffusion est n, qui comprend le nœud initiateur et (n-1) nœuds répondants. Une clé secrète commune, connue par l'initiateur et les répondeurs, est utilisée pour une communication sécurisée dans un groupe de multidiffusion, et La dérivation de la clé est née par l'initiateur et calculée en fonction des entrées fournies par les répondeurs.

Dans cet article [12] , P.Porambage et al développent deux protocoles d'établissement de clés de groupe pour les communications multicast sécurisées entre les périphériques à ressources limitées dans l'IdO. Les principales conditions de déploiement et les exigences de chaque protocole sont décrites en fonction des scénarios spécifiques de l'application d'IdO.

Le protocole 1 , l'initiateur injecte les messages de diffusion (c'est-à-dire sur l'ensemble du réseau) pour démarrer l'établissement de la clé, seuls les membres légitimes du groupe de multidiffusion peuvent continuer le reste Du processus de dérivation clé. Le Protocole 2 exploite les concepts ECIES pour établir une clé secrète partagée entre le groupe de multidiffusion.

Pour l'établissement de la clé, le nombre de transactions de messages entre l'initiateur et les nœuds répondants est quatre pour le protocole 1 et deux pour le protocole 2. En outre, le nombre d'opérations effectuées à chaque extrémité, le nombre de transactions de messages et les frais généraux sont Aussi moins dans le protocole 2 que celui du protocole 1. Ceci augmente l'efficacité et la performance du deuxième protocole proposé. Cependant, dans les deux protocoles, la clé de

groupe doit être rétablie après l'ajout d'un nouveau nœud ou la suppression d'un nœud existant. Dans les deux protocoles, afin de fournir une authentification de groupe et d'initiateur, la clé de groupe est dérivée avec la contribution des membres du groupe de multidiffusion (c'est-à-dire que la clé de groupe est dérivée en décalant les composants clés de chaque membre). Il s'agit d'une assurance implicite que tous les nœuds contribuent et autorisent la clé du groupe final. Cependant, dans le protocole 1, les membres du groupe fournissent une plus grande contribution à la dérivation de la clé avec un degré plus élevé, alors que dans le protocole 2, l'initiateur effectue la majorité des opérations.

Discussion et critiques

Cet article analyse deux mécanismes d'établissement de clés de groupe sécurisé pour la multidiffusion dans les RCSF dans le contexte des applications d'IdO. Selon les résultats d'évaluation, les consommations d'énergie de calcul et de communication des deux protocoles sont tolérables par les nœuds capteurs à ressources limitées. Les propriétés de scalabilité de ces protocoles garantissent le soutien de changements fréquents du groupe de multidiffusion. Le protocole 2 surpasse toujours le protocole 1 en termes de consommation d'énergie. Le protocole 1 est plus approprié pour les applications d'IdO distribuées, qui nécessitent que les membres du groupe contribuent fortement au calcul des clés et ont besoin d'un plus grand caractère aléatoire. Étant donné que le coût énergétique du côté répondeur est très faible, le protocole 2 est plus adapté aux applications centralisées d'IdO , où la plupart du temps les opérations cryptographiques sont effectuées par Une entité centrale et des nœuds de bord ont des profils énergétiques très faibles. Les deux protocoles proposés s'appliquent aux scénarios de communication un-à-plusieurs (1 : n) et ils devraient être étendus à des scénarios de communication de plusieurs à plusieurs (m : n) Obtenant des résultats quantitatifs complets pour les tests en temps réel.

2.5 Comparaison des approches étudiées

Nous comparons les travaux que nous avons analysés comme suit (Table 2.1) :

TABLE 2.1 – Illustre la comparaison entre les travaux présentés précédemment

Articles		**Résistance aux attaques**	**Scalabilité**	**Prise en compte de la contrainte d'énergie**
Architecture centralisée	**MR.Abdmeziem et D.Tandjaoui [5]**	Non	Oui	Oui
Architecture décentralisée	**I.Romdhani, MR.Abdmeziem et D.Tandjaoui [6]**	Non	Oui	Oui
	Y.Zhang et J.Pengfei [7]	Non	Oui	Oui
Architecture distribuée	**S.Sciancalepore et al [8]**	Oui	Moyenne	Oui
	P.Porambage et al [9]	Oui	Oui	Oui
	P.Porambage et al [10]	Non	Oui	Oui
	B.Jiana et E.Xu [11]	Oui	Oui	Oui
Architecture hybride	**P.Porambage et al [12]**	Non	Oui	Oui

2.6 Conclusion

Pour conclure ce chapitre, la gestion des clés est l'un des secteurs les plus importants dans la sécurité d'IdO, beaucoup de travaux ont été effectuées afin d'avoir un schéma performant qui assure un niveau élevé de sécurité et conserve l'énergie.

Dans ce chapitre, nous avons présenté un état de l'art sur la sécurité dans l'IdO. Nous avons étudié quelques protocoles de gestion de clés, et nous avons fait une classification des solutions analysées, qui nous seront utiles par la suite pour l'amélioration d'une solution spécifique ou la conception d'un système de gestion des clés.

Chapitre 3

Proposition et simulation

3.1 Introduction

La protection des communications est l'une des tâches principales dans l'IdO, nécessitant des mécanismes de sécurité efficaces. Dans la littérature, plusieurs solutions ont été proposées. Chacune d'elle présente un point faible qui limite sa performance. Dans ce chapitre, nous proposons une solution pour le problème de la sélection manuelle des tiers (TP_i) par les CNs.

3.2 Amélioration proposée

Dans cette section, nous proposons l'amélioration du système de gestion de clés proposé par Abdmeziem et Tandjaoui [6]. Une approche pour l'´établissement d'un canal de communication sécurisé entre un nœud à ressources limitées et un serveur distant. En effet, cette approche [6] ne permet pas aux nœuds contraints (CN) de sélectionner automatiquement les tiers. De plus, il y'a une seule clé pré-partagée entre chaque nœud contraint (CN) et le tiers (TP_i) à contacter.

Les notations suivantes sont utilisées dans l'approche proposée (Table 3.1) :

TABLE 3.1 – Les différentes notations [5].

Notation	**Description**
CN	Nœud contraint
UN	Nœud non contraint (Serveur distant)
TP_i	Tiers
CA	Autorité de certification
N_x	Nonce généré par le nœud X
$K_{x,y}$	Clé partagée entre X et Y
K_x	Clé publique du nœud X
$[data]_x$	Données (data) chiffrées avec la clé K

3.2.1 Motivations

L'IdO est un mélange de plusieurs technologies qui forment un réseau hétérogène. Les objets d'un tel environnement sont souvent confrontés à effectuer de différents services. De plus, ces objets possèdent des capacités spécifiques (statique ou mobile, alimenté par une batterie, ressources matérielles, capteurs, etc.) et à chacune d'elle possède des contraintes particulières (durée de vie, etc.) [4]. Ces aspects rendent le mécanisme de gestion des clés dans l'IdO plus complexe.

Afin de fournir plus de sécurité, notre approche consiste à sélectionner automatiquement les tiers, quand les nœuds CNs désirent partager le secret S avec le serveur distant, où chaque CN est pré chargé par une clé pré-partagée $K_{CN,Serveur}$ entre CN et le serveur de gestion des clés, ce dernier possède l'ensemble des identités des CNs. L'approche permet de : 1) garantir un gain de temps pour la phase d'initialisation grâce à la sélection automatique des tiers, 2) la désignation automatique des tiers ne nous oblige pas à enregistrer toutes les clés des tiers au niveau des nœuds contraint ,ce qui nous fait gagner un espace mémoire considérable, 3) notre approche nous permet de gagner en sécurité grâce à la possibilité de mise à jour des clés à chaque étape.

3.2.2 Hypothèses

Notre approche repose sur une architecture centralisée, qui s'appuie sur l'utilisation d'un serveur de gestion des clés non limité par ses capacités énormes de calcul et de stockage, appelé aussi le gestionnaire des clés.

Dans le cadre de notre travail, nous admettons que l'IdO assure le suivi des signes cliniques des patients par la mise en place des objets sur le corps humain. Ceci est une hypothèse dérivée par l'approche améliorée [5].

Nous considérons dans le modèle du réseau les composants suivants :

— **Nœud contraint (CN) :** à la capacité d'effectuer un chiffrement symétrique.

— **Les tiers (TP_i) :** à la capacité d'effectuer des opérations cryptographiques asymétriques.

— **Le serveur de gestion des clés :** suffisamment puissant pour effectuer des opérations de calcules (génération des clés, supporter le cryptage asymétrique et symétrique).

— **Station de base(SB) :** joue le rôle d'une passerelle entre les nœuds CNs et le serveur de gestion des clés. Elle n'a pas de contraintes sur les capacités de calcule, de stockage et d'énergie.

— **Le serveur distant (UN) :** suffisamment puissant pour supporter le cryptage asymétrique.

— **Autorité de certification (AC) :** une entité de confiance qui fournit des informations cryptographiques authentifiées aux tiers, au serveur distant et au serveur de gestion des clés.

Initialement chaque nœud possède la clé pré-partagée entre ce dernier et le serveur de gestion des clés.

3.2.3 Modèle physique

Nous proposons une approche du modèle de gestion des clés [5], basé sur la cryptographie symétrique et asymétrique pour sécuriser les communications. Notre objectif principal est de sélectionner les tiers automatiquement, ce qui nous permettra la mise à jour des clés à chaque tour, minimiser le nombre de messages envoyés .Gain d'espace mémoire.

Nous considérons deux scénarios :

- Dans le premier scénario les CNs et les tiers peuvent contacter directement le gestionnaire des clés (CNs, TP_is et le serveur dans la même zone) (voir la figure 3.1).

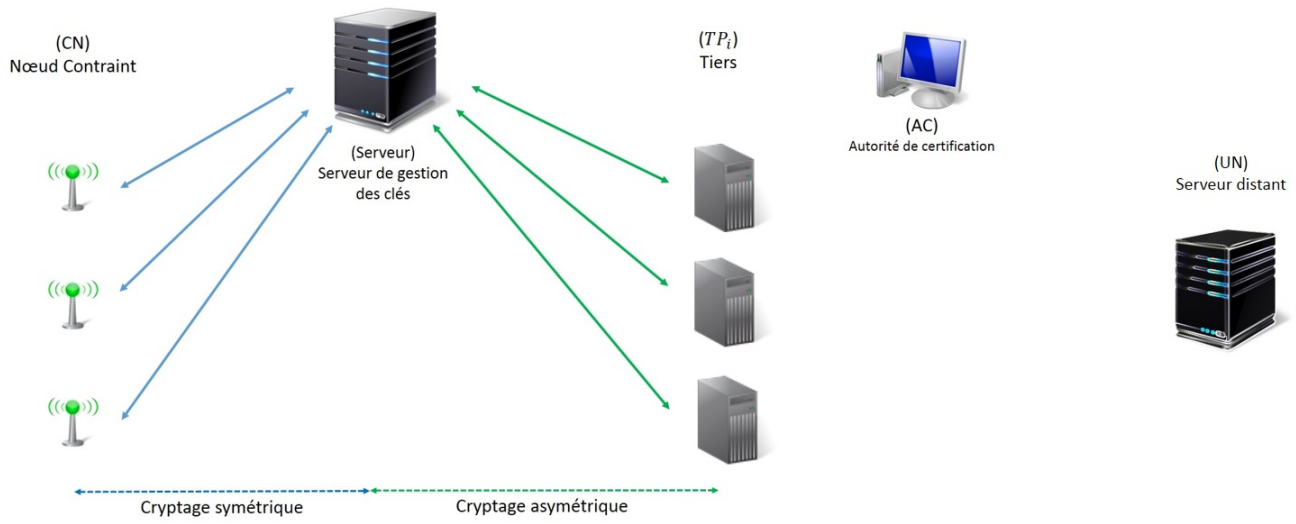

FIGURE 3.1 – Scenario 1

- Dans le deuxième scénario ils existent des CNs et des TP_is qui ont une distance suffisamment grande (sont éloignés géographiquement), ce qui nous a amené à installer des stations de bases qui jouent le rôle d'une passerelle entre les CNs, TP_is et le gestionnaire des clés (voir la figure 3.2).

3.2.4 Fonctionnement

— Les nœuds CNs initient l'échange en contactant le serveur de gestion des clés afin de partager le secret S avec le serveur distant (UN).

— - Le serveur de gestion des clés sélectionne un tiers (TP_i) à contacter pour chaque nœud contraint (CN).

— Le serveur génère des clés pré-partagées pour chaque couple (CN, TP_i).

— Le serveur de gestion des clés envoie à chaque CN la clé et l'identifiant du TP_i à contacter.

— Le serveur envoie les clés et les identifiants des CNs pour les TP_is.

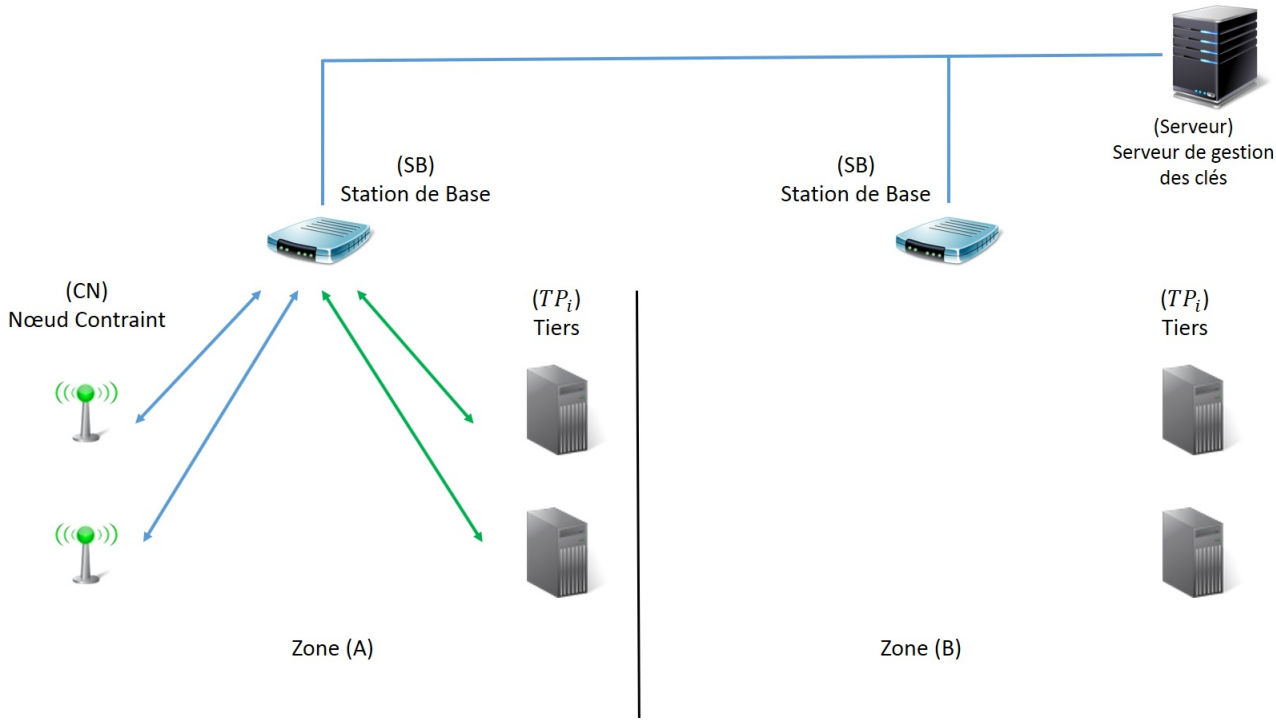

FIGURE 3.2 – Scenario 2

Le reste du fonctionnement est similaire à ce qui a été proposé dans l'article [5].
La figure 3.3 illustre les différentes phases de notre approches.

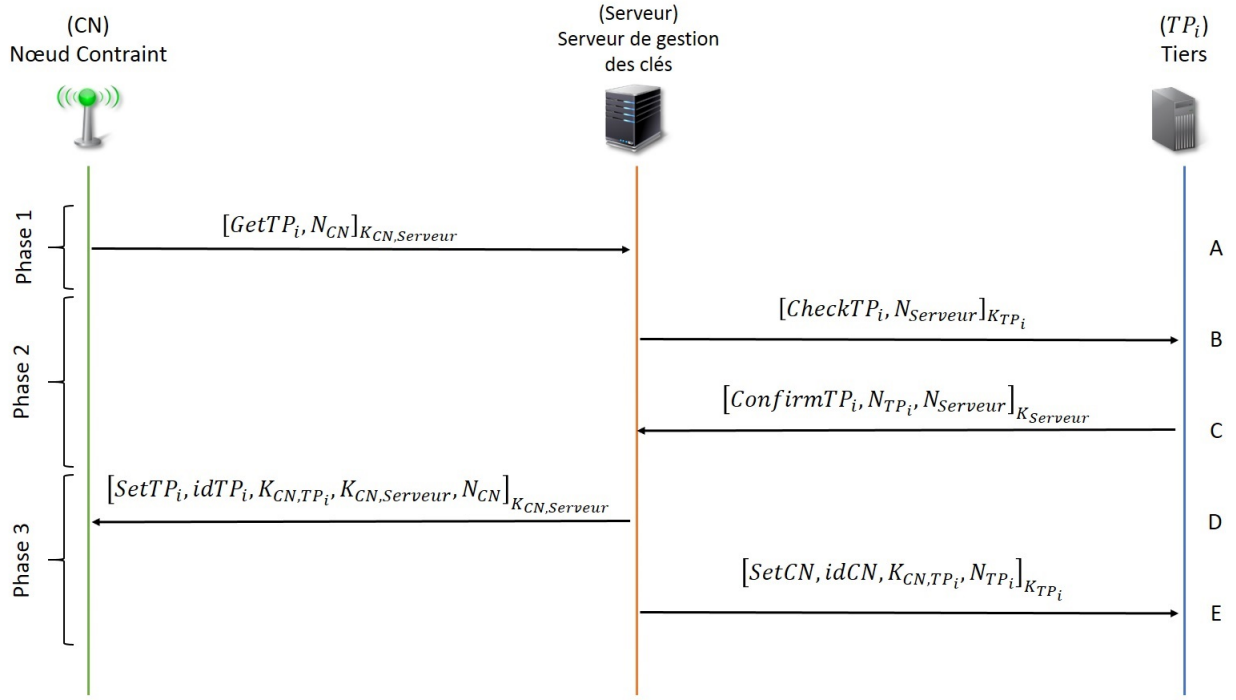

FIGURE 3.3 – Illustrations des différents messages échangés.

Différentes phases et échanges des messages :

Phase 1 : Chaque nœud CN initie l'échange en envoyant un message $GetTP_i$. Le but est d'avoir le TP_i à contacter et la clé de communication K_{CN,TP_i}. Le message (message A) comprend un code d'authentification (MAC) ,il est crypté à l'aide de $K_{CN,Serveur}$. Les Nonces sont inclus dans les messages échangés pour empêcher les attaques de répétition.

Phase 2 : Lors de la réception du (message A) le gestionnaire des clés envoie le message $CheckTP_i$ (message B) aux TP_is qui sont dans la même zone géographique des CNs. Les TP_is expriment leur volonté de faire partie du protocle par l'envoie du message $ConfirmTP_i$ (message C) au serveur de gestion des clés.

Phase 3 : Le serveur de gestion des clés vérifie les identifiants des TP_is. En suite, il génère des clés pré-partagées K_{CN,TP_i} pour chaque couple (CN, TP_i), puis envoie par la suite deux messages, un message $SetTP_i$ aux nœuds CNs (message D qui comprend un code d'authentification (MAC) et qui est crypté à l'aide de $K_{CN,Serveur}$) et un autre message $SetCN$ aux tiers TP_is (message E).

3.2.5 Analyse de sécurité

Confidentialité Les données échangées entre les différents composants impliqués dans notre approche sont gardées confidentielles. Le cryptage symétrique est utilisé en fonction des clés pré-partagées (entre chaque nœud CN et le gestionnaire des clés, une seule clé pré-partagée). Le cryptage asymétrique est utilisé pour sécuriser la communication entre les tiers et le gestionnaire des clés.

Intégrité et authentification les données échangées sont authentifiées, grâce à l'utilisation du MAC dans le cryptage symétrique, et la signature numérique dans le cryptage asymétrique. L'objectif est de s'assurer que les données n'ont pas été modifiées. Le programme garantit également que les TP_is impliqués peuvent prouver leur authenticité au serveur distant (UN) ou au serveur de gestion des clés.

Scalabilité le modèle de réseau permet d'intégrer de nouveaux objets. Le nouvel objet est pré-chargé par une clé pré-partagée $K_{CN,Serveur}$ entre lui et le serveur de gestion des clés, puis on ajoute son identité à l'ensemble des identités des CNs dans le serveur de gestion des clés. Aucune opération n'est requise concernant le TP_i, serveur de gestion des clés, ou les serveurs distants qui seront impliqués dans le protocole.

Résistance pour compromettre le secret S échangé, un attaquant doit avoir toutes les clés d'échanges K_{CN,TP_i}, car le secret S est divisé en plusieurs parties distribués par les CNs au TP_is.

D'où il est difficile de collecter les clés K_{CN,TP_i} car elles changent à chaque fois que les CNs désirent partager le secret S.

Overhead les nœuds contraints ne sont impliqués que dans des primitives de cryptage symétrique, beaucoup moins consommatrices de ressources que celles qui sont asymétriques. Toutes les opérations asymétriques sont déchargées aux tiers qui sont beaucoup plus puissants. Limiter les demandes de calcul pour les nœuds contraints diminue leur consommation d'énergie et augmente ainsi leur durée de vie.

3.3 Simulation

La simulation informatique, ou simulation numérique, est une série de calculs effectués sur un ordinateur et reproduisant un phénomène physique. Elle aboutit à la description du résultat de ce phénomène, comme s'il s'était réellement déroulé. Cette représentation peut être une série de données, une image ou même un film vidéo [26].

3.3.1 Paramètres de simulation

Pour simuler notre approche, nous avons utilisé Java qui est un langage de haut niveau, la simulation (Figure 3.4) est réalisée avec 8 nœuds contraints, dispersés aléatoirement, entre chaque deux nœuds contraints une distance qui varie entre un intervalle de 0.2m et 1m, et la simulation se fait dans un environnement en trois dimensions (3D).

Les nœuds sont fixés et les 8 serveurs de gestion des clés n'ont pas la même distance par rapport aux nœuds (des emplacement différents), afin d'avoir de différents scénarios (le premier serveur a une distance de 2 m par rapport aux nœuds et pour le reste des serveurs on incrémente à chaque fois de 1m).

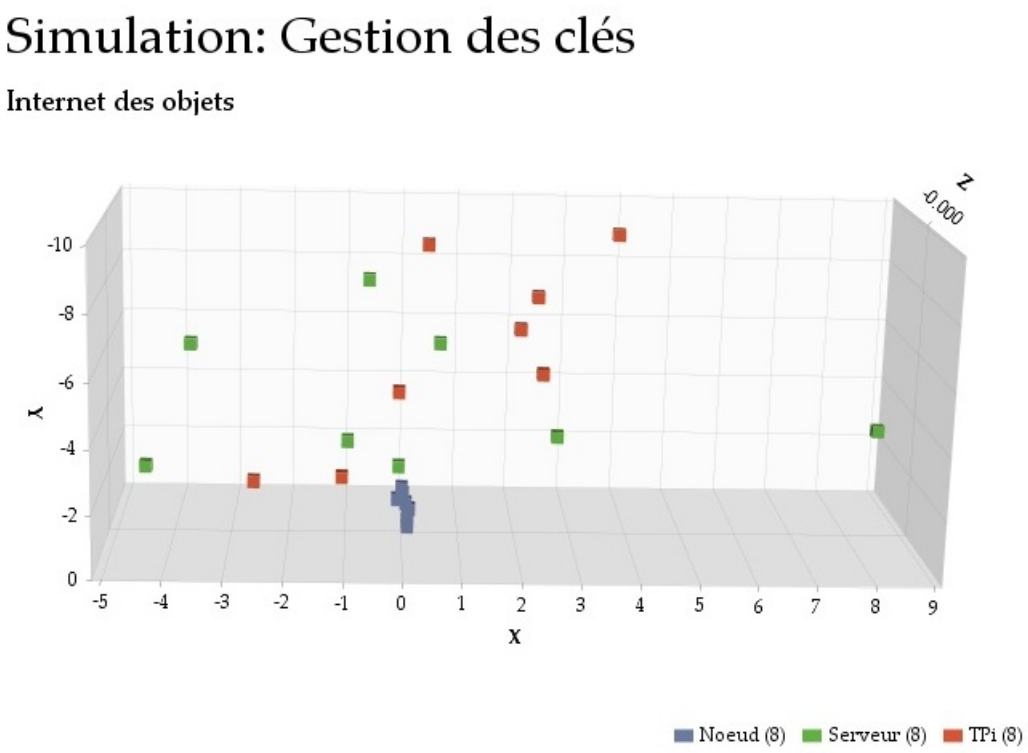

FIGURE 3.4 – Illustration des différents emplacements des entités impliquées dans notre approche.

3.3.2 Analyse des performances

Le protocole amélioré fournit plus de sécurité, comme cela est analysé dans la section 3.2.5. Dans ce qui suit nous nous intéressons à évaluer l'impact énergétique engendré par les messages supplémentaires qui nous permis d'automatiser la sélection des tiers (TP_is)

Consommation d'énergie

La ressource énergétique doit être prise en compte dans n'importe quelle application de l'IdO, on a mesuré et comparé la consommation d'énergie de notre approche avec le protocole proposé par Abdmeziem et Tandjaouiont [5].

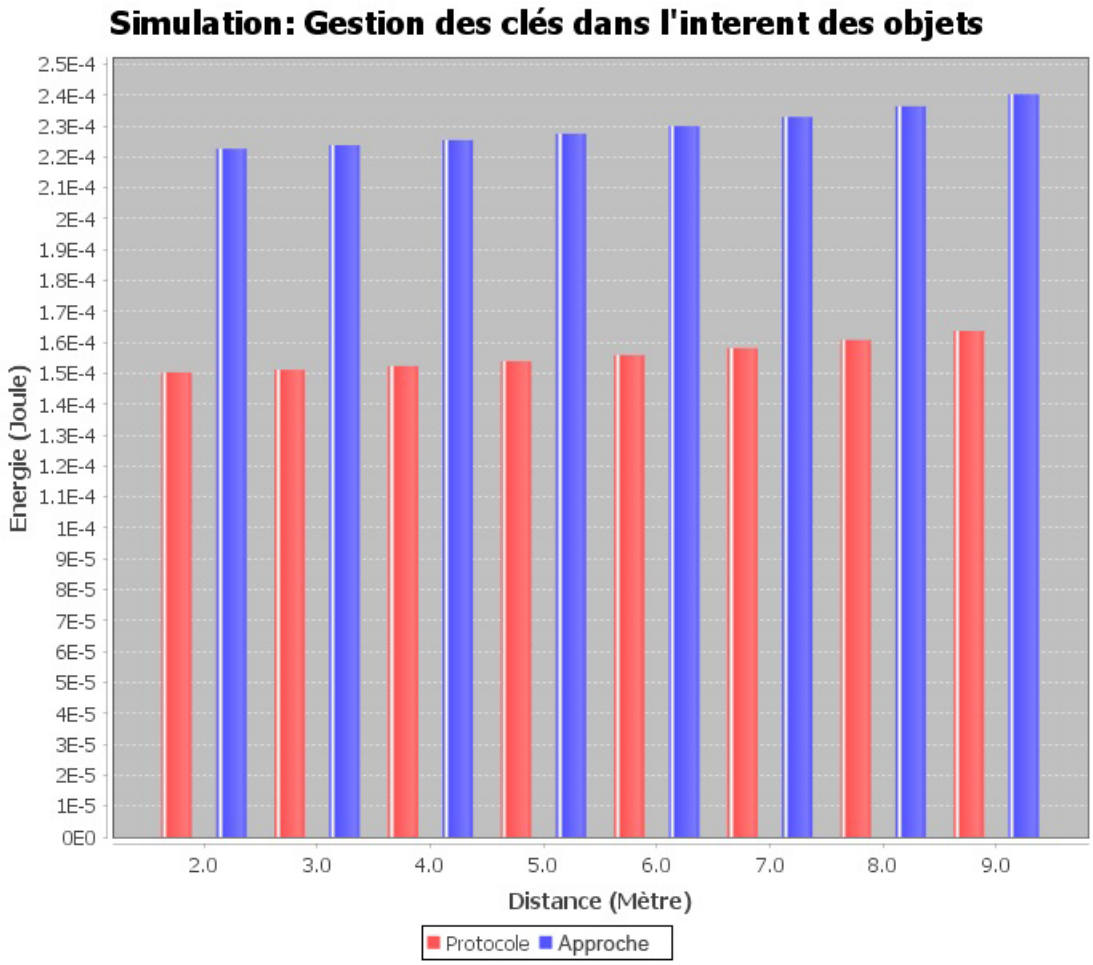

FIGURE 3.5 – L'histogramme représente la consommation énergétique pour les deux protocoles.

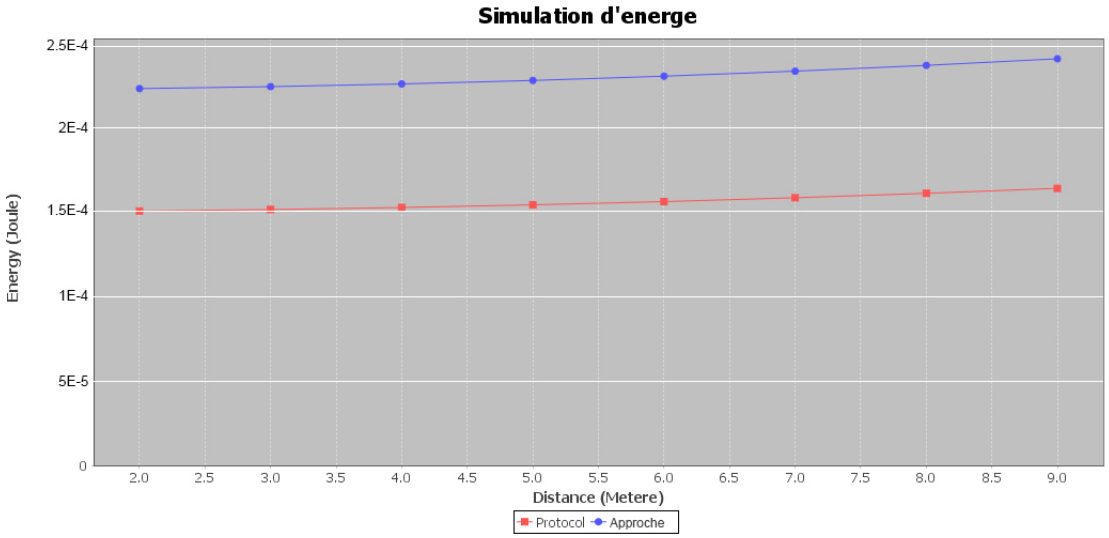

FIGURE 3.6 – La consommation énergétique pour les deux protocoles.

Heinzelman et al [27] proposent un modèle radio de consommation d'énergie. Ainsi, les énergies nécessaires pour émettre $E_{Tx}(k,d)$ et recevoir $E_{Rx(k)}$ des messages sont données par :

— Pour émettre un message de k bits vers un récepteur loin de d mètres, l'émetteur consomme :
$E_{Tx}(k,d) = (E_{elec} \times k) + (E_{amp} \times k \times d^2)$

— Pour recevoir un message de k bits, le récepteur consomme :
$E_{Rx}(k) = E_{elec} \times k$

E_{elec} et E_{amp} représentent respectivement l'énergie de transmission électronique (de valeur 50 nJ/bit) et d'amplification (de valeur 100 pJ/bit/m^2).

Note : 1 J = 10^{-9} nJ = 10^{-12} pJ.

Les figures 3.5 et 3.6 montrent l'énergie consommée par un nœud du réseau des deux protocoles (le protocole [5] et le notre approche) en fonction de la distance. La simulation illustre que la consommation d'énergie de notre approche augmente presque d'un tiers par rapport à celle du [5], ceci est dû aux messages engendrés par la sélection automatique des tiers. Cela nous a permis d'avoir un protocole facile à utiliser et résistant aux attaques grâce aux changements de clés qui représente un facteur important dans la sécurité. En plus, dans un réseau sur un corps humain, la contrainte de consommation d'énergie est moins importante par rapport aux autres domaines d'applications, car les objets sont physiquement accessibles donc facilement rechargeables.

3.4 Conclusion

Sans une forte sécurité l'utilisation d'IdO dans n'importe quel domaine d'application aurait des conséquences indésirables. Établir une communication sécurisée implique l'établissement et la distribution des clés pour crypter et authentifier les messages. La gestion des clés dans l'IdO est le problème le plus délicat de la cryptographie.

Notre approche est une amélioration du protocole présenté dans l'article [5]. L'apport principal est l'automatisation de la sélection des tiers (TP_is), ce qui a permis un gain d'espace mémoire, car notre approche a réduit le nombre d'identifiants et de clés des TP_is à enregistrer.

Notre approche a aussi renforcé la sécurité en permettant une mise à jour des clés à chaque étape et équilibrant la charge sur les différents TP_is.

En contre partie notre approche, consomme jusqu'à 30% de plus d'énergie par rapport au protocole initial.

Conclusion générale

Le travail consigné dans ce mémoire a été le fruit d'une étude menée dans le contexte d'internet des objets en particulier et ce, relativement au problème de sécurité. Nous avons mis en avant les caractéristiques essentielles et les notions fondamentales d'internet des objets, et nous avons étudié aussi les notions de sécurité.

L'ensemble des protocoles de gestion des clés proposés pour l'IdO se basent généralement sur la cryptographie symétrique et asymétrique pour sécuriser les communications. Nous avons étudié un ensemble de ces protocoles de gestion des clés qui permettent d'offrir des services de sécurités pour n'importe quel système basé sur la communication et nous avons mis une classification aux solutions étudiées.

De cette étude, résulte notre contribution consistant en une amélioration d'un modèle de gestion des clés hybride qui permet la sélection automatique des tiers. Cette amélioration nous permet de faciliter l'utilisation du modèle proposé, d'avoir un gain d'espace mémoire, et un changement de clé à chaque fois que les nœuds contraints désirent partager le secret S avec le serveur distant, ceci rend le protocole résistant aux attaques (donc fournir plus de sécurité). Cependant la consommation d'énergie augmente de 30% par rapport au protocole initial, mais en revanche on gagne en terme de sécurité.

Concevoir un protocole efficace de gestion des clés demeure encore un domaine de recherche plus ouvert, comme perspective de notre travail est d'utiliser des méthodes de démonstration formel pour vérifier les caractéristiques de la sécurité et étudier d'autre métriques comme la durée de vie.

Bibliographie

[1] `https://nicholaskellettdotcom.files.wordpress.com/2016/04/1600px-internet_of_things-e1461017506926.jpg`, (consulté le 12 Janvier 2017).

[2] D.Evans. *L'Internet des objets. Comment l'évolution actuelle d'Internet transforme-t-elle le monde ?.* Livre Blanc, Cisco IBSG, États-Unis, Avril 2011.

[3] `http://blog.octo.com/modeles-architectures-internet-des-objets/`, (consulté le 12 Janvier 2017).

[4] N.Merrani et N.Khimoum. *Simulation et évaluation de protocoles de gestion de clés dans les réseaux de capteur.* Mémoire d'ingénieur d'état en informatique de l'université Abderrahmane Mira Bejaia, 2009.

[5] MR.Abdmeziem et D.Tandjaoui. An end-to-end secure key management protocol for e-health applications. *Computers & Electrical Engineering*, 44 :184–197, 2015.

[6] MR.Abdmeziem, D.Tandjaoui et I.Romdhani. A decentralized batch-based group key management protocol for mobile internet of things (dbgk). In *Computer and Information Technology ; Ubiquitous Computing and Communications ; Dependable, Autonomic and Secure Computing ; Pervasive Intelligence and Computing (CIT/IUCC/DASC/PICOM), 2015 IEEE International Conference on*, pages 1109–1117. IEEE, 2015.

[7] Y.Zhang et J.Pengfei. An efficient and hybrid key management for heterogeneous wireless sensor networks. In *Control and Decision Conference (2014 CCDC), The 26th Chinese*, pages 1881–1885. IEEE, 2014.

[8] S.Sciancalepore et al. Key management protocol with implicit certificates for iot systems. In *Proceedings of the 2015 Workshop on IoT Challenges in Mobile and Industrial Systems*, IoT-Sys '15, pages 37–42, New York, NY, USA, 2015. ACM.

[9] P.Porambage et al. Proxy-based end-to-end key establishment protocol for the internet of things. In *Communication Workshop (ICCW), 2015 IEEE International Conference on*, pages 2677–2682. IEEE, 2015.

[10] P.Porambage et al. Two-phase authentication protocol for wireless sensor networks in distributed iot applications. In *Wireless Communications and Networking Conference (WCNC), 2014 IEEE*, pages 2728–2733. IEEE, 2014.

[11] B.Jiana et E.Xu. An energy-efficient security node-based key management protocol for wsn. In *Applied Mechanics and Materials*, volume 347, pages 2117–2121. Trans Tech Publ, 2013.

[12] P.Porambage et al. Group key establishment for enabling secure multicast communication in wireless sensor networks deployed for iot applications. *IEEE Access*, 3 :1503–1511, 2015.

[13] D.Christin, A.Reinhardt, P.Mogre et R.Steinmetz. Wireless sensor networks and the internet of things : Selected challenges. *Proceedings of the 8th GI/ITG KuVS Fachgespräch Drahtlose sensornetze*, pages 31–34, 2009.

[14] `http://www.futura-sciences.com/tech/definitions/internet-internet-objets-15158/`, (consulté le 12 Janvier 2017).

[15] `http://www.objetconnecte.com/quelle-difference-m2m-iot/`, (consulté le 12 Janvier 2017).

[16] Y.Ait Mouhoub et F.Bouchebbah. *Proposition d'un modèle de confiance pour l'Internet des objets.* Mémoire master de l'université Abderrahmane Mira Bejaia, 21 Juin 2015.

[17] B.Billet. *Système de gestion de flux pour l'Internet des objets intelligents.* Thèse de doctorat de l'université de Versailles Saint-Quentin-En-Yvelines, 2015.

[18] C.Llorens, L.Levier, D.Valois et B.Morin. *Tableaux de bord de la sécurité réseau.* Editions Eyrolles, 2011.

[19] S.Atmani. *Protocole de sécurité Pour les Réseaux de capteurs Sans Fil.* PhD thesis, Université de Batna 2, Juillet 2010.

[20] Y.Challal. Réseaux de capteurs sans fil. *University of Technology in compiegne, France*, 2008.

[21] R.Roman, C.Alcaraz, J.Lopez et N.Sklavos. Key management systems for sensor networks in the context of the internet of things. *Computers & Electrical Engineering*, 37(2) :147–159, 2011.

[22] T.Schlossnagle. *Scalable internet architectures.* Kindle Edition, 2007.

[23] B.Daghighi, M.Kiah, S.Shamshirband et M.Rehman. Toward secure group communication in wireless mobile environments : Issues, solutions, and challenges. *Journal of Network and Computer Applications*, 50 :1–14, 2015.

[24] E.Cario. *Systèmes distribués.* Cours de licence informatique de l'université de Pau et des Pays de l'Adour UFR Sciences Pau, 2008.

[25] S.Reed et G.Solomon. Polynomial codes over certain finite fields. *Journal of the society for industrial and applied mathematics*, 8(2) :300–304, 1960.

[26] Y.Makhloufi et Z.Rezgui. *Etude des mécanismes de gestion des clés dans les réseaux de capteur sans fil proposition d'un protocole Hybride basé sur la stéganographie.* Mémoire master de l'université Abderrahmane Mira Bejaia, 2013.

[27] W.Heinzelman, A.Chandrakasan et H.Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. *System sciences, Proceedings of the 33rd annual Hawaii international conference*, pages 10–pp, 2000.

Résumé

L'Internet des objets (IdO) ouvre la voie vers une multitude de scénarios basés sur l'interconnexion entre le monde physique et le monde virtuel : domotique, santé, ville intelligente, sécurité, etc. Une des contraintes principales dans l'IdO est la protection des communications, pour cela, l'IdO nécessite des mécanismes de sécurité efficaces. La plupart des protocoles de gestion des clés proposés dans la littérature se basent sur des mécanismes de chiffrement symétrique et asymétrique. Dans le présent travail, nous avons étudié dans un premier temps, quelques différents protocoles de gestion des clés existants et nous avons amélioré l'un de ces protocoles afin d'automatiser sa gestion des clés.

Mots clés : gestion des clés, internet des objets, chiffrement, sécurité.

Abstract

The Internet of Things (IoT) opens the way to a multitude of scenarios based on the interconnection between the physical world and the virtual world : smart house, health, smart city, security, etc. One of the main constraints in the IoT is the protection of communications. For this, the IoT requires effective security mechanisms. Most key management protocols proposed in the literature are based on symmetric and asymmetric encryption mechanisms. In the papers, we first studied some existing key management protocols and we improved one of these protocols in order to automate its key management.

Keywords : key management, internet of things, encryption, security.