République Algérienne Démocratique et Populaire
Ministre De l'Enseignement Supérieur et de la recherche scientifique

Université A/MIRA-Bejaia
Faculté des Sciences Exactes
Dépatement d'Informatique

MÉMOIRE DE MASTER RECHERCHE

En

Informatique

Option

Réseaux et Systèmes Distribués

Thème

Internet des objets avec sécurité des communications Device to Device

Réalisé par : M^{lle} **BRAHAM Hamama**

M^{lle} **SOUAMI Sylia**

Président	M^{me}	**HAMZA**	**Lamia**
Examinateur	M^{me}	**OUYAHIA**	**Samira**
Encadrant	M^{r}	**SADI**	**Moustapha**

Année universitaire 2020/2021

⁕ *Remerciements* ⁕

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

Nous exprimons nos sincères gratitudes à notre encadreur M^r **SADI Moustapha** pour la haute qualité de son encadrement, son suivi, sa disponibilité et ses conseils.

Nous témoignons également énormément de gratitude envers tous les membres de jury qui ont fait l'honneur d'évaluer, examiner et enrichir notre modeste travail.

Nous remercions aussi l'ensemble des enseignants du département d'Informatique pour tout ce qui nous a été transmis tout au long de notre formation.

Nos remerciements les plus chaleureux à nos parents, pour leur soutien, les encouragements et leurs sacrifices.

Enfin, Nous remercions toutes les personnes intéressées par ce travail, en espérant qu'elles puissent y trouver des explications utiles pour leurs propres travaux.

Merci à toutes et à tous.

⁂ *Dédicaces* ⁂

***J**e dédie ce travail*

A celle qui a attendu avec patience les fruits de sa bonne éducation et de ses dévouements

A ma chère mère

A celui qui s'est changé la nuit en jour pour m'assurer les bonnes conditions

A mon cher père

A mes frères et sœurs qui ont partagé avec moi mes échecs et mes succès

*A ma chère sœur **Hanifa** pour son soutien tout au long de mes études*

*A **Sylia**, chére amie avant d'être binôme*

A tous mes proches et amis pour leur présence et tous les moments de bonheurs passés à leurs cotés

B.Hamama

⁂ *Dédicaces* ⁂

J*e dédie cet événement marquant de ma vie*

A ***mes chers parents****, qui n'ont jamais cessé de formuler des prières à mon égard, de me soutenir et de m'épauler pour que je puisse atteindre mes objectifs.*

A ***mes grands parents****, qui je souhaite une longue vie pleine de santé.*

A mes chers frères et sœur, à qui je souhaite un avenir radieux plein de réussite.

A mes chers amis et particulièrement à ma binôme ***Hamama*** *avec lesquels j'ai trouvé l'entente dont j'avais besoin.*

A toute personne qui occupe une place dans mon cœur.

A tous ceux qui me sont chers.

S.Sylia

Table des matières

Table des figures

Notations et symboles

ACK	Acquittement
AVISPA	Automated Validation of Internet Security Protocols and Applications
BS	Base Station
CAS+	Common Authentication Specification
CL-AtSe	Constraint-Logic-based Attack Searcher
CN	Central Network
D2D	Device to Device
D2I	Device to Infrastructure
DH	Déffie-Hellman
DHMKE	Diffie-Hellman-Markle Key Exchange
eNB	Evolved NodeB
GHz	GigaHertz
HLPSL	High-Level Protocol Specification Language
HMAC	Hash-Based Message Authentication Codes
HTTP	HyperText Transfer Protocol
IdO	Internet des objets
IEEE	Institute of Electrical and Electronics Engineers
IF	Intermediate Format
IoT	Internet of Things
IoV	Internet of Vehicles
IP	Internet Protocol
ISM	Industriel, Scientifique et Médical
IT	Information Technology
LTE	Long Term Evolution
LTE-A	Long Term Evolution-Advanced
M2M	Machine to Machine
MAC	Message Authentication Codes
Mbps	Megabits per second
MD	Malveillant Dispositif
MD5	Message Digest 5

MEMS	Micro Electromechanical System
MIT	Masschusetts Institute of Technology
MITM	Man-in-the-middle
NFC	Near Field Communication
OC	Objet Connecté
OFMC	On-the-fly Model-Checker
RF	Radio Frequency
RFID	Radio Frequency Idendification
SATMC	The SAT-based Model-Checker
SAT TA4SP	Tree Automata based on Automatic Approximations for the Analysis of Security Protocols
SHA	Secure Hach Algorithm
SPAN	Security Protocol ANimator for AVISPA
TIC	Technologie de l'information et de la communication
TLA	Temporal Logic of Actions
UE	User Equipment
V2V	Vehicle to Vehicle
Wi-Fi	Wireless Fidelity
WSN	Wireless Sensor Networks

Introduction générale

Actuellement, la tendance en technologies consiste à se servir des techniques sans fil tel que l'Internet des objets (IdO) qui a été intégrée dans plusieurs domaines comme : l'armé, l'industrie, l'agriculture, etc. L'IdO en tant qu'une évolution de l'Internet actuelle permet une amélioration considérable de notre mode de vie, c'est un terme très vaste et riche. Il nous fait imaginer un monde entier qui est relié et peut se communiquer grâce à l'échange d'informations entre ses objets.

La communication Device to Device (D2D) devient le moyen le plus pratique pour fournir un déploiement rapide et une connexion sans fil autogérée afin de répondre aux différents besoins des fonctionnalités. Elle offre une possibilité plus large de recherches approfondies dans la construction de la future communauté de réseaux sans fil tels que l'IdO dont les objets connectés utilisent la communication D2D.

La technologie D2D permet d'établir des liaisons de communication directes entre appareils à la place de la transmission de données via une station de base BS (Base Station). D'où, il est nécessaire de prendre en compte les exigences de sécurité qui doivent être bien traitées avant que la communication D2D ne soit acceptée et mise en œuvre, afin d'assurer le bon fonctionnement et la sécurité du réseau.

Malheureusement, l'usage des méthodes de chiffrement seulement ne suffit pas pour garantir le secret des données confidentielles. Tous les jours, de nouvelles failles sont découvertes sur des protocoles cryptographiques, il est donc indispensable de vérifier automatiquement la sécurité de ces protocoles avant leurs mises en service. Cette vérification se fait à l'aide des outils de vérification automatiques qui s'appuient sur le modèle formelle tel que AVISPA (Automated Validation of Internet Security Protocols and Applications).

Le sujet de notre projet, est la vérification formelle et automatique d'un protocole de sécurité dans la communication D2D basée sur les dispositifs de l'IdO à l'aide de l'outil SPAN (Security Protocol ANimator for AVISPA) AVISPA, notre travail est composé de quatre chapitres structurés comme suit :

Le premier chapitre sera consacré à la présentation d'une manière générale de l'IdO et l'impact qu'elle va avoir sur notre mode de vie.

Pour ce qui est du deuxième chapitre, nous allons présenter des généralités et l'aspect sécurité associés à la communication D2D.

Dans le chapitre qui suit, nous présenterons un aperçu général de quelques travaux réalisés pour une communication D2D sécurisée, ensuite, on va se concentrer sur l'étude de quelques protocoles d'échange de clés dans la communication D2D.

Dans le dernier chapitre, nous allons vérifier formellement les protocoles de sécurité D2D étudiés dans le chapitre précédent à l'aide de l'outil SPAN AVISPA en se basant sur le langage CAS+ (Common Authentication Specification).

Chapitre 1

Internet des objets

1.1 Introduction

Internet des objets (IdO) ou en anglais the Internet of Things (IoT) représente aujourd'hui une partie majeure de notre vie quotidienne. Des milliards d'objets intelligents et autonomes, à travers le monde sont connectés et communiquent entre eux, tout le temps et partout et idéalement depuis n'importe quelle plate-forme.
L'IdO est le fruit du développement et de la combinaison de différentes technologies. Il englobe presque tous les domaines de la technologie d'information IT (Information Technology) actuels tel que les villes intelligentes, les systèmes machine à machine M2M (Machine to Machine), les véhicules connectés, les réseaux de capteur sans fil WSN (Wireless Sensor Networks), etc [17].
Ce chapitre est consacré à définir l'IdO, présenter l'évolution d'Internet et son impact dans le monde, citer ses différents domaines d'applications, ses technologies, expliquer l'architecture d'IdO et la notion d'objet connecté, et enfin ses avantages et ses défis.

1.2 Définition de l'Internet des objets

Il n'existe pas de définition standard, unifiée et partagée de l'Internet des objets, certaines définitions traitent les aspects techniques de l'IdO, tandis que d'autres se concentrent plutôt sur les usages et les fonctionnalités.
L'IdO est une infrastructure dynamique d'un réseau global, qui relie et combine les objets (physiques ou virtuels) avec l'Internet, en suivant les protocoles qui assurent leurs communication et échange d'informations à travers une variété de dispositifs [18].
D'un point de vue conceptuel, l'IdO affecte à chaque objet une identification unique sous forme d'une étiquette lisible par des dispositifs mobiles sans fil, afin de pouvoir se communiquer les uns avec les autres. Ce réseau crée une passerelle entre le monde physique et le monde virtuel [28].
D'un point de vue technique, l'IdO consiste l'identification directe et normalisée (adresse IP,

protocole http, etc.) d'un objet physique grâce à l'utilisation d'un système d'identification électronique (puces RFID (Radio Frequency Idendification), processeur et communication Bluetooth etc.) et sans besoin de saisir manuellement le code de l'objet [28].

1.3 Qu'est-ce qu'un objet connecté (OC) ?

L'IdO repose avant tout sur les objets connectés, qu'ils soient matériels ou logiciels. A cette dichotomie, il faut ajouter les humains qui associent les capacités matérielles et logicielles. L'objet connecté est d'abord un objet qui a une fonction mécanique et/ou électrique propre, il peut soit être conçu directement «connectable», soit il est déjà existant et la connectivité est rajoutée à postériori. Il a la capacité de capter une donnée grâce aux capteurs qu'il possède et de l'envoyer, via le réseau Internet ou d'autres technologies, pour que celle-ci soit analysée et visualisée sur des tableaux de bord dédiés [26]. Les objets connectés interagissent avec leur environnement par le biais de capteurs : température, vitesse, humidité, vibration, etc.

1.4 Évolution d'Internet et son impact dans le monde

Le terme "Internet of Things" est mentionné pour la première fois en 1999 par Kevin Ashton, cofondateur de l'Auto-ID Center au MIT (Masschusetts Institute of Technology), Kevin Ashton rassemble ses idées et sa vision dans une présentation appelée "That "Internet of Things" Thing", afin d'attirer l'attention des directeurs de l'entreprise Procter et Gamble sur les puces RFID [40]. Aujourd'hui, Kevin Ashton est considéré comme l'inventeur de la notion d'Internet des objets. L'IdO est ensuite évolué avec des technologies sans fil (Wi-Fi par exemple), des MEMS (Micro Electromechanical System), des micros services et d'Internet.
En 2003, la population mondiale a frôlé les 6 milliards d'individus et un demi-milliard d'appareils connectés à l'Internet, le résultat de la division du nombre d'appareils par la population mondiale (0,08) montre qu'il y avait moins d'appareil connecté par personne [42] comme le montre la figure 1.1.
En raison de l'explosion des Smartphones et des tablettes, le nombre d'appareils connectés à Internet a atteint 12,5 milliards en 2010, alors que la population mondiale était de 6,8 milliards. En 2020, il existe environ 50 milliards d'objets connectés dans le monde [42]. Ce chiffre devrait continuer à augmenter il dépassera les 75 milliards d'objets en 2025.
Le nombre de capteurs connectés à Internet pourrait augmenter de plusieurs millions, voire de plusieurs milliards du fait que tout ce qui existe se connecte (animaux, lampes, maisons, personnes, chaussures, arbres, etc.).

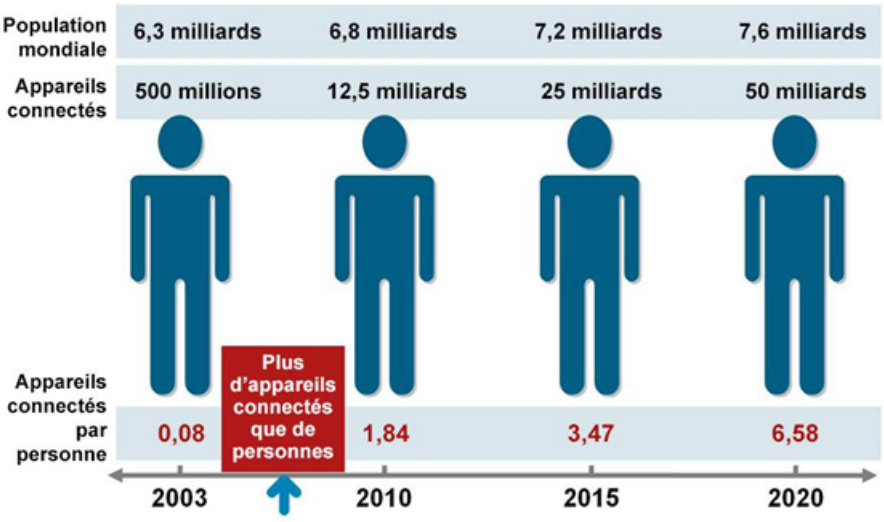

FIGURE 1.1 – L'évolution des objets connectés [41].

1.5 Domaines d'application de l'Internet des objets

Nous constatons que le concept de l'IdO est en pleine explosion vu que nous avons de plus en plus besoin dans la vie quotidienne d'objets intelligents capables de rendre l'atteinte de nos objectifs plus facile. Ainsi, ses domaines d'applications peuvent être variés, comme le montre la figure 1.2.

Parmi ces espaces intelligents, on peut citer [2][3][9][23] :

- **Les villes intelligentes (Smart Cities) :** l'IdO permettra une meilleure gestion des réseaux divers qui alimentent nos villes (eaux, électricité, gaz, etc.) en permettant un contrôle continu en temps réel et précis. Le concept Smart Cities offre à ses habitants une qualité de vie maximale avec une consommation de ressources minimale grâce à une combinaison intelligente des infrastructures (énergie, transport, communication,etc.) aux différents niveaux hiérarchiques (ville, quartier, bâtiment, etc.).
- **La santé (Smart Health) :** dans le domaine de santé, les objets connectés peuvent servir à plusieurs choses. On peut par exemple les utiliser pour suivre la tension, le rythme cardiaque, la qualité de respiration ou encore la masse graisseuse. Ceci permettra ainsi de faciliter la télésurveillance des patients à domiciles, et apporter des solutions pour l'autonomie des personnes à mobilité réduite. C'est presque 90% des services de santé qui intègrent les objets connectés dans leur matériel médical pour augmenter leur productivité et améliorer les soins apportés aux patients ainsi la collaboration entre soignants.
- **L'industrie :** l'industrie n'est pas en reste sur l'usage de l'IdO et des bénéfices que celui-ci lui apporte. Dans le cadre des problématiques rencontrées dans le domaine industriel, l'usage des objets connectés est très spécifique et permettra un suivi total des produits, de la chaîne de production jusqu'à la chaîne logistique, de transformation des processus d'entreprise, de traçabilité qui permet aux usines d'améliorer l'efficacité de ses opérations, d'optimiser la production et d'améliorer la sécurité des employés.

- **L'énergie (Smart Energy) :** l'IdO dans le cadre de l'énergie, répond à des problématiques majeures, elle permet aux innombrables appareils qui composent le réseau électrique de partager des informations en temps réel pour une distribution et une gestion plus efficaces de l'énergie.
- **La domotique ou maison connectée (Smart Homes) :** la domotique regroupe l'ensemble des technologies permettant l'automatisation des équipements d'un habitat. Elle vise à apporter des fonctions de confort : commandes à distance, gestion d'énergie (optimisation de l'éclairage et du chauffage etc.), sécurité (comme les alarmes) et de communication (contacts et discussion avec des personnes extérieures). Les services offerts par la domotique assurent la protection des personnes et des biens en domotique par la prévenir des risques d'accident (incendie, fuite de gaz, etc.), facilitent la vie quotidienne surtout pour les personnes âgées ou handicapées, réduisent la consommation d'énergie grâce à la réactivité maîtrisée d'une maison intelligente.
- **Le transport (Smart Transport) :** presque tous les véhicules vendus aujourd'hui dans le monde renferment déjà des capteurs et de moyens de communication pour traiter la congestion du trafic, la sécurité, la pollution et le transport efficace des marchandises, etc. Des voitures connectées aux systèmes de transport/logistique intelligents pour communiquer de façon autonome avec d'autres véhicules ou une centrale de surveillance, l'IdO peut sauver des vies, minimiser l'impact des véhicules sur l'environnement, prévenir les accidents et réduire les coûts d'assurance.

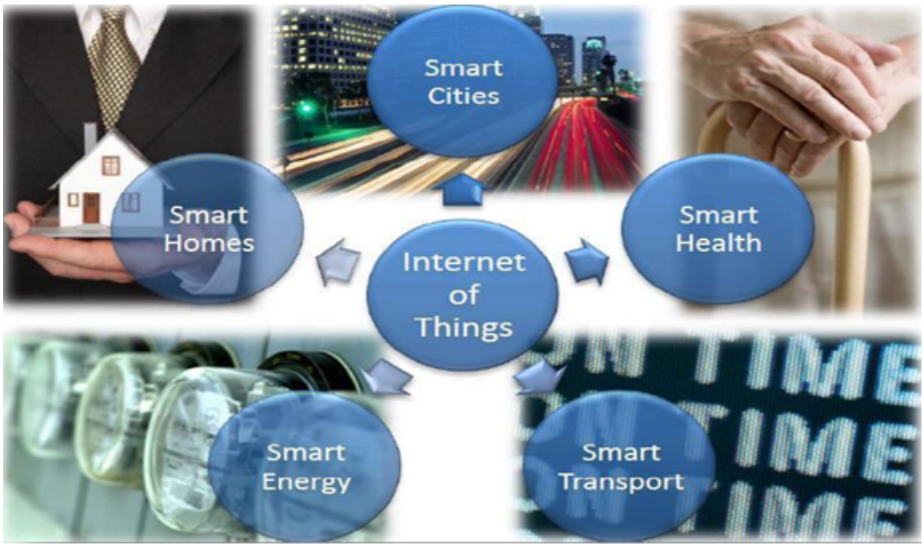

FIGURE 1.2 – Les domaines d'application d'Internet des objets [12].

1.6 Technologies d'Internet des objets

Il existe plusieurs technologies nécessaires utilisées dans le fonctionnement et l'interconnexion des différents objets connectés, nous mettons l'accent seulement sur quelques-unes. Elles sont définies ci-dessous [4][34][39] :

- **RFID :** est une technologie sans fil qui est utilisée pour l'identification des objets, elle

englobe toutes les technologies qui utilisent des ondes radio pour identifier automatiquement des objets ou des personnes. C'est une méthode utilisée pour stocker et récupérer des données à distance grâce à une étiquette (appelée aussi tags, marqueurs, identifiants ou transpondeurs) qui émet des ondes radio. Elles sont placées sur les éléments que l'on veut identifier d'une manière unique. L'étiquette contient des informations stockées électroniquement qui peuvent être lues à distance, à partir de là l'adoption de la technologie RFID s'est avérée nécessaire pour identifier les objets intelligents de façon unique.

- **M2M :** est la combinaison des technologies de l'information et de la communication (TIC), avec des objets intelligents et communicants permettant à ces derniers d'interagir entre eux sans intervention humaine avec le système d'information d'une organisation ou d'une entreprise. L'un des types qu'on peut considérer dans la communication M2M lorsque des équipements utilisateurs sont à proximité immédiats et ont de petites quantités de données à transmettre entre eux est la communication Device To Device (D2D), la combinaison de cette dernière avec l'IdO permettra d'établir un véritable réseau interconnecté contenant divers dispositifs.
- **WSN :** c'est un réseau de capteurs sans fil avec un grand nombre de nœuds, qui peut être une technologie nécessaire au fonctionnement de l'IdO. Ces nœuds ont des capteurs capables de récolter et de transmettre des données environnementales d'une manière autonome. Chaque nœud possède une capacité de traitement et peut contenir différents types de mémoires, un émetteur-récepteur RF (Radio Frequency) et une source d'alimentation, comme il peut aussi tenir compte des divers capteurs et des actionneurs.
- **Bluetooth :** est une technologie sans fil adaptée pour les télécoms de courte portée et permet d'obtenir des débits de l'ordre de 1 Mbps, utilise les ondes radio de bande de fréquence de 2.4 GHz. Il a une pénétration universelle dans l'espace de l'appareil mobile, il est largement intégré dans les ordinateurs personnels, les Smartphone, et les accessoires grand public. Le mode opératoire du Bluetooth garantit une liaison sans fil à la fois stable et fiable qui se configure rapidement et facilement.
- **Wi-Fi :** caractérisé par la norme IEEE 802.11, un standard international fonctionne avec des ondes radio dans une bande de fréquence de 2,4 ou 5 GHz, il permet de relier des équipements informatiques et de téléphonie mobile dans un réseau sans fil haut débit. A l'heure actuelle, la 802.11n s'impose comme la norme Wi-Fi la plus utilisée dans le contexte privé et professionnel. Cette norme offre un débit élevé, de l'ordre de centaines de mégabits par seconde, ce qui est très bien pour les transferts de fichiers, mais peut-être trop consommateur d'énergie pour de nombreuses applications de l'IdO.

1.7 Architecture générale de l'IdO

L'architecture d'un système IdO est composée de plusieurs niveaux qui communiquent entre eux pour relier le monde tangible des objets au monde virtuel des réseaux et du cloud. Tous les projets n'adoptent pas une architecture formellement identique, néanmoins il est pos-

sible de schématiser le parcours de la donnée.

Étant donné que l'IdO a connecté des milliards d'appareils et utilise de nombreuses technologies de calcul et de communication, une architecture clairement stratifiée serait bien pour comprendre l'IdO à un niveau élevé. Il existe un modèle à 5 couches pour l'IdO [29][31][36] comme le montre la figure 1.3 :

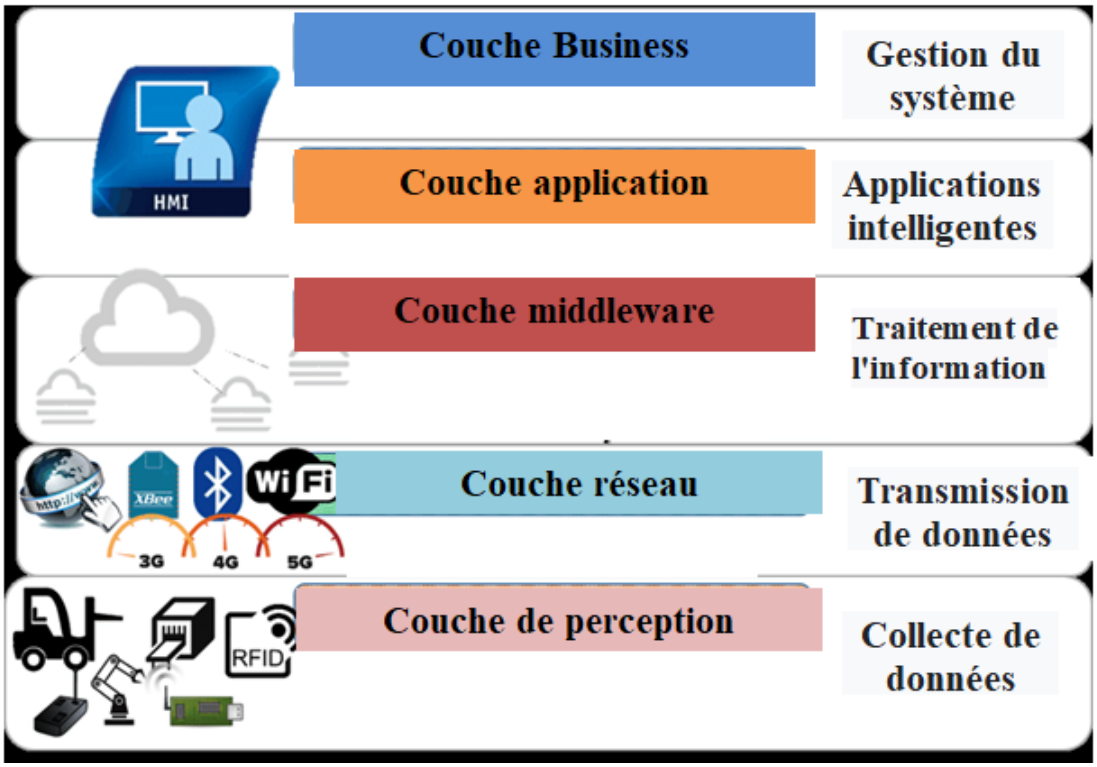

FIGURE 1.3 – Architecture générale d'IdO [8].

- **Couche de perception :** est une couche périphérique /objet. Elle possède des capteurs et actionneurs qui détectent et recueillent des informations sur l'environnement, ces informations collectées sont envoyées à la couche suivante, c'est-à-dire la couche réseau, pour un traitement ultérieur des données.
- **Couche réseau :** est également appelée «couche de transmission», responsable de la connexion, du transport et du traitement des données en toute sécurité issues des capteurs et actionneurs.
- **Couche middleware :** cette couche est en charge de la gestion des services et d'une base de données. Elle reçoit les données de la couche réseau et les enregistre dans la base de données. Elle traite ensuite les données pour prendre les décisions correspondantes.
- **Couche d'application :** offre la gestion des applications basées sur les données traitées dans la couche middleware et est chargée de fournir à l'utilisateur des services spécifiques et applications intelligentes .
- **Couche Business :** gère le système IdO global, y compris les applications et les services. Elle aide les entreprises à créer des modèles commerciaux, à obtenir des résultats d'analyse et à déterminer les stratégies futures.

1.8 Fonctions d'un objet connecté

Les technologies permettant l'IdO se composent essentiellement de quatre fonctions principales : détection, actionneurs, contrôle et communication qui ont une grande analogie avec le corps humain [11][29] comme les montre la figure 1.4 :

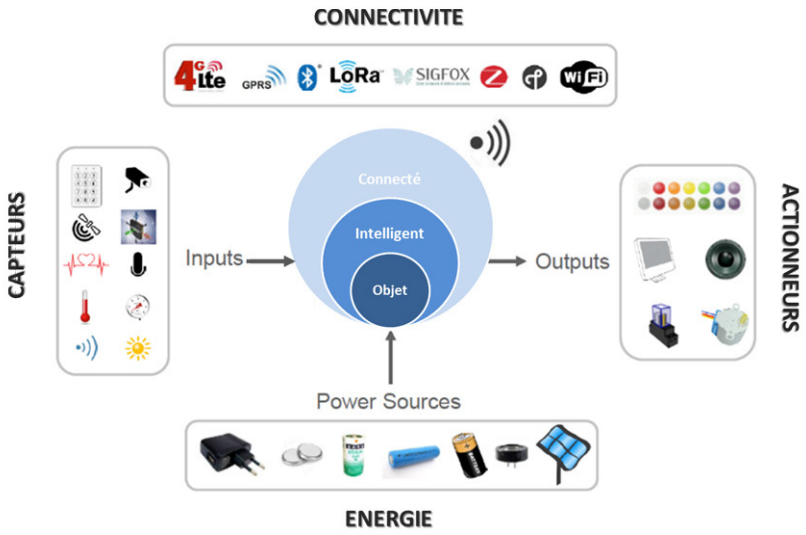

FIGURE 1.4 – Les fonctions principales d'un objet connecté [11].

Capteurs : sont des composants essentiels et intelligents dans un système IdO, permettant de transformer une grandeur physique observée (température, luminosité, mouvement, son, etc.) en une grandeur digitale utilisable par des logiciels. Les objets connectés ont souvent la fonction de captation de ces grandeurs physiques sur leurs lieux d'utilisation.
Actionneurs : sont des dispositifs matériels qui transforment une donnée digitale en phénomène physique pour créer une action en convertissant de l'énergie électrique en une énergie utile, ils sont en quelque sorte l'inverse du capteur. On site quelques exemples comme les éléments de chauffage ou de refroidissement, les haut-parleurs, les moteurs, les lampes etc.
Source d'énergie : l'énergie est un des grands défis des objets connectés, tant pour garantir la plus grande durée de vie possible sans maintenance, il existe quatre types :

- Alimentation filaire : pour les objets ayant accès à une prise de courant.
- Piles ou batteries : pour ceux qui n'y ont pas accès ou de manière occasionnelle (recharge).
- Capteurs d'énergie : «Energy Harvesting» qui apporte une solution à l'autonomie et la gestion des piles qui est l'une des principales problématiques dans l'IdO en prolongeant la durée de vie des objets à très faible consommation, c'est le cas par exemple pour les objets de l'électroniques vestimentaires, les réveils solaires, etc.
- Objets passifs sans fil : qui sont alimentés par les ondes électromagnétiques des lecteurs (RFID, NFC (Near Field Communication), etc).

Connectivité : la connectivité de l'objet est assurée par une petite antenne Radio Fréquence

qui va permettre la communication de l'objet vers un ou plusieurs réseaux. Les objets pourront d'une part remonter des informations telles que leur identité, leur état, une alerte ou les données de capteurs, et d'autre part recevoir des informations telles que des commandes d'action et des données. Le module de connectivité permet aussi de gérer le cycle de vie de l'objet, c'est-à-dire, l'authentification et l'enregistrement dans le réseau, la mise en service, la mise à jour et la suppression de l'objet du réseau.

1.9 Avantages et défis de l'IdO

1.9.1 Avantages

L'IdO devra faire partie de notre quotidien proche et sera appliqué dans divers domaines. Il présente de nombreux points positifs [22] :

- Amélioration de la qualité de service dans différents domaines d'applications dans la vie quotidienne, par exemple : la voiture autonome connectée interagit avec son environnement, elle communique avec d'autres véhicules, l'infrastructure routière, les piétons, les centres de données distants et d'autres entités ;
- Automatisation : l'idée générale de l'IdO implique une communication directe entre des dispositifs, appareils et autres matériels distincts, sans intervention humaine ;
- Connectivité : des connexions améliorées au sein d'un même réseau à l'échelle mondiale permettent d'accéder facilement à diverses informations ;
- Amélioration de la sécurité du travail : la maintenance programmée est également très avantageuse pour garantir la sécurité du travail et le respect des réglementations requises ;
- Télésurveillance : dans le domaine médical, les fabricants d'équipements médicaux équipent de plus en plus leurs appareils avec des capteurs IdO pour collecter les données des patients et permettre la surveillance de la santé et l'administration des médicaments à distance et même de réaliser une chirurgie à distance où le chirurgien et le patient se trouveraient à deux endroits différents ;
- Gain du temps : l'IdO aide les gens à accomplir leurs tâches quotidiennes. Cela permet de gagner un temps précieux. Au lieu de faire des tâches monotones tous les jours, on peut les remplacer par une simple navigation sur le web pour commander des produits, contrôler l'état des objets et/ou endroits connectés.

1.9.2 Défis

- Sécurité des données : assurer la sécurité des données recueillies et transmises par les dispositifs IdO est un défi, car leur utilisation évolue et se développe. Bien que la cybersécurité soit une priorité absolue, les dispositifs Ido ne sont pas toujours inclus dans la stratégie. Les appareils doivent être protégés contre les manipulations physiques, les

attaques logicielles sur Internet, les attaques sur le réseau et les attaques matérielles [45] ;

- Confidentialité des données est une autre préoccupation, notamment parce que les dispositifs IdO sont utilisés dans des secteurs plus sensibles tels que la santé et la finance. Les lois sur la confidentialité des informations entrent également en vigueur dans le monde entier, ce qui signifie que non seulement il est judicieux de protéger les données des individus, mais que les entreprises sont légalement tenues de le faire [36] [45] ;
- Fiabilité du capteur : durée de vie limitée ;
- Les périphériques, applications et services IdO nécessitent des correctifs de sécurité et des mises à jour pour se protéger contre les vulnérabilités connues ;
- Interaction avec des appareils utilisant plusieurs protocoles sans fil ;
- Alimenter des milliards d'appareils connectés.

1.10 Conclusion

L'IdO en tant qu'une évolution de l'Internet actuelle est considéré comme l'une des technologies de pointe dans le monde qui permet une amélioration considérable de notre mode de vie. Suite à cette évolution, certains processus existants dans l'environnement physique sont améliorés et d'autres sont apparus.

Dans ce qui suit, nous allons nous intéresser à la communication D2D dans l'IdO, notamment, l'aspect sécurité qui représente notre problématique de recherche.

Chapitre 2

La communication Device to Device (D2D)

2.1 Introduction

La communication D2D (Device to Device) est considérée comme un nouveau paradigme qui sera mis en oeuvre dans les prochaines générations de réseaux mobiles pour fournir des performances élevées dans le réseau cellulaire, améliorer la couverture, fournir une efficacité spectrale, augmenter la capacité du système et des débits de données élevés [19].
La communication D2D est l'une des techniques prometteuses pour le système de communication sans fil et utilisée dans nombreux domaines différents tels que les services sociaux et les applications (jeux, applications militaires, etc.). La sécurité est l'une des principales préoccupations des communications D2D. Par conséquent, il est nécessaire de prendre en compte les exigences de sécurité dans la conception des communications D2D afin d'assurer la sécurité et le bon fonctionnement du réseau.
Dans ce chapitre, nous allons présenter des généralités sur la communication D2D. Nous commencerons d'abord par sa définition. Ensuite, nous présenterons son fonctionnement, ses différentes formes ainsi que ses avantages et ses limites. Enfin, nous détaillerons le concept de sécurité dans la communication D2D.

2.2 Définition de communication Device to Device

Un système cellulaire conventionnel dépend toujours de l'infrastructure, avec communication D2I (Device to Infrastructure). Toutes les parties de cette communication doivent passer par les stations de bases BS (Base Station) et le système ne permet pas aux dispositifs de communiquer directement entre eux, quel que soit l'emplacement de l'appareil de l'utilisateur. Même si deux appareils sont très proches, le trafic de routage est acheminé via le réseau central. En raison de cette incapacité, la possibilité d'échanger des données entre utilisateurs mobiles

est limitée [5]. L'une des méthodes utilisées pour surmonter ce problème consiste à établir une communication D2D permettant à deux utilisateurs mobiles de communiquer entre eux sans passer par la BS ou le point d'accès AP (Access Point), et ce qui permettra de relayer les connexions directes entre appareils établies via des ressources cellulaires ou des alternatives via les technologies Wi-Fi/ Bluetooth.
La figure 2.1 montre les deux modes de communication D2D, avec et sans infrastructure :

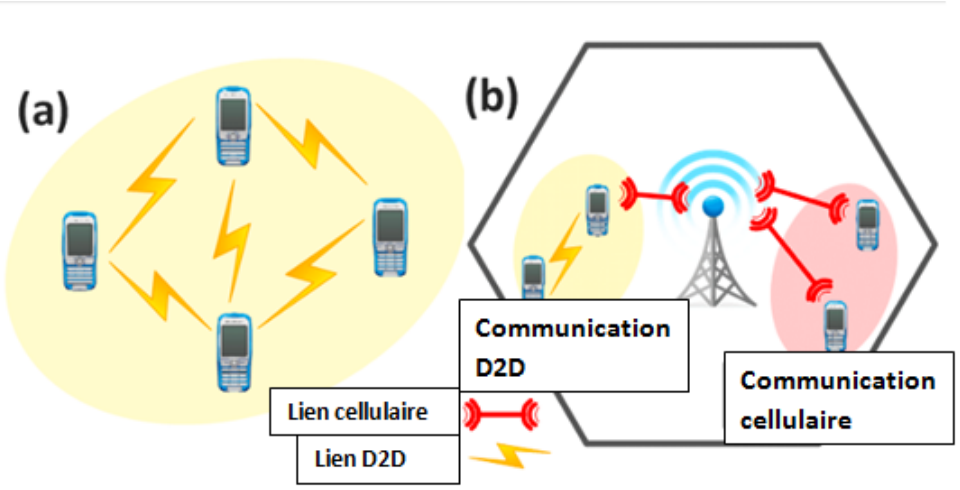

FIGURE 2.1 – Communication D2D : (a) sans infrastructure (b) avec infrastructure [7].

2.3 Fonctionnement du D2D

La communication entre périphériques peut être réalisée selon plusieurs modes de fonctionnement, en fonction de divers scénarios. Selon la situation, le mode de fonctionnement le plus approprié sera choisi pour établir une transmission efficace [33]. La figure 2.2 montre ces différents cas possibles pour la communication D2D :

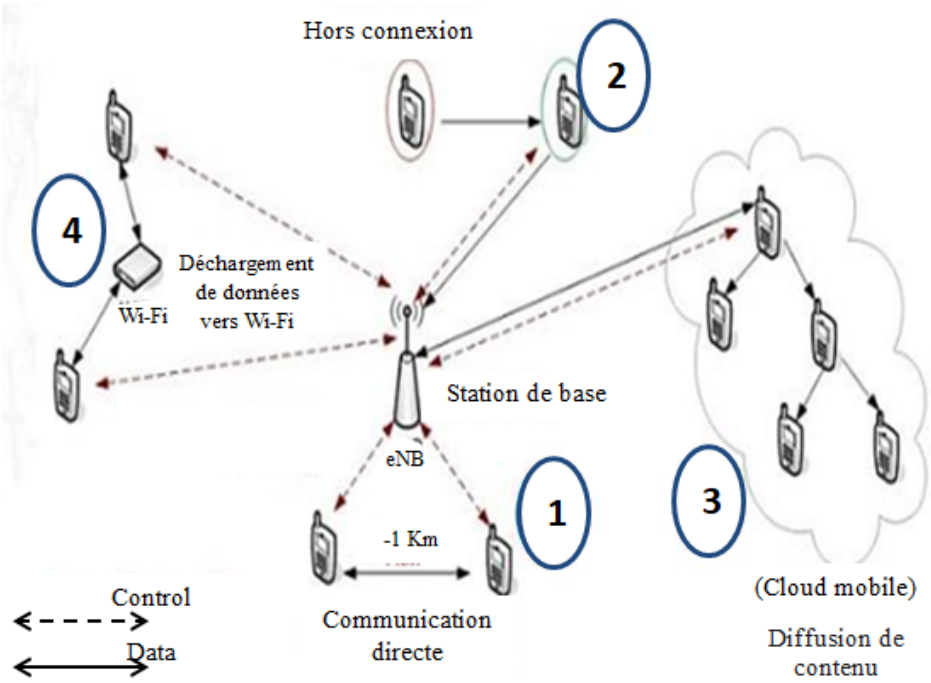

FIGURE 2.2 – Scénarios de communication D2D [33].

- **Scénario-1 :** si deux périphériques se trouvent à proximité, ils peuvent commencer la communication, par exemple en partageant des données. Cela contribue à améliorer le débit de données, à réduire la consommation d'énergie des périphériques et à réduire la charge totale des BS. Le contrôle sera géré par la BS.
- **Scénario-2 :** en absence de connexion réseau mobile active ou de l'insuffisance dans la réception de signaux, les périphériques D2D compatibles peuvent établir une interface de communication alternative avec les périphériques environnants qui sont des BS mobiles connectés. Cela aidera le nœud sans couverture à maintenir une connexion au réseau mobile.
- **Sénario-3 :** plusieurs périphériques peuvent se connecter à un périphérique disposant d'une connexion active à la BS, puis étendre ce réseau en ajoutant une connexion à plusieurs périphériques. Tous les appareils de ce petit Cloud mobile recevront les mêmes données sous forme de publicité ou de messages de la source.
- **Sénario-4 :** dans ce cas, plusieurs périphériques sont déchargés vers une connexion de données Wi-Fi. Les signaux de commande destinés aux appareils seront traités par la BS.

  Le déchargement Wi-Fi offre un débit de données beaucoup plus élevé, une moindre consommation d'énergie et évite les surcharges de trafic des BS.
- **Autres scénarios :** la communication entre périphériques peut être efficacement mise en œuvre pour une communication entre machines, chaque machine pouvant communiquer avec d'autres machines à proximité. La communication entre dispositifs (D2D) est utilisée de la même manière dans les communications entre véhicules V2V (Vehicle to Vehicle) et dans les applications de communication entre homologues. Dans tous ces cas, un nœud

se connecte à la BS principale (station émettrice) et les autres périphériques forment un petit réseau pour communiquer entre eux.

2.4 Formes de la communication D2D

Un réseau cellulaire peut être considéré comme un réseau à deux niveaux constitué des niveaux de macrocellules et d'appareils. Le niveau macrocellule implique des communications entre les BS et les équipements utilisateurs comme dans un système cellulaire classique. Le niveau de l'appareil implique la communication D2D où un appareil se connecte directement à un autre appareil ou réalise sa transmission à l'aide d'autres appareils. Lors de la réalisation de communications au niveau de l'appareil, il y a quatre principaux types de communications que l'on va citer et en détaillant ci-après [14][43] :

- Dispositif de relais avec établissement de liaison contrôlé par l'opérateur tel que l'appareil relaie ses informations via d'autres appareils. Dans cette situation, l'appareil a une mauvaise couverture soit à l'intérieur de la cellule, soit l'appareil se trouve au bord de la cellule. Cela permet à l'appareil d'atteindre une qualité de service plus élevée ou une plus grande autonomie. L'eNodeB (Evolved NodeB) est responsable de l'allocation des ressources et de l'établissement des liens. Et pourra authentifier le dispositif de relais et le chiffrement afin de conserver les privilèges.
- Une communication D2D directe avec établissement de liaison contrôlé par l'opérateur où les appareils source et de destination sont capables de se communiquer et d'échanger des données sans avoir besoin d'un eNodeB. Toutefois, cette BS doit toujours créer des liaisons de contrôle pour la gestion des ressources radio. L'opérateur contrôle les adresses des liens, l'authentification, le contrôle de la connexion et l'allocation des ressources.
- Dispositif de relais avec établissement de liaison contrôlé par appareil, ce mode n'utilise pas la BS pour établir et gérer un lien de contrôle. Par conséquent, les appareils source et de destination sont responsables de la gestion de la communication. Ils coordonnent la communication entre eux pour réaliser la transmission de données en utilisant des dispositifs de relais.
- Une communication D2D directe avec établissement de liaison contrôlé par appareil, dont ce mode n'a pas une entité centrale (par exemple, serveur ou eNodeB) pour surveiller l'allocation des ressources entre les appareils. Les appareils source et de destination ont une communication directe entre eux. Par conséquent, les périphériques source et de destination doivent utiliser leurs ressources afin de minimiser les interférences avec d'autres périphériques qu'il soit du même niveau ou du niveau macrocellule.

2.5 Classification de la communication D2D

La technique D2D a été classée en deux catégories, en bande et hors bande. La principale différence entre les deux est la bande de spectre de fréquences dans laquelle la communication D2D fonctionne [1][20][35]. La figure 2.3 montre les classes de la communication D2D :

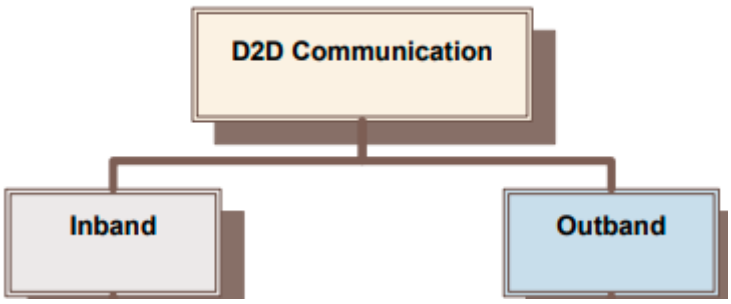

FIGURE 2.3 – Classification de la communication D2D [15].

2.5.1 Communication D2D en Bande (Inband)

En communication Inband, le D2D partage le spectre cellulaire sous licence avec d'autres utilisateurs cellulaires du réseau LTE-A (Long Term Evolution-Advanced), ce dernier signifie une norme de réseau de communication mobile quatrième génération. L'infrastructure réseau, c'est-à-dire que eNodeB contrôle totalement ou partiellement les utilisateurs de D2D. L'eNodeB est responsable de la découverte des équipements D2D potentiels, de l'établissement de la liaison sur la base des informations sur l'état du canal, de l'affectation des ressources radio, en liaison montante ou descendante, du contrôle de l'alimentation en fonction d'un certain niveau seuil prédéfini, ainsi que la coordination des interférences entre les utilisateurs cellulaires et les utilisateurs de D2D.

2.5.2 Communication D2D hors bande (Outband)

Les communications D2D Outband utilisent des bandes de fréquences différentes de celles utilisées pour les communications cellulaires. Dans le cas de communications cellulaires 4G, les communications D2D ne vont pas donc utiliser le spectre du LTE (Long Term Evolution) qui est une norme de communication 3G, mais plutôt d'autres bandes de fréquences c'est à dire un spectre sans licence comme les bandes relatives au Wi-Fi ou encore les bandes ISM (industriel, scientifique et médical).
Le principal avantage de cette catégorie de D2D est qu'elle élimine les interférences entre les utilisateurs D2D et les utilisateurs cellulaires. L'interférence provient de d'autres appareils électroniques sans fil tels que le Bluetooth et le Wi-Fi qui fonctionnent dans la même bande sans licence, avec cette technique il n'est pas possible de contrôler les interférences de l'opérateur. Outband D2D est en outre catégorisé en types «contrôlé» et «autonome». Dans la catégorie

contrôlée, l'interface radio D2D est contrôlée par le réseau cellulaire. Dans la catégorie Autonome, le réseau cellulaire ne contrôle que les liaisons cellulaires alors que les liaisons D2D sont contrôlées par les utilisateurs eux-mêmes. Cependant, l'inconvénient majeur est son interférence intersystème incontrôlable due à la présence d'autres entités communicantes, par exemple, les périphériques Wi-Fi et Bluetooth qui fonctionnent dans la même bande sans licence. Par conséquent, le partage de spectre sans licence ne pourrait pas fournir un environnement contrôlable stable, et peut entraîner une congestion et une mauvaise expérience de la qualité de service, mais également affecter le débit global du réseau. En outre, la sécurité de la transmission D2D et la coordination des communications sur deux bandes différentes avec des interfaces radio indépendantes posent un problème crucial de gestion de l'énergie.

2.6 Applications de la communication D2D

La communication D2D est appliquée dans plusieurs domaines comme le montre la figure 2.4 [15] :

- **Services locaux :** dans le service local, les données des utilisateurs sont directement transmises entre les terminaux et n'impliquent pas le côté réseau, par exemple les applications de médias sociaux, qui sont basées sur un service de proximité. Avec les fonctions de découverte et de communication D2D, un utilisateur peut trouver d'autres utilisateurs à proximité et partager des données ou jouer à des jeux avec eux.
- **Communications d'urgence :** en cas de catastrophe naturelle comme les ouragans, les tremblements de terre, etc., le réseau de communication traditionnel peut ne pas fonctionner en raison des dommages causés. Ce problème peut être résolu en introduisant la communication D2D. Bien que l'infrastructure du réseau de communication puisse être endommagée, un réseau sans fil peut toujours être configuré entre les terminaux sur la base de la connexion D2D.
- **Amélioration de l'IdO :** l'un des objectifs du développement de la communication mobile est d'établir un vaste réseau interconnecté contenant divers types de terminaux, c'est également l'un des points de départ du développement de l'IdO dans le cadre de la communication cellulaire. En combinant D2D avec l'IdO, un réseau sans fil véritablement interconnecté sera créé. La communication V2V dans l'Internet des véhicules IoV (Internet of Vehicles) est un exemple d'amélioration de l'IdO basée sur D2D. Lorsqu'il roule à grande vitesse, un véhicule peut avertir les véhicules à proximité en mode D2D avant de changer de voie ou de ralentir.

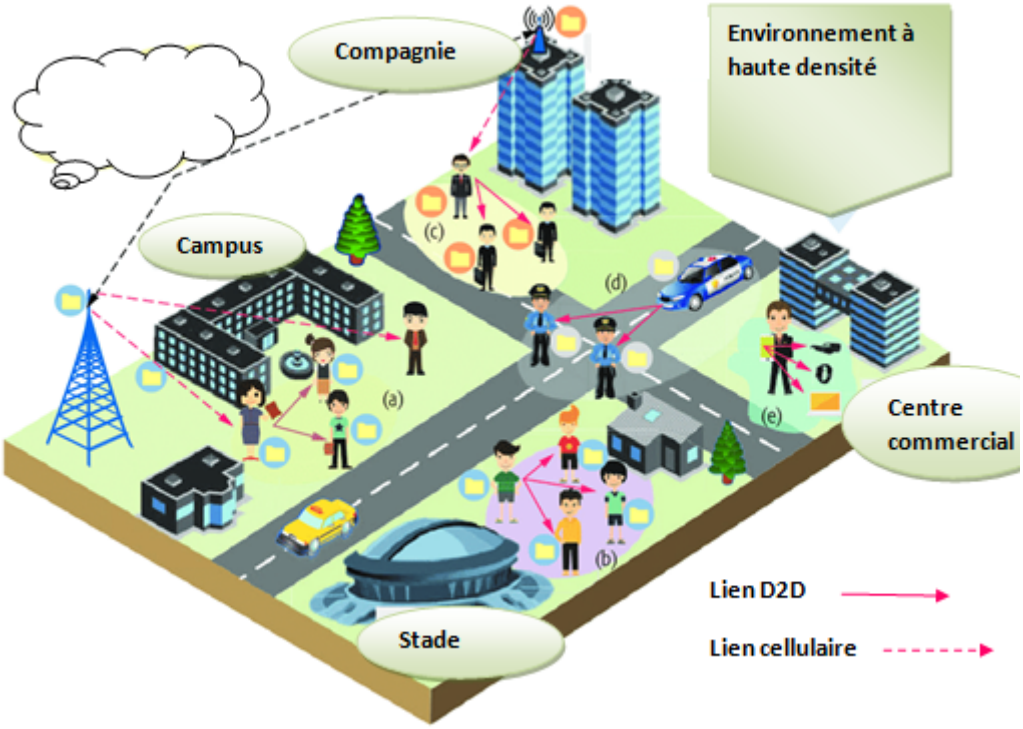

FIGURE 2.4 – Les applications de la communication D2D [44].

2.7 Avantages et limites de la communication D2D

2.7.1 Avantages

La communication D2D peut apporter les principaux avantages suivants [27] :

- Transmissions à haut débit pris en charge également par des appareils situés à distance depuis le BS/AP ;
- Des communications fiables également en cas de panne du réseau, comme cela peut être le cas dans les scénarios catastrophiques ;
- L'économie d'énergie puisque les appareils à proximité immédiate peuvent interagir à un niveau de puissance d'émission plus faible ;
- Un déchargement du trafic qui réduit le nombre de connexions cellulaires ;
- La connectivité hétérogène tenant compte du fait que les communications directes entre les appareils ne reposent pas seulement sur une interface radio cellulaire, mais peuvent être établies au moyen de technologies radio alternatives ;
- Extension de réseau sans ajouter de matériel complexe comme les stations de base.

2.7.2 Limites de la communication D2D

La communication D2D se voit limitée par plusieurs obstacles, on peut citer [33] :

- Les communications D2D nécessitent des techniques de gestion des ressources complexes pour gérer efficacement les périphériques sans interférence ;

- Des mécanismes efficaces de découverte des périphériques, et des procédures de gestion de la mobilité ;
- La puissance de transmission du signal vers un dispositif particulier doit être augmentée à partir de la station de base pour surmonter les interférences environnantes ;
- La communication D2D étant un protocole basé sur la proximité, la distance entre les appareils est limitée en raison de la consommation électrique ;
- Des techniques de cryptage de haut niveau et un protocole de transmission doivent être mis en œuvre pour protéger les informations concernant les utilisateurs.

2.8 Sécurité dans la communication D2D

Malgré tous les avantages des communications D2D, la sécurité est l'une des principales préoccupations qui doivent être bien traitées avant que la technique D2D ne soit largement acceptée et mise en œuvre, en raison de la nature de diffusion de la communication sans fil. Les canaux sans fil sont considérés comme vulnérables à diverses attaques qui remettent en question les principes de base de la sécurité : authentification, confidentialité et disponibilité [38].

2.8.1 Terminologie de base

La sécurité informatique est l'ensemble des moyens techniques qui visent à empêcher l'utilisation non-autorisée. Elle se base sur certains principaux termes cités ci-dessous [21] :

Cryptologie : il s'agit d'une science mathématique comportant deux branches : la cryptographie et la cryptanalyse.

Cryptographie : la cryptographie est l'étude des méthodes donnant la possibilité d'envoyer des données de manière confidentielle sur un support donné.

Cryptanalyse : opposée à la cryptographie, elle a pour but de retrouver le texte clair à partir de textes chiffrés en déterminant les failles des algorithmes utilisés.

Chiffrement : pour réaliser la sécurité dans n'importe quel modèle de communication, il est important de chiffrer les messages transmis. Le chiffrement consiste à transformer une donnée (texte, message, etc.) afin de la rendre incompréhensible par une personne autre que celui qui a crée le message et celui qui en est le destinataire. La fonction permettant de retrouver le texte clair à partir du texte chiffré porte le nom de déchiffrement.

Texte chiffré : est le résultat de l'application d'un chiffrement à un texte clair.

Clé : il s'agit du paramètre impliqué et autorisant des opérations de chiffrement et/ou déchiffrement. Dans le cas d'un algorithme symétrique, la clé est identique lors des deux opérations. Dans le cas d'algorithmes asymétrique elle diffère pour les deux opérations.

Entité (agent) : quelqu'un ou quelque chose qui envoie, reçoit ou modifie de l'information. Elle peut être une personne physique ou morale, un ordinateur, etc.

Expéditeur : entité qui expédie lors d'une transmission d'une information vers une certaine destination.

Récepteur : entité destinée à recevoir l'information dans une transmission à deux parties.

Adversaire : entité malveillante qui n'est pas l'expéditeur ni le récepteur et qui tente de déjouer la sécurité d'une transmission à deux parties, en pratique les adversaires sont appelés attaquants.

Protocole : est une spécification de plusieurs règles pour un type de calcul et de communication particulier.

Canal : moyen de transport de l'information d'une entité à une autre.

Canal sécuritaire : canal qui n'est pas physiquement accessible à un adversaire.

Canal sécurisé : canal ou l'adversaire n'a pas la possibilité de lire, de modifier ou d'effacer.

Cryptage symétrique (chiffrement à clé secrète) : est un chiffrement en utilisant une seule clé pour chiffrer et déchiffrer les données, cette clé doit être partagée avec le destinataire.

Cryptage asymétrique (chiffrement à clés publiques) : est un cryptage qui nécessite deux clés pour fonctionner (les clés existent par paires). Tout d'abord, une clé publique afin de chiffrer les données, et une clé privée utilisée pour décrypter ces données.

Intégrité des données : c'est un service qui garantit que les données n'ont pas été altérées pendant la transmission.

Confidentialité : est la garantie que l'information d'un nœud n'est rendue accessible ou révélée qu'à son destinataire.

Disponibilité : donne une assurance sur la réactivité et le temps de réponse d'un système pour transmettre une information d'une source à la bonne destination.

Authentification
D'une information : prouver qu'une information provient de la source annoncée.
D'une personne : prouver que l'identité est bien celle annoncée.

Authentification mutuelle : processus dans lesquelles deux parties, en général un client et un serveur s'authentifient. Cette authentification permet à chaque partie de connaître l'identité de l'autre. Dans le cadre d'une authentification mutuelle, le serveur demande également un certificat au client. Fonctionnalité également appelée authentification bidirectionnelle.

2.8.2 Objectifs de sécurité dans la communication D2D

Pour sécuriser les communications D2D, des solutions de cryptographie sont nécessaires pour crypter les messages pendant leur transmission via des canaux sans fil. De nombreux algorithmes de cryptage ont été bien développés pour une meilleure sécurité des messages cryptés, mais tous nécessitent que deux appareils se mettent d'accord sur un secret partagé.
En raison de la nature de diffusion des canaux sans fil, les communications sans fil telles que le Wi-Fi et le Bluetooth sont vulnérables à une variété d'attaques qui défient les trois principes de base de la sécurité, à savoir : la confidentialité, l'intégrité et la disponibilité, dont un attaquant a un contrôle total sur le canal sans fil. Ce dernier peut entendre, intercepter et modifier n'importe quel message, et peut également initier une conversation avec n'importe quel autre utilisateur [38].
Un moyen simple d'établir un secret partagé entre deux appareils est que les deux utilisateurs finaux du D2D vont configurer d'une manière interactive une clé secrète via une négociation humaine (comme passer un appel téléphonique s'ils sont à distance), mais ce n'est pas difficile pour l'attaquant de percevoir ce faible secret. Ce qui a poussé les chercheurs à proposer un protocole de Diffie-Hellman (DH) pour établir une clé secrète suffisamment sécurisée qui permet à deux individus de s'entendre sur cette clé secrète, même s'ils ne peuvent échanger des messages que sur des supports publics. Cependant l'inconvénient de ce protocole sa vulnérabilité à ce qu'on appelle l'attaque man-in-the-middle (MITM). La raison essentielle pour laquelle cette attaque est possible est qu'il n'y a pas d'authentification mutuelle entre ces deux appareils. Pour assurer l'authentification souhaitée, une solution intuitive est que les deux appareils mettent la clé secrète obtenue à une fonction de hachage à sens unique, pour générer une valeur de hachage h(K), puis comparer la valeur de hachage via un canal de confiance. Si le processus d'authentification mutuelle est l'accompli, alors les deux appareils peuvent confirmer qu'ils ont établi une clé secrète partagée les uns avec les autres.

2.8.3 Concepts de cryptographie

2.8.3.1 Protocole de Diffie-Hellman (DH)

Le cryptosystème DH est le plus ancien système à clé publique encore en usage, qui permet à deux individus de s'entendre sur une clé secrète, même s'ils ne peuvent échanger des messages que sur chaînes publiques. Le protocole d'accord de clé DH fonctionne comme suit [38] :
Supposons que p et g soient publiquement connus de deux appareils A et B, A et B génèrent tous les deux aléatoirement une valeur a et b respectivement. A calcule $g^a mod\ p$ et l'envoie à B, en conséquence, B calcule $g^b mod\ p$ puis l'envoie à A. A la dernière étape, A calcule $s = (g^b)^a mod\ p$ à partir de la valeur reçue par B, de même B calcule $s = (g^a)^b mod\ p$ à partir de la valeur reçue par A , ce qui est le même nombre que celui obtenu par A.
$(g^a)^b mod\ p$ représente le secret partagé entre A et B, il peut donc être par la suite utilisé comme

clé secrète commune sans que personne ne puisse la calculer pour les communications futures. Le protocole d'accord clé DH est vulnérable à ce qu'on appelle l'attaque MITM.

2.8.3.2 L'attaque man-in-the-middle (MITM)

MITM est une technique de piratage informatique qui a pour but d'intercepter les communications entre deux entités. Dans une transaction entre eux une troisième entité est capable de recevoir les messages des deux et de transmettre d'autres messages. Ainsi elle peut changer les messages concernant l'échange de clés sans que les deux entités s'en aperçoivent [21].
Une interception MITM cible généralement les utilisateurs de boîtes email professionnelles, d'applications bancaires, de sites marchands dans le but de voler leurs identifiants, numéros de cartes et comptes bancaires, etc.
Le protocole d'accord clé DH est vulnérable à cette attaque, puisque g^a et g^b sont transmis sur le canal public, il n'y a aucun moyen pour l'appareil A de savoir avec certitude si g^b vient de l'appareil B, et inversement. L'appareil A établira un secret partagé avec celui qui transmet g^b, et il pourrait certainement ne pas être l'appareil B. Cette attaque peut être évitée à l'aide des protocoles cryptographiques basés sur DH authentifié, qui peut l'empêcher en procédant à une authentification mutuelle [38].

2.8.3.3 Fonction de hachage

Les fonctions de hachage cryptographiques sont des primitives qui prennent des messages de longueur aléatoire comme entrées et produisent des sorties de longueur fixe plus courtes [37], d'où il n'est pas possible de revenir à la chaîne de données initiale.
Les fonctions de hachage sont généralement définies comme H : M $\rightarrow$ D qui prennent un message m $\in$ M en entrée et produisent un digest (haché ou condense) d $\in$ D en sortie tel que d = H(m).
Ces fonctions sont utilisées, par exemple pour la vérification de l'intégrité des messages transmis. On crée pour cela une empreinte du message à transmettre, puis on transmet à la fois le message et l'empreinte. À la réception du message, on calcule l'empreinte du message reçu et on la compare à l'empreinte initiale. Si les deux empreintes correspondent, c'est que le message n'a pu être modifié [24].
Les algorithmes de hachage les plus utilisés actuellement sont :

MD5 (MD signifiant Message Digest) créant une empreinte digitale de 128 bits.

SHA (Secure Hach Algorithm) pouvant être traduit par algorithme de hachage sécurisé créant des empreintes d'une longueur de 160 bits.

2.8.3.4 Codes d'authentification de message

Un code d'authentification de message, ou MAC (Message Authentication Code), est une fonction cryptographique destinée à vérifier l'intégrité de données et à en authentifier l'origine. MAC est le résultat d'une fonction de hachage à sens unique dépendant d'une clé secrète, c'est-à-dire, il calcule à partir d'un message de longueur arbitraire un résumé de longueur fixe qui s'appelle un haché. Mais, contrairement aux fonctions de hachage, ce résumé dépend aussi d'une clé secrète.
Une méthode de calcul de MAC à partir de fonctions de hachage plus élaborées et plus sûres est HMAC (Hash-Based Message Authentication Codes). La méthode HMAC peut être utilisée avec n'importe quelle fonction de hachage itérative telle que MD5 ou SHA [24].
Une pratique courante avec les fonctions de calcul de MAC consiste à tronquer la sortie pour ne garder comme MAC qu'un nombre réduit de bits. Avec HMAC, on peut choisir de ne retenir que quelques bits de gauche par exemple [24].
Dans la méthode HMAC si le périphérique ne connaît pas la clé, alors toutes les balises HMAC ressemblent à des chaînes de bits complètement aléatoires, même si le périphérique connaît ou sélectionne le message reçu [37].

2.9 Conclusion

Dans ce chapitre, nous avons présenté les notions de base ainsi que le domaine de sécurité associés aux communications D2D. Malgré les progrès réalisés dans le domaine de sécurité grâce aux travaux qui ont été effectués afin de sécuriser les communications D2D, les communications sans fil restent vulnérables à une variété d'attaques qui défient les principes de base de la sécurité.
Dans le chapitre suivant, nous intéressons à l'étude de quelques travaux réalisés dans le contexte de la sécurité des communications D2D présentés dans les articles [6][37][38].

Chapitre 3

Echange de clés dans la communication D2D

3.1 Introduction

La protection des communications D2D est l'une des tâches principales du bon fonctionnement et du succès des services D2D. La plupart des études et recherches tendent à rendre leurs solutions applicables et utiles. Dans la littérature, les chercheurs ont proposés diverses protocoles et techniques pour permettre des communications D2D sécurisées, le plus couramment utilisé est le protocole d'échange de clés DH.
Ce chapitre sera consacré à l'étude de quelques travaux récents réalisés dans le contexte de la sécurité des communications D2D, où nous avons pris l'essentiel afin de nous aider dans notre étude.

3.2 Travaux connexes

Shen et al. dans leur article [38], ont étudié les exigences et les défis de sécurité pour les communications D2D, et ont présenté une solution sécurisée et efficace représentée sous un protocole d'accord de clé, qui permet à deux appareils d'établir une clé secrète partagée pour les communications D2D tout en exigeant une quantité minimale d'informations à authentifier mutuellement pour empêcher l'attaque MITM. Leur approche est basée sur le protocole d'accord clé DH qui permet à deux individus de s'entendre sur une clé secrète, même s'ils ne peuvent échanger des messages que sur des supports publics, et le schéma d'engagement permettant à un utilisateur de s'engager sur une valeur choisie tout en la cachant aux autres avec la possibilité de révéler la valeur de l'engagement plus tard.
Ils ont également intégré leur protocole d'accord clé dans le protocole Wi-Fi Direct existant et l'ont mis en œuvre en utilisant de vrais smartphones. Le résultat de la mise en œuvre montre que le protocole proposé est efficace et atteint un niveau élevé de convivialité.

Alam et al. dans [6], ont proposé un protocole de distribution de clé pour LTE basé sur les communications D2D afin d'assurer une communication sécurisée pour trois types de scénarios : le déchargement du réseau, les réseaux sociaux et le sauvetage en cas de catastrophe. Ils ont classé les scénarios D2D et les cas d'utilisation en trois types précédents en fonction de la disponibilité d'un réseau et l'existence d'une application d'utilisateur.
Ces deux aspects affectent la gestion du réseau tel que, les règles de découverte d'appareils, les flux de signaux de contrôle, etc., ce qui entraîne par conséquent des exigences de sécurité différentes. Ils ont fourni un aperçu de l'architecture de sécurité, des exigences de sécurité pour ces trois types. De plus, avec la réutilisation des fonctions et algorithmes existants, ils ont proposé des solutions d'authentification et de gestion de clés pour ces trois types de scénarios.
Dans la technique proposée, le réseau central CN (Central Network) est considéré comme la principale unité fonctionnelle qui contrôle la génération et la distribution de la clé de session entre deux équipements UEs (User Equipment) de communication D2D. Afin de surmonter le risque de fuite de la transmission des clés de session entre le CN et les UEs, le CN commence par les XORs des clés de deux UEs D2D et envoie la clé XOR-ed à chaque UE. Une fois la clé XOR-ed reçue, chaque UE dérive une autre clé à l'aide de sa propre clé. Après cela, chaque UE reçoit une paire de clés de session pour protéger les communications D2D.
Le schéma XOR de distribution de clés proposé offre une protection supplémentaire, puisque les informations transmises sont non seulement cryptées, mais aussi XOR-ed. Il utilise séparément deux clés pour la communication bidirectionnelle. Cependant, cette opération XOR ne peut pas éliminer toujours le risque d'interception de clé de session. Si une partie de la clé de session est divulguée, l'autre partie peut être calculée facilement par l'opération XOR. La façon la plus sûre est de ne pas transmettre la clé de session entre le CN et les UEs ou entre les UEs.

Dans l'article [37], les auteurs ont proposé trois nouveaux protocoles d'échange de clés pour une communication D2D sécurisée dans un réseau cellulaire. Ces protocoles sont basés sur l'échange de clés standard basé sur DH et d'autres fonctions cryptographiques, mais ils diffèrent par le rôle de l'eNodeB dans le processus d'authentification. Les auteurs ont examiné les cas d'utilisation du déchargement du trafic et de réseautage social. Puis, ils ont présenté une analyse détaillée des menaces pour les protocoles proposés qui abordent la vulnérabilité typique des scénarios D2D, c'est-à-dire, l'attaque MITM, et plus en plus sont sécuritaires et complexes.
Ils ont considéré les deux cas d'utilisation des communications D2D les plus courants de déchargement du trafic et de réseautage social. Dans le scénario de déchargement du trafic, les deux équipements UEs sont connectés au même eNodeB et que les applications de chacun des dispositifs ne nécessitent pas de lien D2D. Cependant, le réseau utilise ce scénario pour réduire la charge sur le noyau et le réseau d'accès en assurant une liaison D2D entre les deux UEs. Dans le scénario réseautage social, les applications de chaque appareil nécessitent un lien D2D entre elles.
Pour les protocoles proposés qui impliquent seulement les deux UEs et l'eNodeB en utilisant

des clés publiques DH pour les deux UEs D2D, sont indiquées par g^a et g^b, une fonction de hachage clé, désormais appelée fonction de contrôle, qui associe à une valeur de contrôle. De plus, MAC ou HMAC pour la génération des clés.
Dans notre étude, nous nous sommes basées sur l'article précédent, on utilise l'outil d'analyse SPAN AVISPA que nous allons présenter dans le chapitre suivant, afin de vérifier formellement les deux protocoles d'échange de clés proposés dans l'article pour créer un lien D2D sécurisé entre les équipements.

3.3 Contexte de notre travail

Dans cette section, nous détaillerons les deux premiers protocoles d'échange de clés proposés dans l'article [37], en portant une petite modification au deuxième protocole qui est l'envoie de la clé K sur un canal privé au lieu de l'envoyer sur un canal public afin de vérifier la sécurité de cette clé entre les dispositifs, de plus nous vérifierons la sécurité du premier protocole avec trois dispositifs.

3.3.1 Premier protocole

Dans ce protocole, ils ont utilisé une fonction de vérification pour l'authentification, et une fonction MAC régulière pour éviter les attaques inutiles. Les étapes détaillées de ce protocole sont les suivantes :

- Deux appareils échangent leurs clés publiques DH sur le canal public. Les clés reçues par UE-1 et UE-2 sont dénotées respectivement par g^b et g^a (dans notre code on les a noté respectivement par g^{Nb}, g^{Na}) ;
- Les deux dispositifs envoient un accusé de réception (ACK) à l'eNodeB au sujet de l'échange mutuel. Après cette étape, chaque échange se fait via un canal dédié crypté ;
- L'eNodeB répond à l'un ou l'autre des dispositifs pour lancer le processus d'authentification. Ici dans l'illustration, ils ont choisi UE-2 pour lancer l'authentification ;
- L'UE-2 génère une clé K aléatoire de 16-20 bits en utilisant une fonction MAC régulière qui utilise une clé plus courte (16-20 bits). Cette clé est utilisée pour générer la valeur de contrôle à l'aide de la fonction de contrôle $\mathrm{Ck}(g^a \parallel g^b)$. La clé et la valeur de contrôle sont envoyées à l'eNodeB ;
- Lors de la réception des valeurs, l'eNodeB envoie la clé et la valeur de contrôle à l'UE-1.
- L'UE-1, avec l'aide de K reçue, calcule la valeur de $\mathrm{Ck}(g^a \parallel g^b)$ et la compare avec la valeur qu'il reçoit ;
- Selon le match, l'UE-1 envoie un ACK d'acceptation ou rejet à l'eNodeB, que l'eNodeB envoie à l'UE-2. Si l'ACK est une acceptation, alors un lien D2D est établi entre eux.

3.3.2 Deuxième protocole

Dans ce protocole, l'eNodeB dispose d'une fonction de comparaison supplémentaire pour plus de sécurité au niveau du codage avec une complexité de calcul légèrement supérieure. Les étapes détaillées de ce protocole sont les suivantes :

- Deux appareils UE-1 et UE-2 échangent leurs clés publiques DH sur le canal public, les clés reçues sont dénotées respectivement par g^b et g^a (et g^{Na}, g^{Nb} dans notre spécification CAS+). Ensuite, un ACK est envoyé par les deux appareils à l'eNodeB au sujet de l'échange mutuel ;
- L'eNodeB répond à l'un ou l'autre des dispositifs pour lancer le processus d'authentification. nous choisissons UE-2 pour lancer l'authentification. Cela se fait via le canal dédié crypté ;
- L'UE-2 génère une clé aléatoire K (16-20 bits) et l'envoie par le canal privé à UE-1 qui attend et la reçoit ;
- Grâce à la clé K reçue et la fonction de contrôle Ck($g^a \parallel g^b$), l'UE-1 calcule la valeur de contrôle et envoie ces deux valeurs à l'eNodeB associée ;
- L'UE-2 à l'aide de sa clé K qu'il a précédemment générée et de la fonction de contrôle Ck($g^a \parallel g^b$), calcule la valeur de contrôle et envoie ces deux valeurs à l'eNodeB associée ;
- À ce stade, l'eNodeB compare les clés reçues et les valeurs de contrôle correspondantes et envoie un ACK d'acceptation ou rejet aux deux appareils ;
- Si l'ACK est une acceptation sur les deux appareils, un lien D2D est établi entre eux.

3.3.3 Premier protocole avec trois dispositifs

Dans ce protocole on a ajouté un troisième dispositif UE-3 dont le principe reste le même avec celui du premier protocole. Les étapes de ce protocole sont les suivantes :

- Trois appareils échangent leurs clés publiques DH sur un canal public. Les clés reçues par UE-1 sont g^b et g^c, par UE-2 g^a et g^c et par UE-3 g^a et g^b (g^{Na}, g^{Nb}, g^c dans notre spécification CAS+). Puis un accusé de réception ACK est envoyé par les trois dispositifs à l'eNodeB au sujet de l'échange mutuel ;
- L'eNodeB répond à l'un des dispositifs pour lancer le processus d'authentification, et nous choisissons UE-1 ;
- L'UE-1 génère une clé K aléatoire en utilisant une fonction MAC régulière qui utilise une clé plus courte (16-20 bits). Cette clé est utilisée pour générer la valeur de contrôle à l'aide de la fonction de contrôle Ck($g^a \parallel g^b \parallel g^c$). La clé et la valeur de contrôle sont envoyées à l'eNodeB ;
- Lors de la réception des valeurs, l'eNodeB envoie la clé et la valeur de contrôle à UE-2 et UE-3.
- L'UE-2 et UE-3 à l'aide de la clé K reçue, calculent la valeur de Ck($g^a \parallel g^b \parallel g^c$) et la comparent avec la valeur qu'ils aient reçu. UE-2 et UE-3 envoient un ACK d'acceptation

ou rejet à l'eNodeB, que l'eNodeB envoie à l'UE-1. Si l'ACK est une acceptation donc un lien D2D est établi entre eux.

3.3.4 Analyse des menaces pour les protocoles étudiés

La sécurité de tout protocole est évaluée par la manière dont il répond à une menace externe par un adversaire. D'où, les auteurs dans [37] ont analysé les deux protocoles proposés sous la plus commune attaque MITM, et c'est ce qu'on va vérifier après en utilisant SPAN AVISPA. Les étapes intermédiaires de défense aux attaques à chaque transition sont décrites ci-dessus :

3.3.4.1 Protocole-1 contre l'attaque MITM

Un tiers adversaire malveillant dispositif noté MD est utilisé pour démontrer l'attaque du MITM. Dans cet exemple, il est supposé que l'UE-1 démarre l'échange.

- Sur le canal public, l'UE-1 et le MD peuvent tous les deux recevoir g^b. MD reçoit la clé et propage une version modifiée de $g^{b'}$ sur le canal ;
- Les deux clés g^b et $g^{b'}$ peuvent être reçues par l'UE-1 car il s'agit d'un canal public. Le reste du protocole fonctionne sans problème si l'UE-1 reçoit g^b ;
- Dans la première étape, le MD intercepte l'échange de la clé DH. puisqu'il est exécuté par le canal public ;
- Le MD reçoit g^a de l'UE-1 et envoie une clé modifiée $g^{a'}$ à l'UE-2 ;
- Comme il s'agit d'un canal public, l'UE-2 peut recevoir à la fois g^a et $g^{a'}$. Dans les deux scénarios, il envoie sa clé DH g^b sur un canal public ;
- Dans la situation où il reçoit $g^{b'}$, l'UE-1 génère une clé aléatoire K en utilisant une fonction MAC. Cette clé est utilisée pour générer la valeur de vérification en utilisant la fonction de vérification $\mathrm{Ck}(g^a \parallel g^{b'})$, la clé et la valeur de vérification sont envoyées à l'eNodeB.
- Après avoir reçu les valeurs, l'eNodeB envoie à la fois la clé et la valeur de contrôle à l'UE-2 ;
- L'UE-2, à l'aide du K reçue, calcule la valeur de $\mathrm{Ck}(g^{a'} \parallel g^b)$ et la compare à celle reçue. Comme la valeur de la clé DH est modifiée, la comparaison aboutit à une non-correspondance et la communication s'arrête.

3.3.4.2 Protocole-2 contre l'attaque MITM

Ce scénario est similaire à celui du premier protocole. Après l'échange de clés DH, la valeur de vérification est calculée à la fois sur les dispositifs UE-1 et UE-2, et envoyée à l'eNodeB. Ceci est expliqué comme suit :

- Dans la première étape, puisque l'exécution se fait via un canal public, le MD intercepte l'échange de la clé DH ;

- Le MD reçoit g^a de l'UE-1 et envoie un $g^{a\prime}$ modifiée à l'UE-2 ;
- Comme il s'agit d'un canal public, l'UE-2 peut recevoir à la fois g^a et $g^{a\prime}$. Dans les deux scénarios, il envoie sa clé DH g^b par le canal public ;
- Sur le canal public, l'UE-1 et le MD peuvent tous deux recevoir le g^b. Que le MD reçoive la clé et propage une version modifiée $g^{b\prime}$ sur le canal ;
- Les deux clés g^b et $g^{b\prime}$ peuvent être reçues par l'UE-1. Encore une fois puisqu'il s'agit d'un canal public. Le reste du protocole fonctionne parfaitement si l'UE-1 reçoit g^b, la clé DH originale de l'UE-2 ;
- Après que l'eNodeB a envoyé l'ACK d'initiation, la clé K est envoyée par l'UE-2 sur le canal privé ;
- Puisque les clés DH sont modifiées, les valeurs de contrôle générées par les deux dispositifs, Ck($g^a \parallel g^{b'}$) et Ck($g^{a'} \parallel g^b$) seront différentes, ce qui entraînera une non-concordance au niveau de l'eNodeB ;
- Ensuite, un ACK de rejet est envoyé aux deux dispositifs par l'eNodeB et la communication se termine.

3.3.4.3 Protocole-1 avec trois dispositifs contre l'attaque MITM

Un tiers adversaire malveillant dispositif noté MD est utilisé pour démontrer l'attaque du MITM. Dans cet exemple, il est supposé que l'UE-1 démarre l'échange.

- Sur le canal public, l'UE-1 et le MD peuvent tous les deux recevoir g^b et g^c. MD reçoit les clés et propage une version modifiée de $g^{b'}$ et $g^{c'}$ sur le canal ;
- Les clés g^b, $g^{b'}$, g^c et $g^{c'}$ peuvent être reçues par l'UE-1 car il s'agit d'un canal public. Le reste du protocole fonctionne sans problème si l'UE-1 reçoit g^bet g^c ;
- Les clés g^c et $g^{c'}$ peuvent être reçues par l'UE-2 car il s'agit d'un canal public. Le reste du protocole fonctionne sans problème si l'UE-2 reçoit g^c ;
- Les clés g^b et $g^{b'}$ peuvent être reçues par l'UE-3 car il s'agit d'un canal public. Le reste du protocole fonctionne sans problème si l'UE-3 reçoit g^b ;
- Dans la première étape, le MD intercepte l'échange de la clé DH. puisqu'il est exécuté par le canal public ;
- Le MD reçoit g^a de l'UE-1 et envoie une clé modifiée $g^{a'}$ à l'UE-2 et l'UE-3 ;
- Comme il s'agit d'un canal public, l'UE-2 et l'UE-3 peuvent recevoir à la fois g^a et $g^{a'}$. Dans les deux scénarios, ils envoient respectivement leur clés DH g^b et g^c sur un canal public ;
- Dans la situation où il reçoit $g^{b'}$ et $g^{c'}$, l'UE-1 génère une clé aléatoire K en utilisant une fonction MAC. Cette clé est utilisée pour générer la valeur de vérification en utilisant la fonction de vérification Ck($g^a \parallel g^{b'} \parallel g^{c'}$), la clé et la valeur de vérification sont envoyées à l'eNodeB ;
- Après avoir reçu les valeurs, l'eNodeB envoie à la fois la clé et la valeur de contrôle à l'UE-2 et l'UE-3 ;
- L'UE-2 et L'UE-3 à l'aide du K reçue, calculent respectivement la valeur de Ck($g^{a'} \parallel$

$g^b \parallel g^{c'}$) et Ck($g^{a'} \parallel g^{b'} \parallel g^c$), puis la comparent à celle reçue. Comme la valeur de la clé DH est modifiée, la comparaison aboutit à une non-correspondance et la communication s'arrête.

3.4 Conclusion

Dans ce chapitre, on a donné un aperçu général des travaux réalisés dans le domaine de sécurité des communications D2D. Ensuite, on s'est concentré sur l'étude des deux premiers protocoles d'échange de clés présentés dans l'article [37], ainsi que l'analyse des menaces.
Dans le chapitre qui suit, nous intéressons à la vérification de ces protocoles de sécurité D2D en utilisant une plateforme de validation automatique SPAN AVISPA.

Chapitre 4

Vérification des protocoles de sécurité D2D avec l'outil SPAN AVISPA

4.1 Introduction

Les chercheurs ont découvert des failles de sécurité sur plusieurs protocoles cryptographique publiés, pour cela il est important de vérifier leurs sécurité automatiquement avant leurs mises en service car leur sécurité n'est pas garantie par l'usage des méthodes de chiffrement seulement, mais aussi par une vérification automatique et formelle [25].
Dans le domaine de vérification automatique des protocoles de sécurité, il y a plusieurs analyseurs de protocoles, mais la plateforme AVISPA (Automated Validation of Internet Security Protocols and Applications) est l'analyseur le plus connu qui modélise un grand nombre de protocoles [10].
Dans ce chapitre nous présentons notre travail qui s'articule sur la vérification des protocoles de sécurité D2D, pour cela nous avons utilisé une plateforme de validation automatique SPAN (Security Protocol ANimator for AVISPA) AVISPA utilisant un langage abstrait formel de spécification CAS+ qui permet la spécification modulaire afin de vérifier les propriétés de sécurité.

4.2 Outil de vérification formelle SPAN AVISPA

AVISPA a été développé en 2004 par Basin et al. dans le cadre d'un projet européen [16]. C'est une plateforme gratuite qui permet l'analyse et la validation de manière automatique des protocoles de sécurité. Une telle validation repose sur la spécification des protocoles de sécurité à l'aide d'un langage d'entrée appelé HLPSL (High-Level Protocol Specification Language). Ces spécifications sont réécrites dans un format intermédiaire IF (Intermediate Format) grâce à un traducteur. AVISPA utilise quatre autres outils (back-ends) qui prennent en entrée le format IF, et qui offrent la possibilité de faire quatre analyses différentes du même protocole.

Avec AVISPA on peut vérifier n'importe quel langage convertible en format intermédiaire IF. Finalement, soit une attaque est identifiée, soit le protocole est considéré comme fiable [32].
L'outil AVISPA est également compatible avec l'outil graphique SPAN, possède une présentation des résultats sous forme de graphe et de trace d'attaque compréhensible et rendre les choses plus accessible au grand public il offre un environnement de simulation visuel plus convivial où les résultats sont plus facile à interpréter [16].
L'interface graphique de l'environnement SPAN AVISPA est montrée sur la figure 4.1

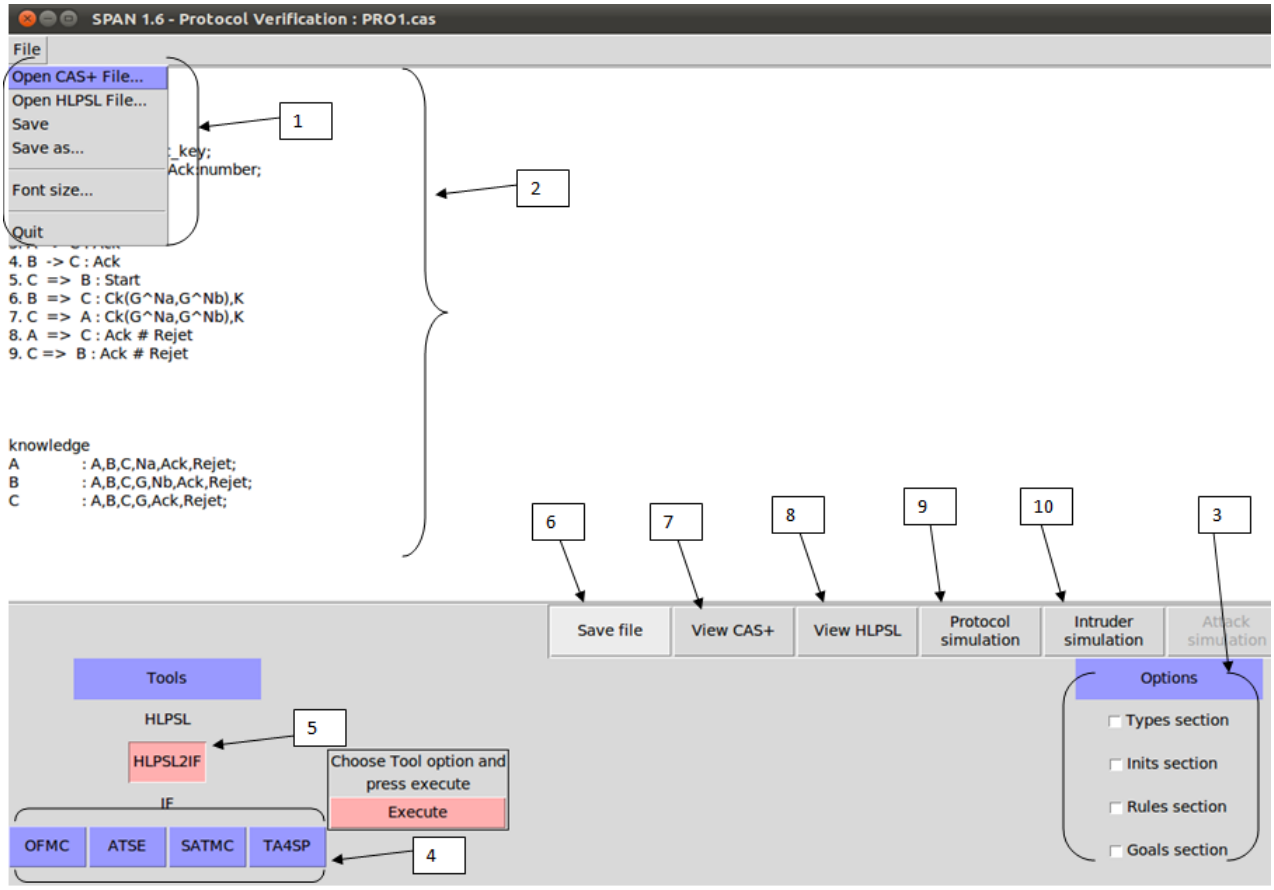

FIGURE 4.1 – Interface graphique de SPAN AVISPA.

1 : Ouvrir ou enregistrer une spécification HLPSL ou quitter l'application

2 : Zone de la spécification CAS+

3 : Les options de l'outil sélectionné

4 : Outils de vérification

5 : Translation de HLPSL à une forme intermédiaire

6 : Enregistrer le fichier modifié HLPSL ou CAS+

7 : Voir le code CAS+

8 : Voir le code HLPSL

9 : Trace du protocole en mode normal

10 : Trace du protocole en mode intrus

Dans l'outil SPAN on trouve deux modes de simulation, en mode normal, les seules transitions qu'on voit sont celles où l'émetteur et le récepteur du message sont d'accord sur le message comme montrés dans la figure 4.2. Dans le cadre de droite (1) ainsi que dans la fenêtre à gauche (2), SPAN affiche les messages déjà envoyés. Dans le cadre de gauche (3), on voit les envois de messages qu'on peut déclencher d'un double-clique. S'il n'y a plus de transitions, c'est qu'on est arrivé à la fin du protocole ou qu'il y a une erreur dans la spécification. Et en mode intrus, tous les messages peuvent être envoyés à l'intrus et chaque message qu'il reçoit enrichit ses connaissances [16].

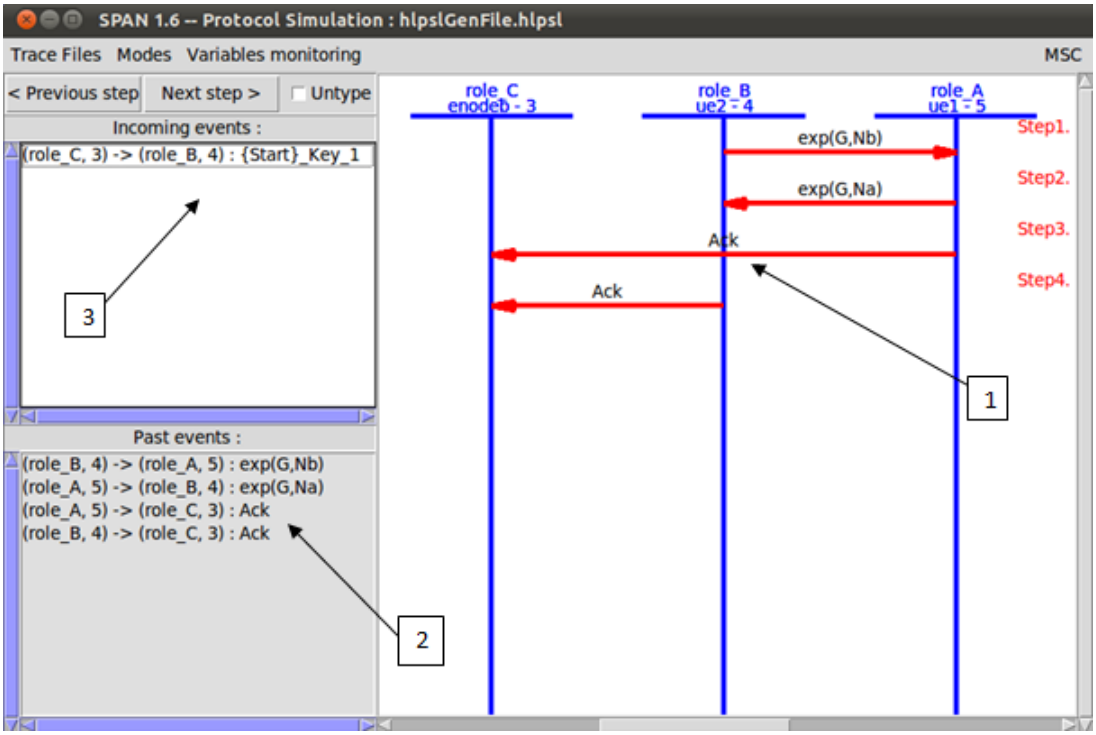

FIGURE 4.2 – Trace d'un protocole en mode normal.

4.3 Architecture d'AVISPA

L'outil supporte quatre moteurs de vérifications dis "backends" permettent de vérifier les propriétés de sécurité, selon les besoins. Ils prennent en entrée le format intermédiaire IF [25] :

- **OFMC (The On-the-fly Model-Checker) :** implémente des techniques symboliques correctes et également complètes. Il supporte la spécification des opérateurs à propriétés algébriques tels que le OU exclusif ou encore l'Exponentielle.
- **CL-AtSe (Constraint-Logic-based Attack Searcher) :** c'est un outil basé sur les contraintes. Il permet de faire une traduction d'une spécification d'un protocole de sécurité sous forme de relations de transition au format IF, vers un ensemble de

contraintes qui peuvent être utilisées pour trouver des attaques sur le protocole en question.

- **SATMC (The SAT-based Model-Checker) :** utilise un état transitoire et recherche les éventuelles violations d'un protocole donné. La sortie est une représentation mathématique de la violation et la transforme en attaque.
- **SAT TA4SP (Tree Automata based on Automatic Approximations for the Analysis of Security Protocols) :** montre la vulnérabilité d'un protocole en faisant une estimation des capacités de l'intrus. Cette méthode permet de savoir si un certain état est accessible ou non et que l'intrus peut savoir certaines connaissance ou non et ainsi de conclure l'absence d'attaque sur le secret pour des scénarios exécutés un nombre indéterminé de fois.

La structure de l'outil AVISPA est représentée sur la figure 4.3.

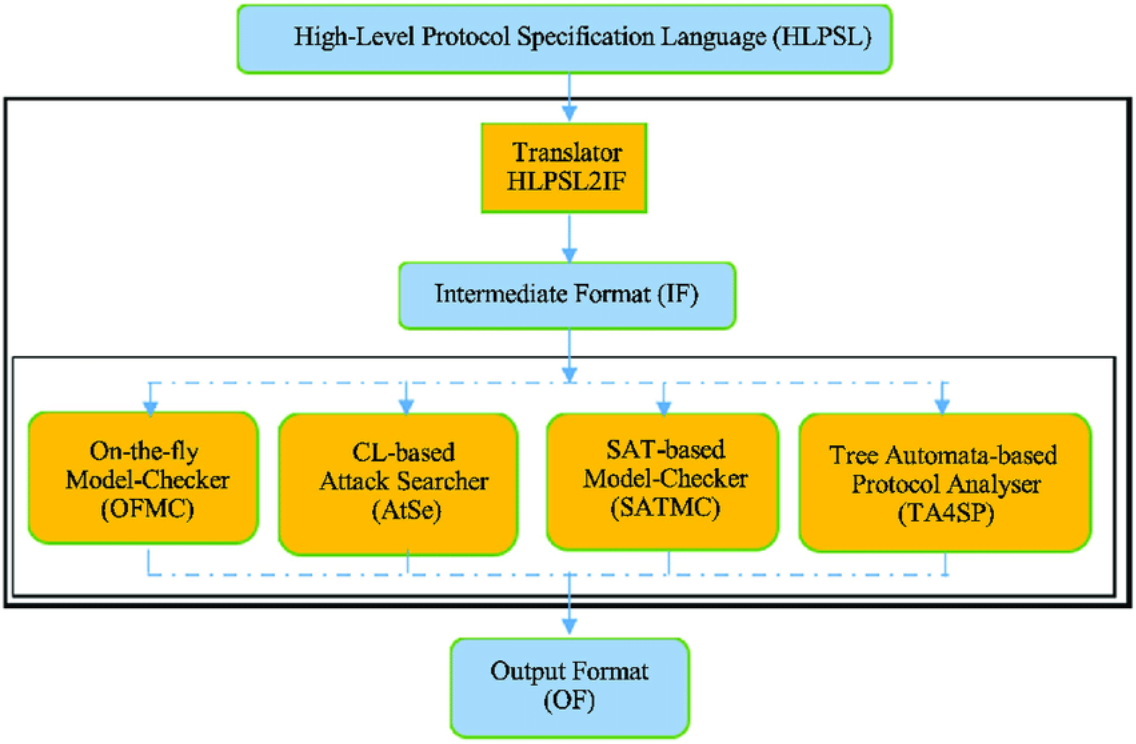

FIGURE 4.3 – Architecture d'AVISPA [25].

4.4 Langage de spécification HLPSL

HLPSL est un langage de spécification de protocoles de sécurité inspiré des travaux de Lamport à propos de la logique temporelle des actions TLA (Temporal Logic of Actions). La représentation des protocoles de sécurité est basée sur des systèmes d'états/transitions sur lesquels sera effectuée la vérification des propriétés de sûreté qui sont exprimées en logique temporelle linéair [25]. Dans les spécifications HLPSL, sont représentées toutes les entités qui peuvent intervenir dans le protocole en question. Cette description est faite dans un fichier qui a pour extension .hlpsl et qui fera l'objet d'une traduction à l'aide du traducteur hlpsl2if

afin de générer un fichier .if. Ensuite, les différents back-end utilisent ce format pour vérifier le protocole spécifié [30].

4.5 Langage de spécification CAS+

CAS+ (Common Authentication Specification) est un langage simple à lire et à écrire, mais dont le niveau de précision dans la spécification égale celui de HLPSL ce langage se fait en six parties [16] :

- **Déclaration d'identifiants (identifiers) :** déclarer toutes les valeurs, agents et clés apparaissant dans la section « messages », ils doivent être de l'un des types suivants : user, public_key, symetric_key, function, number. Les noms de variables doivent commencer par une majuscule.
- **Messages (message) :** ils permettent de décrire les étapes de communication dans le protocole.
  i. Si $\rightarrow$ Ri : Mi
  i représente le numéro de l'étape, Si et Ri des utilisateurs (émetteur et récepteur) et Mi le message. La flèche représente le type de canal utilisé. Il peut être du type Dolev-Yao ($\rightarrow$), du type protégé en lecture et écriture ($\Longrightarrow$), ou du type protégé en écriture ($\rightsquigarrow$).
- **Connaissances des agents (knowledge) :** dans cette partie, pour chaque agent, nous déclarons les informations qu'il connaît avant de commencer la session de protocole. Ces informations connues ne peuvent être que des variables de la section "identifiers" précédente. On considère que chaque utilisateur connait implicitement son nom. Cette section ne doit rien indiquer sur l'intrusion.
- **Sessions (sessions_instances) :** il s'agit d'une session standard des identifiants qui sont des constants arbitraires choisis pour instancier les variables. Il est aussi possible d'instancier l'intrus en utilisant l'identifiant «i». Les sessions s'exécutent en parallèle.
- **Connaissances de l'intrus (intruder_knowledge) :** dans cette session l'intrus a la possibilité de connaitre les identifiants.
- **Objectifs de la vérification (Goals) :** ce sont les propriétés de sécurité que l'on souhaite tester dans notre protocole. On en distingue trois familles de buts : la confidentialité (secrecy), l'authenticité (authentification), l'intégrité (integrity).

4.6 Étapes de vérification d'une spécification CAS+

- On commence par la spécification du protocole à tester grâce au langage CAS+, ainsi que les propriétés à vérifier ;
- On lance AVISPA à l'aide de l'interface graphique SPAN tout en précisant le simple fichier CAS+, il nous est demandé si on souhaite générer HLPSL. Disons oui. S'il y a des erreurs dans notre fichier CAS+, elles sont affichées dans la fenêtre d'édition. Sinon,

le fichier CAS+ se charge et le HLPSL est généré (et par défaut AVISPA utilise OFMC backend) ;

- Lancer l'exécution du protocole afin de vérifier s'il est sécurisé ou non ;
- Simuler le protocole en cliquant sur le bouton « Protocol simulation» dont les échanges de messages peuvent être visualisés graphiquement ;
- Une fois que nous sommes satisfaits de notre spécification CAS+ dans la simulation de protocole, nous enregistrons le fichier HLPSL généré.

4.7 Les hypothèses de vérification

Dans le cadre de la modélisation des protocoles de sécurité, il est nécessaire de modéliser également l'intrus, c'est-à-dire, de définir son comportement et de le limiter. Pour cela, les hypothèses utilisées sont rassemblées sous le nom de « modèle de Delev-Yao ». Le modèle de Dolev-Yao est un des premiers modèles formels pour la vérification de protocoles cryptographiques. Ce modèle est basé sur deux hypothèses importantes qui sont [10] :

- **Le chiffrement parfait :** dans ce modèle on fait l'hypothèse de cryptographie parfaite, ainsi les messages chiffrés sont donc analysés comme des boites noires contenant un message qu'il n'est possible d'ouvrir que si la clé de déchiffrement est connue. C'est à dire un intrus ne peut déchiffrer un message m chiffré avec une clé k que s'il possède l'inverse de cette clé.
- **L'intrus est le réseau :** l'intrus peut intercepter et remplacer les messages envoyés par les acteurs honnêtes du protocole, et leur envoyer des messages sous une fausse identité. Donc, on considère que l'intrus a le contrôle total du réseau. Il connait toutes les données publiques des agents, dispose des privilèges et des clés des agents malhonnêtes. Ainsi, l'intrus peut déchiffrer un message s'il connaît la clé de déchiffrement, il peut chiffrer un message avec n'importe quelle clé en sa possession et il est capable de mémoriser, effacer, construire en envoyer tous les messages s'il a la clé.

Dolev et Yao ont modélisé les capacités de l'intrus par un modèle de déduction c'est-à-dire qui représente tous les messages qu'un intrus peut fabriquer par un ensemble de règles. Pour montrer qu'un protocole respecte une propriété donnée, il faut fournir une preuve formelle que quelles que soient les actions de l'attaquant, la propriété désirée est maintenue pour le protocole [13].

4.8 Vérification formelle des deux protocoles de sécurité D2D

4.8.1 Spécification formelle du premier protocole en langage CAS+

Nous avons exécuté sur SPAN le code du programme écrit en CAS+ détaillé ci-dessus, dans notre code, la fonction de contrôle Ck($g^a \parallel g^b$)est notée par Ck($g^{Na} \parallel g^{Nb}$) :

```
protocol protocole1;
identifiers
A,B,C   : user;
Ck          : function;
K           : symmetric_key;
Na, Nb, G,Start,Rejet,Ack:number;

messages
1. B -> A : G^Nb
2. A -> B : G^Na
3. A  -> C : Ack
4. B  -> C : Ack
5. C  =>  B : Start
6. B  =>  C : Ck(G^Na,G^Nb),K
7. C  =>  A : Ck(G^Na,G^Nb),K
8. A  =>  C : Ack # Rejet
9. C =>  B : Ack # Rejet

knowledge
A           : A,B,C,Na,Ack,Rejet;
B           : A,B,C,G,Nb,Ack,Rejet;
C           : A,B,C,G,Ack,Rejet;


session_instances
 [A:ue1,B:ue2,C:enodeb,Na:a,Nb:b,Ack:ack,Start:start,Rejet:rejet,G:g,K:k];

intruder_knowledge
 ue1,ue2,enodeb,g;

goal
secrecy_of K [A,B];
```

FIGURE 4.4 – Code du premier protocole en CAS+.

4.8.1.1 Simulation du premier protocole

On lance tout d'abord le code source écrit en CAS+, une fois l'exécution s'est déroulée sans erreurs, on peut simuler le protocole entre les deux entités. La figure suivante présente les différentes étapes de la simulation du protocole :

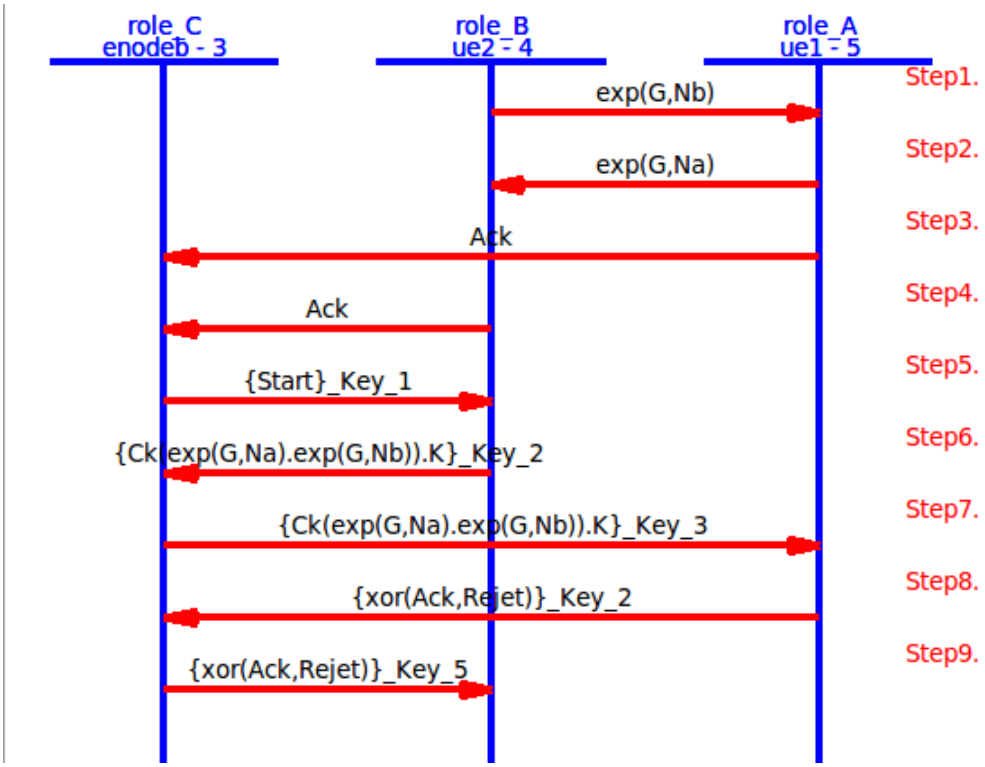

FIGURE 4.5 – Trace du premier protocole.

4.8.1.2 Résultat de la vérification

Le but de ce test est de trouver les vulnérabilités de sécurité dans le design proposé. L'outil AVISPA analyse et range le résultat suivant en deux catégories : safe pour les protocoles restés sûrs, et unsafe pour les protocoles contenant des failles. AVISPA vérifie s'il existe une attaque lors de l'exécution du protocole. En réalité, si un protocole est non sécurisé, AVISPA donnera la trace détaillée de l'attaque et il nous montre comment des dommages peuvent être faits.
Nos vérifications sont réalisées par le back-end OFMC et le résultat que nous avons obtenu est le suivant :

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/hlpslGenFile.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.14s
  visitedNodes: 64 nodes
  depth: 10 plies
```

FIGURE 4.6 – Résultat OFMC du premier protocole.

- La première section SUMMARY indique si le protocole est sécurisé ou non, ou si l'analyse n'a pas été concluante. Dans notre cas le résultat est SAFE ;
- La deuxième section DETAILS décrit sous quelles conditions le protocole est déclaré sûr ou non, sous quelles conditions une attaque est trouvée et finalement pourquoi l'analyse n'a pas été concluante ;
- La section PROTOCOL rappelle le nom du protocole analysé ;
- La section GOAL présente le but de l'analyse, comme par exemple la confidentialité de la clé de chiffrement des données ;
- La section BACKEND désigne le traducteur des spécifications HLPSL.

Donc l'outil AVISPA nous a démontré que notre vérification ne contient pas de faille de sécurité. Les résultats nous confirment qu'il n'y a aucune brèche de sécurité, notamment la confidentialité. Le rapport obtenu dans la figure justifie que le protocole est sécurisé. Il est bien protégé contre les différentes attaques telles que MITM.

4.8.2 Spécification formelle du deuxième protocole en langage CAS+

Dans le code du deuxième protocole présenté sur la figure 4.7, nous avons les deux clés K1 et K2 qui représentent la même clé K d'où K= K1= K2 (K1 et K2 sont des instanciations de K), afin d'éviter les problèmes de compilation au niveau d'OFMC.

```
protocol test;

identifiers
A,B,C                  :user ;
Ck                     :function;
K1,K2 ,K  ,N           :symmetric_key;
Ack,Start ,Reject, Na,Nb,G   :number;

 messages
1. A-> B    : G^Na
2. B-> A    : G^Nb
3. A-> C    : Ack
4. B-> C    : Ack
5. C-> B    : {Start}N
6. B => A   : K
7. A => C  : K1, Ck(G^Na,G^Nb)
8. B => C  : K2, Ck(G^Na,G^Nb)
9. C => B   : Ack# Reject
10.C =>  A  : Ack# Reject

knowledge
A   : A, B, C,Na,Ack,Reject,G;
B   : A,C,B,Nb,Ack ,Reject,G;
C   : C,A,B ,Ack,Reject;

session_instances

[A:ue1,B:ue2 ,C:enodeb,G:g,Ack:ack,Reject:reject,
 Na:a,Nb:b,Start:start,K:k,K1:k,K2:k];

intruder_knowledge
ue1,enodeb,ue2;

goal
secrecy_of K [A,B];
secrecy_of K1[A,B];
secrecy_of K2[A,B];
```

FIGURE 4.7 – Code du deuxième protocole en langage CAS+.

4.8.2.1 Simulation du protocole

On lance tout d'abord le code source écrit sous CAS+, après l'exécution sans erreurs du protocole, on lance la simulation en cliquant sur le bouton « Protocol simulation» dont les échanges de messages entre les entités peuvent être visualisés graphiquement comme sont représentés dans la figure qui suit :

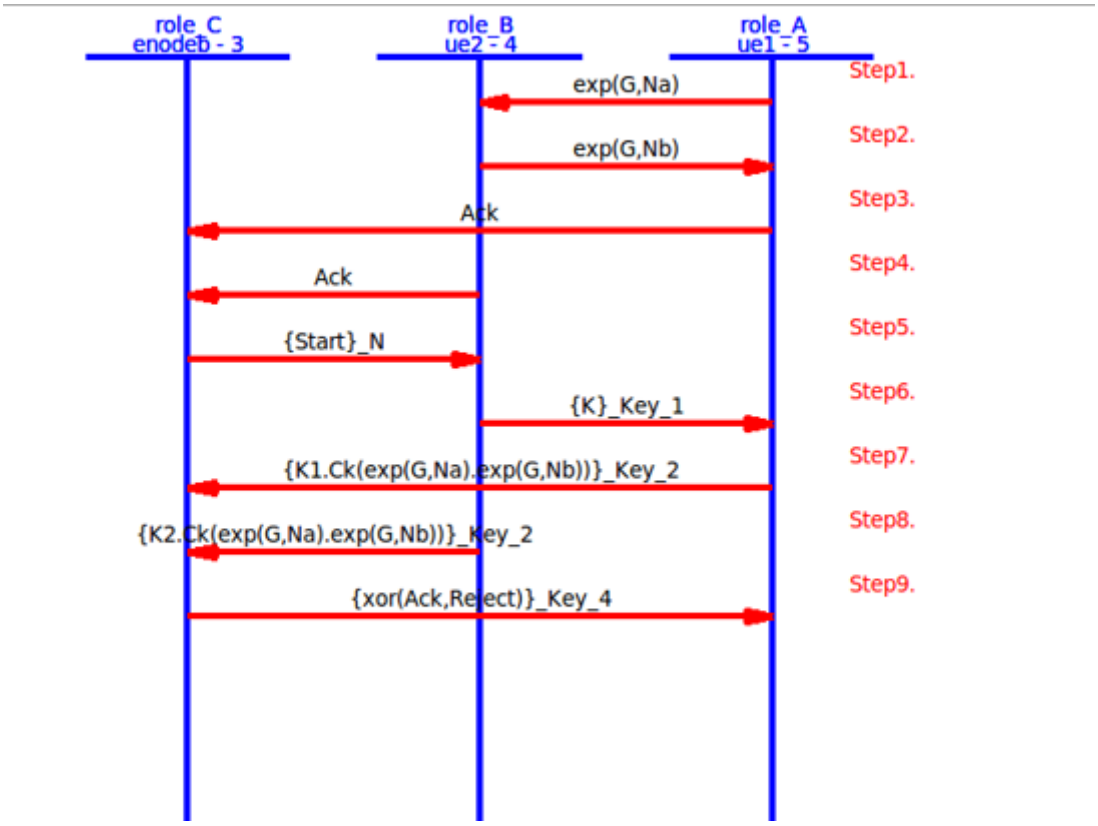

FIGURE 4.8 – Trace du deuxième protocole.

4.8.2.2 Résultat de vérification

Après avoir chargé le fichier, nous allons l'exécuter et faire une interprétation des résultats. Les résultats fournis ci-dessus sont ceux générés par l'outil OFMC :

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/hlpslGenFile.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.20s
  visitedNodes: 48 nodes
  depth: 11 plies
```

FIGURE 4.9 – Résultat OFMC du deuxième protocole.

Ce résultat signifie qu'il n y a pas d'attaque détectée pour la confidentialité de la clé K vérifiée par (Secrecy of K [A, B], Secrecy of K1 [A, B], Secrecy of K2 [A, B]). On peut ainsi déduire que le diagnostic de la plateforme AVISPA pour ce protocole est sûr et sécurisé.

4.8.3 Spécification formelle du premier protocole avec trois dispositifs en langage CAS+

Comme le deuxième protocole, la clé K= K1= K2 dans le but d'éviter les problèmes de compilation au niveau d'OFMC. La figure 4.10 présente le code du programme écrit en CAS+ :

```
protocol protocole1;
identifiers
A,B,C,D   : user;
Ck        : function;
K,K1,K2      : symmetric_key;
Na, Nb,Nc, G,Start,Rejet,Ack:number;

messages
1.B->C :G^Nb
2. C -> B : G^Nc
3.A -> B : G^Na
4. B -> A : G^Nb
5. A -> C : G^Na
6. C -> A : G^Nc
7. B  -> D : Ack
8. C  -> D : Ack
9. A -> D  : Ack
10. D  =>  A : Start
11. A  =>  D : Ck(G^Na,G^Nb,G^Nc),K
14. D  =>  C : Ck(G^Na,G^Nb,G^Nc),K1
15. C =>  D : Ack # Rejet
12. D  => B  :  Ck(G^Na,G^Nb,G^Nc),K2
13. B  =>  D : Ack # Rejet
16. D =>  A : Ack # Rejet

knowledge
A          : A,B,C,D,G,Na,Ack,Rejet;
B          : A,B,C,D,G,Nb,Ack,Rejet;
C          : A,B,C,D,G,Nc,Ack,Rejet;
D     : A,B,C,D,Ack,Rejet;

session_instances
 [A:ue1,B:ue2,C:ue3,D:enodeb,Na:a,Nb:b,Nc:c,Ack:ack,Start:start,Rejet:rejet,G:g,K:k,K1:k,K2:k];

intruder_knowledge
 ue1,ue2,ue3,enodeb,g;

goal
secrecy_of K [A,B,C];
secrecy_of K1 [A,B,C];
secrecy_of K2 [A,B,C];
```

FIGURE 4.10 – Code du premier protocole avec trois dispositifs en CAS+.

4.8.3.1 Simulation du protocole

De la même manière que le premier protocole, on exécute le code source écrit en CAS+, une fois l'exécution s'est déroulée sans erreurs, on peut simuler le protocole entre les trois entités afin de visualiser graphiquement les différents messages échangés. La figure suivante représente les différentes étapes de la simulation du protocole :

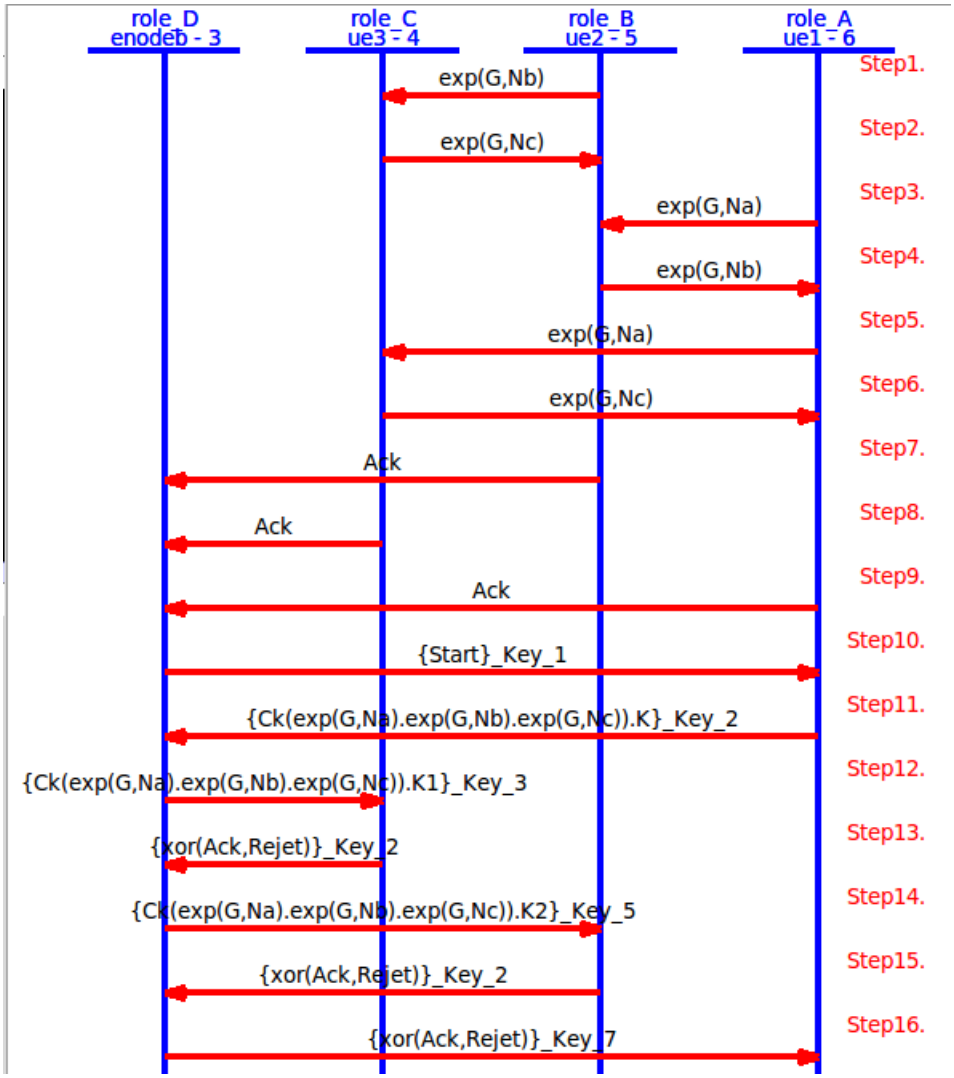

FIGURE 4.11 – Trace du premier protocole avec trois dispositifs.

4.8.3.2 Résultat de vérification

Après avoir chargé le fichier, nous allons l'exécuter et faire une interprétation des résultats. Les résultats fournis ci-dessus sont ceux générés par l'outil OFMC :

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/hlpslGenFile.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 4.68s
  visitedNodes: 0 nodes
  depth: 1000000 plies
```

FIGURE 4.12 – Résultat OFMC du premier protocole avec trois dispositifs.

Ce résultat signifie en clair qu'il n y a pas d'attaque détectée pour la confidentialité de la clé K vérifiée par (Secrecy of K [A, B, C], Secrecy of K1 [A, B, C], Secrecy of K2 [A, B, C]). On peut ainsi déduire que le diagnostic de la plateforme AVISPA pour ce protocole est sûr et sécurisé.

4.9 Conclusion

La vérification formelle est une méthode qui permet la description mathématique d'un protocole afin de vérifier la fiabilité des protocoles cryptographiques.
Dans ce chapitre, nous avons présenté quelques notions de base des outils de vérification SPAN AVISPA et le langage CAS+, ainsi que la vérification formelle des protocoles cryptographiques : la spécification formelle, la simulation et les résultats de vérification obtenus. Nous avons montré l'importance de cette vérification pour assurer la propriété de confidentialité dans les communications D2D.
Suite à nos vérifications, nous avons constaté que les protocoles étudiés [37] sont non vulnérables vis-à-vis des attaques, notamment l'attaque MITM, autrement dit, ils sont sécurisés et sûrs.

Conclusion générale et perspectives

Notre travail a été réalisé dans le cadre de la technologie des objets connectés, cette technologie offre à l'être humain un confort supplémentaire qui facilite sa vie quotidienne plus que jamais. De ce fait, un objet connecté généralement utilise une communication D2D. Cette dernière, est devenue une solution attrayante pour améliorer les performances des réseaux cellulaires. Cependant, l'un des problèmes qu'on peut rencontrer dans les communications D2D est la problématique de la sécurité.

Ce mémoire, a pour objectif de vérifier formellement les propriétés de sécurité des protocoles D2D à l'aide d'outils automatiques SPAN AVISPA, qui sont utilisés dans divers domaines grâce à leurs efficacités.

Dans notre travail, nous avons présenté quatre chapitres. Le premier chapitre consiste à faire une recherche d'une manière générale sur l'Internet des objets et nous avons mis en avant les concepts essentiels de ce dernier.

Dans le chapitre suivant, nous avons présenté le principe général de la communication D2D, et nous avons abordé le problème de la sécurité qui représente les pierres cruciales des communications D2D dans l'IdO.

Le troisième chapitre, est consacré à l'étude de quelques travaux récents réalisés dans le contexte de la sécurité des communications D2D. Le dernier chapitre, a concerné la vérification formelle des protocoles de sécurité D2D avec l'outil SPAN AVISPA qui offre un langage modulaire et expressif pour spécifier des protocoles et des propriétés de sécurité, nous nous sommes basées sur le langage CAS+.

Dans ce dernier chapitre, nous avons vérifié formellement les deux premiers protocoles d'échange de clés pour une communication D2D sécurisée proposés dans [37]. Suite à nos vérifications, nous avons constaté que ces protocoles sont sûrs et la propriété de confidentialité est assurée. Ensuite, nous avons ajouté au premier protocole un troisième dispositif afin de nous montrer que ce protocole est plus efficace et offre une sécurité contre l'attaque MITM, pratiquement applicable et convient parfaitement à la communication D2D.

Ce travail nous a été d'un grand apport pédagogique sur le plan théorique et pratique, puisqu'il nous a permis d'améliorer nos connaissances déjà acquises, et acquérir de nouvelles connaissances telles que, se familiariser avec l'environnement de travail SPAN AVISPA que nous avons découvert pour la première fois.

Comme perspectives, nous envisageons d'évaluer la résistance des protocoles étudiés face aux différents types d'attaques avec d'autres outils de validation. À l'avenir, nous prévoyons de tester les protocoles sur un banc d'essai D2D réel et d'apporter même quelques améliorations dans le but éternel de sécuriser de façon maximale les objets.

Bibliographie

[1] Difference between inband d2d vs outband d2d communication. https ://www.rfwireless-world.com/Terminology/InBand-D2D-communication-vs-Outband-D2D-communication.html, consulté le 05 mars 2021.

[2] Smart city ou la ville intelligente. http://www.objectif-bim.com/index.php/technologie-bim/ville-intelligente-smart-city, consulté le 10 juillet 2021.

[3] Domaines d'applications de l'iot, travaux et risques. https://wikimemoires.net/2019/09/domaines-d-applications-de-l-iot, consulté le 24 juin 2021.

[4] Internet of things (iot) protocols you need to know about. https://www.rs-online.com/designspark/eleven-internet-of-things-iot-protocols-you-need-to-know-about, consulté le 28 juin 2021.

[5] Mohd Hirzi Adnan and Zuriati Ahmad Zukarnain. Device-to-device communication in 5g environment : Issues, solutions, and challenges. *Symmetry*, 12(11) :1762, 2020.

[6] Muhammad Alam, Du Yang, Jonathan Rodriguez, and Raed A Abd-Alhameed. Secure device-to-device communication in lte-a. *IEEE Communications Magazine*, 52(4) :66–73, 2014.

[7] Rawan Alkurd, Raed M Shubair, and Ibrahim Abualhaol. Survey on device-to-device communications : Challenges and design issues. In *2014 IEEE 12th International New Circuits and Systems Conference (NEWCAS)*, pages 361–364. IEEE, 2014.

[8] Liliana Antao, Rui Pinto, Joao Reis, and Gil Gonçalves. Requirements for testing and validating the industrial internet of things. In *2018 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW)*, pages 110–115. IEEE, 2018.

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

[10] Noureddine Chikouche and Mohamed Benmohammed. Vérification automatique des protocoles d'authentification des systèmes rfid. *Proc. of STIC'09*, 2009.

[11] Connectwave. la référence des objets connectés professionnels. https://www.connectwave.fr, consulté le 15 Avril 2021.

[12] Alain Coulon. L'internet des objets un gisement à exploiter. *La Lettre d'ADELI n*, 2010.

[13] D Dolev and A Yao. On the security of public key products : Department of computer science, 1981.

[14] Scott Fowler, Yuan Li, Alberto Pollastro, and Stefano Napoli. Simple network design and power allocation for 5g device-to-device communication. In *2014 IEEE 19th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, pages 203–207. IEEE, 2014.

[15] Pimmy Gandotra and Rakesh Kumar Jha. Device-to-device communication in cellular networks : A survey. *Journal of Network and Computer Applications*, 71 :99–117, 2016.

[16] Simon Debras Hamdi Nasser, Michel Fangayoumani. Vérification formelle d'un protocole de sécurité à l'aide d'un outil d'analyse automatique des failles de sécurité. Technical report, École Nationale de l'Aviation Civile.

[17] Mohamed Tahar Hammi. *Sécurisation de l'Internet des objets*. PhD thesis, Université Paris-Saclay (ComUE), 2018.

[18] M Han and H Zhang. Business intelligence architecture based on internet of things. *Journal of Theoretical & Applied Information Technology*, 50(1) :90–95, 2013.

[19] Magri Hicham, Noreddine Abghour, and Mohammed Ouzzif. Device-to-device (d2d) communication under lte-advanced networks. *International Journal of Wireless & Mobile Networks (IJWMN) Vol*, 8, 2016.

[20] A Inam and X Gui. Device to device (d2d) communication : Interference management perspective. *Technical Journal*, 22(2), 2017.

[21] Messai Mohamed Lamine. Securité dans les reseaux de capteurs sans-fil. *Memoire de Magistere en Informatique Ecole Doctorale d'Informatique de bejaia*, 2008.

[22] Djeffal Leila. Gestion dynamique du spectre pour l'internet des objets (iot). Master's thesis, Université Mohamed Khider – Biskra, 2019.

[23] Team Lesleudis. 10 applications de l'internet des objets qui révolutionnent la société. https ://blog.lesjeudis.com/10-applications-de-l-internet-des-objets-qui-revolutionnent-la-societe, consulté le 20 juin 2021.

[24] Cédric Llorens, Laurent Levier, Denis Valois, and Benjamin Morin. *Tableaux de bord de la sécurité réseau*. Editions Eyrolles, 2011.

[25] Mohamedi Malika and Ikerbane Samia. Vérification automatique d'un protocole de sécurité dans les systèmes rfids à base d'outils avispa & span. Master's thesis, Université Mouloud Mammeri, 2016.

[26] Dihia Megtit, Tedjini Dahmane. *Réalisation d'une serre agricole intelligente et contrôlable à distance par Interne*. PhD thesis, Université Abou Bakr Belkaid– Tlemcen, 2017/2018.

[27] Leonardo Militano, Giuseppe Araniti, Massimo Condoluci, Ivan Farris, and Antonio Iera. Device-to-device communications for 5g internet of things. *EAI Endorsed Transactions on Internet of Things*, 1(1), 2015.

[28] Daniele Miorandi, Sabrina Sicari, Francesco De Pellegrini, and Imrich Chlamtac. Internet of things : Vision, applications and research challenges. *Ad hoc networks*, 10(7) :1497–1516, 2012.

[29] Karim Moussi. Detecteur de gaz toxiques en utilisant l'iot. Master's thesis, université Abderahmene Mira. Bejaia, 2020.

[30] Gaspard Nono, Louenkam Guy. *Problématique de l'intégration de la voix et du texte dans un environnement sans fils : Application à la téchnologie Bluetooth.* PhD thesis, Université de Yaoundé I, 2012/2013.

[31] Keyur K Patel, Sunil M Patel, et al. Internet of things-iot : definition, characteristics, architecture, enabling technologies, application et future challenges. *International journal of engineering science and computing*, 6(5), 2016.

[32] Fatima Rabehi. *La gestion des groupes dans les manets aspects sécurité et gestion de clés.* PhD thesis, Université des Sciences et de la Technologie Mohamed Boudiaf Oran, 2010.

[33] Rajiv. How does device to device communication works. https://www.rfpage.com/how-does-device-to-device-communication-works, consulté le 16 fevrier 2021.

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

[35] Ghazanfar Ali Safdar, Masood Ur-Rehman, Mujahid Muhammad, Muhammad Ali Imran, and Rahim Tafazolli. Interference mitigation in d2d communication underlaying lte-a network. *IEEE Access*, 4 :7967–7987, 2016.

[36] Imad Saleh. Internet des objets (ido) : Concepts, enjeux, défis et perspectives. *Internet des objets*, 1(1) :5, 2017.

[37] Ravindranath Sedidi and Abhinav Kumar. Key exchange protocols for secure device-to-device (d2d) communication in 5g. In *2016 Wireless Days (WD)*, pages 1–6. IEEE, 2016.

[38] Wenlong Shen, Weisheng Hong, Xianghui Cao, Bo Yin, Devu Manikantan Shila, and Yu Cheng. Secure key establishment for device-to-device communications. In *2014 IEEE Global Communications Conference*, pages 336–340. IEEE, 2014.

[39] John A Stankovic. Wireless sensor networks. *computer*, 41(10) :92–95, 2008.

[40] Harald Sundmaeker, Patrick Guillemin, Peter Friess, Sylvie Woelfflé, et al. Vision and challenges for realising the internet of things. *Cluster of European research projects on the internet of things, European Commision*, 3(3) :34–36, 2010.

[41] Rahim Tafazolli. *Technologies for the Wireless Future : Wireless World Research Forum (WWRF).* John Wiley & Sons, 2006.

[42] Abdelkrim Taleb, Omar Mankouri. Programmation de la sécurité internet des objet, etude de cas module wifi electric imp. Master's thesis, Universite Abou Bekr Belkaid Tlemcen UABT, mai-2016.

[43] Mohsen Nader Tehrani, Murat Uysal, and Halim Yanikomeroglu. Device-to-device communication in 5g cellular networks : challenges, solutions, and future directions. *IEEE Communications Magazine*, 52(5) :86–92, 2014.

[44] Sachin Umrao, Abhishek Roy, and Navrati Saxena. Device-to-device communication from control and frequency perspective : A composite review. *IETE Technical Review*, 34(3) :286–297, 2017.

[45] Rob Van Kranenburg and Alex Bassi. Iot challenges. *Communications in Mobile Computing*, 1(1) :1–5, 2012.

Résumé

L'Internet des objets représente aujourd'hui une partie majeure de notre vie quotidienne, est un concept qui repose sur l'idée que tous les objets seront connectés à l'Internet pour un mode de vie beaucoup plus sophistique.
Nous nous sommes focalisées plus précisément sur la communication Device To Device dans l'IdO, qui a suscité de l'intérêt en tant que technologie prometteuse pour les réseaux sans fil, elle favorise l'utilisation de communication directe entre appareils sans passer par la station de base. Une des contraintes principales dans la conception des communications D2D est la sécurité des données entre les objets connectés, pour cela, il est nécessaire de prendre en considération des protocoles cryptographiques efficaces afin d'assurer la sécurité et le bon fonctionnement du réseau.
Dans ce contexte, l'objectif de ce mémoire était de vérifier des protocoles de sécurité D2D à base de l'outil de vérification formelle SPAN AVISPA qui nous a permis de montrer que les protocoles proposés sont sécurisés et sûrs.

Mots-clés : IdO, D2D, protocoles cryptographiques, vérification formelle, SPAN AVISPA.

Abstract

The Internet of Things today represents a major part of our daily life, is a concept based on the idea that all objects will be connected to the Internet, for a much more sophisticated lifestyle.
We focused specifically on Currency To communication Dvice To Device in the IoT, which has generated interest as a promising technology for wireless networks, it promotes the use of direct communication between devices without going through the base station. One of the main constraints in the design of D2D communications is the security of data between connected objects, for this, it is necessary to consider effective cryptographic protocols in order to ensure the security and smooth operation of the network.
In this context, the objective of this brief was to verify a D2D security protocol based on the formal verification tool SPAN AVISPA which allowed us to show that the proposed protocols are safe and secure.

Keywords : IoT, D2D, cryptographic protocols, the formal verification, SPAN AVISPA.