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 d'Informatique

MÉMOIRE DE MASTER RECHERCHE

En Informatique

Option
Réseaux et Systèmes Distribués

Thème

Authentificaiton dans l'IoT

Présenté par :

ACHOUR Raouf
MAKHLOUFI Naima

Soutenu le 02 Juillet 2017 devant le jury composé de :

Président	**Pr. TARI Abdelmalek**	**U. A/Mira Béjaia.**
Examinateur	**M. SKLAB Youcef**	**U. A/Mira Béjaia.**
Examinateur	**Mme. BENNAI Sofia**	**U. A/Mira Béjaia.**
Encadreur	**Pr. BOUKERRAM Abdellah**	**U. A/Mira Béjaia.**
Co Encadreur	**M. ELSAKAAN Nadim**	**U. A/Mira Béjaia.**

Béjaia, Juillet 2017.

“ Remerciements ”

Louange à dieu, le miséricordieux, sans lui rien de tout cela n’aurait pu être.

Nous exprimons notre vive reconnaissance à notre Encadreur le

Pr. BOUKERRAM Abdellah.

Notre reconnaissance va également à notre Co-Encadreur M. ELSAKAAN Nadim pour ses précieux conseils, ses orientations, sa disponibilité, sa sympathie et le temps qu’il nous a patiemment consacré malgré ses différentes responsabilités.

Nous remercions chacun des membres du jury pour l’intérêt porté à notre travail en acceptant de l’examiner et de l’enrichir avec leurs propositions.

Enfin, nous adressons nos plus sincères remerciements à tous nos proches et amis qui nous ont soutenus et encouragés tout au long de notre cursus universitaire et de la réalisation de ce mémoire.

Merci à toutes et à tous.

Naima, Raouf.

* Dédicaces *

À mes parents qui ont partagé avec moi mes échecs et mes succès.

À mes frères Mohamed Rafik et Hani.

À la très chère Sofia-Lydia.

À mes amis Mehdi, Nadim, Abdeslam et Halim pour leur soutien et leur aide.

À ma binôme Naïma.

À la Team Phoenix pour son soutien et ses encouragements.

À tous ceux qui compte pour moi et ceux pour lesquels je compte.

Raouf A.

* Dédicaces *

Je dédie ce mémoire À

Mon père, qui peut être fier de trouver ici le résultat de longues années de sacrifices et de privations pour m'aider à avancer dans la vie. Puisse Dieu faire en sorte que ce travail porte son fruit. Merci pour les valeurs nobles, l'éducation et le soutien permanent venu de toi.

Ma mère, qui a œuvré pour ma réussite, de par son amour, son soutien, tous les sacrifices consentis et ses précieux conseils, pour toute son assistance et sa présence dans ma vie, reçoit à travers ce travail aussi modeste soit-il, l'expression de mes sentiments et de mon éternelle gratitude.

Mes sœurs pour avoir contribué à la réussite de ce travail d'une manière indirecte d'y avoir apporté tant d'humeur et d'amour et pour tout le soutien moral prodigué dans les moments les plus difficiles.

Mes cousins et cousines, tentes et oncles.

Tous mes proches grands et petits.

Mes ami(e)s avec lesquels j'ai partagé mes moments de jolies et de bonheurs.

Mon binôme Raouf.

Tous mes enseignants.

Naima M.

Table des matières

Table des figures

Liste des tableaux

Liste des abréviations

AMQP	**A**dvanced **M**essage **Q**ueuing **P**rotocol
ARPANET	**A**dvanced **R**esearch **P**rojects **A**gency **N**etwork
CoAP	**C**onstrained **A**pplication **P**rotocol
DNS	**D**omain **N**ame **S**ystem
DNSSEC	**D**omain **N**ame **S**ystem **S**ecurity **E**xtensions
DoS	**D**enial **O**f **S**ervice
ECC	**E**lliptic **C**urve **C**ryptography
EPC	**E**lectronic **P**roduct **C**ode
EPCIS	**E**lectronic **P**roduct **C**ode **I**nformation **S**ervices
ETSI	**E**uropean **T**elecommunications **S**tandards **I**nstitute
GPS	**G**lobal **P**ositioning **S**ystem
HTTP	**H**yperText **T**ransfer **P**rotocol
HTTPS	**H**yperText **T**ransfer **P**rotocol **S**ecure
HVAC	**H**eating, **V**entilation and **A**ir-**c**onditioning
IEEE	**I**nstitute of **E**lectrical and **E**lectronics **E**ngineers
IETF	**I**nternet **E**ngineering **T**ask **F**orce
IdO	**I**nternet **d**es **O**bjets
Ipv6	**I**nternet **P**rotocol **v**ersion 6
IoT	**I**nternet **o**f **T**hings
ITU	**I**nternational **T**elecommunication **U**nion
LEACH	**L**ow-**E**nergy **A**daptive **C**lustering **H**ierarchy
LLN	**L**ow power **L**ossy **N**etworks
LS	**S**pecialty **l**ine

MQTT	**M**essage **Q**ueue **T**elemetry **T**ransport
M2M	**M**achine **T**o **M**achine
ONS	**O**bject **N**aming **S**ervice
PIR	**P**rivate **I**nformation **R**etrieval
P2P	**P**eer-**t**o-**p**eer
RBAC	**R**ole-**B**ased **A**ccess **c**ontrol
REST	**R**epresentational **S**tate **T**ransfer
RFID	**R**adio **F**requency **I**dentification
RPL	**R**outing **P**rotocol for **L**ow power and **L**ossy Networks
RSA	**R**ivest, Adi **S**hamir et Leonard **A**dleman
SOAP	**S**imple **O**bject **A**ccess **P**rotocol
TCP/IP	**T**ransmission **C**ontrol **P**rotocol / **I**nternet **P**rotocol
TLS	**T**ransport **L**ayer **S**ecurity
UDP	**U**ser **D**atagram **P**rotocol
URI	**U**niform **R**esource **I**dentifier
VPN	**v**irtual **p**rivate **n**etwork
WM-RSA	**W**ireless sensor **M**ote **R**ivest, Adi **S**hamir et Leonard **A**dleman
WSN	**W**ireless **s**ensor **n**etworks
W3C	**W**orld **W**ide **W**eb **C**onsortium
XMPP	**E**xtensible **M**essaging and **P**resence **P**rotocol
6loWPan	**IPv6** **L**oW **P**ower **w**ireless **A**rea **N**etworks

Introduction générale

L'Internet a subi de graves changements depuis son premier lancement à la fin des années 1960 en tant que résultat de l'ARPANET. Un réseau initial à quatre nœuds s'est rapidement transformé en un réseau fortement interconnecté et auto-organisé qui construit la base quotidienne pour les entreprises, la recherche et l'économie. Au cours des années 1990, un certain nombre de termes émergeaient pour saisir de nouvelles formes d'interactions personnelles et commerciales. La prochaine révolution sera l'interconnexion entre les objets pour créer un environnement intelligent nommé Internet des Objets (IdO) ou IoT pour Internet of Things en anglais. L'IoT est considéré comme la troisième vague de technologies de l'information juste après Internet et les réseaux de communication mobiles. Il est apparu comme l'un des paradigmes de communication les plus puissants du XXIe siècle.

L'IoT peut être résumée dans une phrase "Un réseau mondial d'entités interconnectées hétérogènes (RFID, capteurs, actionneurs, smartphones, montres, etc.) doté de capacités auto configurables basées sur des protocoles de communication standard et interopérables". Dans la plupart des cas, ces entités hétérogènes, les "Objets", ont une contrepartie localisable, adressable et lisible sur Internet et sont en mesure d'interagir entre eux et de coopérer avec leurs voisins pour atteindre des objectifs communs. Dans ce réseau, les objets physiques et virtuels ont des identités, des attributs physiques, des personnalités virtuelles et des interfaces intelligentes qui sont intégrés au réseau d'une façon transparente. Actuellement, il y a 9 milliards d'appareils interconnectés et devrait atteindre 24 milliards d'appareils d'ici 2020.

L'IoT promet d'être l'une des tendances majeures de notre ère numérique. Ces dernières années, une abondance de solutions a émergé pour interconnecter des Objets intelligents pour des systèmes avec des objectifs différents. IoT est alimenté par les progrès récents d'une variété de dispositifs et de technologies de communication. Il a pour but de fournir une infrastructure informatique facilitant les échanges d'Objets d'une manière sûre et fiable.

Un défi majeur consiste à permettre l'établissement d'une clé cryptographique partagée de façon sûre et légère, entre les Objets ou les nœuds de capteurs et l'utilisateur en dehors du réseau. L'établissement d'une clé est l'aspect le plus difficile de la sécurité cryptographique, c'est le processus par lequel des clés sont produites, enregistrées, protégées, transférées, chargées, employées, et détruites. Actuellement aucune entreprise ne peut se passer d'outils informatiques, et très souvent un réseau informatique de taille plus ou moins importante est mis en œuvre. Le nombre de machines dans ces réseaux peut parfois devenir extrêmement élevé, la maintenance ainsi que la gestion de ces parcs informatiques deviennent alors des enjeux cruciaux, d'autant plus qu'une panne du réseau peut parfois avoir des conséquences catastrophiques. À travers ce mémoire, nous proposons une solution de relai en cas de panne du serveur, basé sur le choix d'un algorithme d'élection et d'un protocole d'authentification léger et adapté aux ressources de l'IoT.

Ce mémoire est organisé comme suit. Dans le premier chapitre, nous présenterons les concepts généraux, la sécurité et les services relatifs au domaine de l'IoT. Dans le deuxième, quelques solutions récemment proposées sur l'axe de l'authentification et de l'établissement de clé dans l'IoT et quelques algorithmes d'élection où nous ferons une étude bibliographique sur les protocoles de gestion de clé dans l'IoT, et sur les algorithmes d'élection. Dans le chapitre 3 nous présenterons notre proposition. Le quatrième chapitre sera consacré à la simulation de la proposition.

Chapitre 1

Concepts fondamentaux de l'IoT

1.1 Introduction

L'IoT caractérise des Objets physiques connectés ayant leur propre identité numérique et capable de communiquer les uns avec les autres. Ce réseau crée en quelque sorte une passerelle entre le monde physique et le monde virtuel. Dans ce chapitre, nous présentons les différents concepts de base qui serviront à clarifier les termes utilisés tout au long du document. Nous présentons d'abord quelques concepts fondamentaux de l'IoT, son fonctionnement, ses domaines d'application, quelques vulnérabilités et menaces relatives à son déploiement, ainsi que quelques services de sécurité tels que la confidentialité, l'authentification et l'identification.

1.2 L'internet des Objets

1.2.1 Définition

De nos jours, environ deux milliards de personnes partout dans le monde utilisent Internet pour naviguer sur le Web, envoyant et recevant des courriels, accédant à des contenus et services multimédias, jouant à des jeux, utilisant des applications de réseaux sociaux et bien d'autres tâches. Alors que de plus en plus de personnes auront accès à une telle infrastructure mondiale d'information et de communication, un autre grand bond en avant vient, lié à l'utilisation d'internet comme plate-forme mondiale pour permettre aux machines et aux Objets intelligents de communiquer,

de dialoguer, calculer et coordonner [36]. Dans l'IoT, tout Objet est potentiellement connecté à Internet est capable de communiquer avec d'autres Objets. Ceci engendre de nouveaux risques liés notamment à la confidentialité, l'authenticité et l'intégrité des données échangées entre les Objets [7].

1.2.2 Cycle de vie d'un Objet connecté dans l'IoT

La vie d'un Objet commence quand il est fabriqué. En raison des différentes zones d'application (HVAC, éclairage, sécurité, etc.), les nœuds sont adaptés à une tâche spécifique. Il est donc peu probable qu'un seul constructeur fabrique tous les nœuds pour la même structure. Par conséquent, l'interopérabilité ainsi que l'amorçage de confiance entre les nœuds de différents fournisseurs sont importants [23].

Dans l'IoT, les Objets intelligents passent par trois étapes : la phase préparatoire, la phase opérationnelle et la phase de maintenance [44].

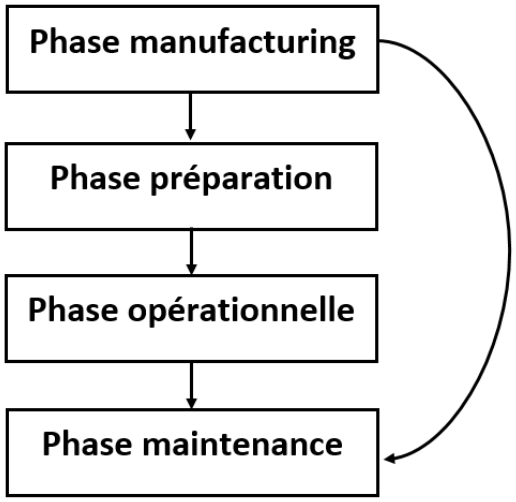

FIGURE 1.1 – Cycle de vie d'un appareil dans l'IoT.

- La phase préparatoire (bootstrapping) : déploiement des Objets (capteurs, tags), leur configuration avec les informations nécessaires, par exemple les identificateurs, les clés de sécurité, etc.
- La phase opérationnelle : l'Objet connecté se met à réaliser sa mission qui diffère d'une application à une autre.
- La phase de maintenance : effectue des mises à jour, règle les problèmes en faisant d'éventuelles réparations des Objets en cas de défaillances par exemple. Il est même possible de remplacer des Objets et redémarrer à nouveau à partir de la phase préparatoire.

1.2.3 Fonctionnement de l'IoT

1.2.3.1 Technologies de l'IoT

L'IoT permet l'interconnexion des différents Objets intelligents via l'Internet. Ainsi, pour son fonctionnement, plusieurs systèmes technologiques sont nécessaires. "L'IoT désigne diverses solutions techniques (RFID, TCP/IP, technologies mobiles, etc.) qui permettent d'identifier des Objets, capter, stocker, traiter, et transférer des données dans les environnements physiques, mais aussi entre des contextes physiques et des univers virtuels" [43]. En effet, bien qu'il existe plusieurs technologies utilisées dans le fonctionnement de l'IoT, nous mettons l'accent seulement sur quelques-unes qui sont, selon Han et Zhanghang, les technologies clés de l'IoT. Ces technologies sont les suivantes : RFID, WSN et M2M, et elles sont définies ci-dessous [43].

- **RFID :** est une technologie sans fil qui est utilisée pour l'identification des Objets[14], elle englobe toutes les technologies qui utilisent des ondes radio pour identifier automatiquement des Objets ou des personnes. C'est une technologie qui permet de mémoriser et de récupérer des informations à distance grâce à une étiquette qui émet des ondes radio. Il s'agit d'une méthode utilisée pour transférer les données des étiquettes à des Objets, ou pour identifier ces Objets à distance. L'étiquette contient des informations stockées électroniquement pouvant être lues à distance [43].

- **WSN :** est un ensemble de nœuds qui communiquent sans fil et qui sont organisés en un réseau coopératif. Chaque nœud possède une capacité de traitement et peut contenir différents types de mémoires, un émetteur-récepteur RF et une source d'alimentation. Il peut aussi tenir compte des divers capteurs et actionneurs [46]. Comme son nom l'indique, le WSN constitue un réseau de capteurs sans fil qui peut être une technologie nécessaire au fonctionnement de l'IoT.

- **M2M :** est l'association des technologies de l'information et de la communication avec des Objets intelligents dans le but de donner à ces derniers les moyens d'interagir sans intervention humaine avec le système d'information d'une organisation ou d'une entreprise [43].

1.2.3.2 Composants de l'IoT

L'IoT n'est pas une technologie, mais un système où l'intégration de tous les composants induit une complexité que l'interopérabilité diminue mais n'évite pas. La gestion des interfaces y est déterminante. Voici les principaux systèmes technologiques nécessaires au fonctionnement de l'IoT [6] :

Type de systèmes	**Identification (y compris lecteurs)**	**Capteurs**	**Connexion**	**Intégration**	**Traitement de données**	**Réseaux**
Enjeux	Reconnaître chaque objet de façon unique et recueillir les données stockées au niveau de l'objet.	Recueillir des informations présentes dans l'environnement pour enrichir les fonctionnalités du dispositif.	Connecter les systèmes entre eux.	Intégrer les systèmes pour que les données soient transmises d'une couche à l'autre.	Stocker et analyser les données pour lancer des actions ou pour aider à la prise de décisions.	Transférer les données dans les mondes physiques et virtuels.
Technologies Anciennes (exemples)	Codes barre, solutions RFID simples	Thermomètre, hydromètre…	Câble, Bluetooth, Infrarouge, WiFi, Satellites…	Middlewares …	Excel, ERP, CRM…	Internet, Ethernet…
Technologies Récentes (exemples)	Solutions RFID complexes, Surface Acoustic Waves, puces optiques, ADN	Capteurs miniaturisés nanotechnologies	Bluetooth, Near Field Communication (NFC), WiFi...	Middlewares évolués	Datawarehouse 3D (compatible avec les puces RFID), Web sémantique…	Réseau EPCglobal…

TABLE 1.1 - Composants de l'IoT [6]

Lier un Objet ou un lieu à Internet est un processus plus complexe que la liaison de deux pages Web. L'IoT exige sept composants [6] :

- Une étiquette physique ou virtuelle pour identifier les objets et les lieux. Quelques systèmes d'étiquetage sont décrits ci-dessous. Pour permettre aux étiquettes physiques plus petites d'être localisées, elles doivent être embarquées dans des marqueurs visuels.
- Un moyen de lire les étiquettes physiques, ou de localiser les étiquettes virtuelles.
- Un dispositif mobile tel qu'un téléphone cellulaire, un assistant personnel ou un ordinateur portable.
- Un logiciel additionnel pour le dispositif mobile.
- Un réseau sans fil de type 2G, 3G ou 4G afin de permettre la communication entre le dispositif portable et le serveur contenant l'information liée à l'objet étiqueté.
- L'information sur chaque objet lié. Cette information peut être contenue dans les pages existantes du Web, les bases de données comportant des informations de type prix, etc.
- Un affichage pour regarder l'information sur l'objet lié. À l'heure actuelle, il est probable que ce soit l'écran d'un téléphone mobile.

1.2.3.3 Protocoles de fonctionnement de l'IoT

De nombreuses normes IoT sont proposées pour faciliter et simplifier les tâches des programmeurs d'applications et des fournisseurs de services. Différents groupes ont été créés pour fournir des protocoles, y compris les efforts menés par le W3C, IETF, EPCglobal, IEEE et l'ETSI [3].

L'IoT ambitionne de faire communiquer chaque système avec tous les autres au moyen de protocoles communs. La mise en application à une large échelle du concept d'IoT apparaît largement tributaire d'une standardisation de la communication entre Objets dite M2M [18].

- Au niveau de la couche de Liaison, le standard **IEEE 802.15.4** est plus adapté que l'Ethernet aux environnements industriels difficiles.
- Au niveau réseau, le standard **6loWPan** a réussi à adapter le protocole Ipv6 aux communications sans fil entre nœuds à très faible consommation.
- Au niveau routage, l'IETF a publié en 2011 le standard **RPL**.
- Au niveau de la couche application le protocole **CoAP** qui tente d'adapter HTTP, beaucoup trop gourmand aux contraintes des communications entre nœuds à faible consommation.

• **CoAP Constrained Application Protocol :** est un protocole de couche d'application pour les applications IoT. Il définit un protocole de transfert Web basé sur les fonctionnalités HTTP, est lié à UDP (et non TCP) par défaut qui le rend plus approprié pour les applications IoT. En outre, CoAP modifie certaines fonctionnalités HTTP pour répondre aux exigences de l'IoT telles que la faible consommation d'énergie et le fonctionnement en présence de liens à perte et bruyants. CoAP a été conçu sur la base de REST qui représente un moyen plus simple d'échanger des données entre les clients et les serveurs via HTTP [3]. REST peut être considéré comme un protocole de connexion cacheable qui repose sur l'architecture sans serveur apatride. Il est utilisé dans les applications de réseaux sociaux et mobiles et élimine l'ambiguïté en utilisant les méthodes HTTP get, post, put et delete. Il permet aux clients et aux serveurs d'exposer et de consommer des services Web comme le protocole d'accès aux Objets simples (SOAP), mais de manière plus simple en utilisant les identificateurs de ressources uniformes (URI). CoAP vise à permettre à de minuscules appareils à faible puissance, le calcul et les capacités de communication à utiliser les interactions RESTful. Avec CoAP, les interactions entre services web de l'Internet des PC et de l'Internet des Objets deviennent bien plus simples à réaliser, une passerelle applicative assez légère (correspondance entre les commandes REST et CoAP) se charge de l'adaptation d'un monde à l'autre [3].

• **MQTT Message Queue Telemetry Transport :** représente un protocole de messagerie idéal pour les communications IoT et M2M. Il vise à connecter des

périphériques et des réseaux intégrés aux applications et au middleware. MQTT utilise le modèle de publication/souscription pour offrir une flexibilité de transition et une simplicité d'implémentation. Il convient aux périphériques à ressources limitées qui utilisent des liens peu fiables ou à faible bande passante. MQTT est construit en haut du protocole TCP. Il se compose de trois composants ; abonnés, éditeurs et courtiers. De nombreuses applications utilisent MQTT telles que les soins de santé, la surveillance, le compteur d'énergie et la notification de Facebook. Par conséquent, le protocole MQTT permet d'acheminer les périphériques de petite taille, à faible consommation et à faible mémoire dans des zones vulnérables et réseaux à faible bande passante [3].

- **XMPP Protocole de messagerie et de présence extensible :** est une norme de messagerie instantanée IETF (IM) qui est utilisée pour les conversations multiparties, les appels vocaux et vidéo et la téléprésence. Il permet aux utilisateurs de communiquer entre eux en envoyant des messages instantanés sur Internet quel que soit le système d'exploitation qu'ils utilisent. XMPP permet aux applications de messagerie instantanée d'accéder à l'authentification, au contrôle d'accès, à la mesure de la confidentialité, au cryptage hop-by-hop et à la compatibilité avec d'autres protocoles. Beaucoup de fonctionnalités XMPP en font un des protocoles préférés par la plupart des applications de messageries instantanées et pertinentes dans le cadre de l'IoT. Il fonctionne sur une variété de plateformes basées sur Internet de manière décentralisée. XMPP est sécurisé et permet d'ajouter de nouvelles applications au-dessus des protocoles de base [3].

- **AMQP Message avancé Protocole de mise en file d'attente :** est un protocole de couche d'application standard ouvert pour l'IoT se concentrant sur des environnements axés sur les messages. Il requiert un protocole de transport sécurisé comme TCP pour échanger des messages. Il prend en charge une communication fiable via des primitives de garantie de livraison de messages, en définissant un protocole au niveau du fil, les implémentations AMQP peuvent interopérer entre elles. Les communications sont traitées par deux composants principaux : échanges et files

d'attente de messages. Les échanges sont utilisés pour acheminer les messages vers les files d'attente appropriées. Le routage entre les échanges et les files d'attente des messages repose sur certaines règles et conditions prédéfinies. Les messages peuvent être stockés dans les files d'attente, puis envoyés au récepteur par la suite. AMQP prend également en charge le modèle de communication publier/souscrire [3].

1.2.4 Domaines d'application

L'idée émergente de l'IoT trouve rapidement sa voie tout au long de notre vie moderne visant à améliorer notre qualité de vie en connectant de nombreux périphériques intelligents, technologies et applications. Les potentialités offertes par l'IoT et son aspect ubiquitaire permettent le développement de nombreuses applications. Dans l'ensemble, l'IoT permettrait l'automatisation de tout ce qui nous entoure. Cependant, seules quelques applications sont actuellement déployées. Nous citerons quelques domaines que l'IoT touche ci-dessous.

- **La domotique**

La domotique est l'ensemble des techniques utilisées dans l'habitation qui permet de centraliser le contrôle des différents systèmes d'une maison. Le principe de la domotique est de faire en sorte qu'une maison devienne intelligente, indépendante et qu'elle réfléchisse par elle-même. Tous ces principes sont possibles grâce à l'IoT qui permet de connecter les appareils maison à un réseau et de les piloter à distance. La domotique a pour but d'améliorer le confort quotidien en automatisant ou en gérant à distance les tâches récurrentes.

- **L'agriculture**

L'usage des Objets connectés se démocratise dans l'agriculture. De nombreuses améliorations en découlent, concernant la gestion des engins agricoles, la maîtrise de l'irrigation ou la gestion optimisée des intrants, que la surveillance de la croissance

des plantes ou encore la prévention des risques météo. De quoi renouveler en profondeur les pratiques de cette activité ancestrale, grâce à l'analyse des données récoltées et au pilotage de plus en plus fin des exploitations.

- **Smart cities**

Le terme "Smart Cities" (ou villes intelligentes en français) est utilisé pour désigner l'écosystème cyber physique émergeant par le déploiement d'une infrastructure de communication avancée et de nouveaux services sur des scénarios à l'échelle de la ville. Grâce à des services avancés, il est en effet possible d'optimiser l'utilisation des infrastructures physiques de la ville (par exemple, les réseaux routiers, le réseau électrique, etc.) et la qualité de vie des citoyens.

- **La santé**

Les patients porteront des capteurs médicaux qui surveillent des paramètres tels que la température corporelle, la pression artérielle, l'activité respiratoire. D'autres capteurs portables (accéléromètres, gyroscopes) ou fixes (proximité) seront utilisés pour recueillir les données utilisées pour surveiller les activités des patients dans leur milieu de vie. Les informations seront agrégées localement et transmises aux centres médicaux éloignés, qui pourront effectuer une surveillance à distance avancée et seront capables de prendre des mesures rapides au besoin.

- **La nanotechnologie**

Les nanotechnologies pourraient régler de nombreux problèmes pour l'instant insolubles avec des Objets connectés "normaux". Vêtements connectés, pare-brise intelligents permettant le système "tête haute" avec affichage de l'information à hauteur du conducteur, etc. C'est un fait. Les nanotechnologies donnent la réplique à l'IoT, et inversement. Néanmoins, de nombreuses questions restent en suspens.

Peut-on connecter les nanotechnologies à Internet ? Peut-on réduire la taille des routeurs au nanomètre ? Quelle quantité de données serait remontée par des dispositifs de ce genre ? Est-il possible de faire un réseau maillé à partir de ces nano robots ?

- **L'environnement**

Dans ce domaine, un rôle clé est joué par la capacité de détecter de manière répartie et autogérée les phénomènes naturels et les processus (température, vent, précipitations, hauteur des rivières, etc.), ainsi que l'intégration transparente de ces données hétérogènes.

- **Sécurité et surveillance**

La surveillance de sécurité est devenue une nécessité pour les bâtiments d'entreprise, centres commerciaux, usine, parkings et de nombreux autres lieux publics. Tout en préservant la vie privée des utilisateurs, des capteurs ambiants peuvent être utilisés pour surveiller la présence de produits chimiques dangereux. Des capteurs de surveillance du comportement des personnes peuvent être utilisés pour évaluer la présence de personnes qui agissent de manière suspecte.

La figure suivante représente différents domaines d'application de l'IoT :

FIGURE 1.2 – Domaines d'application de l'IoT [3].

1.3 Sécurité de l'IoT

1.3.1 Vulnérabilités et menaces de sécurité dans l'IoT

L'IoT est l'intégration de multiples réseaux hétérogènes, il devrait traiter les problèmes de compatibilité entre les différents réseaux qui sont sujets aux problèmes de sécurité. Les diverses applications potentielles de l'IoT, l'hétérogénéité de ses

technologies habilitantes et sa forte dimension humaine et socioéconomique rendent sa sécurité un sujet difficile et complexe.

La sécurité représente un élément essentiel pour permettre l'adoption généralisée des technologies et des applications de l'IoT. Sans garantie en termes de confidentialité et d'authenticité, les parties prenantes concernées seront peu susceptibles d'adopter des solutions IoT à grande échelle. L'IoT est plus susceptible d'être attaqué que l'Internet puisque des milliards d'appareils de plus produiront et consommeront des services. L'ubiquité de l'IoT amplifiera les menaces classiques de sécurité qui pèsent sur les données et les réseaux, mais en plus, le rapprochement du monde physique et du monde virtuel à travers l'IoT ouvre la voie à de nouvelles menaces qui pèseront directement sur l'intégrité des Objets eux-mêmes, les infrastructures et processus (monde physique), et la vie privée des personnes. On distingue trois grandes catégories de menaces qui sont décrites ci-dessous :

- **Menaces sur les données et le réseau**

Le développement rapide de la numérique pose d'ores et déjà plusieurs défis qui se retrouvent à une échelle paroxysmique dans le domaine de l'IoT. En captant le mode de vie des utilisateurs, les Objets communiquant exacerbent les questions de vie privée et les problématiques de souveraineté et d'exploitation des données. Par ailleurs, qui dit connecter un Objet, implique de le rendre vulnérable et pose donc d'importantes questions en termes de cybercriminalité [15].

Le système IoT produit également une grande quantité de données connues sous le nom de Big Data. Ainsi, les Big Data ont leurs propres problèmes de gestion et de sécurité. Sans mesures particulières, les données stockées sur ces dispositifs seraient alors accessibles, y compris des données cryptographiques qui permettraient d'accéder à d'autres données sensibles ou jouer des rôles sensibles dans les systèmes complexes les hébergeant. Les attaques physiques telles que la destruction, la délocalisation ou le masquage des nœuds peuvent rendre inaccessible la ressource fournie par le nœud. De plus, les secrets cryptographiques stockés à l'intérieur du

nœud peuvent être extraits, ce qui permet des attaques de répétition, une injection de paquets, un clone ou une reprogrammation de nœud. Sur la couche physique, les attaques DoS peuvent être lancées en bloquant ou en altérant le signal radio. À la couche de lien, une attaque sur la disponibilité du réseau consiste à inonder le réseau avec de gros paquets pour occuper toute la bande passante. L'injection de paquets peut également conduire à l'épuisement de la batterie ou à la collision des paquets, suivie d'une perte de paquets [26].

- **Menaces sur la vie privée**

Les capteurs connectés à Internet et intégrés dans notre entourage récoltent des informations qui nous sont privées. À titre d'exemple ; l'état de santé, la localisation géographique, le contenu du réfrigérateur, etc. Ces capteurs apprennent avec le temps le comportement, les préférences et les habitudes de leurs utilisateurs, du coup, ces utilisateurs demandent à avoir le droit dont on protège leur vie privée contre toute fuite d'informations qu'ils jugent critiques sur Internet. En d'autres mots, les utilisateurs doivent savoir qui utilise quoi comme données les concernant et pour quelle raison. D'autre part, il est même nécessaire de permettre aux utilisateurs de l'IoT d'autoriser la récupération de leurs données par des tierces parties (pour faire des statistiques par exemple) ou simplement la refuser. Tous les pronostics envisagent le développement d'une informatique ambiante avec potentiellement des dizaines d'Objets par personne y compris dans leur sphère privée et intime. Ces Objets de l'espace personnel sont géo localisables, peuvent communiquer avec d'autres Objets à travers des réseaux spontanés, peuvent écouter ce que dit la personne, peuvent filmer la personne et/ou son environnement et peuvent même enregistrer son rythme cardiaque, son rythme respiratoire, la température de son corps.

Des questions légitimes se posent sur le devenir de cette masse de données personnelles. Sans régulation stricte, une protection accrue de la privacy, un degré élevé de contrôle des Objets par les usagers, l'adoption de l'IoT serait un échec. Dans son rapport sur l'IoT ? l'ITU a pointé du doigt ces menaces potentielles. Elle conclut que la protection de la privacy ne doit pas se limiter à des solutions

technologiques, mais doit comprendre des mesures juridiques [7].

- **Menaces sur le système et les Objets physiques**

Tous les dispositifs de l'IoT ont la mémoire faible et les ressources limitées en termes de calcul, ainsi ils sont plus vulnérables à plusieurs attaques. Lorsque les périphériques sont connectés et mis en service dans le réseau, le matériel de saisie, la sécurité et les paramètres de domaine peuvent être écoutés [33]. IoT apporte une nouvelle surface d'attaque ; des attaques physiques sur ou à travers différents dispositifs. [ii] Le système IoT se compose de technologies hétérogènes, ce qui augmente la complexité dans la détermination et la compréhension des exigences de sécurité. Il comprend également des appareils mobiles nécessitant une mobilité. Cette mobilité augmente les risques et les problèmes de sécurité [38]. Du point de vue des services, le problème principal concerne la façon d'intégrer les fonctionnalités et/ou les ressources fournies par les Objets intelligents dans ces services [36].

1.3.2 Attaques dans l'IoT

L'IoT est vulnérable à un nombre considérable d'attaques. Il existe diverses attaques sur des schémas d'authentification d'utilisateurs distants tels que le dictionnaire, men-in-the-middle, le texte en clair, la carte à puce perdue, la modification, le déni de service (DOS), la divulgation de clé de session, l'emprunt d'identité, l'initié, etc. Ces attaques peuvent être gênantes pour un utilisateur légitime lors de l'accès à un système dans un but spécifique. Une attaque de dictionnaire tente de deviner des mots de passe communs basés sur le dictionnaire. Une attaque men-in-the-middle est implémentée pour reconnaître l'information. Une attaque en clair est utilisée lorsque le texte chiffré est volé. Une attaque perdue de carte à puce est introduite lorsqu'une carte à puce est perdue, puis un attaquant peut appliquer des procédures pour acquérir l'information. Une attaque de modification est implémentée pour modifier les informations ; en d'autres termes, l'attaquant modifie les informations puis retransmet les données à nouveau [30]. Ces attaques sont présentées dans le tableau 1.2

Nom de l'attaque	But et résultat de l'attaque	Menace	Active ou Passive
DoS	- Saturer un serveur ou bloquer le trafic. - rendre un service non disponible.	- Intégrité. - disponibilité. - confidentialité.	Active
Man-in-the-Middle	- Intercepter les communications entre deux parties contrôler la conversation. - écouter, modifier ou supprimer des données.	-Intégrité. -confidentialité.	Active
L'usurpation d'identité	- vol d'identité. - réaliser des actions frauduleuses. - prendre délibérément l'identité d'une autre personne vivante.	- Confidentialité. - Authentification.	Active
Flooding	- épuiser la mémoire et l'énergie des noeuds - Saturer le réseau	- Intégrité. - disponibilité.	Active
Les attaques de cartes à puce	- pouvoir accéder aux informations et aux secrets contenus dans la carte(code PIN, clé(s) secrète(s)cryptographique(s), etc...).	- physiques. - logicielles.	Active
Wardriving	- Utilisé pour pouvoir accéder à internet au nom d'une autre personne. - Parcourir tous les lieux où le Wifi est déployée afin de découvrir toutes les bornes Wifi existantes noter l'adresse géographique.	- Confidentialité.	Passive
Sniffing	- Capturer les trames circulent local et afficher leur contenus (entêtes des sur un réseauprotocoles, id des user, MDP non crypté, etc.).	- Confidentialité.	Passive

TABLE 1.2 – Type d'attaques dans l'IoT

1.3.3 Mécanismes de sécurité

Le terme sécurité englobe un large éventail de concepts différents. En premier lieu, il se réfère à la prestation de base des services de sécurité, dont la confidentialité, l'authentification, l'intégrité, l'autorisation, la non-répudiation et la disponibilité. Ces services de sécurité peuvent être mis en œuvre au moyen de différents mécanismes cryptographiques tels que des blocs de chiffrement, des fonctions de hachage ou des algorithmes de signature. Pour chacun de ces mécanismes, une infrastructure de gestion de clés solide est fondamentale pour gérer les clés cryptographiques requises. Cependant, la sécurité doit non seulement se concentrer sur les services de sécurité requis, mais aussi sur la façon dont ils sont réalisés dans le système global et comment les fonctionnalités de sécurité sont exécutées.

1.3.3.1 La cryptographie

La cryptographie est sans doute la technique la plus utilisée dans le cadre des réseaux filaires et des réseaux sans fil traditionnels disposant d'une capacité de calcul et de mémoire conséquente. Les solutions de cryptographie sont réputées comme des solutions sures qui répondent à l'ensemble des problèmes liés à la sécurité des données. Les spécificités des réseaux de capteurs, à savoir une faible puissance de calcul et une mémoire limitée à laquelle se rajoute la problématique de préservation de l'énergie, sont des freins considérables à l'utilisation des systèmes cryptographiques courants réputés surs (SSL, RSA, etc.) [17].

On distingue deux types de cryptographie ; la cryptographie symétrique à clé secrète et la cryptographie asymétrique ou à clé publique.

- **Cryptographie symétrique**

Cette Cryptographie utilise une clé commune pour le chiffrement et le déchiffrement. La clé est partagée entre l'émetteur et le destinataire et doit être communiquée par un des correspondants au second. Si un attaquant intercepte cette clé pendant la communication, il sera en mesure de déchiffrer les messages

transmis. La sécurité de ce type de chiffrement repose sur la non-divulgation d'un secret commun entre les deux correspondants [27]. Les solutions de chiffrement à clés symétriques sont exploitables au sein des réseaux de capteurs et apportent une réelle solution pour la sécurité du réseau. Cependant, si le chiffrement à clé symétrique est possible au sein des réseaux de capteurs, la sécurité totale de ce type de solution reste à démontrer. D'une part parce que le chiffrement à clé symétrique ne garantit pas l'authentification des données comme peut le faire la signature numérique des chiffrements à clés publiques, et d'autre part, parce que se pose le problème de la distribution des clés de chiffrement. Ainsi, si le protocole IEEE 802.15.4 spécifie la méthode de chiffrement à utiliser, à aucun moment elle ne spécifie comment doivent être gérées les clés et comment permettre l'authentification des données [17].

- **Cryptographie asymétrique**

Cette cryptographie utilise une paire de clés dont l'une peut être rendue publique. Ce type de chiffrement renforce la sécurité, car la clé privée n'a pas besoin d'être communiquée entre les correspondants. Si un attaquant connaît la clé publique et est capable d'intercepter les messages chiffrés entre les correspondants, il n'est pas en mesure de déchiffrer ces messages. Cette résistance est due à la difficulté à factoriser les grands nombres entiers. La sécurité de ce type de chiffrement repose essentiellement sur la non-divulgation de la clé privée [27]. Les mécanismes de cryptographie asymétriques sont potentiellement de bonnes solutions de sécurité quand le nombre de nœuds est très élevé comme les réseaux de capteurs. Ils posent moins de problèmes pour la gestion des clés. Cependant, la cryptographie asymétrique utile à l'établissement de clés partagées entre nœuds n'est pas réaliste. Les demandes en temps de calcul et en mémoire sont si fortes dans ce cas en plus de la nécessité d'une infrastructure centralisée pour la gestion des clés.

Des travaux de recherche essaient d'optimiser des algorithmes asymétriques comme RSA, appelé WM-RSA pour les capteurs de type [48]. Les résultats sont meilleurs que ceux de RSA, mais le temps d'exécution est toujours de l'ordre de la seconde, temps qui n'est pas envisageable sur certaines applications des réseaux de

capteurs. En plus, la taille des clés pose toujours des problèmes de stockage (1024 bits pour RSA). Dans cette optique, des travaux récents tentent d'apporter une réponse avec l'utilisation de cryptographie à clé publique basée sur les courbes elliptiques (ECC). Les courbes elliptiques assurent le même niveau de sécurité que RSA avec des clés plus courtes [21].

1.3.3.2 Fonctions de hachage

En cryptographie, les fonctions de hachage sont surtout utilisées pour optimiser et renforcer les performances des algorithmes de chiffrement et ainsi assurer des services de sécurité tels que le contrôle d'accès (authentification et autorisation) et l'intégrité à travers des relations d'égalité, d'égalité probable, non-égalité, etc. Ainsi, on peut noter :

- La réduction de la taille des données à chiffrer (par exemple en signature numérique) permettant d'accélérer les calculs.
- La génération de nombres pseudo-aléatoires utilisés dans diverses applications cryptographiques.
- Les fonctions de hachage aident aussi à la génération de clé pour le chiffrement et à la distribution de clés en cryptographie symétriques.

1.3.3.3 La signature

Procédé d'identification du signataire d'un document électronique, basé sur l'utilisation d'un algorithme de chiffrement, qui permet de vérifier l'intégrité du document et d'en assurer la non-répudiation. Le système de signature peut signer un nombre illimité de messages et la taille de la signature augmente logarithmiquement en fonction du nombre de messages signés. La taille de la signature dans un système typique peut varier de quelques centaines d'octets à quelques kilo-octets, et la génération d'une signature peut nécessiter quelques centaines à quelques milliers de calculs de la fonction de cryptage classique sous-jacente.

1.3.3.4 Échange de clés

Les approches de gestion de clés à base de pré-déploiement ne seraient applicables qu'à une échelle réduite et limitée à des usagers expérimentés. Nous croyons que la situation devrait évoluer dans le contexte de l'IoT vers des méthodes de gestion de clés basée sur une cryptographie peu coûteuse, ne nécessitant pas une intervention humaine. Ainsi, la cryptographie à base de courbes elliptiques [17], entre autres, devrait jouer un rôle important dans les systèmes de gestion de clés. Par ailleurs, l'ubiquité des Objets de l'IoT et la difficulté de leur procurer une protection physique et/ou une surveillance permanente, les exposent au risque de compromission physique par des intrus. Ceci peut avoir un impact important si les intrus réussissent à récupérer les clés cryptographiques qui seraient dans la mémoire des Objets corrompus et à les utiliser pour mener des attaques diverses sur le réseau et les autres Objets. Afin de réduire les conséquences de cette vulnérabilité due à l'intégration des Objets de l'IoT au monde physique, il serait nécessaire que la gestion de clés soit résiliente. En d'autres mots tolérants à la compromission des Objets [7].

Deng et al.[13] définissent deux propriétés qui doivent être vérifiées dans la conception d'un système résilient de gestion de clés :

- la propriété d'opacité : un adversaire ne devrait pas avoir la capacité de déduire d'autres clés utilisées dans le réseau en compromettant un nombre réduit d'Objets.
- la propriété d'inoculation : un adversaire ne devrait pas pouvoir introduire un Objet non légitime dans le réseau en compromettant un nombre réduit d'Objets.

Deng et al. pensent que ce type de propriétés devrait être pris en compte dans la conception d'une gestion de clés résiliente pour l'IoT [13]. Des solutions de sécurité doivent permettre aux capteurs connectés d'échanger, en toute sécurité, des clés cryptographiques avec les autres dispositifs connectés à internet. Cependant, un échange

de clés avec une faible dissipation énergétiques est vivement recommandé dans l'IoT [44].

1.3.3.5 Authentification mutuelle

L'authentification mutuelle est une notion de sécurité également importante pour un protocole d'échange de clé authentifiée. Le but est qu'après chaque session, le participant qui accepte la clé échangée veut s'assurer que son partenaire l'accepte aussi. Puisqu'il est impossible d'obtenir à la fois que "A accepte implique B accepte" et que "B accepte implique A accepte", on change un peu le but. Intuitivement, un participant accepte lorsque toutes les étapes sont consistantes et lui permettent d'extraire la clé partagée. Un participant termine le protocole s'il n'y a plus de message à échanger [5].

1.3.4 Services de sécurité dans l'IoT

Dans l'IoT, chaque entité virtuelle et physique est transmissible, adressable et accessible via Internet. Ces entités virtuelles et physiques produisent une communication transparente et un service collaborant avec des utilisateurs et d'autres dispositifs créant des réseaux orientés service [33].

Un service est une collection de données et de comportements associés pour accomplir une fonction ou une caractéristique particulière d'un dispositif ou d'une partie d'un dispositif. Il peut faire référence à d'autres services primaires ou secondaires et/ou à un ensemble de caractéristiques qui composent le service. Les systèmes IoT intègrent de manière transparente les Objets physiques, les données et les périphériques. Dans ce scénario, les services agissent comme des ponts par lesquels ces "Objets intelligents" interagissent les unes avec les autres de façon automatisée et avec une intervention aussi humaine que possible [12]. Un système IoT nécessite l'authentification et l'autorisation des utilisateurs et des périphériques. L'authentification vérifie l'identité des utilisateurs ou des périphériques dans un système IoT et l'autorisation fournit les privilèges nécessaires à l'entité autorisée.

La confidentialité des données crypte les données pour fournir la protection contre son utilisation non prévue [38].

Nous allons définir par la suite trois services de sécurité que nous jugerons utiles et importants dans la réalisation de notre travail qui sont ; la confidentialité, l'authentification et l'identification.

1.3.4.1 La confidentialité

Si notre utilisation du Web aujourd'hui peut déjà être source d'une connaissance étendue de nos comportements individuels, la traçabilité des Objets de la vie quotidienne serait encore infiniment plus sensible [50]. L'un des défis majeurs à relever pour l'IoT est celui d'assurer la confidentialité et la sécurité des données échangées en raison de l'hétérogénéité inhérente des Objets connectés à internet et de la capacité de surveiller et de contrôler les Objets physiques. Si l'on estime qu'il existe déjà des problèmes de sécurité avec un milliard de smartphones pourtant placés sous contrôle humain, quelle sera la situation avec mille milliards d'Objets autonomes connectés, recueillant des informations sur notre santé, notre conduite automobile ou les paramètres de notre habitat ? [3]

La confidentialité garantie que seules les entités autorisées peuvent accéder et modifier les données [36]. Les questions de confidentialité des données peuvent se poser tant au cours de la collecte des données que lors de la transmission et du partage des données [2].

- **Importance**

La protection des renseignements personnels est l'un des domaines les plus sensibles dans le contexte de l'IoT. La question de la protection de la vie privée est donc cruciale [33]. Elle comprend la dissimulation de ces renseignements ainsi que la capacité de contrôler ce qui se passe avec cette information. Le droit à la vie

privée peut être considéré comme un droit de l'homme fondamental et inaliénable, ou comme un droit ou une possession personnelle [49]. La confidentialité et la sécurité sont donc essentielles, ainsi que la propriété des données. Notez, ce ne sont pas des questions IoT, mais des questions générales sur internet qui sont amplifiées par la croissance de nouvelles applications. Ces problèmes existent déjà pour l'interne, et l'industrie. Les organismes gouvernementaux commencent lentement à les reconnaître et à prendre des mesures [4].

- **Solutions dans les réseaux classiques**

Le respect des exigences de confidentialité des clients est assez difficile. Un certain nombre de technologies ont été développées afin d'atteindre les objectifs de confidentialité de l'information. Ces technologies d'amélioration de la protection de la vie privée (PET ; Technologies d'amélioration de la confidentialité) peuvent être décrites brièvement comme suit :

– **Les réseaux privés virtuels (VPN) :** sont des extranets établis par des groupes proches de partenaires commerciaux. Comme seuls les partenaires ont accès, ils promettent d'être confidentiels et d'avoir de l'intégrité. Cependant, cette solution ne permet pas un échange d'informations global dynamique et n'est pas pratique pour les tiers au-delà des frontières de l'extranet [49].

– **Transport Layer Security (TLS) :** basé sur une structure de confiance mondiale appropriée pourrait également améliorer la confidentialité et l'intégrité de l'IoT. Toutefois, comme chaque étape de délégation ONS nécessite une nouvelle connexion LS, la recherche d'informations serait affectée négativement par de nombreuses couches supplémentaires [49].

– **Les extensions de sécurité DNS (DNSSEC) :** utilisent la cryptographie à clé publique pour signer des enregistrements de ressources afin de garantir l'authenticité de l'origine et l'intégrité des informations livrées. Toutefois, la

DNSSEC ne peut assurer l'authentification de l'ONS que si l'ensemble de la communauté internet l'adopte [49].

- **Onion Routing :** crypte et mélange le trafic Internet à partir de nombreuses sources différentes, c'est-à-dire les données sont enveloppées dans plusieurs couches de cryptage, en utilisant les clés publiques des routeurs onion sur le chemin de transmission. Ce processus empêcherait l'appariement d'un paquet de protocoles internet particulier à une source particulière. Cependant, le routage d'onion augmente les temps d'attente et entraîne ainsi des problèmes de performance [49].

- **Les systèmes privés de récupération d'information (PIR) :** cachent quel client est intéressé par les informations, une fois que les EPCIS ont été localisés. Cependant, des problèmes d'évolutivité et de gestion des clés, ainsi que des problèmes de performance, se poseraient dans un système globalement accessible tel que l'ONS, ce qui rend cette méthode peu pratique [49].

- **Peer-to-Peer (P2P) systems :** est une autre méthode pour augmenter la sécurité et la confidentialité, qui montre généralement une bonne évolutivité et de performance dans les applications, P2P pouvait être basé sur des tables de hachage distribuées (DHT) [49].

- **Contrôle d'accès :** assurer la confidentialité dans les systèmes de gestion de la connaissance. Une approche standard, qui correspond bien aux caractéristiques des environnements IoT, est représentée par le contrôle d'accès basé sur les rôles (RBAC) [36].

- **Cloud computing :** est une autre méthode pour l'assurer la confidentialité, si les périphériques IoT transmettent des données au cloud via la connexion HTTP, l'intégrité des données est atteinte. Dans le cas de HTTPS, la confidentialité et à l'intégrité sont atteintes à la fois [19].

- **Discussion**

La confidentialité dans le partage et la gestion des données peut se produire parce qu'une grande partie de l'information transmise (par exemple l'emplacement GPS) peut être sensible, mais elle peut également être requise pour permettre le bon fonctionnement des applications utiles en temps réel telles que l'analyse du trafic [2].

Des solutions coutumiers pour assurer la confidentialité des données peuvent ne pas être appliquées directement aux contextes IoT, en raison de deux principaux facteurs limitatifs. Le premier concerne la quantité de données générées par de tels systèmes et concerne par conséquent les problèmes d'extensibilité. Le second concerne la nécessité de contrôler l'accès aux données d'une manière en ligne et flexible, les droits d'accès changeant au moment de l'exécution et étant appliqués aux flux de données dynamiques [36].

L'analyse des données collectées à partir de dispositifs intelligents et diverses déductions peut porter atteinte à la vie privée des individus ou des groupes. Une politique de confidentialité efficace doit être conçue et appliquée par le système. La politique de confidentialité pourrait être mise à jour dynamiquement par les administrateurs de politiques autorisés. Une fois la politique configurée, le système doit veiller à ce que toute politique de confidentialité ne soit pas respectée [37].
L'expérience que nous avons aujourd'hui de l'internet tend, en effet, à accréditer la thèse selon laquelle l'utilisateur privilégie le bénéfice retiré du service rendu au quotidien face au risque d'atteinte à sa vie privée, tout au moins jusqu'à ce qu'intervienne un incident [50]. La confidentialité des personnes et des Objets doit être assurée pour empêcher l'identification et le suivi non autorisés [34].

Pour ce qui concerne la confidentialité, un autre problème se pose : les algorithmes de chiffrement consomment rapidement trop de ressources (et donc de bat-

terie). L'idée est alors de choisir des algorithmes de chiffrement à la fois robustes et légers en consommation de ressources.

1.3.4.2 L'authentification

L'authentification est une fonctionnalité importante et critique dans le contexte de l'IoT pour permettre une communication sécurisée entre les périphériques [33]. En effet, un service d'authentification fournit la preuve que l'identité d'un Objet ou le sujet a l'identité qu'elle prétend avoir [32]. Le terme "authenticité" désigne la propriété qui garantit qu'un partenaire de communication est bien celui qu'il prétend être. L'authentification est la première barrière de sécurité qui permet d'empêcher un utilisateur non autorisé d'accéder aux nœuds. Une personne non autorisée pourrait très bien accéder à un nœud donné, sans attendre qu'il envoie des données à un nœud puits. Il est donc requis d'avoir une authentification forte auprès de chaque nœud.

Les mécanismes d'authentification sont capables d'empêcher les Utilisateurs d'accéder aux données des nœuds capteurs et de garantir la sécurité des données de manière efficace. L'authentification consiste à permettre à l'utilisateur légitime d'accéder aux ressources ainsi que de les refuser à une personne malveillante. Après l'authentification, on a le contrôle d'accès qui permet de restreindre l'accès à l'utilisateur authentifié aux seules données dont il a les privilèges [51].

- **Importance**

L'Authentification est connue comme un élément central pour faire face aux problèmes de sécurité et de confidentialité dans les réseaux informatiques. Elle peut empêcher les utilisateurs non autorisés d'accéder aux ressources, empêcher les utilisateurs légitimes d'accéder aux ressources d'une manière non autorisée, et permettre aux utilisateurs légitimes d'accéder aux ressources de la manière autorisée

[31].

- **Solutions**

Les solutions à considérer pour assurer l'authentification sont :

– **Mitiger les attaques par déni de service :** Il s'agit d'un gros problème dans les réseaux de capteurs dû aux ressources limitées des capteurs. Les faibles ressources des capteurs posent d'énormes problèmes dans tous les domaines de recherche sur les WSNs, dont la sécurité bien sûr. Le Déni de Service abusant justement des ressources des systèmes semble être l'attaque parfaite pour une personne mal-intentionnée.

– **Détection et révocation de nœuds compromis :** L'authentification des utilisateurs doit mettre en place des mécanismes pour détecter activement et ainsi révoquer un nœud compromis

– **Assurer la disponibilité des capteurs :** Les faibles ressources des capteurs posent d'énormes problèmes dans tous les domaines de recherche, dont la sécurité bien sûr. Le Déni de Service abusant justement des ressources des systèmes semble être l'attaque parfaite pour une personne mal-intentionnée. On a déjà du mal à se protéger des dénis de service distribués avec les serveurs web. Il est encore plus facile de faire tomber un réseau de capteurs.

- **Discussion**

Il y a eu beaucoup de scientifiques cryptographiques travaillant dans le domaine de l'authentification des utilisateurs distants à l'aide de la conception du système de carte à puce. Une authentification d'utilisateur utilisant un système de carte à puce contient généralement deux rôles : l'utilisateur et le serveur ; et trois protocoles : enregistrement, connexion et authentification, et changement de mot de passe. Dans le principe de la conception du protocole, pour garantir la confidentialité des

connexions, il ne peut pas révéler l'identité de l'utilisateur [8]. Les limitations de ressources et capacités des Objets embarqués miniaturisés rendent difficile l'utilisation des algorithmes cryptographiques actuels en raison de leur consommation en termes de calcul et de mémoire [7].

Nous croyons que l'émergence d'une cryptographie robuste et peu coûteuse en termes de ressources combinée à des avancées technologiques d'auto-récupération de l'énergie ("energy harvesting") permettrait de surmonter ces difficultés à moyen terme. En effet, plusieurs travaux de recherche ont montré que la cryptographie à base de courbes elliptiques [42] offrait un niveau de robustesse de la sécurité semblable à la cryptographie asymétrique classique avec l'avantage d'être peu coûteuse en termes de ressources (mémoire, calcul, bande passante). Par ailleurs, des développements récents ont démontré la possibilité de récupérer de l'énergie ("energy harvesting"), sous certaines conditions, de l'environnement des Objets communicants.

1.3.4.3 L'identification

L'IoT inclut divers Objets comme les ordinateurs, les nœuds de capteurs, les personnes, les véhicules, les médicaments, les livres, etc. Ces Objets doivent être identifiés de façon unique pour fournir un moyen de communication entre eux [33]. Chaque Objet doit être clairement identifié avant d'interagir avec le réseau. Les méthodes d'identification sont donc utilisées pour fournir une identité claire pour chaque Objet dans le réseau [3].

Selon des scénarios spécifiques, les Objets peuvent avoir besoin d'être identifiés de manière unique ou d'être identifiés comme appartenant à une classe donnée. Cela pourrait se faire essentiellement de deux façons. Le premier consiste à marquer physiquement un Objet au moyen de RFID, code QR ou similaire. De cette façon, un Objet peut être lu au moyen d'un dispositif approprié, renvoyer un identifiant pouvant être recherché dans une base de données pour récupérer l'ensemble des caractéristiques (description) qui lui est associé. La deuxième possibilité est de fournir un Objet avec sa propre description : s'il est équipé de moyens de com-

munication sans fil, il pourrait communiquer directement sa propre identité et les caractéristiques pertinentes. Ces deux approches ne s'excluent pas mutuellement et peuvent se compléter [36].

- **Importance**

L'identification est cruciale pour l'IoT afin de faire correspondre les services avec leur demande. L'identification permet de connaitre l'identité d'une entité alors que l'authentification permet de vérifier cette identité.

- **Solutions proposées**

Différents schémas d'identité ont été proposés pour l'IoT et il est prévu qu'il est douteux d'avoir une identification commune globalement. Beaucoup de méthodes d'identification sont disponibles pour l'IoT comme les EPC, la RFID, Short-OID, u-Code, la NFC, etc. De plus, l'adressage des Objets IoT est essentiel pour différencier l'ID d'Objet et son adresse, les méthodes d'adressage des Objets IoT incluent IPv6 et IPv4. 6LoWPAN fournit un mécanisme de compression sur les en-têtes IPv6 qui rend l'adressage IPv6 approprié pour les réseaux sans fil de faible puissance , donc l'adressage aide à identifier les Objets de façon unique [3, 33]

– **RFID :** L'identification par radiofréquence (RFID) est l'une des technologies les plus importantes utilisées dans l'IoT, car elle peut stocker des données sensibles, communiquer sans fil avec d'autres Objets et identifier/tracer des Objets automatiquement [22]. La technologie RFID est un outil clé pour l'IoT, car elle permet l'identification simultanée d'un grand nombre d'Objets avec des étiquettes coût-efficacité [2]. Dans le domaine des soins de santé, la technologie RFID est utilisée au sein de l'IoT et les applications courantes, y compris le suivi de localisation des actifs médicaux [22].

– **6LoWpAN :** Définit en particulier des mécanismes d'encapsulation et

de compression d'en-têtes permettant aux paquets Ipv6 d'être envoyés ou reçus via le protocole de communication pour réseaux radio IEEE 802.15.4. Le standard 6Lowpan ne prévoit pas de fonctions de sécurité en plus de celles potentiellement mises en œuvre au niveau du 802.15.4 et de Ipv6 [18]. 6lowpan permet une intégration complète de WSN dans l'internet [41].

– **IPv6 :** Une des technologies préconisées à l'IETF pour l'interconnexion des réseaux de l'IoT est IPv6. Un des avantages majeurs est l'exploitation de l'immense capacité d'adressage de 128 bits d'IPv6 ce qui répondrait aux besoins d'adressage à très large échelle d'un IoT qui comporterait potentiellement plusieurs dizaines de milliards d'Objets [7].

– **Identification biométrique :** R. Greenstadt et J. Beal ont proposé l'utilisation d'une imprégnation des Objets puis une identification biométrique continue pour la protection des Objets. Cette identification biométrique peut être diverse et variée comme les empreintes, l'image de la rétine, la fréquence de la voix, le mouvement, la reconnaissance du visage, etc. L'objectif est de permettre une reconnaissance assez naturelle du propriétaire de l'Objet et ainsi éviter un tas de failles et d'attaques de sécurité par de tierces parties non légitimes à manipuler les Objets [20].

- **Discussion**

Les progrès de la RFID et du WSN contribuent de façon significative au développement de l'IoT. En outre, de nombreuses autres technologies et dispositifs tels que les codes à barres, les Smart Phones, les réseaux sociaux et les réseaux informatiques sont utilisés pour former un réseau étendu pour supporter IoT [10].

Il est important qu'une étiquette RFID ayant été attachée à un Objet puisse, ultérieurement, être désactivée afin de permettre aux clients de décider s'ils souhaitent utiliser l'étiquette. Les étiquettes RFID peuvent être soit désactivées en les

mettant dans un filet de protection appelé "cage de Faraday" qui est impénétrable par des signaux radio de certaines fréquences, soit en les "tuant", c'est-à-dire en les supprimant et en les détruisant [49]. Nous croyons que l'identification robuste et scalable jouera un rôle déterminant dans la sécurité de l'IoT. En effet, elle permettrait de compenser la vulnérabilité des Objets livrés sans protection à un environnement potentiellement hostile. Cette absence de protection physique encouragera alors les attaques à base de compromission physique des Objets pour accéder aux données cryptographiques sensibles qui ouvrent la voie à des attaques par escalade de privilèges.

La vision orientée vers les Objets est de loin la vision dominante aujourd'hui, et la technologie RFID est souvent (à tort) supposée être synonyme d'IoT. Il est important de noter que si la technologie RFID continue à être un facteur très important de ce phénomène (en particulier à cause de l'identité unique fournie par l'EPC), ce n'est certainement pas la seule technologie qui peut être utilisée pour la collecte de données [2].

En ce qui concerne IPV6, les ressources limitées des Objets, notamment en termes d'énergie, et l'environnement de connectivité intermittente (LLN : Low power Lossy Networks ") rendent difficile l'implémentation de cette technologie. Une des voies explorées aujourd'hui pour la communication dans les environnements LLN à énergie et connectivité faibles est l'adaptation d'IPv6 à ces environnements à travers une série de protocoles comme 6LowPAN et RPL [7].

Tableau résumant les trois services décris ci-dessus :

Service	Solutions proposées	Remarques
Confidentialité	- VPN - TLS - DNS - Onion routing - PIR - Contrôle d'accès - Cloud computing	-La confidentialité est nécessaire pour protéger les données sensibles. -Difficulté d'appliquer les solutions directement au contexte de l'IoT en raison de l'extensibilité et de contrôle d'accès. -Une politique de confidentialité doit être appliquée. -En général l'utilisateur privilégié du bénéfice du service face au risque de sa vie privée.
Authentification	- Mitiger les attaques par déni de service. - Utilisation les techniques de détection d'intrusions et d'authentification.	-Les limites de ressources dans l'IoT rendent difficile l'utilisation des algorithmes cryptographiques en raison de leur consommation en termes de calcul et de mémoire. - Des recherches sont en cours afin de rendre ces algorithmes peu couteux et robustes à la fois.
Identification	- RFID - 6LoWpAN - IPv6 - Identification biométrique	- L'identification est cruciale pour l'IoT afin de faire correspondre les services avec leur demande. - L'identification permet de connaitre l'identité d'une entité. - Une identification robuste et scalable jouera un rôle déterminant dans la sécurité de l'IoT. - Les ressources limitées de ces objets rendent aussi difficile l'implémentation de ces technologies.

TABLE 1.3 – Résumé des services décris en haut

1.4 Conclusion

L'IoT est sujet à de nombreuses menaces et attaques. Pour faire face à la plupart de ces menaces, les mécanismes de sécurité tels que la cryptographie et la signature numérique sont des moyens très puissants. Mais, pour assurer la sécurité dans l'IoT, les systèmes cryptographiques doivent être associés à une gestion de clé efficace.

Ce chapitre se composait de deux parties. La première a été consacrée à la présentation de l'IoT, ses domaines d'application ainsi que son fonctionnement. Puis, nous avons mis dans une deuxième partie, l'accent sur quelques concepts liés à la sécurité, où nous avons défini quelques mécanismes de sécurité et nous avons détaillé trois services de sécurité que nous avons a jugé importants.

Le chapitre suivant sera consacré à l'étude de quelques travaux récents réalisés dans le contexte de l'authentification et de l'établissement de clés dans l'IoT. Puis, nous citerons dans une deuxième partie quelques algorithmes d'élection.

Chapitre 2

Taxonomie des protocoles d'authentification pour l'IoT

2.1 Introduction

Plusieurs obstacles importants restent à combler pour la réalisation de la vision de l'IoT dont le principal est la sécurité. La plupart des études et recherches tendent à rendre leurs solutions applicables et utiles. Dans le domaine de la sécurité, les chercheurs ont proposé diverses solutions pour permettre des communications sécurisées entre les Objets.

Ce chapitre est divisé en deux parties. La première sera consacrée à la présentation, l'analyse et à la critique de quelques solutions d'authentification dans le but d'établir des clés secrètes dans l'IoT. Pour cela, nous commençons d'abord par déterminer nos critères d'analyse suivie par les protocoles en question et d'une classification des solutions étudiées. Par la suite, nous présenterons une description et une discussion de ces dernières. En fin de cette première partie, nous établirons une comparaison des travaux analysés dans un tableau. La seconde partie est une introduction sur le concept de l'élection et des algorithmes d'élection.

2.2 Protocoles d'authentifications étudiés

Les protocoles d'authentification ont chacun leur propre manière d'authentifier un utilisateur ou une machine. Ils utilisent différents algorithmes et différentes techniques. Cependant, ils ont tous quasiment le même principe de fonctionnement qui est à base de clés.

2.2.1 Critères de comparaison des solutions

L'établissement de critères s'impose pour une bonne étude critique des protocoles proposés pour l'authentification et la gestion de clé. Dans le but de mener une étude objective des solutions proposées dans la littérature, et en tenant compte des besoins et des contraintes spécifiques de l'IoT, cela nous permettra de cibler les défis auxquels nous ferons face lors de notre contribution.

Notre travail s'inscrit dans le cadre de la proposition d'une technique d'authentification qui consiste en un protocole de gestion de clé.

2.2.1.1 Résistance aux attaques

Les techniques de sécurité doivent résister tant que possible aux attaques. Plus la résistance est forte, plus les solutions sont meilleures.

2.2.1.2 Consommation énergétique

L'énergie est sans doute le facteur limitatif le plus important par rapport aux capacités d'un nœud capteur. La réserve énergétique limitée de chaque nœud capteur doit être gérée raisonnablement pour prolonger autant que possible sa durée de vie et ainsi que celle du réseau.

2.2.1.3 Complexité de mémoire de stockage

Un capteur ou un objet sont des petits dispositifs avec une mémoire très réduite et un espace de stockage limité. Donc pour construire un mécanisme de sécurité efficace, il est nécessaire de limiter la taille du code de l'algorithme de sécurisation.

2.2.1.4 Accord de clé

Un cryptage doit être défini à l'aide d'une clé de session partagée secrètement entre deux parties pour communiquer de manière sécurisée sur un réseau ouvert non sécurisé. Une clé de session est négociée en utilisant un protocole d'accord clé, dans lequel toutes les parties impliquées indépendamment et individuellement forment et influencent une clé.

2.2.1.5 Authentification mutuelle

Dans un système d'authentification, l'authentification mutuelle est d'une grande importance. Lorsqu'un utilisateur tente de se connecter à un réseau ou un serveur, il a besoin pour être sûr de la légitimité du serveur, car un adversaire pourrait se faire passer pour un serveur légitime en exécutant une attaque man-in-the-middle. En revanche, un serveur doit également être sûr de la légitimité de l'utilisateur.

2.2.1.6 Scalabilité

On dit qu'un réseau a une bonne scalabilité quand on peut facilement augmenter le nombre de nœuds ou d'Objets et qu'il n'y a aucun problème lors de cet ajout au réseau. Il il est plus intéressent que le réseau assure le bon fonctionnement quel que soit le nombre de nœuds.

2.2.2 Étude et critique de quelques protocoles d'authentification

Dans les protocoles étudiés, nous les classerons en deux catégories. D'une part, nous avons des systèmes qui utilisent la cryptographie symétrique ; qui utilise une clé commune pour le chiffrement et le déchiffrement. Cette clé est partagée entre l'émetteur et le destinataire et doit être communiquée par un des correspondants au second correspondant. D'autre part, d'autres systèmes qui utilisent la cryptographie asymétrique qui utilise une paire de clés dont l'une peut être rendue publique. La clé privée n'a pas besoin d'être communiquée entre les correspondants.

2.2.2.1 Protocoles basés sur la cryptographie symétrique utilisant une carte à puce

- **Protocole d'authentification et d'accord clé basé sur la cryptographie symétrique et une carte à puce**

S. Kumari et al.[28] ont proposé un système d'authentification avec accord clé basé sur la cryptographie symétrique, et carte à puce (sc), qui vise à préserver l'anonymat des utilisateurs. Le système proposé comprend également quatre phases : phase d'enregistrement, phase de connexion, phase d'authentification et phase de changement de mot de passe. Premièrement un utilisateur $\boldsymbol{U_i}$ désire d'être un utilisateur légal du système doit s'inscrire auprès du serveur de services $\boldsymbol{S_i}$, $\boldsymbol{Ui}$ choisit son identité $\boldsymbol{ID_i}$et un mot de passe $\boldsymbol{PW_i}$. Lorsqu'un utilisateur enregistré, $\boldsymbol{U_i}$ veut accéder aux services de , il insère sa carte à puce$\boldsymbol{SC_i}$ dans un périphérique terminal et saisit son identité $\boldsymbol{ID_i}$ et mot de passe $\boldsymbol{PW_i}$. Ensuite, $\boldsymbol{SC_i}$ effectue quelques étapes pour calculer la demande de connexion. Quand la réception de la demande de connexion le serveur $\boldsymbol{S_i}$ et $\boldsymbol{U_i}$ effectuent des étapes différentes pour une authentification mutuelle réussite et avoir une clé de session commune. La phase de changement de mot de passe est réalisée lorsque $\boldsymbol{U_i}$ souhaite remplacer son ancien mot de passe$\boldsymbol{PW_i}$par un nouveau, la carte $\boldsymbol{SC_i}$ décide si la demande de changement

de mot de passe est acceptable ou non.

Discussion et critique

Le régime proposé assure l'authentification mutuelle et est résiste à diverses attaques, y compris Attaque initiale, attaque de ***DoS*** et il fournit un accord clé de session, mais n'est pas complètement exempt d'attaques de sécurité, il souffre de l'attaque de supposition de mot de passe de carte à puce où un attaquant peut lancer avec succès une attaque de supposition de mot de passe de carte à puce et de la violation de l'anonymat ou un attaquant peut obtient l'IDi de l'utilisateur. Le protocole proposé est couteux en mémoire et en consommation d'énergie.

- **Un système d'authentification et d'accord clé basé sur une carte à puce et un nœud de passerelle pour l'IoT**

M. Turkanovie et al.[47] ont présenté un nouveau système d'authentification à deux facteurs et d'accord clé pour les ***WSN*** à l'aide d'une carte à puce, basé sur l'idée Internet of Things, le schéma proposé est basé uniquement sur la cryptographie symétrique, il utilise uniquement des calculs XOR et hache, il permet aux utilisateurs distants de négocier en toute sécurité une clé de session avec un nœud général de capteur. Le plan de Turkanovi'c et al. Se compose de quatre phases ; pendant la phase de pré-déploiement, chaque nœud de capteur régulier ***Sj*** est prédéfini avec son identité SID_j et une clé de mot de passe sécurisée générée au hasard X_{GWN}-S_j, qui est partagée avec le ***GWN*** nœud de passerelle et stocké dans la mémoire du nœud. Le ***GWN*** est prédéfini avec une clé de mot de passe hautement sécurisée générée de manière aléatoire ***Xgwn***.

Après le déploiement, deux phases d'enregistrements distincts sont nécessaires ; la première, est la phase d'enregistrement de la ***GWN*** et un nœud de capteur, leur but est que l'utilisateur s'inscrive auprès du ***WSN***, l'inscription se fait à l'aide d'une carte à puce, personnalisée par la ***GWN*** à la fin de la phase, lorsque l'utilisateur

est enregistré avec le ***WSN***, il peut ensuite se connecter à chaque nœud de capteur régulier sur demande et exécuter la phase d'authentification, la deuxième étape d'enregistrement est entre un utilisateur et le ***GWN***, leur but est d'enregistrer tous les nœuds de capteurs réguliers déployés avec la ***GWN*** et avec le réseau, après une inscription réussie, la ***GWN*** et le nœud de capteur régulier peuvent mutuellement s'authentifier et ainsi être certain de la légitimité de l'autre.

Après la phase d'enregistrement, l'utilisateur U_i peut se connecter à un nœud de capteur désiré du réseau en déclenchant une phase d'authentification, pour que la phase d'authentification soit initiée, l'utilisateur doit d'abord se connecter, le schéma utilise une carte à puce (SC) pour que l'utilisateur puisse s'inscrire et s'authentifier.

La phase d'authentification est déclenchée par l'utilisateur avec un message d'authentification à la fin d'une phase de connexion exécutée avec succès, l'utilisateur envoie le message d'authentification à un nœud de capteur souhaité du réseau et non la ***GWN***, le but de cette phase est de négocier une clé de session secrète entre l'utilisateur et le nœud de capteurs de manière à la fois contribuera individuellement à la clé de session avec un nonce secrètement choisi. Après avoir négocié avec succès la clé de session, ils peuvent l'utiliser pour communiquer en toute sécurité dans une affaire cryptée. Afin de parvenir à la négociation de clé de session sécurisée, une méthode d'accord de clé léger est proposée qui implique une authentification mutuelle entre toutes les parties.

Discussion et critique

Le schéma proposé est également sûr et léger, car il utilise la cryptographie symétrique, ce qui est hautement souhaitable par l'architecture à contraintes de ressources du WSN. Le système garantit des caractéristiques importantes, la protection par mot de passe, le changement de mot de passe et l'ajout de nœud dynamique. En outre, il offre une grande sécurité, car il est résilient pour reporter

les attaques d'adversaires privilégiées, infractions de cartes à puce, les attaques d'usurpation d'identité, les attaques d'identification quand plusieurs utilisateurs connectés avec la même connexion, les attaques de contournement ***GWN***, les attaques de changement de mot de passe et les attaques ***DoS***. Il assurant une meilleure performance au sein de l'environnement de ressources limitées.

Ce schéma n'est pas adapté aux applications pratiques en raison de faiblesses de sécurité où a quelques défauts de sécurité et il est susceptible à quelques attaques cryptographiques, il souffre d'une attaque par carte à puce et d'une attaque man-in-the-middle. En outre, le régime ne prévoit pas de non-traçabilité, ne fournit pas l'anonymat des nœuds, il ne résiste pas à l'attaque de la devinette d'identité, Attaque hors ligne de détection de mot de passe et attaque d'usurpation d'utilisateur, l'anonymat des utilisateurs, le système ne garantit pas l'authentification mutuelle entre toutes les parties.

Le système proposé consomme plus de stockage, où il nécessite plus d'espace de stockage à partir d'un ***GWN*** ou d'un nœud de capteur et beaucoup d'énergie environ. Ce protocole pose un problème de confidentialité lorsqu'un grand nombre de personnes est impliqué dans l'échange de clé (car il faudrait n (n - 1)/2 clés secrets). Le système ***GWN*** devait conserver la clé partagée de chaque nœud capteur enregistré qui signifie immédiatement que le nombre de nœuds utilisés dans le WSN est limité par la mémoire du ***GWN***.

- **Un système d'authentification et d'accord clé basé sur une carte à puce et un nœud de passerelle pour l'IoT (UAKAS)**

M. Farash et al.[16] ont proposé un système d'authentification d'utilisateur et d'accord-clé (UAKAS) nouveau repose sur le schéma de Turkanov'c et al., qui aborde et élimine toutes les lacunes et vulnérabilités de [47]. Le schéma proposé permet la même fonctionnalité, mais améliore le niveau de sécurité. La différence vitale et une amélioration de schéma proposé par Turkanovi 'et al. est le fait que les

clés partagées $\boldsymbol{X_{GWN}\text{-}S_j}$ peuvent être supprimées de la mémoire du $\boldsymbol{GWN}$ et de la $\boldsymbol{S_j}$ après la phase l'enregistrement réussi, car ils ne sont utilisés que pour cette phase.

Le schéma proposé permet à un utilisateur aléatoire de se connecter à n'importe quel nœud de détection spécifique du WSN et de conserver une communication sécurisée avec celui-ci. Le schéma proposé consiste en une étape de pré-déploiement, d'enregistrement, de connexion, d'authentification, de changement de mot de passe et de nœud dynamique. Les nouvelles phases d'enregistrement, de connexion, d'authentification et de changement de mot de passe, Les principes de la phase de pré-déploiement du schéma proposé sont les mêmes que ceux de Turkanovi 'et al. Pendant la phase de pré-déploiement, le nœud de passerelle GWN génère et stocke un mot de passe partagé $\boldsymbol{X_{GWN}\text{-}S_j}$ pour chaque nœud capteur $\boldsymbol{Sj}$. La clé partagée $\boldsymbol{X_{GWN}\text{-}S_j}$ est utilisée aux fins de la phase d'enregistrement, cette phase est requise pour une phase d'enregistrement réussie. Il est considéré comme une initiation du WSN. Chaque nœud de capteur est requis pour s'inscrire auprès du $\boldsymbol{GWN}$ après son déploiement dans le champ afin que le $\boldsymbol{GWN}$ authentifie et vérifie sa légitimité. Avant que n'importe quel utilisateur puisse se connecter avec un nœud de capteur spécifique, il doit s'inscrire auprès du WSN.

La phase d'inscription du schéma proposé est divisée en deux parties, phase d'inscription de l'utilisateur et phase d'inscription du nœud capteur, le processus d'enregistrement de l'utilisateur se fait via un canal sécurisé par le $\boldsymbol{GWN}$ et le résultat est un utilisateur enregistré avec une carte à puce. Avant que l'authentification puisse être démarrée, l'utilisateur Ui doit se connecter. Ceci est effectué hors ligne via la carte à puce SC. Après une connexion réussie la SC prépare le processus d'authentification, le but de la phase d'authentification est de permettre à un utilisateur de négocier une clé de session secrète avec un nœud capteur spécifique sans contacter le $\boldsymbol{GWN}$. La clé de session sera ensuite utilisée pour la communication sécurisée (demande, réponse) entre l'utilisateur et le nœud du

capteur et sera construite par les deux parties.

Discussion et critique

La différence vitale et l'amélioration de schéma proposé par Turkanovic et al. permet à un ***GWN*** d'ajouter de nombreux nœuds supplémentaires au réseau sans remplir sa mémoire, car il est continuellement libéré après l'avoir enregistré. Les auteurs ont montré que le schéma proposé fournit un accord clé, ainsi est résistant à une attaque d'usurpation d'utilisateur, à un problème d'anonymat d'utilisateur, une attaque répétée, une attaque initiale privilégiée, et une attaque de déni de service, l'attaque de carte intelligente volée, l'attaque de man-in-the middle.

Ce protocole n'assure pas l'authentification mutuelle, ne pouvait pas résister à une attaque de carte à puce, à une attaque de déconnexion de mot de passe hors ligne, à une attaque d'information temporaire spécifique et ne désactive pas la traçabilité des utilisateurs. L'architecture de réseau utilisée dans ce protocole est inefficace pour les environnements où l'énergie est un problème majeur. Le temps de fonctionnement total de ce protocole est 0.0128 ms, le coût total de calcul est 32 Th donc est couteux en mémoire et en termes de consommation d'énergie.

2.2.2.2 Protocoles basés sur la cryptographie asymétrique basés sur ECC

- **Un système d'authentification et d'établissement de clé utilise des certificats implicites (CA) basés sur ECC**

Les auteurs de [42], ont proposé un protocole d'authentification et d'établissement de clé pour les WSN dans les applications IoT distribuées basé sur ECC, utilisant un certificat implicite et le DTLS, la solution proposée s'inspire d'un système de certificats implicite ECQV et d'un mécanisme d'échange de clés ECDH. Le protocole d'authentification proposé contient deux phases ; la première phase est la phase d'enregistrement. Au cours de cette phase, le réseau

obtient des références de sécurité d'une partie de confiance ; la deuxième phase est la phase d'authentification, qui lance une communication sécurisée entre deux entités de réseau à l'aide des références de sécurité obtenues.

Le rôle de la phase d'enregistrement est d'établir une communication authentifiée, où les nœuds de capteurs et les utilisateurs finaux devraient posséder des certificats implicites pour des suites de chiffrage particulières, les nœuds de capteurs demandent des informations de sécurité et des certificats de l'autorité de certification ($\boldsymbol{AC}$). Cette autorité de certification délivre des certificats pour le nœud de capteur et vérifie la validité des certificats qu'elle a fournis. La phase d'authentification est effectuée entre un nœud de capteur et un serveur ou entre deux nœuds de capteurs, d'abord, le client envoie un message suites de chiffrement où les certificats implicites sont composés au serveur. Le serveur soit il accepte cette suite de chiffrement et répond par un message, soit il supprime la poignée de main.

Le client envoie son certificat accompagné d'une notation cryptographique aléatoire et de la valeur MAC. Si la vérification $\boldsymbol{MAC}$ est réussie, le serveur calcule la clé publique du client ($\boldsymbol{Qu}$) en utilisant le certificat reçu CertU et la clé publique ($\boldsymbol{QCA}$) de l'autorité de certification. Le serveur utilise sa clé privée dv et la clé publique du client Qu pour calculer un secret commun entre deux parties $\boldsymbol{Kuv = dvQu}$. En conséquence, le serveur envoie son certificat $\boldsymbol{Certv}$, nonce $\boldsymbol{Nv}$ et valeur $\boldsymbol{MAC}$. Semblable au côté du serveur, le client vérifie $\boldsymbol{MAC}$, calcule la clé publique du serveur et dérive la clé commune en utilisant sa propre clé privée et la clé publique du serveur $\boldsymbol{K_{uv}=d_u Q_v}$. Enfin, les deux nœuds de capteurs peuvent s'authentifier entre eux et établir une clé secrète commune et un lien de communication sécurisé qui peut être utilisé pour assurer d'autres acquisitions de données entre le client et le serveur.

Discussion et critique

Le schéma d'authentification proposé est léger et peut être facilement déployé

dans les dispositifs à ressources restreintes, ainsi qu'une sécurité raisonnablement élevée. En raison de la petite taille des certificats, il est possible de déployer dans les WSN hétérogènes compatibles avec IoT. En outre, le schéma d'authentification proposé prend en charge l'addition du nouveau nœud (il est scalable) et la mobilité des périphériques de bord et des utilisateurs finaux. L'authentification proposée est basée sur ECC, l'avantage de l'utiliser est qu'il fournit une sécurité égale pour RSA, mais avec moins de frais généraux. L'autorité de certification (***CA***) a été utilisée pour délivrer des certificats, après avoir obtenu leur propre certificat, les nœuds peuvent se déplacer et changer leur emplacement. ***CA*** peut valider l'identité des capteurs et communiquer avec d'autres entités du réseau. Le schéma proposé il offre une meilleure sécurité, il est sécurisé contre les attaques de déni de service (***DoS***), cependant, a toujours des limites. Par exemple, les attaques de capture de nœud peuvent créer une quantité limitée de dommages à l'ensemble du réseau. Le schéma proposé.

La performance du schéma proposé montre qu'il offre une consommation d'énergie et des frais généraux moins élevés, et qu'il consomme moins de mémoire sur chaque nœud capteur.

- **Mécanisme d'authentification des objets via un coordinateur (*C*) intégré à l'efficacité énergétique pour l'Internet des objets basé sur (*ECDH*)**

S. Patel, et al.[39] ont proposé un système d'authentification mutuelle basée sur la cryptographie a la courbe elliptique (ECC) et un modèle de contrôle d'accès basé sur la capacité pour l'IoT, le schéma proposé est basé sur la courbe elliptique Diffie-Hellman (ECDH) pour établir une clé secrète partagée entre deux nœuds. Le protocole EMA est décrit dans la figure 2.1.

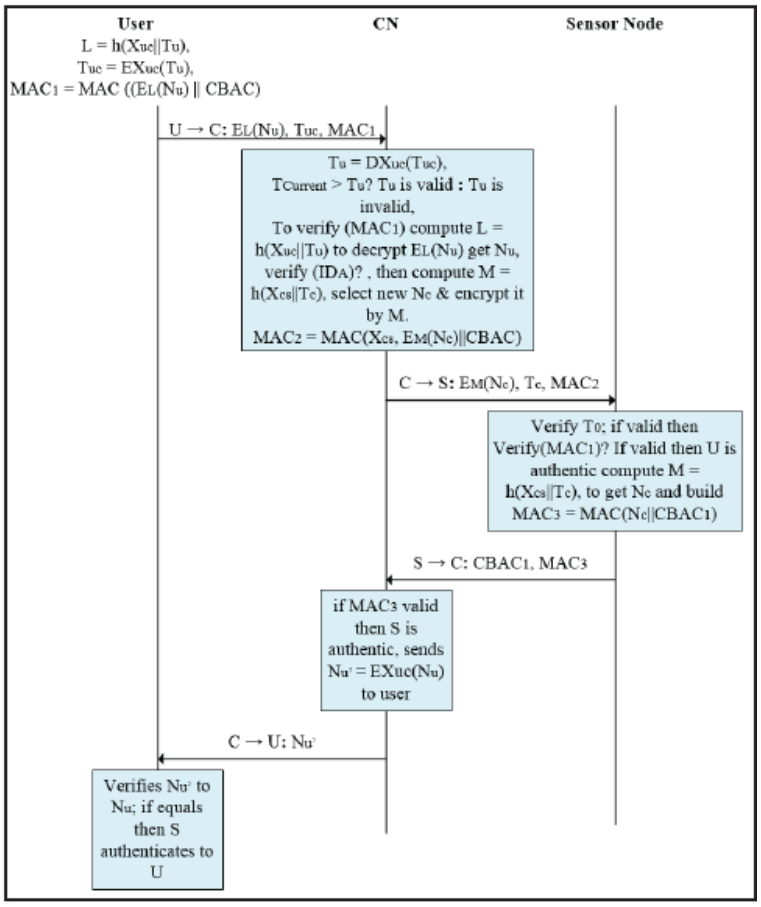

FIGURE 2.1 – le protocole EMA [39].

Discussion et critique

Le protocole proposé fonctionne bien avec la mobilité des objets hétérogènes, la topologie du réseau dynamique et les ressources limitées dans l'environnement IoT. Le protocole proposé est sécurisé contre plusieurs attaques, l'évaluation du protocole par l'outil AVISPA montre qu'il peut refuser efficacement les attaques telles que man-in-middle, DoS et attaque de replay.

Le protocole proposé est efficace sur le plan énergétique pour l'application des IOT, la méthodologie présentée fournit une solution ECC basée sur sa faible exigence de mémoire, sa grande évolutivité et sa facilité de déploiement dans l'environnement IoT. Il a moins de frais de communication au niveau du capteur pour obtenir l'authentification mutuelle des dispositifs avec une approche de contrôle d'accès basée sur les capacités.

- **Établissement une clé de groupe pour l'activation de la communication multidiffusion sécurisée pour les applications IoT basé sur ECC**

Les auteurs de [40] ont proposé un protocole de gestion clés pour assurer une communication de multidiffusion sécurisée pour l'IoT, le protocole est basé sur les opérations de ECC et s'appliquent aux scénarios de communication un-à-plusieurs (1 : n), il utilise les calculs XOR, hache et la signature numérique. Ce protocole exploite les fonctionnalités simplifiées dans ECIES et il crée des frais généraux inférieurs sur les nœuds capteurs en raison de moins de transactions de messages et d'un nombre réduit d'opérations au répondeur.

Trois étapes pour qu'une clé secrète partagée soit connue par l'initiateur et les autres membres du groupe de multidiffusion. Tout d'abord, l'initiateur $\boldsymbol{L}$ détermine l'ensemble des nœuds de capteurs par leur identité qui doit être inclus dans le groupe multicast particulier $\boldsymbol{U}$ et commence la communication. Il calcule la clé du groupe $\boldsymbol{k}$, et le code $\boldsymbol{Auth}$, puis il envoie un message de multidiffusion pour le groupe $\boldsymbol{U}$ avec une signature numérique. Lorsque le nœud capteur $\boldsymbol{Uj}$ reçoit le message de diffusion, il vérifie d'abord s'il est inclus dans le multiplexage $\boldsymbol{U}$ et vérifie la signature numérique. Si les deux sont correctement vérifiés, il calcule la clé $\boldsymbol{K}$, de même, tous les nœuds du groupe doivent effectuer les mêmes calculs pour dériver la clé de groupe. Alors $\boldsymbol{U_j}$ vérifie le code $\boldsymbol{Auth}$ et si est correctement vérifié, la clé de groupe $\boldsymbol{k}$ est authentifiée. Chaque nœud de capteur doit envoyer un message d'accusé de réception pour terminer la poignée de main. Plus tard, en vérifiant le message d'accusé de réception, l'initiateur peut assurer l'authenticité du

membre du groupe particulier et la dérivation précise de la clé de groupe $\boldsymbol{k}$.

Discussion et critique

Le protocole proposé assure l'authentification mutuelle, la confidentialité, d'intégrité, une communication sécurisée, une bonne robustesse, une bonne sensibilisation aux contraintes et une bonne évolutivité tout en générant la clé de groupe secrète. Les attaques de déni de service (***DoS***) sont également atténuées à cause de la signature numérique qui est utilisée pour chaque message transmis.
Le protocole proposé est efficace sur le plan énergétique, la consommation de mémoire est réduite, car aucune liste d'abonnés ne doit être conservée et le nombre de transactions de messages entre l'initiateur et un membre du groupe répondant est uniquement deux messages.
Le protocole proposé n'assure pas la mobilité des objets, ne s'applique pas aux scénarios de communication plusieurs à plusieurs (m :n), il s'applique uniquement aux scénarios de communication (1 :n).

- **Établissement de clé sécurisée simple et efficace basé sur ECC et utilise la technologie OpenID**

Le travail présenté dans [31] fournit une approche basée sur la Cryptographie de Courbe Elliptique (ECC) pour l'établissement de clés et le modèle RBAC pour définir les politiques de contrôle d'accès. La proposition utilise la technologie OpenID et les entités centrales configurables à des fins d'authentification. Le protocole proposé capable d'empêcher les utilisateurs non autorisés d'accéder aux données des nœuds capteurs sur la couche de perception IoT et de garantir efficacement la sécurité des données.

Les auteures ont proposé d'utiliser SKC ou PKC pour l'établissement des clés et la distribution, et comme les systèmes basés sur la PKC et PKC subissent de plusieurs problèmes comme (forte consommation d'énergie, d'un retard considérable

et la nécessité d'une grande mémoire pour stocker des matériaux clés), les auteurs ont cru que la solution ECC est solide pour être considérée.

La procédure de demande complète pour accéder à un "Objet" est montrée dans la figure 2.2.

Les trois étapes requises pour établir une clé de session pour deux entités sont : Tout d'abord, la RA qui est responsable de l'objet produira un $\boldsymbol{P} \in \boldsymbol{G}$ aléatoire et calculera $\boldsymbol{Ps = sP}$ dans Fp. Notez que le $\boldsymbol{S}$ est une clé secrète qui est supposée être assigné avant que la RA n'ait rejoint l'IoT. Pour chaque utilisateur avec IDu, RA générera $\boldsymbol{P_u}$ = h($\boldsymbol{ID}$) et la clé privée de la chose $\boldsymbol{Su = sPu.}$ Deuxièmement, l'utilisateur génère une clé privée éphémère a et calcule $\boldsymbol{Qu = aSu}$ et $\boldsymbol{Qu'= aP}$. Puis l'utilisateur enverra un message d'authentification $\{\boldsymbol{IDu}, \boldsymbol{Qu}, \boldsymbol{h\ (IDu \| IDt \| Qu \| Qu')}\}$ à la RA. Une fois que RA reçoit le message, RA calculera $\boldsymbol{Qu" = s\ Qu^{-1}}$ et vérifie si $\boldsymbol{h(IDu \| IDt \| Qu \| Qu")}$ égal à $\boldsymbol{h(IDu \| IDt \| Qu \| \ Qu')}$ ou ne pas. Sinon, l'authentification échoue. Sinon, passez à la troisième l'étape. La troisième étape est l'établissement de la clé de session. De même, la RA choisira une clé éphémère aléatoire $\boldsymbol{b}$ et calculera $\boldsymbol{Qt = bP}$ pour l'"Objet" souhaité. La clé de session sera $\boldsymbol{h(abP)}$ en fonction de l'algorithme ECC.

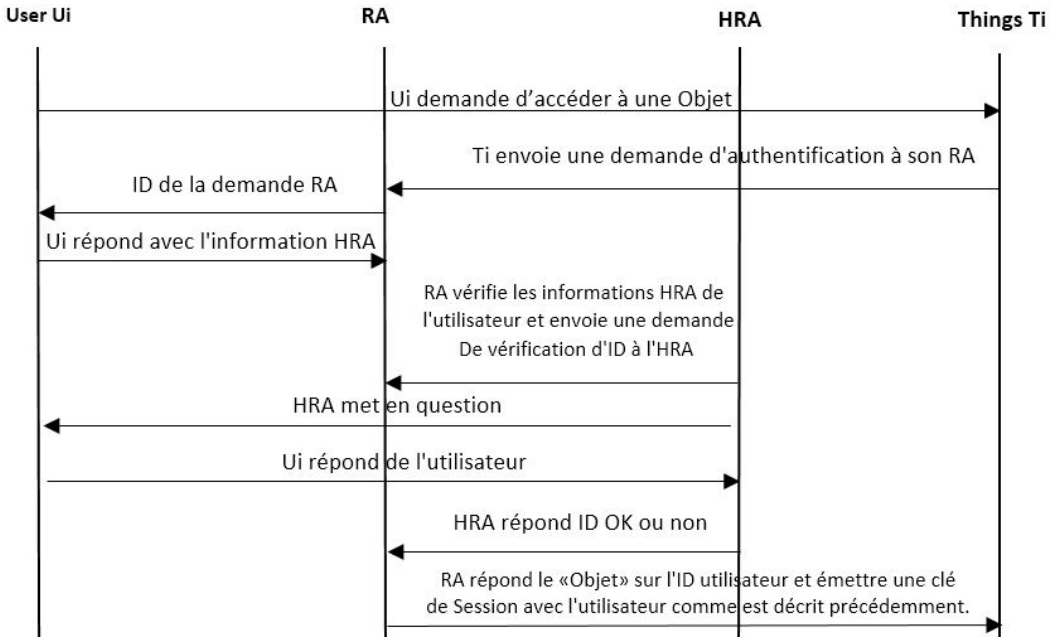

FIGURE 2.2 – La procédure de demande d'accéder à un Objet

Discussion et critique

Le protocole proposé est sûr. Il offre l'anonymat des utilisateurs, et l'établissement de la clé de session sécurisée. Les résultats de l'analyse montrent cette approche peut empêcher les attaques comme l'écoute, l'attaque de man-in-the middle et les attaques de répétition, donc ce protocole est sécurisé de diffère attaques, mais l'évaluation de la sécurité est pratiquement impossible pour l'environnement IoT.

Dans ce protocole l'authentification mutuelle n'est pas traitée, il ne peut pas résister pas au attaque DoS, ce protocole est coûteux pour les nœuds de capteurs dans l'IoT où a un coût élevé lors de l'échange de messages, couteux en mémoire et couteux en termes de consommation d'énergie, car il est basé sur l'algorithme ECC qui est également plus complexe et plus difficile à mettre en œuvre que d'autres

algorithmes cryptographiques et nécessite plus de ressources de traitement. Comme il est limité, car il ne supporte qu'un seul modèle de contrôle d'accès, ce qui impose des restrictions aux applications (SP).

- **Protocole d'authentification basé sur *ECCDH* utilise *KDC***

L'article de Parikshit N. et al.[33] présente un protocole d'authentification mutuelle et est divisé en trois parties ; génération de clé secrète basée sur l'algorithme de cryptographie de courbe elliptique-Diffie Hellman (ECCDH), établissement d'identité, création de capacités pour le contrôle d'accès. Le protocole utilise un ou plusieurs centres de distribution de clés (KDC) de confiance pour générer des paramètres de domaine et d'autres éléments de sécurité, et la partie importante est que ce KDC n'est pas nécessairement toujours en ligne. Le protocole est décrit ci-dessous.

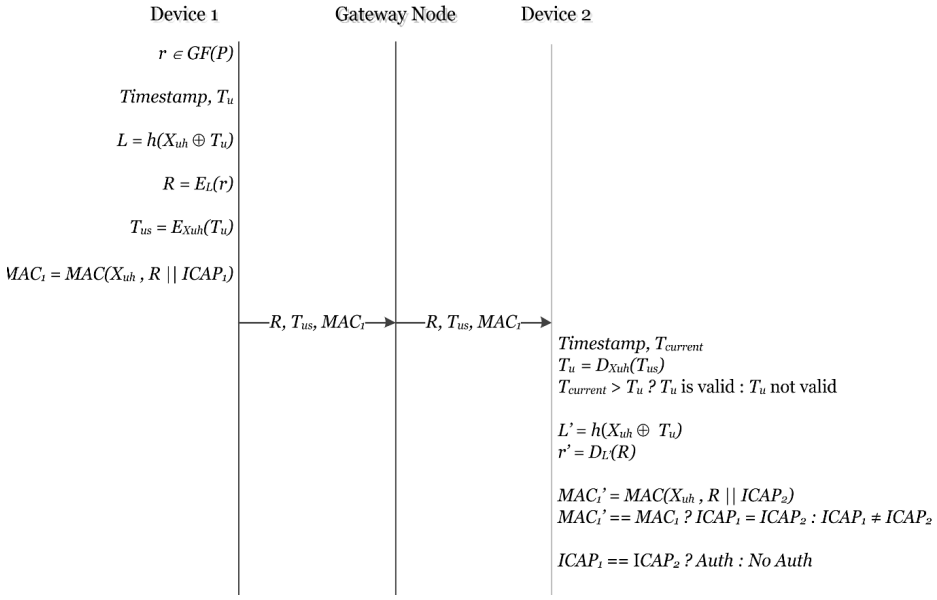

FIGURE 2.3 – One way authentication protocol [33].

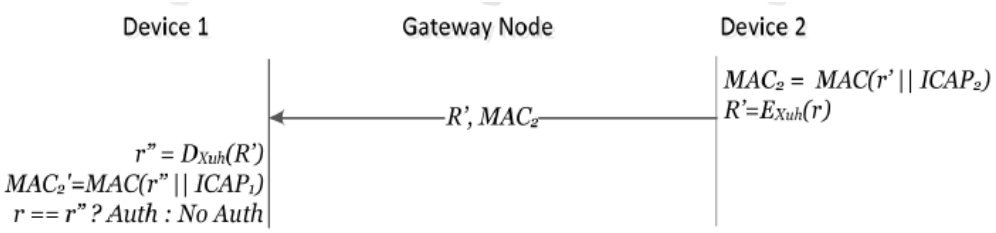

FIGURE 2.4 – Protocol for mutual authentication [33].

Discussion et critique

Il utilise une approche à clé publique compatible avec la limitation d'IOT. L'approche introduit également une technique d'horodatage pour authentifier les messages envoyés entre les périphériques afin d'interdire les attaques de l'homme dans le milieu et de servir de MAC. En outre, le modèle utilise un algorithme RC5 pour chiffrer les données entre les périphériques communicants. Le modèle proposé vise à tirer parti de l'établissement de l'identité mutuelle dans IoT en comblant les lacunes existantes dans les capacités d'authentification et de contrôle d'accès avec le protocole intégré.

Lorsqu'on considère les grandes données multimédias, cette approche souffre de frais généraux supplémentaires en raison de la génération, de la mise à jour et de la distribution des clés. Les auteurs ont basé leur modèle de contrôle d'accès suggéré sur la capacité et l'identité des périphériques. Le système de contrôle d'accès basé sur l'authentification d'identité et la capacité (IACAC) crée la capacité en fonction de l'identité pour accorder l'accès sur le réseau local. Ce schéma n'est toujours pas adapté aux petits appareils au sein de l'IoT. Ce protocole est vulnérable aux attaques man-in-the-middle et l'attaque replay, l'adversaire peut intercepter un message envoyé.

- **Une méthode efficace d'authentification basée sur ECC pour la couche de perception**

Une méthode efficace d'authentification mutuelle était proposée par YE Ning et al.[51] établissant des paires de clés public-privé, basée sur la cryptographie de courbe elliptique (ECC) pour la couche de perception de l'IoT. Cette méthode peut confirmer l'identité des deux côtés de la communication et établir une clé de session. L'authentification comprend deux phases ; une phase d'initialisation, authentification mutuelle où l'utilisateur doit négocier avec BS l'information secrète utilisée pour l'authentification lorsqu'il accède au réseau du capteur. Et une phase d'établissement de clés. Le schéma suivent décrit les différentes étapes.

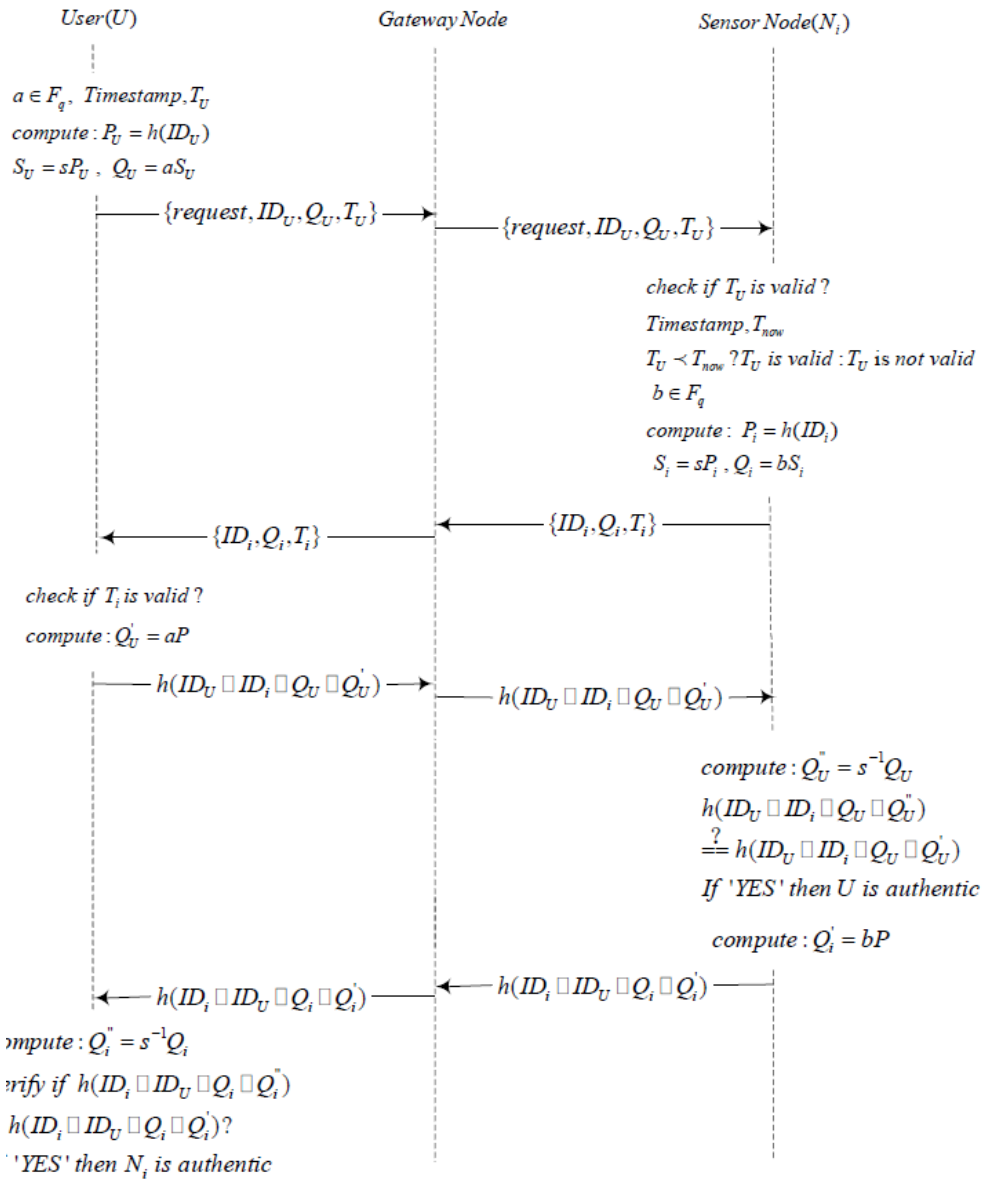

FIGURE 2.5 – Mutual Authentication Key Establishment Phase [51].

Si un attaquant malveillant obtient une clé de session ou capture le trafic réseau d'IoT, la clé de session peut l'identifier en tant que malveillants. Le message de

renvoi sera rejeté en raison de l'identité illégale.

Discussion et critique

L'auteur dit que son authentification résiste aux attaques suivantes ; men in the middle, l'effraction, capture de nœud, Dos, attaques de répétition. Les algorithmes cryptographiques traditionnels ne peuvent pas atteindre le poids léger, le manque d'authentification mutuelle entre les utilisateurs et les nœuds n'est pas adapté à l'environnement ouvert d'IoT. L'authentification mutuelle garantit la sécurité de la communication entre les utilisateurs et les nœuds, dont le processus est simple pour résoudre le problème limité de ressources de la couche de perception IoT. ECC offre de meilleures performances, car il peut utiliser une taille de clé plus petite pour obtenir la même sécurité.

Dans cet article, l'utilisateur déclenche une requête et le nœud complète l'authentification de l'utilisateur pour éviter les attaques malveillantes (étape 4), tandis que dans l'étape 5, l'utilisateur authentifie le nœud. Grâce à l'authentification mutuelle, les relations de confiance entre l'utilisateur et les nœuds seront établies.

- **protocole d'authentification basé sur le hashing et l'extraction des fonctionnalités**

Le protocole de Guanglei Zhao et al.[53] est basé sur l'algorithme de hachage SHA, l'extraction de fonction et la cryptographie de la courbe elliptique ECC. Le modèle proposé combine la méthode de la fonction hasch et l'extraction des fonctionnalités. La quantité d'information initiale peut être réduite grâce à l'extraction de fonctionnalités, elle peut transformer les données d'entrée en ensemble de fonctions afin d'effectuer la tâche souhaitée à l'aide de cette représentation réduite au lieu d'utiliser toutes les données. De plus, cette extraction est irréversible. Ici la clé privée n'est pas stockée dans le nœud, mais est considérée comme résiduelle et est complétée dans le processus d'accord de clé. En plus d'une deuxième clé pour

crypter et décrypter la clé résiduelle. Un nombre RT aléatoire est généré pour éviter les attaques de répétition.

Discussion et Critique

Les auteurs ont présenté un schéma d'authentification mutuelle pour IoT entre les plates-formes et les nœuds terminaux. Le schéma est basé sur le hashing et l'extraction des fonctionnalités. L'extraction des fonctions a été combinée avec la fonction hasch pour éviter toute attaque de collision. Ce système fourni en fait bonne solution pour l'authentification dans IoT. Le processus d'extraction des caractéristiques a les propriétés de l'irréversibilité nécessaire pour assurer la sécurité et est léger qui est souhaitable dans IoT. Le schéma se concentre sur le processus d'authentification lorsque la plate-forme essaie d'envoyer des données aux noeuds terminaux et non pas le contraire. Bien que le système améliore la sécurité tout en réduisant la quantité d'information réduite, cela ne fonctionne que sur la théorie et il n'y a pas de preuve pratique pour l'appuyer.

2.2.3 Classification des protocoles étudiés

Nous classifions les protocoles que nous avons étudiés en deux niveaux. Le premier niveau représente la catégorie de la solution proposée à base de la cryptographie symétrique utilisant une carte à puce et le deuxième niveau à base de la cryptographie asymétrique basée sur ECC. La figure 2.6 montre notre classification.

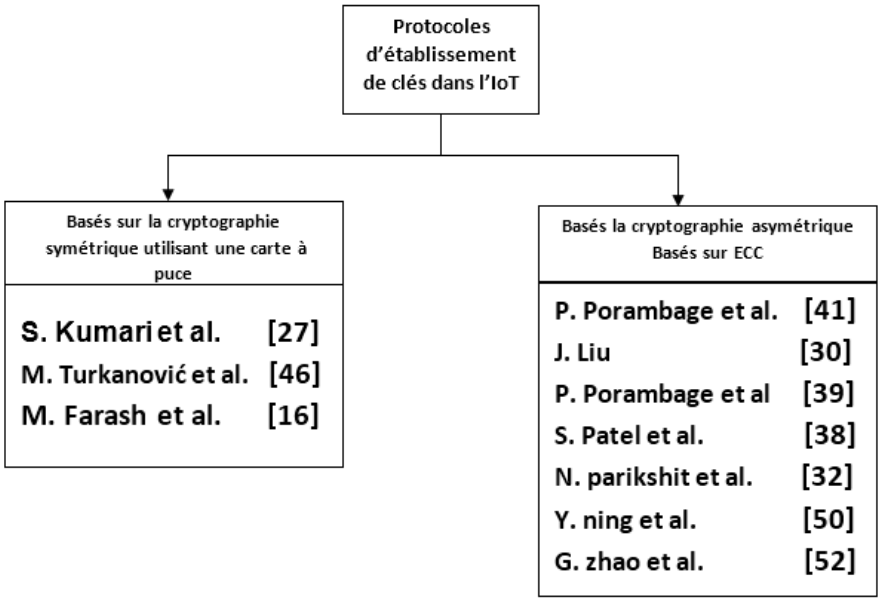

FIGURE 2.6 – Classification des protocoles étudiés pour l'établissement de clé.

2.2.4 Comparaison des protocoles étudiés

Dans cette partie, nous allons faire une comparaison entre les différents protocoles de gestion de clés dans l'IoT étudiés, dans ce chapitre on se basant sur les métriques d'évaluations citées ci-dessus. Le tableau 2.1 illustre la comparaison entre les travaux présentés précédemment.

		Résistance aux Attaques				Scalabilité	Accord de clé	Optimisation de la consommation d'énergie	Complexité de calcul et mémoire de stockage	Anonymat	L'authentification mutuelle
		DOS	man-in-the middle	répétition	Supposition de mot de passe						
basé sur la cryptographie symétrique utilise carte à puce	**M. Turkanović et al. [10]**	Oui	Non	/	non	Oui	Non	Non	19Th 768 bits	Non	Oui
	S.kumari et al. [45]	Oui	/	/	Non	/	Oui	Non	16th + 36t⊕ 768 bits	Non	Oui
	Farash et al. [18]	Oui	Oui	Oui	Non	Oui	Oui	Non	32*Th* 512bit	Oui	Non
basé sur la cryptographie asymétrique basé sur (ECC)	**S.patel et al. [31]**	Oui	Oui	Oui		Oui	Oui	Oui	2TMac +Th +TRC5		Oui
	P.porambage et al. [39]	Oui	/	/		Oui	Oui	Oui	(n+5) PM + (n+2) PA		Oui
	J.liu et al. [7]	Non	Oui	Oui		/	Oui	Non	Oui	Oui	Non
	N.parikshit et al. [G]	Non	Non			/	/	Oui	2Dh+2DMac+2 DRCS		Non
	Y.ning et al [C]	Oui	Oui	Oui		/	/	Non	R+Tnul+2Th		Oui
	G.zhao et al. [ICM]	Oui	Oui	Oui		/	/	Non	Oui		Non
	P. porambage et al. [50]	Oui	Non	/		Oui	Oui	Oui	ROM 22.875 kB RAM 2.871 kB		Oui

Table 2.1 -Tableau comparatif entre les protocoles de gestions de clé

Oui, signifie que le protocole assure la fonction ou résout le problème de manière optimisée, ou peut résister à l'attaque contrairement au Non.

2.3 L'élection

L'IoT est un système distribué composé d'un grand nombre d'objets intelligents (ordinateurs, smartphone, etc.). Ces objets peuvent fonctionner ensemble, mais tous les systèmes sont indépendants. L'élection d'un leader est une nécessité fondamentale pour les systèmes distribués. Lorsqu'un système est choisi comme leader, il devrait fonctionner comme un système de gestion ; prenez des décisions finales et autres. Il existe plusieurs algorithmes électoraux disponibles dans le système distribué.

2.3.1 Introduction sur l'élection

Le problème d'élection est l'un des grands paradigmes de l'algorithmique distribuée. Il a été présenté pour la première fois par LeLann [52]. Ce problème est de partir d'une configuration dans laquelle tous les processus sont dans le même état pour arriver dans une configuration dans laquelle un seul processus est dans l'état "gagnant" et tous les autres dans l'état "perdant" [9].

L'opération d'élection consiste à allouer le processeur à un processus[1]. leur but est de choisir un élément d'un ensemble ; cet élément est appelé élément élu. Il peut être utilisé pour prendre des décisions dans un système réparti ; il peut également centraliser des informations [52].

Donnons deux exemples de l'intérêt de l'élection [45] :

– Dans une application de type maître-esclave, tous les serveurs sont redondants pour le traitement des requêtes de lecture des clients. Cependant les requêtes d'écriture ne sont traitées que par un seul serveur (le maître) qui transmet ensuite les modifications aux autres sites (les esclaves). On pourra, par

exemple, imaginer une gestion de mots de passe qui permet d'ouvrir une session sans passer par le réseau.

– Certaines applications nécessitent une phase d'initialisation exécutée par un seul site. L'exemple le plus simple est celui de la circulation d'un jeton unique entre les sites. Dans ce cas l'initiateur est le premier possesseur du jeton.

2.3.2 Algorithmes d'élection

Un algorithme d'élection est un algorithme qui satisfait les trois propriétés suivantes [9] :

– chaque processus exécute le même algorithme : symétrie complète,
– l'algorithme est décentralisé : une exécution peut être commencée par un nombre quelconque de processus,
– l'algorithme atteint une configuration terminale dans laquelle il existe exactement un processus "gagnant" et tous les autres processus sont "perdants".

La dernière propriété est quelquefois relâchée en : il existe un seul processus gagnant. Le processus gagnant est alors au courant qu'il a gagné l'élection, mais les autres ne savent pas encore qu'ils ont perdu. Dans ce cas, le processus gagnant diffuse le résultat aux autres processus. Dans les algorithmes que nous étudions, chaque processus possède un nom unique qui peut être comparé ($=$, $<$, $>$, $\leq$, $\geq$) avec les identifiants des autres processus du système réparti. Enfin, chaque processus possède une variable statep pouvant contenir les valeurs gagnant et perdant. Certaines fois, la variable statep est égale à dormant avant que le processus n'ait exécuté une étape de l'algorithme et candidat lorsque le processus participe à l'algorithme, mais ne sait pas encore s'il est gagnant ou perdant [9].

2.3.2.1 Élection sur un anneau unidirectionnel

- **Algorithme de LeLann (1977) [29].**

Algorithm 1 Algorithme de LeLann

VAR
Listep sous-ensemble de P init {p}
$etat_p$ (init, leader , perdu, sleep) ;
DEBUT
Si (p est intiateur) alors
$etat_p$:= init ; envoie < tok, p > à $Suivant_p$;
reçoit < tok, q > ;
tant que $(q \neq p)$ **faire**
$Liste_p := Liste_p \, \epsilon\{q\}$;
envoie < tok, p > à $Suivant_p$;
reçoit < tok, p > ;
si p = max ($Liste_p$) **alors** $etat_p$:= leader **else** $etat_p$:= perdu **Fsi**
Fin tant que
Sinon
tant que vrai **faire**
reçoit < tok, q > ;
envoie < tok, q > à $Suivant_p$;
si $etat_p$ = sleep **alors** $etat_p$:= perdu ; **Fsi**
tant que
Fsi ;
FIN.

Complexité de l'algorithme

Cet algorithme possède une complexité en nombre de messages en $O(n^2)$, avec n le nombre de processus. Chaque initiateur calcule l'ensemble des identifiants des processus de l'anneau, l'initiateur avec l'identifiant le plus petit est "gagnant". Chaque initiateur p émet un message <tok ,p> contenant son identifiant, le message <tok, p> est propagé sur l'anneau. Les canaux sont supposés FIFO et un initiateur p doit générer son message <tok, p> avant qu'aucun autre message <tok,q> ne soit

reçu. Lorsqu'un initiateur p reçoit son message <tok,q>, les messages <tok,q> de tous les autres processus ont visité p. p peut alors connaître l'identifiant du processus "gagnant". Lorsqu'un processus reçoit un message <tok,q>, s'il veut participer à l'élection, il commence à exécuter l'algorithme en se considérant comme un initiateur [29].

- Nombre de messages : $O(n^2)$ (N initiateurs * N processus dans l'anneau).
- Temps : 2N - 1.

- **Algorithme de Chang et Robert (1979) [29].**

Cet algorithme s'applique à un anneau unidirectionnel. Chaque site n'émet des messages que vers le site suivant sur l'anneau.

Algorithm 2 Algorithme de Chang et Roberts

```
VAR
etat_p
DEBUT
    Si (p est intiateur) alors
      etat_p := cand ; envoie < tok, p > à Suivant_p
        tant que (etat_p ≠ leader) faire
          reçoit < tok, q > ;
            Si (q = p) then etat_p := leader
            sinon
             Si (q > p) alors
                Si etat_p = cand alors etat_p := perdu
               envoie < tok ; q > à Suivant_p
                FinSi
              FinSi
            FinSi
        Ftant que
    Sinon tant que vrai faire
      reçoit < tok ; q >
      envoie < tok ; q > à Suivant_p
      Si etat_p = sleep alors etat_p := perdu FinSi
    FinSi
FIN.
```

Complexité de l'algorithme

- Nombre de messages : $O(n^2)$.
- Temps : 2N - 1.

2.3.2.2 Élection sur un anneau bidirectionnel

- **Algorithme de Franklin (1982) [45]**

Cet algorithme s'applique à un anneau bidirectionnel. Il se décompose en "tours" où les compétiteurs sont les initiateurs (au même sens que dans l'algorithme précédent). À chaque tour, un initiateur survivant envoie à ses initiateurs voisins (à gauche et à droite) sa candidature. De ce fait, chaque initiateur reçoit les requêtes des initiateurs voisins gauche et droite et ne survit au tour suivant que s'il a la plus petite identité. Le tournoi s'achève quand un initiateur sait qu'il est où sera au prochain tour le seul survivant. Il acquiert cette certitude soit en recevant l'un des messages qu'il a envoyés (ou éventuellement les deux) soit en recevant deux messages d'un même site. Les sites "éliminés" participent à l'algorithme en transmettant les messages qu'ils reçoivent. Comme dans l'algorithme précédent un message de confirmation achève l'algorithme.

Variables du site i

- $suivant_i$: constante contenant l'identité du site successeur de i sur l'anneau.
- $precedent_i$: constante contenant l'identité du site prédécesseur de i sur l'anneau.
- $etat_i$: état du service. Cette variable prend une valeur parmi "repos, en cours, attente, terminée". Cette variable est initialisée à repos.
- $nbreq_i$: nombre de requêtes reçues au cours d'un tour.
- $conc_i$: identité du site dont on reçoit la première des deux requêtes du tour courant. Lorsqu'elle vaut i, cela signifie qu'aucune requête en avance n'est reçue.

- dir_i : booléen indiquant la direction d'où vient la première des deux requêtes du tour courant.
- $concav_i$: identité du site dont on reçoit la première des deux requêtes du tour suivant.
- $chef_i$: identité du site (provisoirement) élu.

Algorithme du site i

Si aucun processus d'élection n'a pas atteint le site i, le service initialise un tel processus. Au cours de ce processus, il propose sa candidature dans les deux sens de l'anneau tant qu'il n'est pas éliminé ou qu'il ne sait pas qu'il est le seul survivant. Dans tous les cas, il attend que le processus d'élection soit terminé pour renvoyer l'identité du site élu. La boucle répéter correspond à la participation du site aux tours successifs. À chaque changement de tour, il faut prendre garde à traiter l'éventuelle requête en avance.

Algorithm 3 Algorithme de Franklin ***leader()***

DEBUT
Si ($etat_i$ == repos) **Alors** $etat_i$ = en cours ;
Répéter
nbreqi=0 ;
Si ($concav_i \neq$ i) **Alors** nbreqi=1 ; $conc_i$=$concav_i$;
Si ($conc_i$< $chef_i$) **Alors** $chef_i = conc_i$; **FSi**
$concav_i$;
Finsi
envoyer à ($suivant_i$,(req, i)) ;
envoyer à($precedent_i$,(req, i)) ;
Attendre($nbreq_i$==2) ;
Jusqu'à ($etat_i$ = en cours) ;
Si $concav_i \neq$ i **Alors**
Si (dir_i) Alors
envoyer à($suivant_i$,(req,$concav_i$)) ;
Sinon
envoyer à($precedent_i$,(req,$concav_i$)) ;
Finsi
Finsi
Finsi Attendre($etat_i$==terminé) ; renvoyer($chef_i$) ;cours ; $chef_i$=i ; $concav_i$=i ;
FIN.

À la réception d'une requête, on met à jour si nécessaire l'identité provisoire du leader. Dans le cas où on est un initiateur survivant (état à en_cours), on enregistre les deux requêtes voisines en prenant garde à mémoriser une requête en avance. Sur la réception de la deuxième, on teste si on a survécu puis si on est le seul survivant. Dans le cas où on est en attente, on retransmet les messages (fonction proxy).

Algorithm 4 Algorithme de Franklin ***sur_réception_de(j,(req,k))***

```
DEBUT
  Si (etat_i == repos || k< chef_i) Alors
    chef_i = k ;
  Finsi
  Si (etat_i == en cours) Alors
    Si (nbreq_i==0) Alors
      conc_i=k ; nbreq_i=1 ; dir_i = (j== precedent_i) ;
    Sinon
      concav_i=k ;
    Sinon
      nbreq_i=2 ;
      Si ( chef_i < i) Alors etat_i = attente ;
      Sinon
        si (conc_i==i || k==conc_i) Alors
          etat_i = terminé ; envoyer à(suivant_i,(conf,i)) ;
      Finsi
    Finsi
  Sinon
    etat_i=attente ;
    Si (j==precedent_i) Alors envoyer à(suivant_i,(req,k)) ;
    Sinon
      envoyer à(precedent_i,(req,k)) ;
    Finsi
  Finsi
FIN.
```

La confirmation du site élu fait un tour de l'anneau.

Algorithm 5 Algorithme de Franklin ***sur_réception_de(j,(conf,k))***

```
DEBUT
  Si (i ≠ k) Alors
    envoyer à(suivant_i,(conf,k)) ; etat_i = terminé ;
  Finsi
FIN.
```

- **Algorithme de Dolev, Klawe et Rodeh [29].**

Dans ce cas, Dolev, Klawe et Rodeh ont mis au point un algorithme d'élection de leader. L'idée est de choisir parmi les différentes stations celle dont l'id est le plus grand. Initialement, chaque station est dans état actif (càd participant au protocole). Dés qu'une station découvre qu'elle n'est pas celle dont l'id est le maximum global, elle passe en mode passif, et n'agit plus alors que comme un relai pour les messages. L'algorithme utilise pour chaque station deux variables entiéres : left et max, ainsi que deux types de messages : <1, i> et <2, i> ou i est un entier. Notons que la variable max contient initialement l'identificateur de la station. L'algorithme, tel que présenté dans l'article original.

Hypothèses et principe

- Hypothèses de l'anneau.
- Calcul du plus grand/petit numéro de processus.
- Initialement tous les processus sont actifs à chaque round chaque processus compare son numéro avec ses voisins.
- Si son numéro est plus petit que l'un de ses voisins (sens de la montre et opposé), passe à l'état passif.
- A moins la moitié des identités actives ne survivent pas à un round ! après au plus logN round l'élection est finie.
- MAIS : nécessite un anneau bidirectionnel ce qui n'est pas dans les hypothèses de départ.

Principe pour un anneau unidirectionnel

- Les processus passifs ne font que relayer les messages, quelque soit leur type.
- Un processus actif envoie son numéro au prochain processus actif (relayé par les processus passifs) dans un message (one).

– Il obtient donc, dans le premier message reçu, le numéro du processus qui le précède.
– Il envoie ensuite un second message < two > avec le numéro reçu.
– Son successeur reçoit ainsi le numéro du processus actif qui le précède à distance 2.

2.3.2.3 Élection sur réseau complet

- **Algorithme de Bully (Garcia-Molina) [29].**

Déclenchement

– Quand un processus P s'aperçoit que le coordinateur ne répond plus à ses requêtes (time-out sur TEMPO), il lance l'algorithme d'élection.

Lancement de l'élection par P

– P Envoi d'un message ELECTION à tous les autres processus dont le numéro est plus grand que le sien Réception d'un message ELECTION depuis P par un processus Q.
– Le processus Q envoie un message ACK à P lui signant qu'il est actif.
– A son tour Q, lance une élection si ce n'est pas déjà fait.

Sur le processus P

– Si aucun processus ne lui répond avant TEMPO, P gagne l'élection et devient le coordinateur.
– Si un processus de numéro plus élevé répond, c'est lui qui prend le pouvoir. Le rôle de P est terminé.

Annonce de l'élu

– Le nouveau coordinateur envoie un message à tous les participants pour les informer de son rôle. L'application peut alors continuer à s'exécuter Réveil d'un processus inactif.
– Déclenche une élection.

– S'il détient le plus grand numéro de processus en cours de fonctionnement, il gagne l'élection et devient le nouveau coordinateur.

2.3.2.4 Élection sur un arbre

- **Algorithme général d'élection 1 [52].**

Dans cette section, nous donnons un algorithme permettant de guider localement l'élection globale. L'algorithme exécuté par chaque sommet v est paramétré par une valeur ($\lambda(\mathbf{v})$) déterminée localement par **v** en utilisant une fonction **F**. L'algorithme utilise aussi une routine attente ?exponentielle ($\lambda(v)$) ; cette routine génère un temps d'attente **T** suivant une distribution exponentielle de paramètre ($\lambda(v)$), la valeur de **T** est donc un réel valant : $\frac{-1}{(\lambda(v))}\, ln(x)$ où x est un réel pris uniformément dans l'ensemble [0,1].

Algorithm 6 Algorithme d'élection aléatoire dans un arbre

Var
$Neigh_v$: ensemble de sommets (voisins de v) ;
rec_v [w] pour chaque w ϵ $Neigh_v$: booléen initialisé à faux ;
(rec_v [w] est vrai si v reçoit un message de w) ;
$\lambda(v)$: le paramètre de la durée de vie, sa valeur initiale est la même pour tous les sommets ;
Tant que w : recv [w] est faux > 1 **faire**
receive < tok, $\lambda(v)$ > from w ;
$\lambda(v)$:= F ($\lambda(v)$, $\lambda(w)$) (F est une fonction quelconque) ;
rec_v[w] := vrai ;
Ftant que
(À présent, il y a un seul sommet w0 avec rec_v [w0] = f aux ; v est alors une feuille) ;
attente - exponentielle($\lambda(v)$) ;
Si un message < tok, p > est arrivé alors $etat_v$:= elu ;
Sinon
Début
$etat_v$:= perdant ;
envoyer < tok, $\lambda(v)$ > à w0 tel que rec_v [w0] est égal à faux ;
Fin ;
Finsi
FIN.

Chaque sommet v a un paramètre ($\lambda(v)$) et une liste recv [ ] de booléens. Chaque élément recv[w] indique si oui ou non v a reçu un message du voisin w. À la réception d'un message $< tok, (\lambda(w)) >$ d'un voisin w, v met à jour son paramètre ($\lambda(v)$) en appliquant la formule ($\lambda(v)$) = F(($\lambda(v)$), ($\lambda(w)$)). Le choix de la fonction F est l'élément déterminant dans l'orientation de l'élection et la distribution de probabilité qui en résulte.

Le processus d'élection est modélisé par un processus de mort à temps continu comme suit : chaque sommet v qui devient une feuille génère une durée de vie selon une loi exponentielle de paramètre ($\lambda(v)$), ce paramètre étant déterminé au moment où v devient feuille. Ce qui est équivalent au fait que si v devient feuille à l'instant

t, alors, la probabilité que la mort de v survienne pendant l'intervalle [t, t + h[est donnée par h($\lambda(v)$) + o(h) quand h tend vers 0 et ceci indépendamment de ce qui s'est passé auparavant ou ailleurs dans l'arbre. Une fois le temps de vie écoulé, la feuille disparaît et transmet son poids au sommet père dans l'arbre. Le processus continue ainsi jusqu'à ce que l'arbre soit réduit à un seul sommet. Ce modèle général laisse libre le choix de la valeur initiale de ($\lambda(v)$) et de la fonction F. La seule contrainte (imposée par le fait que l'on désire avoir un algorithme distribué) est que ($\lambda(v)$) soit déterminée de manière locale quand v devient une feuille. L'idée intuitive de l'algorithme est de contrôler le processus d'élection en contrôlant la valeur de ($\lambda(v)$). Une remarque simple illustre ce propos : si ($\lambda(v)$) augmente, alors la probabilité pour v d'être élu diminue.

- **Algorithme général d'élection 2 [52].**

Principe

- Un (ou plusieurs) processus détecte la panne du coordinateur.
- Il(s) informe(ent) l'ensemble des processus du début de l'algorithme avec un message <wakeup > car la seconde partie de l'algorithme doit être initié par toutes les feuilles.
- Lorsque le message < wakeup > a parcouru tout l'arbre, l'élection commence.
- les feuilles émettent un message.
- le processus ayant le plus grand numéro est élu.

Variables

- wsp : booléen init faux (wsp est vrai si p est réveillé).
- wrp : integer init 0 (compte les messages de reveil reçus) reqp[q] ; 8q 2 Neighp : booléen init faux (vrai si p a reçu un message de q).
- vp : numéro de processus init p (plus grand processus) etatp : (sleep, leader , lost) init sleep.
- Vp : voisins du processus.

Algorithm 7 Algorithme de élection aléatoire dans un arbre -Partie réveil-

Début
Si p est initiateur **alors**
WS_p := vrai ;
Pour q ϵ $Neigh_q$ faire
Envoie <wakeup> à q ;
FinPour ;
Finsi ;
Tant que wr_p <# V_p **faire**
Recoit <wakeup> ;
Si ws_p=faut **alors** ws_p =vrai ;
Pour q ϵ $Neigh_q$ **faire**
envoie <wakeup> à q
Finpour ;
Finsi ;
Fintant que ;
FIN.

Algorithm 8 Algorithme de élection aléatoire dans un arbre -Partie élection-

Début
Tant que #(q :req_p[q]) = faux)> 1 **faire**
Reçoit <tok,r> de q ;
req_p[q] :=vrai ;
v_p :=max(v_p,r) ;
Envoie <tok,v_p > à q_0 tel que req_p[q_0] est faux ;
Reçoit <tok,v_p > de q_0 ;
v_p :=max(v_p,r) ;
si v_p = p **alors**
$State_p$:= leader ;
Sinon
$State_p$:= lost ;
FinSi ;
Pour q ϵ $Neigh_p$, q $\neq$ q0 **faire**
envoie <tok,v_p > à q ;
Finpour ;
FIN.

2.3.2.5 Algorithme basé sur l'extinction [52].

Principe

- Chaque processus initiateur p dise à ses voisins un message marqué par son numéro p.
- Chaque processus dénie la plus grande valeur reçue, initialement la sienne.
- Quand un message arrive, si la valeur contenue est inférieure à la valeur locale, le message est ignoré, si elle est supérieure le processus abandonne sa propre diusion et enregistre la nouvelle valeur reçue comme la plus grande, si la valeur est égale à la valeur locale alors le processus est le leader.
- Complexité messages : N (nombre processus) * M (nombre messages par processus).

2.3.2.6 Élection sur topologie du réseau quelconque

- **Élection dans le cas de la diffusion séquentielle [25].**

Hypothèses

- Une topologie du réseau quelconque est supposée.
- Le réseau est modélisé par un graphe est connexe.
- Les liaisons sont unidirectionnelles.
- Les messages ne se perdent pas et sont délivrés au bout d'un temps fini après leur émission.
- Un processus ne connaît que ses voisins et n'apprend jamais la structure globale du réseau.

Lors de la décision de lancer une élection
faire
Si $état_i$ = initial **alors** lancer_exploration **fsi**
fait

Lors de réception de explorer (k,Z,S) depuis p_j
Faire
cas $pgvu_i > k \rightarrow$ **si** $etat_i$ = initial **alors** lancer_exploration **fsi**
$pgvu_i < k \rightarrow etat_i$:= battu ;
$pgvu_i$:= k ;
$pred_i$:= j ;
Soit Y = $voisins_i$ – Z ;
cas $Y = \Phi \rightarrow succ_i := \Phi$;
cas $S = \Phi \rightarrow$ **envoyer** conclure à p_j ;
$S \neq \Phi \rightarrow$ **envoyer** rebrousser(k, $Z \cup \{i\}$,S) à p_j
fcas
$Y \neq \Phi \rightarrow$Soit x = maximum(Y) ;
$succ_i := \{x\}$;
envoyer explorer(k, $Z \cup \{i\}$, $S \cup Y-\{x\}$) à p_x
fcas
fcas
fait

Lors de réception de *Rebrousser(k,Z,S)* depuis pj
faire
Si $pgvu_i$ = k **alors**
Soit Y = $voisin_i \cap$ S ;
Cas $Y = \Phi \rightarrow$ **envoyer** rebrousser(k,Z,S) à $Pred_i$;
$Y \neq \Phi \rightarrow$**Soit** X = maximum(Y) ;
$Succ_i := Succ_i \cup \{X\}$;
envoyer explorer(k, Z, S-{X}) à Px ;
Fcas
Fsi
Fait

Lors de réception de *Conclure* depuis p_j
Faire
Si ($pgvu_i$ = i) **Alors** $etat_i$ = elu **fsi**
$\forall$ **x** $\in$**($Succ_i \cup Pred_i$)-{j} : envoyer** Conclure à Px ;
Fait

FIGURE 2.7 – Algorithme d'élection séquentielle

Tableau de comparaison des algorithmes d'élection

Algorithme	**topologie**	**complexité**
Lelann	Anneau	O(n^2)
Chang et Robert	Anneau	O(n^2)
Franklin	Anneau	O(n log(n))
Dolev, Klawe et Rodeh	Anneau unidirectionnel	O(n log(n))
Bully	réseaux complets	O(n^2)

TABLE 2.2 – Tableau de comparaison des quelques algorithems d'élection

2.4 Synthèse

À partir du tableau de comparaison des protocoles d'établissement de clés, nous constatons que les protocoles utilisant la cryptographie symétrique sont moins couteux en espace mémoire et sont plus légers, et que les protocoles utilisant la cryptographie asymétrique offrent une meilleure protection des réseaux. Par rapport à la cryptographie symétrique, la cryptographie asymétrique implique des calculs beaucoup plus complexes qui sont dans la plupart des cas très coûteux pour les nœuds. Les protocoles utilisant des certificats garantissent une communication sécurisée, ils offrent une meilleure sécurité, et sont scalables. Les protocoles utilisant des cartes à puce sont vulnérables à plusieurs attaques. Si par exemple cette carte est volée ou perdue, un attaquant peut connaître toutes les informations sensibles stockées dans sa mémoire, comme il peut avoie accès aux paramètres secrets d'autres utilisateurs légitimes, et peut obtenir la clé de session partagée entre chaque utilisateur et n'importe quel nœud de capteur, et peut donc monter une attaque men-in-the-middle.

Pour ce qui est des algorithmes d'élection, il n'existe pas d'algorithme déterministe du leader dans les réseaux pour toutes les topologies. Chaque topologie et structure se caractérise par son style et sa disposition. Ajoutez à ça notre domaine d'étude, l'IoT où les noeuds ont de faibles ressources et capacités. Il est

donc difficile de concevoir un système qui s'adapte à toutes les situations. De ce fait, il ne faut pas généraliser ces algorithmes d'élection, mais plutôt étudier chaque cas séparément et indépendamment des autres.

2.5 Conclusion

Dans ce chapitre nous avons vu différents protocoles d'authentifications utilisant différentes techniques et méthodes, ainsi que quelques algorithmes d'élection, ou nous avons fait une étude et analyse là-dessus, et nous avons soulever certains points importants. Le chapitre suivant fera l'objet de notre contribution qui est basée sur l'authentification des capteurs et des noeuds dans un système IoT.

Chapitre 3

Proposition

3.1 Introduction

Dans le chapitre précédent, nous avons analysé quelques travaux publiés sur l'axe des modèles d'authentification d'une part, et d'autre part nous avons cité certains algorithmes et techniques d'élection que nous avons pu recenser.

Dans l'IoT, appareils et capteurs communiquent ensemble directement par le world wide web et avec n'importe quelle application. Réduire la surface d'exposition des Objets connectés aux attaques est une tâche complexe. Elle requiert une connaissance architecturale de la chaine de valeur qui relie les objets au cloud. Il faut s'intéresser aux objets eux-mêmes, à leurs capteurs et processeurs, aux réseaux locaux et distants, aux protocoles de tout niveau, puis aux serveurs, à leurs logiciels et aux traitements des données qui y sont réalisés. Les besoins sont bien connus depuis des années.

Le présent chapitre traite la traduction pratique de notre proposition théorique afin de la rendre opérationnelle. Il sera consacré à la présentation des notions dont nous avons besoin ainsi qu'à la présentation de notre proposition. L'idée que nous proposons a été suggérée par M. ELSAKAANE Nadim (chercheur de l'université de Béjaia).

3.2 Motivation

Les avantages et les capacités multiples de l'Internet ont permis de donner naissance à l'IoT qui peut être considéré comme étant une véritable révolution dans le monde de la technologie, de l'information et de la communication. Selon le géant de la réseautique Cisco, les appareils connectés atteindront le nombre de 50 milliards en 2020 [35, 11], et intègreront notre vie personnelle et professionnelle dans ses différentes facettes. D'où l'importance de mettre l'accent sur la nécessité de sécuriser les communications et de faire en sorte à toujours garder les systèmes aussi disponibles que possible dans l'IoT, et ce, afin de créer un système à part entière qui répond à nos besoins.

3.3 Algorithme et protocoles utilisés

3.3.1 L'algorithme d'élection

Dans cette partie, nous nous sommes basés sur l'analyse des algorithmes précédents et nous avons opté pour l'implémentation plutôt classique et assez simpliste dans le cas d'un arbre pour effectuer notre élection. L'algorithme se trouve dans le chapitre 2, section 2.3.2.4, deuxième point (Algorithme général d'élection 2). Après des ajustements nous obtenons un algorithme plus adapté a notre situation. Cet algorithme est décrit ci-dessous :

Principe

- Un Noeud détecte la panne du Serveur.
- Il informe l'ensemble des Noeuds du début de l'algorithme avec un message.
- Lorsque le message parcourt tout l'arbre et atteint toutes les feuilles, l'élection commence.
- Les feuilles lancent l'élection et émettent un second message contenant leur propre ID ainsi que les ressources restantes et disponibles.

- Linformation remonte l'arbre et à chaque fois la plus grande valeur est transmise et ce jusqu'à atteindre la racine.
- Pour minimiser le nombre de messages transmis et économiser de l'énergie, le Noeud doit attendre que tous ses fils lui envoient leurs messages afin de pouvoir choisir le plus grand et faire suivre vers la racine.
- Une fois que le message atteint la racine, le Noeud ayant les plus grandes valeurs sera élu.
- Un nouveau message sera transmis alors à tous les Noeuds pour les informer que l'élection est finie et qu'un Noeud a été choisi. Ce dérnier se verra son type changer en Serveur.

3.3.2 Protocole d'authentification

3.3.2.1 Cryptographie à courbe elliptique

La différence des algorithmes de chiffrement à base de courbes elliptiques par rapport aux algorithmes basés sur les entiers comme RSA ou El-Gamal est que, pour les vaincre, il faut résoudre un problème de logarithme discret sur une courbe elliptique est réputé être un problème plus difficile que le problème similaire dans les entiers modulos n. c'est pourquoi on estime qu'une clé de 200 bits (qui mesure, pour une courbe elliptique, la taille du corps fini K de cette courbe) pour les chiffres basés sur les courbes elliptiques est plus sûre qu'une clé de 1024 bits pour le RSA. Comme les calculs sur les courbes elliptiques ne sont pas bien compliqués à réaliser, c'est un gros avantage pour les cartes à puces où on dispose de peu de puissance, et où la taille de la clé influe beaucoup sur les performances.

3.3.2.2 Échange des clés par courbes elliptiques

Il s'agit d'un échange de clés à la manière de Diffie-Hellman, c'est-à-dire sans se les communiquer directement. Alice et Bob se mettent d'accord ensemble et publiquement sur une courbe elliptique E (a,b,K), c'est-à-dire qu'ils choisissent

un corps fini K (par exemple, Z/pZ), et une courbe elliptique y^2=x^3+ax^2+b. Ils choisissent aussi ensemble, et toujours publiquement, un point P situé sur la courbe.

Ensuite Alice et Bob choisissant secrètement KA et KB, Alice envoie à Bob le point de la courbe elliptique KAP, et Bob envoie à Alice KBP, ils sont capables de calculé kA (kBP)=kB(kAP)=(kAkB)P, ce point de la courbe elliptique consiste la clé secrète ((kAkB)P).

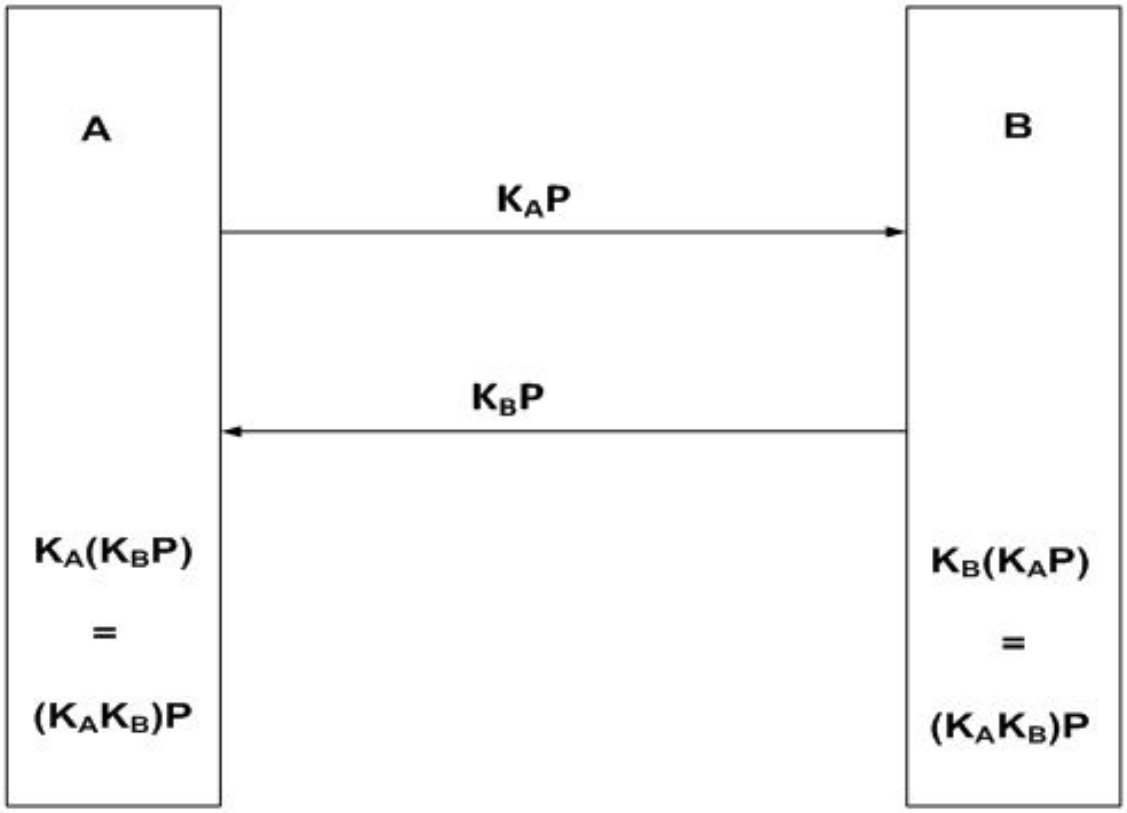

FIGURE 3.1 – Échange de clés

3.3.3 Protocole Leach

Avec le protocole LEACH, les agrégations, les compressions ainsi que le routage des données minimisent la consommation d'énergie en réduisant le flux des données et ainsi les communications globales. Les noeuds d'un réseau sont homogènes et ont les mêmes contraintes d'énergie. Le clustering permet aux noeuds d'effectuer des communications sur de petites distances avec leurs CHs, ces derniers ayant pour tâche de communiquer les résultats de leurs calculs à la station de base. Ceci leur coûte en termes d'énergie. Ce protocole présente d'excellents résultats comparés à d'autres algorithmes de clustering. Dans Leach, les données sont fusionnées pour réduire la quantité d'informations transmises et la consommation d'énergie est partagée sur l'ensemble des noeuds prolongeant ainsi la durée de vie du réseau [24].

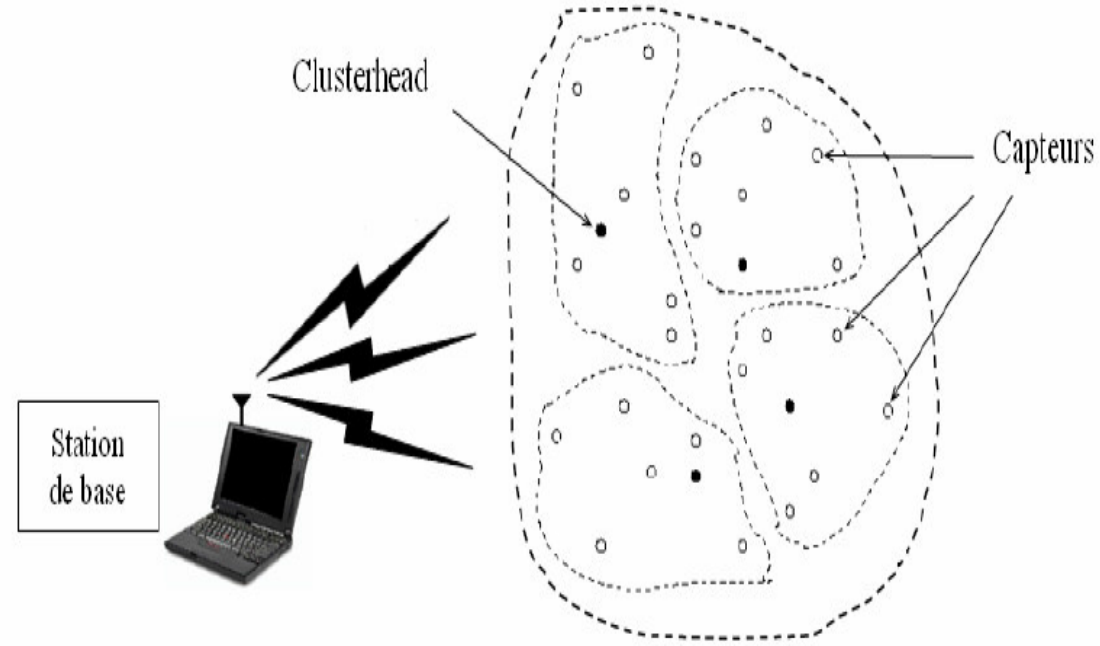

FIGURE 3.2 – Le Clusturing dans un réseau de capteurs [24].

3.3.4 Modèle du système

Dans notre cas d'étude, nous avons choisi de ne pas prendre un modèle défini ou précis d'architecture. Au début, nous avons pensé à utiliser une architecture en étages, soit, dans une tour ou un immeuble, où se trouvent plusieurs types d'activités, ce qui implique, dans ce cas, que chaque étage ou groupe d'étages soit occupé par des entreprises différentes, et chacune utilise son propre système informatique et ses propres serveurs pour gérer l'ensemble de ses activités. Puis, nous avons changé d'avis et avons opté pour une architecture non définit et aléatoire ou nous déploierons notre proposition, et ce, afin de tester notre solution dans un environnement ou les nœuds ne connaissent pas l'architecture et n'ont aucune notion du chemin à prendre pour leur routage, et doivent donc explorer l'environnement afin de pouvoir communiquer et ainsi construire un chemin.

3.3.5 Notre solution :

Dans le cadre de notre travail, nous admettons que l'IoT est un réseau dense composé d'objets hétérogènes déployés aléatoirement et regroupés en communautés. Chaque objet peut jouer le rôle de fournisseur ou demandeur de service. Chaque objet a une identité unique et appartient à une seule communauté gérée par un serveur de confiance. Un objet sollicité pour la réalisation d'un service peut soit accepter ou refuser de le fournir.

Nous supposons aussi dans notre étude que la panne du serveur est déjà survenue, et nous tenterons donc de chercher un nouveau candidat temporaire pour prendre le relai afin que le système ne tombe jamais en panne, de cette façon les activités sur le réseau ne cesseront pas et le système demeurera fonctionnel jusqu'à la réparation ou le remplacement dudit serveur. La première étape de notre proposition débutera juste après la panne.

- **Phase 1**

Dans la première étape de notre proposition, comme nous l'avons dit juste précédemment, nous supposons que la panne est déjà survenue et qu'un des nœuds l'a détectée. C'est donc ce même nœud qui informera tous les autres qu'une panne du serveur a été détectée. Remarquez que le Noeud ayant détecté la panne est légèrement plus que les autres.

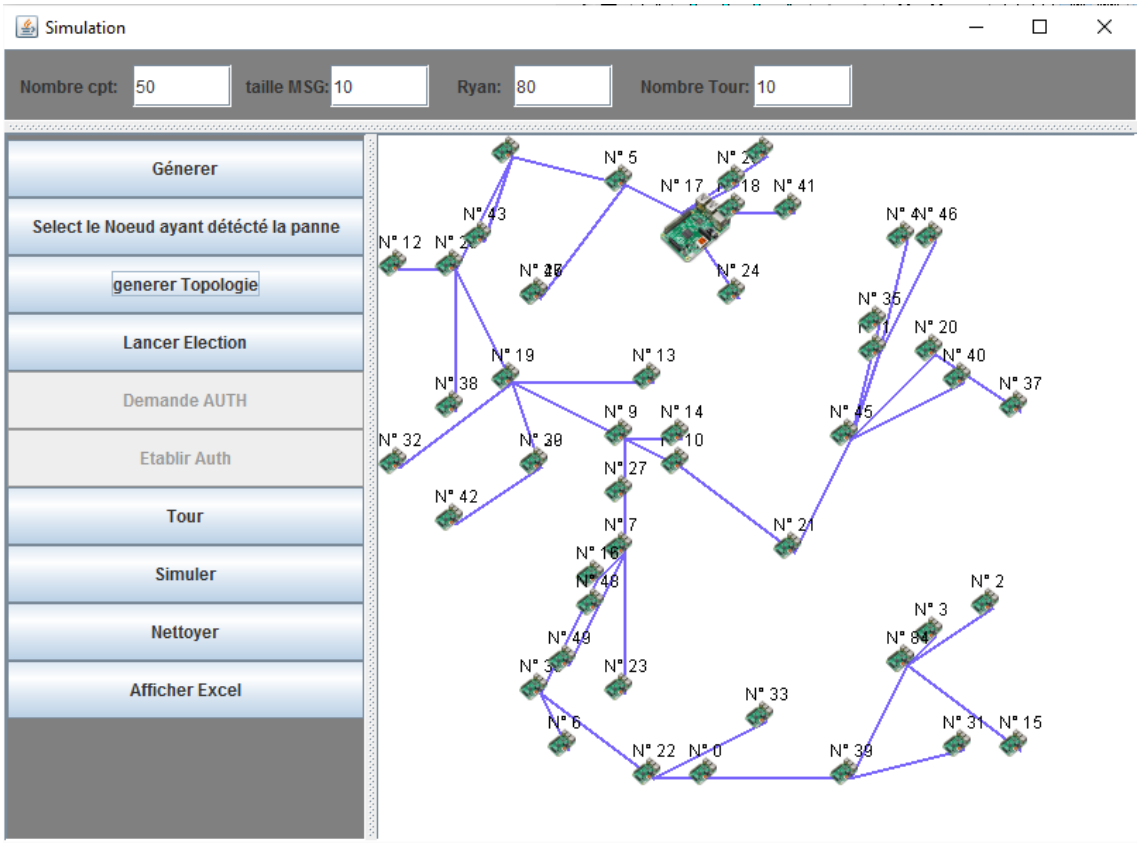

FIGURE 3.3 – Phase 1.1

- **Phase 2**

Le nœud lancera une exploration à l'aveugle à partir de sa position afin de détecter tous les voisins qui sont à sa portée, et ce dans le but de créer une arborescence dont il sera la racine. Cette arborescence permettra non seulement de réduire le nombre de messages transitant, mais aussi de couvrir tous les Noeuds (sauf si des Noeuds sont isolés et ne sont pas à porter), et ce, sans que les messages transitent plusieurs fois et soient obligés de rebrousser chemin. Ainsi nous économiserons aussi de l'énergie. La figure 3.3 illustre cette arborescence.

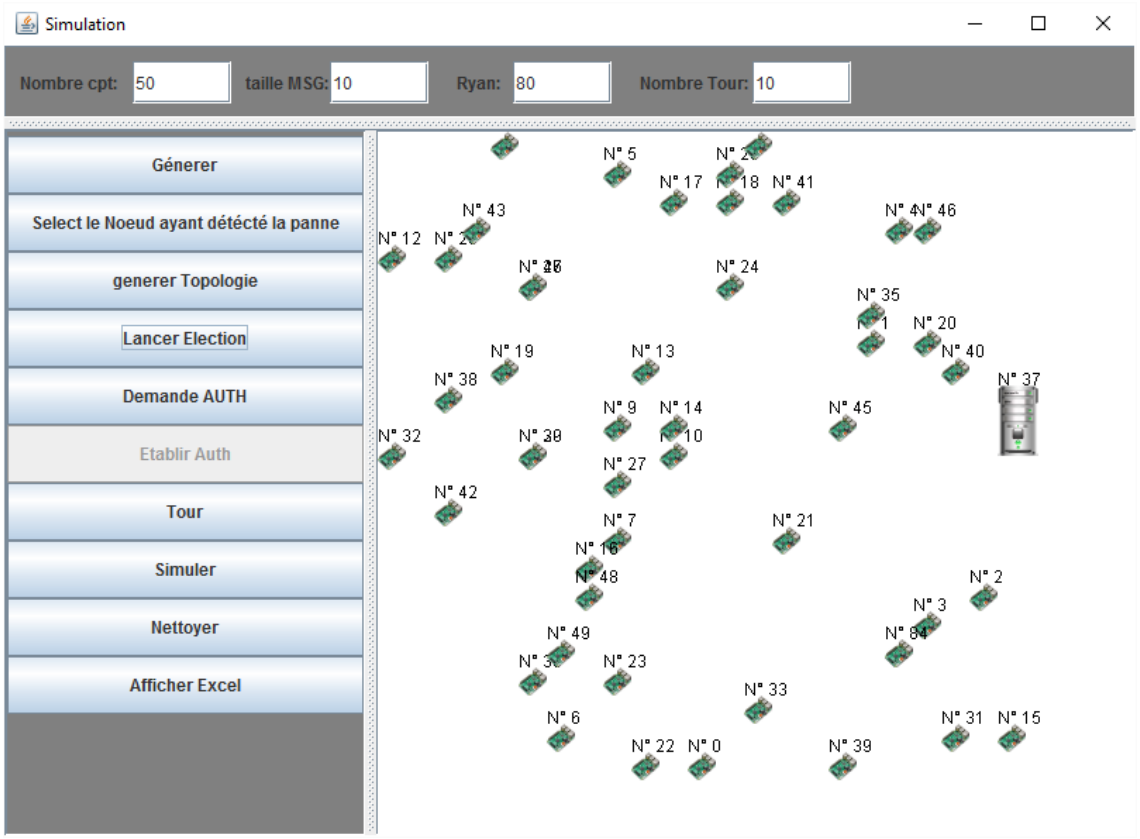

FIGURE 3.4 – Phase 1.2

- **Phase 3**

L'étape suivante est le lancement de l'algorithme d'élection, ou on peut voir et suivre les différentes étapes de l'algorithme. À la fin de l'algorithme, un nouveau serveur est sélectionné.

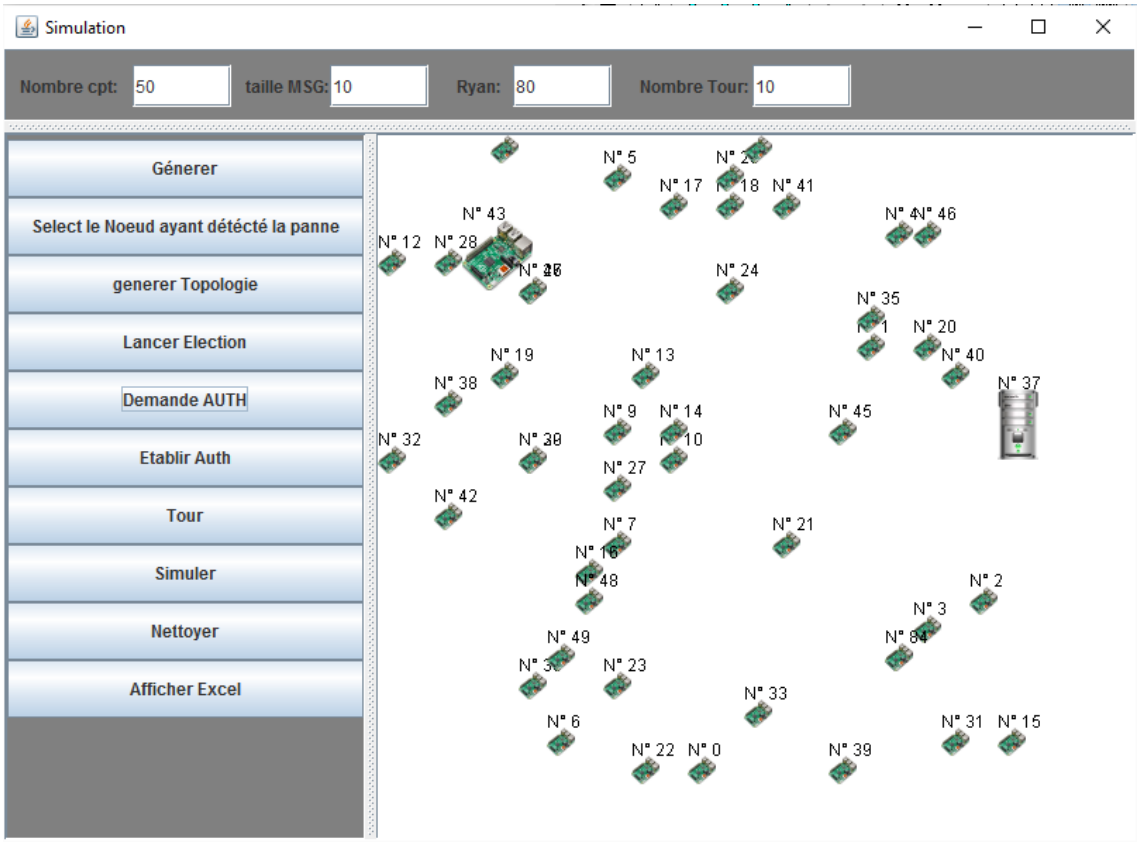

FIGURE 3.5 – Phase 1.3

- **Phase 4**

Maintenant le scénario est le suivant :
- Un Noeud demande à s'authentifier auprès du nouveau serveur
- Il envoie son chiffre, sa clé et l'envoie au serveur.
- le serveur vérifie si la clé et bien chiffré.
- Si c'est le cas, il envoie un ok au Noeud.
- Sinon l'authentification échouera.

Pour le routage de la clé d'authentification, nous avons utilisé le protocole Leach qui calcule le meilleur chemin afin d'économiser le maximum d'énergie. Ce chemin est montré dans la figure ci-dessous.

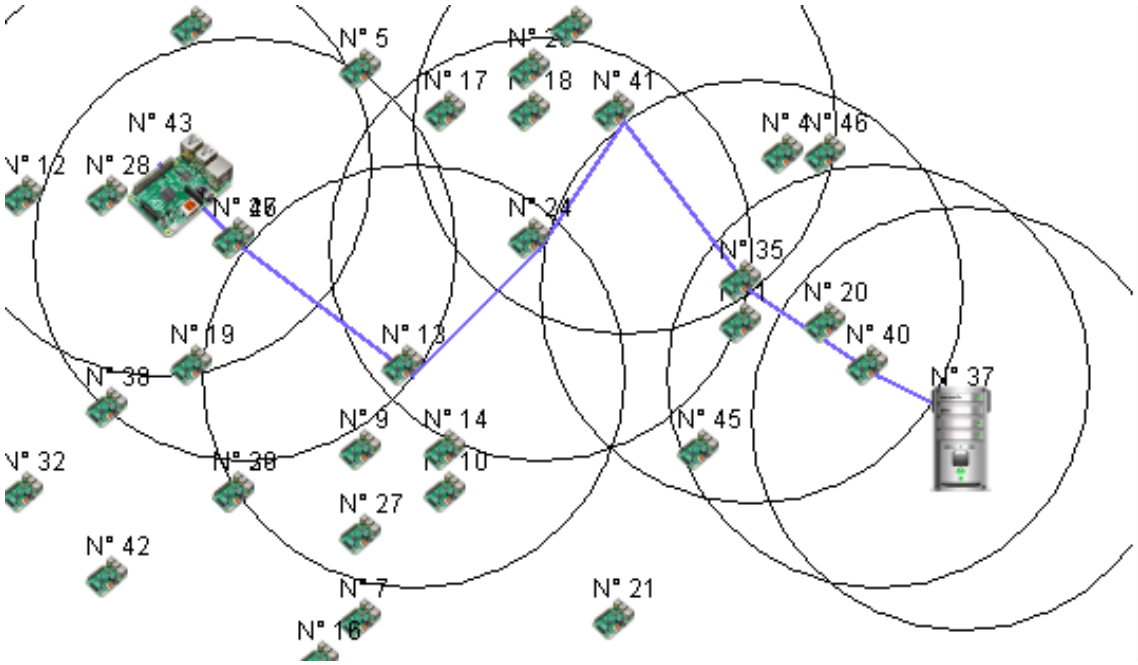

FIGURE 3.6 – Phase 1.4

3.4 Conclusion

Ce chapitre a été consacré à la présentation de notre contribution. Par manque de temps et suite à un changement d'orientation de dernière minute, nous avons pris le risque et le défi de nous lancer dans cette tentative et expérience qui est plus intéressante et enrichissante. De plus, l'IoT est un des domaines qui est le plus d'actualité et qui a un très grand potentiel de par ses applications et ses domaines d'applications.

Notre modèle regroupe différentes techniques et méthodes qui sont l'élection d'un leader afin d'élire un nouveau candidat, un protocole d'authentification qui sert a sécurisé les communications entre les noeuds à base d'ECC ce qui permet d'économiser des ressources surtout dans notre notre domaine d'étude, l'IoT, ou on doit sauvegarder et économiser un maximum, et en dernier, un protocole de routage, Leach, qui optimise le chemin par lequel transitent les messages dans le but d'économiser un maximum d'énergie. Le chapitre suivant fera l'objet des résultats de notre implémentation.

Chapitre 4

Simulation de notre solution

4.1 Introduction

Apès avoir décrit les differentes phases et étapes de notre solution dans le chapitre précédent, une évaluation des performances s'impose afin d'observer et analyser pas une expérimentation et à travers des simulations que nous lancerons dans différents cas, le comportement de notre proposition. Nous commencerons d'abord par définir l'environnement de travail ainsi que les outils utilisés.

4.2 Outils utilisés

4.2.1 IDE Eclipse

Eclipse IDE est un environnement de développement intégré libre (le terme Eclipse désigne également le projet correspondant, lancé par IBM) extensible, universel et polyvalent, permettant potentiellement de créer des projets de développement mettant en œuvre n'importe quel langage de programmation. Eclipse IDE est principalement écrit en Java (à l'aide de la bibliothèque graphique SWT, d'IBM), et ce langage, grâce à des bibliothèques spécifiques, est également utilisé pour écrire des extensions. La spécificité d'Eclipse IDE vient du fait de son architecture totalement développée autour de la notion de plug-in (en conformité avec la norme OSGi) : toutes les fonctionnalités de cet atelier logiciel sont développées en tant que plug-in.

Plusieurs logiciels commerciaux sont basés sur ce logiciel libre, comme par exemple IBM Lotus Notes 8, IBM Symphony ou Websphere Studio Application Developer.

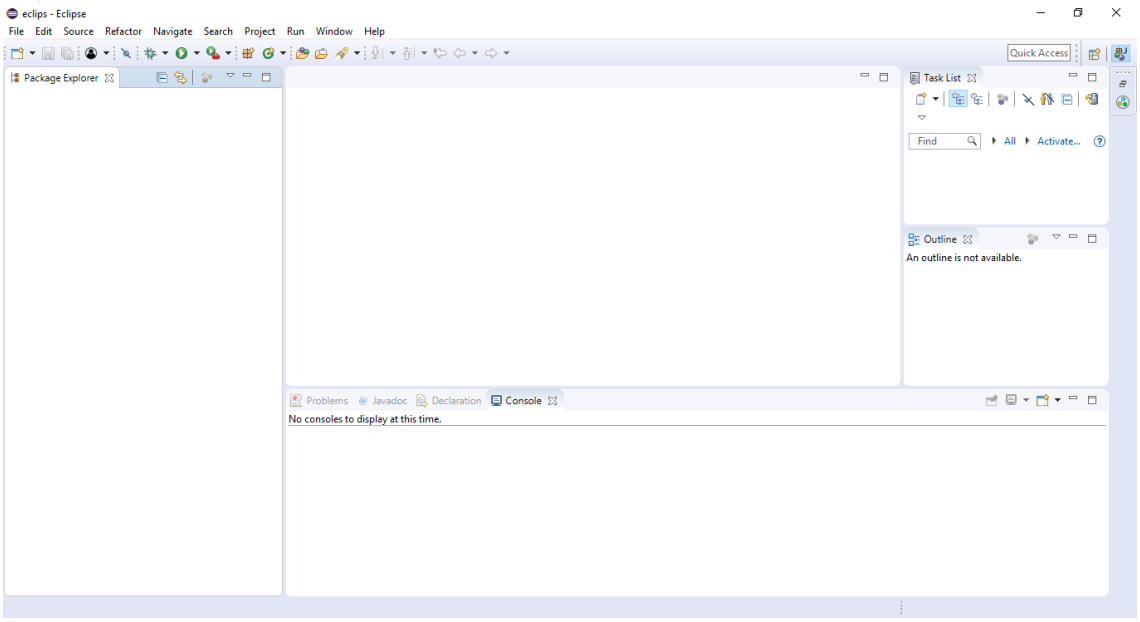

FIGURE 4.1 – Eclipse

4.2.2 Dispositifs dans l'IoT

Du point de vue matériel, IoT est composé de matériel hétérogène, plus encore que dans l'Internet traditionnel. Les appareils IoT peuvent être classés en deux catégories en fonction de leur capacité et de leur performance. La première catégorie se compose de périphériques haut de gamme, qui incluent des ordinateurs simples comme Rasberry Pi et les smartphones. Les périphériques IoT haut de gamme disposent de suffisamment de ressources et de caractéristiques adéquates pour exécuter des logiciels basés sur les systèmes d'exploitation traditionnels tels que Linux ou BSD. La deuxième catégorie se compose de périphériques IoT bas de gamme, qui sont trop limités aux ressources pour exécuter ces systèmes d'exploitation traditionnels. Les exemples populaires de périphériques IoT bas de gamme incluent Arduino, Econotag,

Zolertia Z1, IoT-LAB M3 nœuds, Open-Mote nodes et TelosB motes, certains Dont sont représentés sur la Figure ci-dessous.

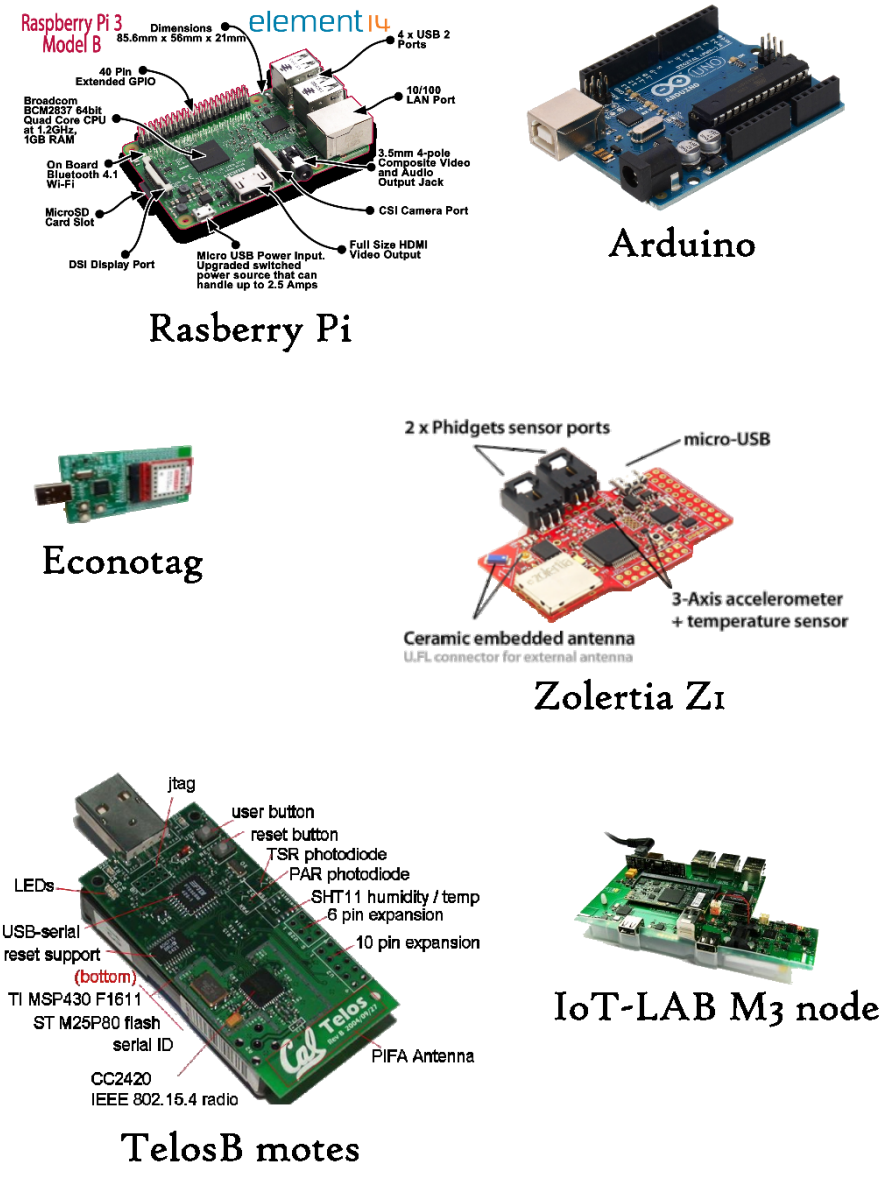

FIGURE 4.2 – Dispositifs dans l'IoT

4.3 Simulation

Nous avons lancé plusieurs simulations avec différents paramètres afin de voir et d'observer le comportement et la réaction des Noeuds dans de multiples cas.

4.3.1 Graphe énergétique

Pour ce qui est de l'énergie des Noeuds, elle sera consommée au fur et à mesure que les testes se prolongerons et qu'on générera de panne. Dans le premier graphe, la simulation a été lancée avec une cinquantaine de Noeuds et suite à 10 tours de panne, d'élection et de tentative d'authentification, et disposés de façon éparpillé et dont certains groupes seront isolés de la majorité des autres. Dans ce cas nous constaterons une très grande consommation énergétique qui s'explique par le fait que le serveur élu se trouvait à l'extrémité du réseau.

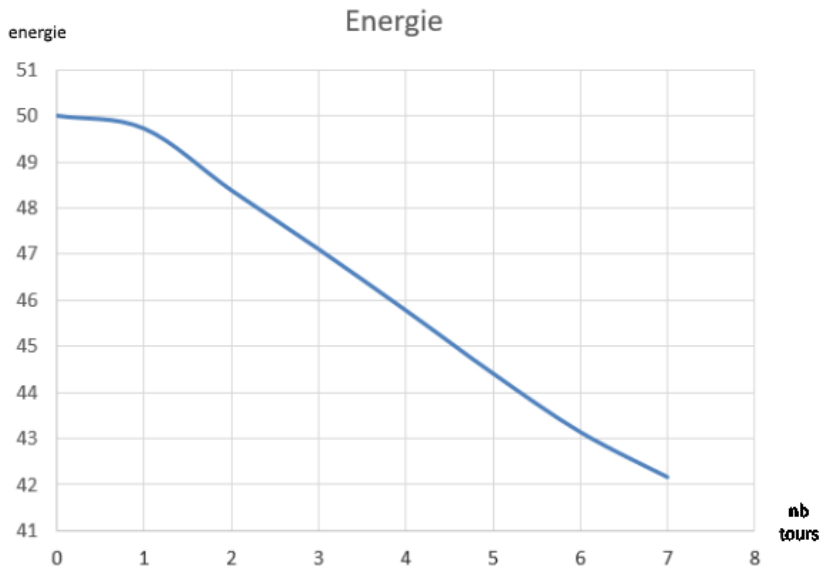

FIGURE 4.3 – Graphe d'énergie 1.

Dans le second grand graphe énergétique, nous remarquerons une consommation plutôt correcte et économe. Cela est dû à une topologie où tous les Noeuds seront reliés les uns au autres (sauf très rarement dans certains tours) et/ou le serveur de relai se trouve plus ou moins au centre du réseau.

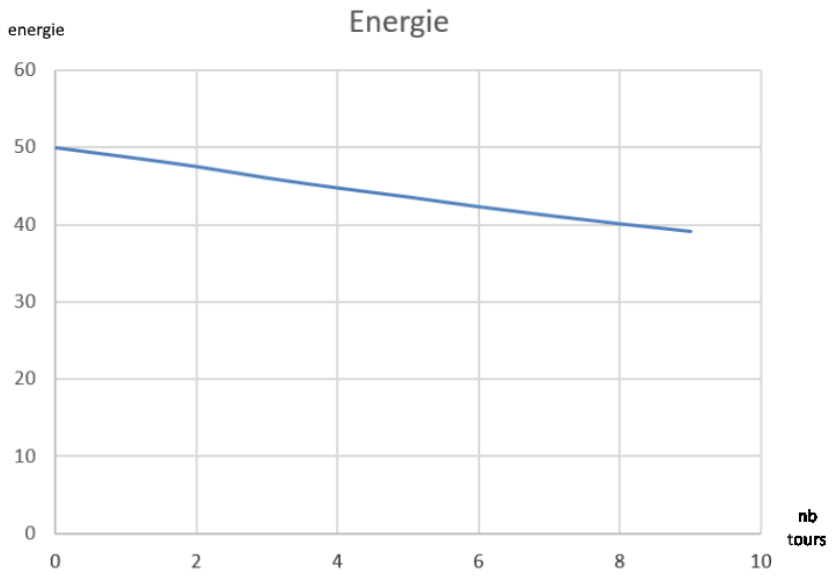

FIGURE 4.4 – Graphe énergétique 2.

4.3.2 Graphe des messages non reçus

Ce graphe représente les messages de demandes d'authentification non reçue par le serveur en cas où le Noeud qui veut s'authentifier soit isolé et ne peut en aucun cas communiquer avec le serveur ou avec un autre Noeud afin de relayer le message. Nous obtiendrons donc un des deux graphes suivants.

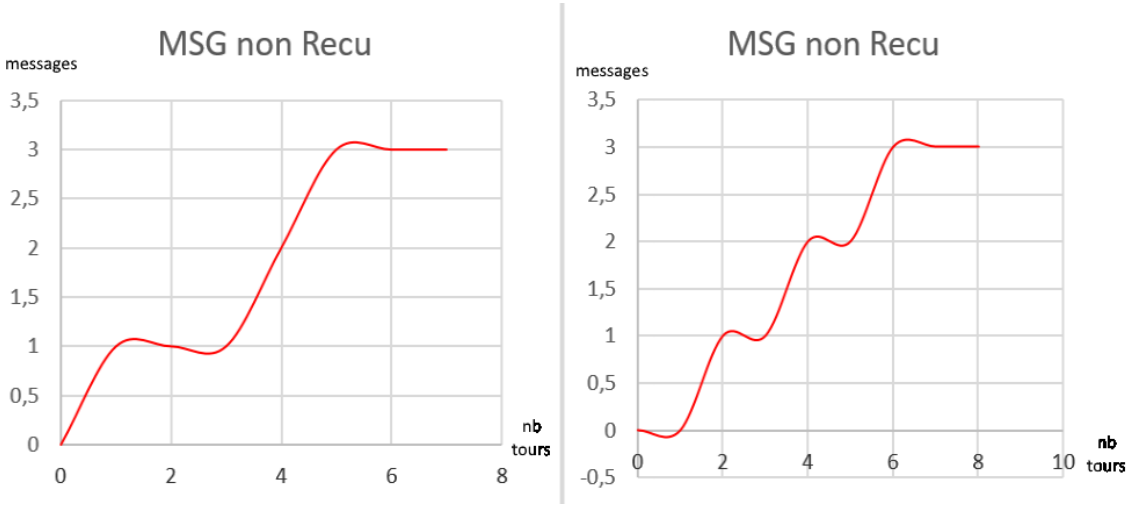

FIGURE 4.5 – Graphe des messages non reçus 1.

En temps normal et dans une disposition semblable à la deuxième décrite dans le graphe d'énergie, on a un graphe nul. Ce qui signifie qu'aucun message n'a été perdu et que toutes les demandes d'authentifications ont atteint le serveur.

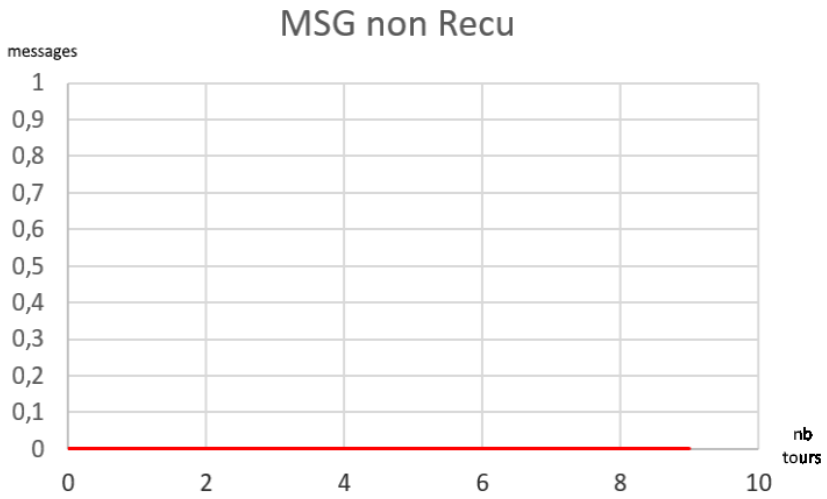

FIGURE 4.6 – Graphe des messages non reçus 2.

4.3.3 Graphe d'attaques

Dans notre simulation, parfois, se génèrent aléatoirement des Noeuds qui n'appartiennent pas au réseau de départ et donc ont une clé différente des autres Noeuds. Ces Noeuds sont considérés comme malicieux et le serveur détecte qu'ils ont une clé différent lorsqu'ils tentent de s'auhentifier. Les graphes suivants montrent la détection de ces attaques au fil du temps. Quand une attaque est générée et que le serveur ne l'a pas détecté, cela signifie que le Noeuds attaquant est isolé et sa demande n'a pas atteint le serveur.

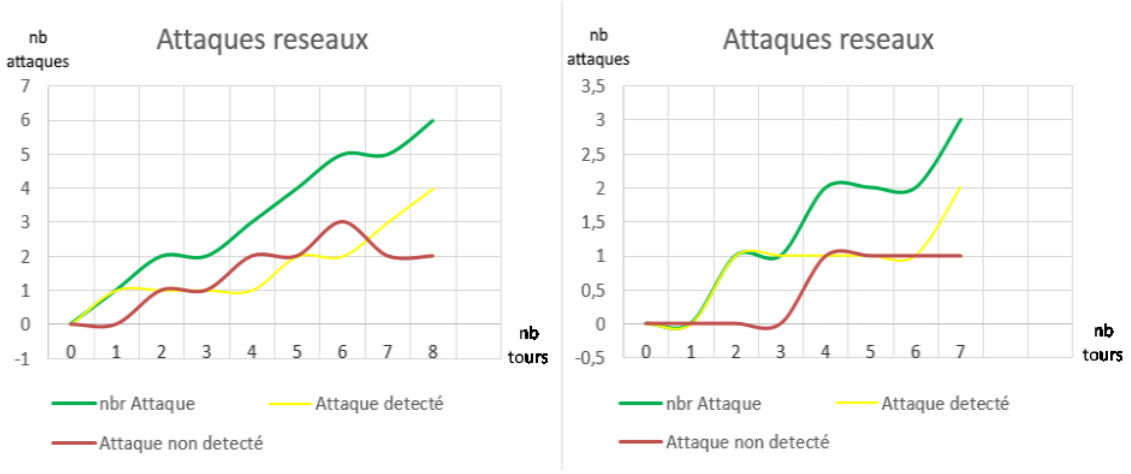

FIGURE 4.7 – Graphe d'attaques 1.

Dans le graphe qui suit, aucun Noeud isolé n'a été détecté.

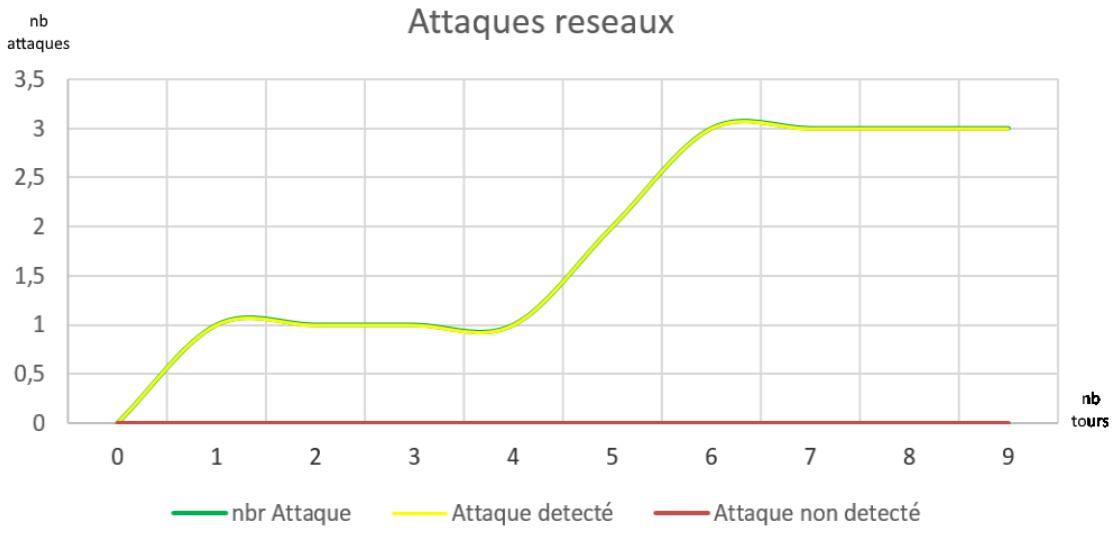

FIGURE 4.8 – Graphe d'attaques 2.

4.4 Conclusion

Cette simulation nous a permis de tester et d'évaluer les performances de notre solution proposée. Après différents tests avec différents paramètres, il en ressort que le nombre de Noeud influe sur la consommation énergitique ; si on n'a pas beaucoup de Noeuds, on risque d'avoir des vides dans la topologie générée et ainsi ne pas couvrir tous les Noeuds. De plus, le protocole Leach serra plus efficace et performant, car il aura à sa disposition plus choix à faire afin de pouvoir économiser un maximum d'énergie. Pour ce qui est des attaques, le serveur peut les détecter si le Noeud en question est à portée des autres.

Conclusion et perspectives

IoT est caractérisée par une forte ubiquité dans le monde physique et une forte omniprésence autour de ses usagers. Les diverses applications potentielles de l'IoT, l'hétérogénéité, ses technologies habilitantes et sa forte dimension humaine et socioéconomique rendent sa sécurité un sujet difficile et complexe. Dans ce travail, nous avons mis en avant les concepts essentiels de l'IoT, ainsi que ses besoins et les défis de la sécurité. Nous avons étudié quelques schémas d'établissement de clé qui permettent d'offrir le service d'authentification et de sécurité. Nous avons aussi recensé quelques algorithmes d'élections qui servent à élire un processus parmi plusieurs dans le but de prendre des décisions dans un système informatique. Notre proposition se base sur un protocole d'authentification et un algorithme d'élection, le protocole d'authentification vise à sécuriser les communications entre les différents objets ou bien entre les Objets et le serveur, et l'algorithme d'élection vise à sélectionner un autre serveur parmi un ensemble de dispositifs pouvant assurer cette fonction en cas de panne.

En guise de perspective, nous tenterons d'améliorer et d'optimiser certains aspects de la proposition que nous suggérons à savoir la première partie qui sert à construire un arbre afin de relier tous les noeuds a eux dans le but d'établir une élection et être sûr qu'on ne néglige pas quelques-uns. Nous testerons aussi de déployer le protocole d'authentification qui aurait dû être utilisé afin de le tester et d'apporter même quelques améliorations dans le but éternel d'optimiser et de sécuriser de façon maximale les Objets.

Bibliographie

[1] `http://www.lirmm.fr/~lafourca/ML-enseign/CNAM/mps.pdf`, (Consulte le 22 juin 2017).

[2] C. Aggarwal, N. Ashish, and A. Sheth. The internet of things : A survey from the data-centric perspective. In *Managing and mining sensor data*, pages 383–428. Springer, 2013.

[3] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash. Internet of things : A survey on enabling technologies, protocols, and applications. *IEEE Communications Surveys & Tutorials*, 17(4) :2347–2376, 2015.

[4] E. Baccelli and D. Raggett. The internet of things and the web of things. ERCIM, 2015.

[5] M. Bellare and P. Rogaway. Entity authentication and key distribution. In *Annual International Cryptology Conference*, pages 232–249. Springer, 1993.

[6] P. Benghozi, S. Bureau, and F. Massit-Folea. L'internet des objets. quels enjeux pour les européens ? 2008.

[7] Y. Challal. *Sécurité de l'Internet des Objets : vers une approche cognitive et systémique*. PhD thesis, Université de Technologie de Compiègne, 2012.

[8] Y. Chen, J. Chou, and I. Liao. Improved on an improved remote user authentication scheme with key agreement. *IACR Cryptology ePrint Archive*, 2016 :23, 2016.

[9] D. Conan. Initiation á l'algorithmique répartie. 2017.

[10] L. Da-Xu, W. He, and S. Li. Internet of things in industries : A survey. *IEEE Transactions on industrial informatics*, 10(4) :2233–2243, 2014.

[11] E. Dave. L'internet des objets comment l'évolution actuelle d'internet transforme-t-elle le monde ? 2011.

[12] J. De-Loof, C. SAP, S. Meissner, A. Nettsträter, A. CEA, M. SAP, and J. Walewski. Internet of things-architecture iot-a deliverable d1. 5-final architectural reference model for the iot v3. 0. 2013.

[13] J. Deng, C. Hartung, R. Han, and S. Mishra. A practical study of transitory master key establishment forwireless sensor networks. In *Security and Privacy for Emerging Areas in Communications Networks, 2005. SecureComm 2005. First International Conference on*, pages 289–302. IEEE, 2005.

[14] D.Jain, P. Krishna, and V. Saritha. A study on internet of things based applications. *arxiv preprint arxiv :1206.3891*, 2012.

[15] P. Cagni et al. *Les Nouveaux Eldorados de l'économie connectée*. 2013.

[16] M. Farash, M. Turkanović, S. Kumari, and M. Hölbl. An efficient user authentication and key agreement scheme for heterogeneous wireless sensor network tailored for the internet of things environment. *Ad Hoc Networks*, 36 :152–176, 2016.

[17] Y. Faye. *Algorithmes d'authentification et de cryptographie efficaces pour les réseaux de capteurs sans fil*. PhD thesis, Université de Franche-Comté, 2014.

[18] S. Feng, J. Cerles, H. Dalmas, T. Dô-Khac, and B. Paulin. *sécurité des objets connectés*. Institut national des hautes études de la sécurité et de la justice, 2014.

[19] K. Gaurav, P. Goyal, V. Agrawal, and S. Rao. Iot transaction security. 2015.

[20] R. Greenstadt and J. Beal. Cognitive security for personal devices. In *Proceedings of the 1st ACM workshop on Workshop on AISec*, pages 27–30. ACM, 2008.

[21] D. Hankerson, A. Menezes, and S. Vanstone. *Guide to elliptic curve cryptography*. Springer Science & Business Media, 2006.

[22] D. He and S. Zeadally. An analysis of rfid authentication schemes for internet of things in healthcare environment using elliptic curve cryptography. *IEEE Internet of Things Journal*, 2(1) :72–83, 2015.

[23] T. Heer, O. Garcia-Morchon, R. Hummen, S. Keoh, S. Kumar, and K. Wehrle. Security challenges in the ip-based internet of things. *Wireless Personal Communications*, 61(3) :527–542, 2011.

[24] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan. An application-specific protocol architecture for wireless microsensor networks. 1 :660–670, 2002.

[25] J. Hélary, A. Maddi, and M. Raynal. *Calcul distribué d'un extremum et du routage associé dans un réseau quelconque*. PhD thesis, INRIA, 1986.

[26] C. Hennebert and J. Dos-Santos. Security protocols and privacy issues into 6lowpan stack : a synthesis. *IEEE Internet of Things Journal*, 1(5) :384–398, 2014.

[27] X. Kauffmann-Tourkestansky. *Analyses sécuritaires de code de carte à puce sous attaques physiques simulées*. PhD thesis, Université d'Orléans, 2012.

[28] S. Kumari, M. Khan-Khurram, and X. Li. An improved remote user authentication scheme with key agreement. *Computers & Electrical Engineering*, 40(6) :1997–2012, 2014.

[29] P. Laurent. *Algorithmique Distribuée, Élection distribuée*. PhD thesis, Université de franche-comté, 2014.

[30] T. Limbasiya and N.Doshi. An analytical study of biometric based remote user authentication schemes using smart cards. *Computers & Electrical Engineering*, 2017.

[31] J. Liu, Y. Xiao, and C. Chen. Authentication and access control in the internet of things. In *Distributed Computing Systems Workshops (ICDCSW), 2012 32nd International Conference on*, pages 588–592. IEEE, 2012.

[32] J. Lopez, R. Oppliger, and G. Pernul. Authentication and authorization infrastructures (aais) : a comparative survey. *Computers & Security*, 23(7) :578–590, 2004.

[33] P. Mahalle, B. Anggorojati, N. Prasad, and R. Prasad. Identity authentication and capability based access control (iacac) for the internet of things. *Journal of Cyber Security and Mobility*, 1(4) :309–348, 2013.

[34] C. Mayer. Security and privacy challenges in the internet of things. *Electronic Communications of the EASST*, 17, 2009.

[35] N. Mehdi. Demain, l'internet des objets. `http://www.strategie.gouv.fr/publications/demain-linternet-objets`, 2015.

[36] D. Miorandi, S. Sicari, F. De-Pellegrini, and I Chlamtac. Internet of things : Vision, applications and research challenges. *Ad Hoc Networks*, 10(7) :1497–1516, 2012.

[37] K. Mohapatra, J. Bhuyan, P. Asundi, and A. Singh. Asolution framework for managing internet of things(iot). *International Journal of Computer Networks & Communications (IJCNC)*, 8(6), 2016.

[38] M. Pasha, S. Shah, and U. Pasha. Security framework for iot systems. *International Journal of Computer Science and Information Security*, 14(11) :99, 2016.

[39] S. Patel, D. Patel, and A. Navik. Energy efficient integrated authentication and access control mechanisms for internet of things. In *Internet of Things and Applications (IOTA), International Conference on*, pages 304–309. IEEE, 2016.

[40] P. Porambage, A. Braeken, C. Schmit, A. Gurtov, M. Ylianttila, and B. Stiller. Group key establishment for enabling secure multicast communication in wireless sensor networks deployed for iot applications. *IEEE Access*, 3 :1503–1511, 2015.

[41] P. Porambage, C. Schmitt, P. Kumar, A. Gurtov, and M. Ylianttila. Pauthkey : A pervasive authentication protocol and key establishment scheme for wireless sensor networks in distributed iot applications. *International Journal of Distributed Sensor Networks*, 2014.

[42] P. Porambage, C. Schmitt, P. Kumar, A. Gurtov, and M. Ylianttila. 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.

[43] R. Saad. *Modèle collaboratif pour l'Internet of Things (IoT)*. PhD thesis, Université du Québec à Chicoutimi, 2016.

[44] S. Sahraoui. *Mécanismes de sécurité pour l'intégration des RCSFs à l'IoT (Internet of Things)*. PhD thesis, Université de Batna 2, 2016.

[45] H. Serge. *L'ELECTION*. PhD thesis, Université de Paris-Dauphine, 2014.

[46] J. Stankovic. Wireless sensor networks. *computer*, 41(10), 2008.

[47] M. Turkanović, B. Brumen, and M. Hölbl. A novel user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor networks, based on the internet of things notion. *Ad Hoc Networks*, 20 :96–112, 2014.

[48] H. Wang and Q. Li. Efficient implementation of public key cryptosystems on mote sensors (short paper). In *International Conference on Information and Communications Security*, pages 519–528. Springer, 2006.

[49] R. Weber. Internet of things-new security and privacy challenges. *Computer law & security review*, 26(1) :23–30, 2010.

[50] M. Weill and M. Souissi. L'internet des objets : concept ou réalité ? In *Annales des Mines-Réalités industrielles*, number 4, pages 90–96. Eska, 2010.

[51] N. Ye, Y. Zhu, R. Wang, and Q. Lin. An efficient authentication and access control scheme for perception layer of internet of things. 2014.

[52] A. Zemmari. *Présentation et analyse de quelques algorithmes distribués probabilistes*. PhD thesis, Université Bordeaux 1, 2009.

[53] G. Zhao, X. Si, J. Wang, X. Long, and T. Hu. A novel mutual authentication scheme for internet of things. In *Modelling, Identification and Control (ICMIC), Proceedings of 2011 International Conference on*, pages 563–566. IEEE, 2011.

RÉSUMÉ

Internet of Things relie des objets à Internet, ce qui permet d'obtenir des idées jamais obtenues auparavant. L'IoT est un vaste réseau de périphériques qui comprend différents dispositifs intelligents et capteurs. Ces "Objets" collectent et échangent des données. Leurs contraintes matérielles ainsi que celle des environnements hostiles dans lesquels ils pourraient être déployés rendent l'IoT vulnérable et nécessitent des mécanismes de sécurité efficaces et peu couteux. La panne des serveurs dans les systèmes distribués est un autre problème dans l'IoT. Dans ce mémoire, nous présentons un état de l'art de quelques protocoles de gestion de clé existants, quelques algorithmes d'élection.

Mots clés : Internet des Objets (IdO), authentification, sécurité, élection.

ABSTRACT

Internet of Things connects objects to the Internet, making it possible to get ideas that have never been available before. The IoT is an extensive network of devices that includes various intelligent devices and sensors. These "Objects" collect and exchange data. Their physical constraints as well as hostile environments in which they could be deployed make IoT vulnerable and require effective and inexpensive security mechanisms. Another problem in the IoT is the failure of servers in distributed systems. In this brief, we present a state of the art of some existing key management protocols, some election algorithms..

Key words : Internet of Things (IoT), Authentication, security, election.