Les protocoles réseau de l'Internet des Objets - vulnérabilités connues -

Bastien Jorge, Bastien Lacroix et Alexander Proux

***Abstract*— L'Internet of Things (IoT), ou l'Internet des Objets (IdO), est considéré comme étant un concept moderne, et à même de révolutionner l'avenir proche. L'intérêt est de créer un environnement composé d'appareils et de systèmes intelligents, pouvant communiquer entre eux par le biais de réseaux informatiques. Ces échanges de données vont ainsi permettre de meilleures prises de décision dans un contexte de plus en plus complexe. Cependant, au fur et à mesure que les échanges se multiplient, la question de sécurité se pose, qui doit prendre en compte la faible puissance de calculs des appareils. Ce document a pour objectif de se focaliser sur les protocoles qui régissent ces échanges, et quelles sont les problématiques de sécurité qui en découlent.**

***Index Terms*— Internet des Objets, sécurité réseau, protocole**

I. INTRODUCTION

Concept apparu dans les années 90, l'Internet des Objets est en passe de devenir un des challenges à venir, et ouvre des perspectives d'évolution en matières technologiques et scientifiques [9].

Aujourd'hui, l'Internet des Objets est principalement utilisé dans des domaines tels que la domotique ou le "*Quantified self*" [1]. Il s'agit de l'extension d'Internet à des objets physiques qui n'ont pas pour vocation première à y être connecté. C'est pourquoi les systèmes embarqués et les capteurs intelligents rentrent dans cette catégorie et participent au développement de l'IdO.

L'intérêt d'IdO est multiple, tant au point de vue des utilisateurs civils que des entreprises : gain d'efficacité, d'économies, de fiabilités... La prise de conscience de ce domaine provoque une rapide évolution des technologies, et une intégration sur des supports très variés et diversifiés : le concept "d'objets" a pour vocation de ne poser aucune limite quant aux possibilités envisageables d'implémenter des outils technologiques dans des éléments du quotidien.

Cependant, l'aspect touchant à la sécurité est une problématique de plus en plus présente dans la conscience collective, notamment sur les questions de données personnelles. L'insertion des technologies dans l'espace personnel induit d'assurer un niveau de confiance des utilisateurs vis-à-vis des objets connectés, à travers la mise en place d'outils garantissant les normes de sécurité attendues de la part des systèmes informatiques.
En 2014, HP a publié une étude démontrant que près de 70% des appareils considéré comme faisant partie de l'IoD sont vulnérables aux attaques [1] : protection de la vie privée, autorisation et authentification insuffisantes, chiffrement inexistant, interface web non sécurisée et firmware non sécurisés...
La même année, la CNIL européenne, du nom de G29, a également porté son attention sur l'IdO, à travers un dossier proposant un cadre de travail sur la réflexion à porter sur le domaine [10]. Il en ressort que les acteurs directement impliqués dans la conception des objets connectés sont les mieux placés pour introduire et prendre en compte les questions de vie privée et de sécurisation des données.

L'*Internet of Things* étant un ensemble d'outils et de technologies, il convient de prendre en compte chaque niveau et aspect d'un objet connecté, modélisé comme un système à part entière. La question des échanges de données fait l'objet de ce document : **comment les flux de données des objets connectés sont-ils vulnérables ?** La suite du document est ainsi organisée de la manière suivante : la section II présente les différents protocoles réseaux qui sont utilisés dans le domaine, et la sécurité qui en découle ; puis dans la section III, nous verrons deux cas d'études présentant des éléments concrets et identifiables impactant au quotidien. Nous terminerons par une conclusion à la section IV.

II. PROTOCOLES RESEAU

D'un point de vue technique, l'IdO peut se baser sur plusieurs protocoles. Chacun d'entre eux présente différents avantages (portée, sécurités intégrées...) mais a également ses propres faiblesses.

Nous allons nous pencher ici sur les principaux d'entre eux ainsi que les typologies d'attaques auxquelles ils peuvent être soumis, dans le but d'effectuer un comparatif en fin de section.

La liste des protocoles qui va suivre ne se veut pas exhaustive, mais elle couvre à elle seule l'immense majorité des protocoles réseau sur lesquels s'appuie l'IdO. Néanmoins, il est à noter que l'étude se limite aux seuls protocoles non propriétaires.

A. Bluetooth

Bluetooth est un standard de communication courte portée basé sur les ondes radio, inventé par *Ericsson* en 1994. Il équipe aujourd'hui la majorité des objets nécessitant d'être reliés sur une courte distance, comme par exemple les téléphones ou les automobiles, en passant par les casques audio et les périphériques informatiques ; *ABI Research*

1. Ensemble des outils et méthodes permettant de mesurer et d'analyser ses données personnelles

estime à 10 milliards le nombre d'objets utilisant cette technologie d'ici 2018 [2]. Son fonctionnement est simple à appréhender : il relie différents périphériques à un hôte (ou plusieurs hôtes entre eux) créant ainsi ce qu'on appelle un *piconet*[2].

Si différentes normes se sont succédées depuis son apparition (pour arriver aujourd'hui à la *4.2*, rendue publique le 2 décembre 2014, *voir* Annexe B), l'architecture en elle-même est restée globalement la même.

Les fonctions principales sont dissociées, on les retrouve donc dans deux groupes architecturaux :

- Le premier, attribuée à l'hôte, qui regroupe des fonctions haut niveau (applicatives, services…)
- Le second affectée au contrôleur, qui se concentre sur les fonctions de communication bas niveau

Ces deux groupes communiquent entre eux de manière standardisée à travers l'*HCI*[3] qui s'intègre pleinement dans l'ensemble des protocoles du noyau *Bluetooth* (cf. *infra* : II-A.1).

1) Les protocoles Bluetooth*:* Les protocoles *Bluetooth* peuvent être divisés en différentes catégories :

- Les protocoles du noyau *Bluetooth* (*voir* Figure 1), regroupant la *baseband*[4] (qui permet la formation d'un *piconet*), le protocole de gestion de liaison (qui établit la connexion entre les éléments *Bluetooth* et fournit une couche de sécurité), le L2CAP[5] (qui segmente et réassemble les paquets), le SDP[6] et le RFCOMM[7], ainsi que le HCI (cf *supra*.).

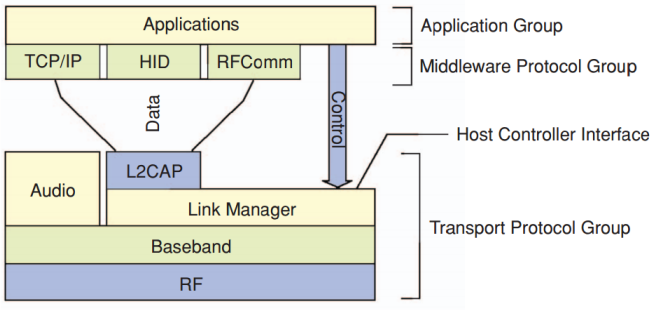


Fig. 1: Protocoles du noyau *Bluetooth* [3]

- Le protocole de téléphonie : *TCS Binary*, qui définit les signaux de contrôle d'appel.
- Des protocoles adoptés, comme TCP/IP.

2. Réseau ad-hoc ultra local, aussi appelé BT-WPAN (pour *Bluetooth Wireless Personal Area Network*)

3. Host Controller Interface

4. Bande de base, technique de transmission dans laquelle le signal est envoyé directement sur le canal sans modulation

5. *Logical link control and adaptation protocol*, couche de contrôle et d'adaptation de liens logiques

6. *Service Discovery Protocol*, protocole de recherche de services

7. *Radio frequency communication*, protocole de communication radio-fréquence

2) Les mécanismes de sécurité de Bluetooth*: Bluetooth* dispose originellement de trois modes de sécurité. Dans le premier mode, **aucune fonction de sécurité** n'est activée, ce qui signifie que tous les dispositifs *Bluetooth* peuvent s'y connecter. Le mode de sécurité 2 implémente une sécurité au **niveau application**, à savoir après la connexion. Le mode 3, en plus d'hériter des fonctionnalités du mode 2, ajoute une sécurité au **niveau liaison** : authentification et chiffrement en amont de la connexion. La version 2.1 de *Bluetooth* introduit un mode 4, dans lequel la connexion ne peut s'effectuer que si la communication est initialisée par l'appareil en mode 4.

Les clés de d'authentification et de chiffrement utilisées sont générées à partir de trois éléments. Une **adresse physique de la carte *Bluetooth***, un équivalent de l'adresse MAC d'une carte réseau, un **code personnel d'identification**, code PIN stocké sur 1 à 16 octets attribué à l'utilisateur, et la **génération d'un nombre aléatoire** sur 128 bits.

3) Les vulnérabilités de Bluetooth*:* Elles sont de deux type : les menaces génériques (DOS, écoute passive, *Man in the Middle*, attaques actives…) sur lesquelles nous ne nous attarderons pas, et des menaces plus spécifiques. La technologie étant maintenant présente dans une immense majorité des objets connectés, la vulnérabilité principale est issue de l'utilisateur lui-même (activation continue du *Bluetooth*, code PIN par défaut laissé à 0000 ou 1234, etc.). Mais il existe d'autres risques :

- Le **blueprinting**, qui consiste à analyser l'équipement attaqué afin de pouvoir cibler les failles propres à ce matériel. Ce type d'attaque est facilitée par l'adresse physique de la carte *Bluetooth* (cf. *supra* : II-A.3) qui contient, sur un certain nombre de bits, une partie propre au fabriquant et à l'élément.
- Le **scan**, qui permet de connaitre les applications communiquant vers l'extérieur.
- Le **Hijacking** est une technique de détournement : il suffit de se connecter au système dont on veut prendre le contrôle via un clavier *Bluetooth* et il sera possible d'interagir directement avec l'équipement visé.
- Le **Bluesnarfing** exploite une vulnérabilité firmware des anciennes versions *Bluetooth* pour accéder aux données du système visé après avoir forcé la connexion, puis récupère les fichiers avec des commandes `GET`.
- Le **Bluebugging**, à l'instar du *Bluesnarfing*, s'en prend aux failles des anciens firmwares pour cette fois-ci utiliser des commandes `AT` et passer des appels surtaxés.
- Le **car whisperer** exploite les failles des voitures proposant un kit main libre intégré, et qui utilisent souvent un code PIN par défaut, ce qui permet une écoute des communications.

B. Wifi

Parue en 1997, la norme *IEEE 802.11* est appelée commercialement Wi-Fi, pour *Wireless Fidelity*. Contrairement

au *Bluetooth* qui fait partie des réseaux WPAN [8], le Wifi est un WLAN [9] : son rayon d'action est largement supérieur (10m à 20m pour le *Bluetooth*, plusieurs dizaines de mètres pour le Wifi).

Ici aussi, la norme a su évoluer (*voir* Annexe C) mais a conservé les mêmes principes de fonctionnement au fil du temps.

1) Les protocoles Wifi: La norme IEEE 802.11 définit les couches basses du modèle OSI :

- La **couche physique**, qui repose sur les principes radio-électriques. Située en dessous de la couche MAC, elle est divisée en deux sous-couches :
  - La **sous-couche PMD** [10] qui gère l'encodage des données et effectue la modulation
  - La **sous-couche PLCP** [11] qui s'occupe de l'écoute du support indique à la couche MAC que le canal est libre.
- La **couche laison**, elle aussi divisée en deux sous-couches distinctes :
  - La **sous-couche LLC** [12], la même que celle utilisée par Ethernet, pour simplifier les relations entre liaisons filaires et Wifi
  - La **sous-couche MAC** assure la gestion d'écoute et d'émission ainsi que le contrôle d'erreur et d'intégrité de la trame

2) Les mécanismes de sécurité du Wifi:

- Pour résoudre le problème de **confidentialité des échanges**, la sécurité du protocole Wifi repose principalement sur le mécanisme de protection **WPA** [13], et depuis le 24 juin 2004, sur **WPA2** (intégré depuis la norme 802.11i). Ce dernier repose sur l'algorithme de chiffrement AES [14] (même si l'algorithme TKIP [15] utilisé dans la première version de WPA est également supporté) afin de chiffrer les communications de l'infrastructure utilisant le Wifi. Il est possible d'améliorer encore la robustesse du WPA en utilisant le deuxième mode de fonctionnement de la norme IEEE 802.11i : le **WPA Enterprise** (par opposition au WPA Personal), qui impose l'utilisation d'une infrastructure d'authentification et d'un contrôleur réseau.
- Wifi permet également de contrôler l'**identité des périphériques** et de les limiter sur son réseau, avec un second mécanisme : le **filtrage d'adresses MAC**. En effet, il est possible de créer une liste blanche des cartes réseaux (identifiées par leur adresse MAC unique) autorisées à se connecter. Dans le cadre de l'IdO le filtrage MAC est toutefois relativement rare.

Néanmoins, pour une sécurité optimale, ces deux mécanismes devraient être mis en place simultanément.

3) Les vulnérabilités du Wifi: Elles sont au final assez peu nombreuses. En effet, sa principale faille réside dans l'utilisation d'anciennes technologies de chiffrement (AES avec TKIP par exemple, ou encore pire, WEP [16] qui peut maintenant être décrypté en moins d'une minute).

Cependant, il existe d'autres procédés permettant de détourner le flux d'information, notamment l'*ARP Spoofing* : il s'agit d'une technique de Man-in-the-Middle consistant à créer un point d'accès malveillant intermédiaire entre le périphérique et l'hôte auquel il se croit connecté. Dans la pratique, cela consiste à se faire passer pour l'hôte en diffusant un réseau Wifi avec le même nom (SSID [17] identique) mais d'une intensité largement supérieure afin d'"écraser" le réseau originel en forçant le périphérique à utiliser le second sans qu'il ne le détecte ; la dernière étape consiste à connecter le point frauduleux à l'hôte afin de créer un pont vers Internet et ainsi pouvoir surveiller les flux, et le cas échéant, les intercepter pour les modifier ou les supprimer.

C. NFC

De nos jours, les utilisateurs préfèrent les équipements numériques leur apportant toujours plus de flexibilité, de simplicité et de mobilité. L'une des principales fonctionnalités des smartphones dans ce domaine est le transfert de données et de communications rapides entre deux appareils électroniques. La technologie NFC [18] a donc été créée pour répondre à ces besoins. NFC est devenu un standard suivant les normes ISO14443 et FeliCA RFID (*Radio-Frequency Identification*) pour les smartphones, de même que pour les appareils similaires, afin d'établir des communications radio entre eux lors d'un contact physique. Il opère donc dans la bande spectrale de 13,56 Mhz et supporte un transfert de donnés variant entre 106, 216 et 424 kbit/s [4].

Il existe 3 modes de fonctionnements pour les équipements NFC [5] :

- Le mode lecture/écriture, un équipement NFC agit à proximité d'un autre appareil pouvant se coupler. On appelle cela le PCD (il s'agit du mode le plus utilisé). L'équipement peut dans ce cas lire et écrire les données stockées sur des transpondeurs passifs compatibles NFC.
- Un mode basé sur une carte d'émulation qui peut s'insérer dans un appareil supportant le protocole NFC et réagissant à proximité par induction à une autre carte du même type. On appelle ce mode PICC.
- Le mode peer-to-peer où les deux dispositifs NFC peuvent réaliser une communication bidirectionnelle pour transférer des données arbitraires.

8. *Wireless Personal Area Network*
9. *Wireless Local Area Network*
10. *Physical Medium Dependent*
11. *Physical Layer Convergence Protocol*
12. *Logical Link Control*, contrôle de la liaison logique
13. *Wifi Protected Access*
14. *Advanced Encryption Standard*
15. *Temporary Key Internet Protocol*
16. *Wired Equivalent Privacy*
17. *Service Set Identifier*
18. *Near Field Communication*

1) Les protocoles de NFC: NFC utilise certaines couches et autres protocoles dans ses trois différents modes (*voir* Figure 2).

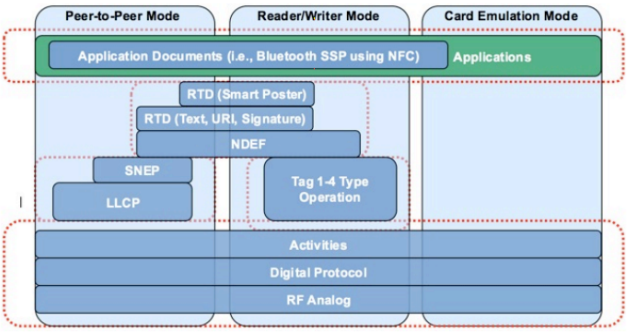


Fig. 2: Extrait d'un tableau comparatif de protocoles [6]

- **LLCP** : *Logical Link Control Protocol*. Vérifie l'identité de l'élément connecté, détermine la taille des paquets acceptables pour la transmission et recherche d'erreurs.
- **NDEF** : *NFC Data Exchange Format*. Message au format binaire qui encapsule une ou plusieurs applications de payement dans une simple structure de message [7].
- **RTD** : *Record Type Definition*. Signature qui protège l'intégrité et l'authenticité de NDEF.

2) Les mécanismes de sécurité de NFC: Les appareils mobiles qui autorisent les transferts de données *system-to-system* [19] peuvent déclencher des problèmes de confidentialité comme le suivi de localisation d'un utilisateur, une redirection vers un site inconnu, les fuites de données indésirables...

Les spécifications de la technologies NFC traitent certains de ces problèmes de vie privée. Par exemple, NFC nécessite d'avoir moins de quatre centimètres de proximité pour établir l'interaction entre deux équipements. A cette distance, les utilisateurs sont en totale connaissance de la personne avec qui ils interagissent. La technologie NFC doit également se désactiver quand l'écran ou le clavier d'un appareil mobile est verrouillé. De plus, la plateforme de l'équipement doit permettre à l'utilisateur de désactiver NFC de la même manière que pour les autres types de communication alternatives (Bluetooth, WiFi) [8].

Bien que la portée de la communication NFC est restreinte à quelques centimètres, ce seul critère ne sécurise pas les communications. Afin d'assurer l'entière confidentialité des échanges, l'algorithme de chiffrement AES est utilisé. Pour chaque transactions, un unique nombre d'identification est assigné au *tag* [20] NFC qui sera utilisé comme clef de chiffrement pour les messages envoyés.

3) Les vulnérabilités de NFC: Les principales menaces sont les attaques classique du type *Man in the Middle*, mais aucune protection n'est fournie contre les attaques *eavesdropping* qui ont pour but la modification des données. Dans un scénario de ce type, l'attaquant utilise une antenne pour enregistrer les communication entre les deux appareils NFC. Bien que les interactions NFC ne se réalisent qu'à quelques centimètres maximum, ce type d'attaque reste réalisable. Dans la plupart des cas l'attaquant modifie les informations échangées afin de la rendre inutile.

D. Autres protocoles

Ces autres protocoles se basent sur le protocole IEEE 802.15.4, technologie sans-fil standard dans l'industrie, avec pour particularités de fonctionner avec peu d'échanges de données, irréguliers et définis autour d'une zone restreinte. Le standard 802.15.4 propose ainsi une base de travail sur les deux premières couches du modèle OSI (Physique et Liaison) pour permettre la mise en place de protocoles plus élevés, avec des fréquences à 2.4 GHz, une distance d'environ 10 mètres et un débit d'environ 250 kb/s.

1) ZigBee: Similaire à la technologie *Bluetooth*, ZigBee fut créé dans l'objectif de proposer un protocole plus simple et moins coûteux, tout en mettant en place des kits de développement à la portée de tous. Limitée à une centaine de mètres, ZigBee s'appuie sur les couches Physique et Liaison (MAC) défini par le protocole 802.15.4 pour implanter au niveau Réseau et Applicatif des intégrations peu gourmandes (*voir* Figure 3).

- Sur la couche Réseau, ZigBee met en place un système similaire à ce qui est communément trouvé dans cette couche (routage, adressage), en utilisant le protocole AODV [21], destiné aux réseaux mobiles, notamment avec la création de routes à la volée.
- Sur la couche Application se situe une interface directe pour les utilisateurs, qui se basent sur les éléments de la couche Réseau, en les associant à des rôles et à des fonctionnalités.

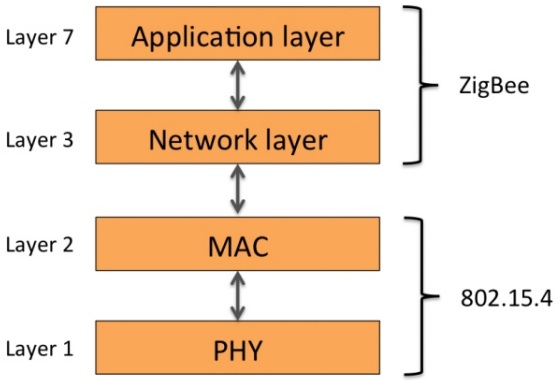


Fig. 3: Modèle de *ZigBee*

19. "Pairing" des équipements pour établir une interaction et échanger des données

20. L'intégrité des informations est stockée dans ces étiquettes par l'utilisation des signatures

21. *Ad hoc On-Demand Distance Vector*

D'un point de vue sécurité, ZigBee utilise beaucoup le chiffrement symétrique AES (clé 128 bits) pour chiffrer l'ensemble des échanges entre un appareil coordinateur dit « de confiance » et les appareils annexes, accompagnés de codes d'intégrité [14].
L'utilisation d'un tel système entraîne des problématiques au niveau de la création et de la distribution de clés de chiffrement [15] [16] [17]. Ainsi, deux problématiques ont pu être observées au niveau de ZigBee :

- *L'absence de révocation des clés* : si un appareil quitte (ou est forcé à quitter) la communication, il est toujours capable de communiquer car la clé est gardée en mémoire dans son système. Que ce soit pour des raisons de maintenance ou de vol, la clé devient récupérable et une falsification / usurpation peut avoir lieu.
- *Un système de certificats contraignant* : ZigBee indique que de nombreuses entités peuvent fournir des certificats, dont ces derniers ont été simplifiés pour être stockés facilement et être plus efficaces (par l'absence de signature par exemple). Cela engendre des facilités pour réaliser de faux certificats et ainsi usurper une identité.

2) 6LoWPAN: 6LoWPAN est l'acronyme de *IPv6 Low Power Wireless Arena Network*, et est un des protocoles les plus utilisés dans le cadre de l'internet des objets. Le nombre important d'appareils connectés à venir (50 milliards en 2020 selon la National Cable & Telecommunications Assocation [19]) implique l'utilisation de l'IPv6, protocole basé sur l'adresse IP à 128 bits. Cependant, la taille importante des en-têtes de ce dernier rend difficile son application directe sur des supports à faible ressource et à réseau limité. Ce problème est résolu par une fragmentation et un re-découpage, couplé à une compression d'en-tête, et situé entre les couches Réseau et Liaison du modèle OSI [11].

De plus, 6LoWPAN propose de nouveaux standards, incluant l'implémentation du *mesh-routing* et une version allégée du protocole sécurisé de découverte des voisins (LSEND [22]. Devenu presque indispensable, la question de la sécurité devient essentielle dans l'utilisation de 6LoWPAN, notamment à travers la mise en place d'un routeur de bordure pour la fragmentation des paquets. En outre, ce dernier est programmé pour sortir de la veille uniquement lorsqu'il est nécessaire, afin d'économiser la batterie du système [12].

Une des failles les plus exploitées consiste ainsi à multiplier l'envoi de requêtes inutiles à un appareil afin de vider sa batterie. L'absence de batterie entraîne la désactivation du routeur de bordure interne, et ainsi empêche toute communication avec l'appareil. La faible puissance de l'appareil rend difficile l'implémentation au niveau Réseau du protocole sécurisé IPsec, d'où l'utilisation du LSEND, basé sur un système de certifications et de signature RSA pour la découverte des voisins proches [20].

22. *Lightweight SEcure Neighbor Discovery Protocol*

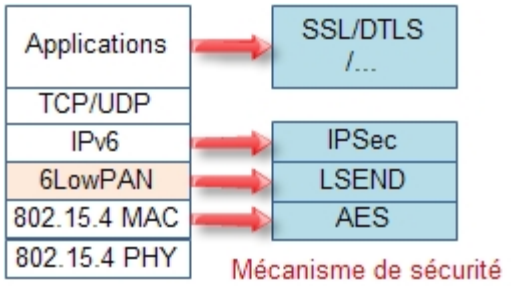


Fig. 4: Sécurité de la pile protocolaire de *6LoWPAN*

Étant un protocole IP, 6LoWPAN est également sensible à différentes attaques [21] liées à cette technologie, comme :

- **Usurpation et altération d'information de routage** : un appareil malicieux modifie des éléments de routages afin de créer des boucles, des longueurs inutiles, des erreurs…
- **Transfert sélectif** : l'appareil malicieux drop des paquets au hasard, afin de rendre le système instable.
- **Sinkhole attack** : un appareil malicieux tente de récupérer l'ensemble des paquets d'une zone pour éviter leur propagation.
- **Sybil attack** : un appareil malicieux se fait passer pour différents types d'appareils, menaçant ainsi le routage global.
- **Wormhole attack** : l'attaquant tente ici de dévier un message en créant un nouveau chemin, moins efficace en terme de routage.

3) Thread: Le monde de l'Internet des Objets est sur le point d'exploser. Chacun y va de ses propres technologies, mais certains mouvement se sont déjà initiés pour tenter d'harmoniser l'ensemble. C'est le cas de Samsung, Google et ARM, qui ont réalisé une alliance afin de créer le protocole *Thread*. Le but est de pouvoir connecter les différents équipements de plusieurs constructeurs principalement dans la domotique [23]. Thread est basé sur de l'IPv6, open-source et sa structure repose sur le protocole 6LoWPAN.

Le futur amènerait ainsi la coopération entre Thread Group et l'alliance ZigBee à s'associer afin de créer une interface au-dessus du reséau Thread, qui devrait s'appeler *ZigBee Cluster Library*. En effet, à l'heure actuelle, Thread gère uniquement les fonctions de réseau d'un dispositif, de sorte que les fabricants d'appareils peuvent choisir ZCL ou d'autre applications en surcouche [24].

E. Comparatif

Le tableau récapitulatif ci-dessous (*voir* Figure 5) compare les quatre principaux protocoles utilisés dans l'internet des objets. Il démontre que ces typologies présentent différents avantages et inconvénients, et leur utilisation dépend du contexte dans lequel il se trouve.

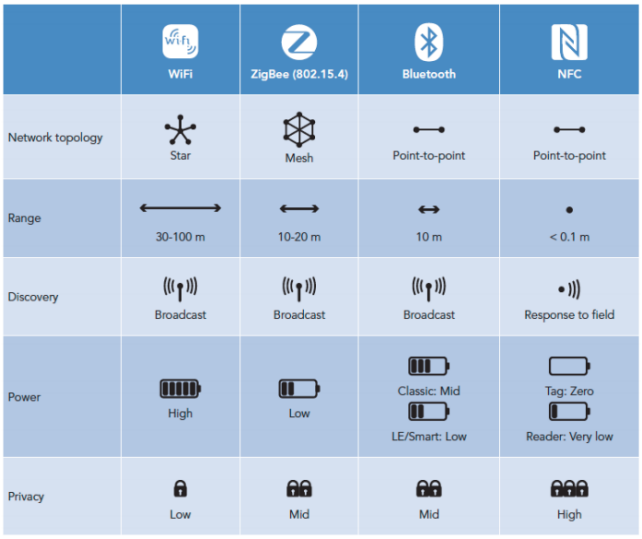


Fig. 5: Extrait d'un tableau comparatif de protocoles [6]

III. ETUDES DE CAS

Il convient de s'attacher à étudier différents cas pratiques d'utilisation d'objets connectés afin de mettre en évidence de nouvelles vulnérabilités, parfois indépendantes des protocoles réseaux listés ci-dessus. Dans un premier temps nous allons nous intéresser aux drones et leur sécurité propre (III-A) avant de mettre en évidence des vulnérabilités plus humanistes en étudiant un cas plus large de la télémaintenance et de la santé (III-B).

A. Drones

Le marché des drones est très récent et a connu une poussée exceptionnelle, le leader dans ce domaine étant actuellement le constructeur *Parrot*. Il faut faire la distinction entre leurs deux types de drone : les drones **utilitaires** et les drones de **prise d'image**. Leurs deux *flagships* [23] sont respectivement le AR-Drone et le Bebob drone. Nous allons ici nous consacrer à celui conçu pour le grand public : l'AR Drone.

La connexion avec l'AR Drone se réalise à l'aide du protocole WiFi par le biais d'une application mobile facile d'utilisation : *free flight* ; le drone se pilote alors à l'aide des accéléromètres du smartphones. Par ailleurs, l'application mobile n'est pas le seul moyen pour se connecter au drone. En effet, les AR Drone fonctionnent exactement comme un routeur, on peut donc se connecter simplement en entrant la commande `telnet 192.168.1.1` sur le port 5551.

Une fois à connecté au drone, aucune sécurité particulière n'a été mise en place par Parrot : les ports 21 (FTP), 23 (telnet en mode root) du drone sont continuellement ouverts et le réseau WiFi mis en place n'est même pas chiffré. La société ayant pleinement conscience des failles se défend par leur volonté de permettre aux développeurs la création et la modification des programmes intégrés au drone. Le système d'exploitation du drone est de type UNIX. Dans le répertoire `/bin` se trouve la plupart des scripts permettant de commander le drone. On peut donc utiliser la commande `sh` pour se mettre dans un environnement Shell. Ainsi, l'accès à tous les dossiers est libre car la connexion se fait par défaut en root. On peut donc librement naviguer dans l'arborescence du drone. Il ne reste plus qu'a exécuter le fichier `program.elf` qui est le programme chargé de gérer à proprement parler le pilotage du drone. C'est ce fichier qui transforme les commandes envoyées par le pilote en suite d'actions à effectuer par le drone [26].

L'absence de sécurité permet également à toute personne de lancer des attaques sans aucune contrainte de droits. Nous allons prendre exemple de l'attaque par désauthentification, qui donne la possibilité à l'attaquant de prendre le contrôle du drone.

1) Attaque par désauthentification: Cette attaque peut se résumer en deux grandes phases :

- Déconnecter le pilote initial pour enlever le contrôle du drone
- Reprendre le contrôle du drone et lui envoyer nos propres commandes

Pour ce faire, on cherche d'abord parmi les réseaux WiFi alentours un *Acces Point* dont l'adresse MAC est de la forme `90:03:b7:*:*:*`. En effet, l'AR Drone est un routeur sans fil qui peut être scanné [24]. Les adresses MAC étant composées de deux parties [25], une adresse comme celle ci-dessus permet de s'assurer qu'un drone est construit par l'entreprise Parrot.

```
# python airoscapy.py mon0
-=-=-=-=-=-= AIROSCAPY =-=-=-=-=-=-
CH ENC BSSID             SSID
11  Y  c0:c1:c0:12:34:56 caesars
11  Y  00:24:01:12:34:56 cosmopolitan
11  Y  c0:c1:c0:12:34:56 encore
09  N  c0:c1:c0:12:34:56 excalibur
09  Y  00:16:b6:12:34:56 flamingo
05  N  00:16:b6:12:34:56 golden nugget
01  Y  e0:91:f5:12:34:56 hard rock
01  Y  00:22:3f:12:34:56 harrahs
07  Y  00:1e:52:12:34:56 luxor
07  Y  c0:3f:0e:12:34:56 mgm
07  Y  00:30:bd:12:34:56 mirage
10  Y  00:25:9c:12:34:56 nyny
11  N  c0:c1:c0:12:34:56 palms
11  Y  00:14:6c:12:34:56 rio
^C
-=-=-=-=-  STATISTICS =-=-=-=-=-
Total APs found: 14
Encrypted APs  : 11
Unencrypted APs: 3
```

Fig. 6: Capture d'écran de Airoscapy.py

Une fois le drone identifié, il suffit d'envoyer le script de désauthentification via le protocole FTP (commande `PUT`). Le fichier se retrouve dans le répertoire `/data/video`. Comme la connexion par défaut s'effectue en root, il est possible d'ajouter les droits d'exécution au le script à l'aide de la commande `chmod`. Il faut enfin se connecter au

23. Les *flagships* d'une marque sont ses produits phare

24. À l'aide par exemple du logiciel *Airoscapy.py*(*voir* Figure 6), un scanner de canaux sans fil passif [27]

25. Les six premiers octets sont caractéristiques du constructeur et les six derniers caractérisent la machine proprement dite (*voir* II-A.3 - Blueprinting)

drone via telnet et lancer l'exécution du script.

Ce drone en particulier n'est pas une exception dans l'industrie, et Parrot n'est pas le seul à faire preuve d'un laxisme étonnant. En effet, les drones de la marque concurrente Dji qui se situent dans la gamme de prix supérieurs souffrent des mêmes faiblesses [28].

Cependant, d'autres constructeurs ont choisi d'implémenter une sécurité d'un autre type : les communications se font alors par radio à 2.4GHz. La sécurisation se fait par saut de fréquence extrêmement rapide[26], les canaux sont répartis dans une bande de fréquence selon une séquence pseudo-aléatoire[29], rendant ainsi la transmission très difficile à intercepter. Néanmoins, il suffit de mettre en place un brouilleur radio afin de neutraliser complètement ce type de drones, et ainsi de le faire chuter s'il est en utilisation.

B. Télémaintenance & santé

S'il est un domaine plus sensible concernant l'IdO et les objets connectés, il s'agit sans doutes de celui impactant directement l'individu. Les vulnérabilités touchant l'IdO sur ce terrain sont plus d'ordre éthique que technologique, mais elles ne sont pas négligeables pour autant.

1) La télémaintenance: La télémaintenance[27] fait partie de ces domaines sensibles.

La **maintenance prédictive** [30] (selon la norme française *NF EN 13306*) est un formidable moyen de réaliser des économies substantielles : plutôt que de mettre en place des routines d'inspection et de remplacement de composants sur une base calendaire, les techniques prédictives surveillent l'équipement pour identifier d'éventuelles défaillances et avertir lorsqu'une intervention est nécessaire. Cependant, il est important de bien prendre conscience qu'il est impossible de se reposer uniquement sur la maintenance prédictive. En effet, 50% des défaillances en fonctionnement ne sont pas identifiées par les concepteurs [31], cela signifie que dans la moitié des cas une intervention d'un technicien sera nécessaire afin de prendre connaissance de la panne.

Dès lors, on peut voir se profiler une question éthique : puisque la maintenance prédictive se base sur le **real-time data analysis**[28] (RTDA), est-il normal que les constructeurs des objets connectés que nous manipulons quotidiennement soient tenus au courant des moindres détails de l'utilisation que nous en faisons ? Bien évidemment la réponse à cette question est négative, mais inconsciemment ou non les utilisateurs exploitent le RTDA mis à disposition par le constructeur en ne gardant en tête que l'aspect positif que cela peut leur apporter.

Enfin, il est une problématique qui commence seulement à provoquer des réactions auprès des utilisateurs puisqu'ils sont de plus en plus concernés de manière directe : il s'agit des **mises à jour automatique**. Cet éveil de conscience est assez récent puisqu'il est apparu avec l'émergence des smartphones (applications de manière générale, OS mobiles…), ou encore plus récemment avec certaines machines dont la migration vers Windows 10 a été forcée [32]. Il est incontestable que pour un constructeur ces mises à jour ont un atout de toute première importance : elles permettent par exemple de combler des failles alors qu'elles viennent d'être découvertes (failles *0-day*) et de s'assurer ainsi de la relative sécurité du matériel. Elles ont un autre avantage pour le constructeur, celui de retirer des fonctions qu'il ne désire plus voir apparaitre dans son matériel. Il n'est pas rare de voir des utilisateurs bloquer ces mises à jour afin de conserver cette fameuse fonction.

2) Le domaine de la santé et les autres informations personnelles: Encore plus sensible que la télémaintenance, le domaine de la santé est un enjeu majeur de la récupération d'information, et elle passe sans aucun doutes par l'IdO.

Basée également sur le RTDA via des utilisations qui peuvent sembler anodines (gps, applications d'aide au suivi sportif…), l'objectif réel est d'emmagasiner des informations personnelles sur l'utilisateur afin de lui proposer des offres ciblées, publicité ou autre.

C'est ici que le bât blesse : des entités telles que les mutuelles complémentaires cherchent à pouvoir accéder à ces informations, de manière à refuser un client "à risque" (qui aurait été identifié à risque cardiaque par son application *running* par exemple). Ce scénario aurait des conséquences inenvisageables sur des personnes malades, ou âgées.

Un autre exemple de risque concernant l'analyses des données personnelles est celui, tout aussi récent, des boites noires intégrées aux véhicules personnels [33]. En effet, certaines compagnies d'assurances n'hésitent plus à proposer à leurs cotisants d'intégrer dans leur automobile une boite noire enregistrant toutes les informations relatives à la conduite, moyennant un avantage financier. La conséquence directe, au delà de la perte totale et consentie de liberté, est que ces compagnies puissent refuser de prendre en charge les dommages et intérêts dus à une victime d'un accident de la route, au motif que le conducteur excédait la limitation de vitesse autorisée d'1km/h.

IV. CONCLUSION

Révolution, "l'avenir technologique", un défi du XXI^{ème} siècle… Les mots ne manquent pas pour décrire le potentiel de l'Internet des Objets. Les enjeux de l'application de ce concept sont multiples et variés : santé, environnement, transport… Peu de domaines ne seront pas affectés par cette vague de systèmes embarqués intelligents, générant ainsi une quantité phénoménale de données. Néanmoins, les problèmes de sécurité soulèvent de nombreuses questions quant à la transmission et l'utilisation de ces données : il est nécessaire d'établir un niveau de confiance envers les utilisateurs pour garder une proximité des objets connectés avec le quotidien. Ce document s'est donc focalisé sur les différents protocoles d'échanges de données entre appareils.

26. *Frequency-hopping spread spectrum*
27. Maintenance à distance d'un système via un moyen de communication
28. Analyse de données en temps réel

Il existe plusieurs technologies capables d'être utilisées dans le cadre de l'IdO et toutes ont pour vocation de permettre à des objets avec faible puissance de calcul et de stockage de pouvoir interagir avec son environnement, composé d'autres systèmes. Certains proviennent de l'univers des portables et mobiles, se focalisant surtout sur une certaine mobilité (*Wi-Fi*, *Bluetooth*), tandis que de nouvelles implémentations prennent mieux en compte la limite de ressources pour proposer de nouveaux standards (*ZigBee*, *6LoWPAN*).

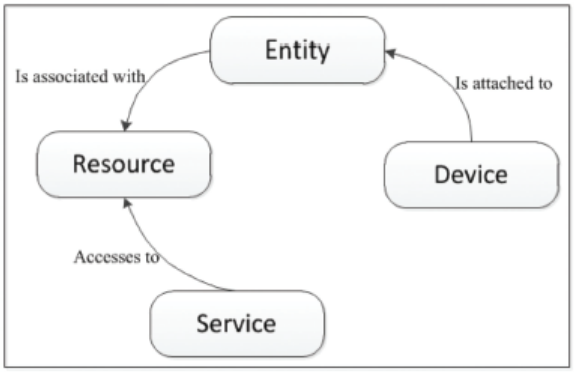
Cependant, la problématique de la sécurité est toujours présente quels que soient la technologie ou le protocole utilisés, et fait souvent partie intégrante du design de ce dernier. En combinant plusieurs mécanismes (cryptographie, habilitation, certificats), des parades sont possibles pour contrer certaines attaques touchant au réseau, et ainsi proposer une première défense pour protéger des menaces. De par leur nature connectée et mobile, les systèmes embarqués restent cependant fragiles et à risque d'être physiquement touchés, nécessitant de penser à la sécurité dans son ensemble. Nous observons que les développements techniques liés à l'IdO restent encore très récents, mais les premières approches de sécurité montrent des prémices prometteuses.

La sécurité technique fait partie des premières problématiques prises en compte par les constructeurs et fabricants : empêcher l'utilisation malicieuse d'un appareil est primordial pour éviter des conséquences désastreuses sur le plan humain comme économique. Ces données peuvent également être une source d'information sur un environnement physique, et ainsi en déceler les faiblesses. La forte production de données nous force alors à repenser la notion de vie privée, et pose des questions éthiques quant au stockage et à l'utilisation de ces données. En conséquent, le concept du *Big Data* rejoint les objets connectés, qui l'alimentent de manière conséquente, à travers la collecte de données massives sur les usagers.

Le potentiel de l'IdO est indéniable, et peut se révéler grandement positif pour ses utilisateurs de différentes manières, mais il nécessite de repenser la sécurité sur tous les plans, en commençant d'abord par les protections techniques des flux de données.

APPENDIX

A. *Modèle abstrait d'un objet connecté*



B. *Évolution du* Bluetooth *et de ses fonctionnalités [34]*

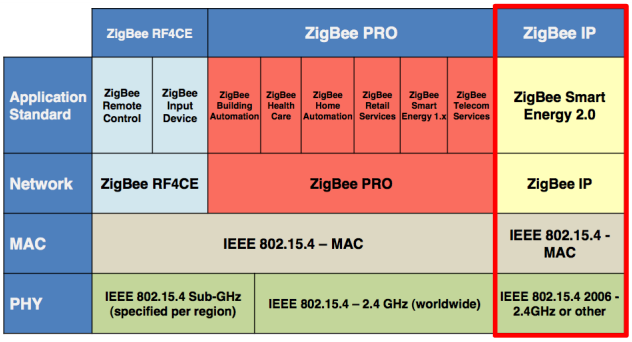
Bluetooth Versions	Year	Faster connection	SSP	Security Mode 4	Bug fixes	Error detection	Synchronization	Data Rate	L2CAP	HCI for AMP	Security for AMP	Power consumption
1.1	2002				X							
1.2	2003	X				X	X					
2.0	2004							X				
2.1	2007		X	X								X
3.0	2009							X	X	X	X	X
4.0	2010											X
		New Features			Enhancement Features							

C. *Évolution du Wifi et de ses fonctionnalités [35]*

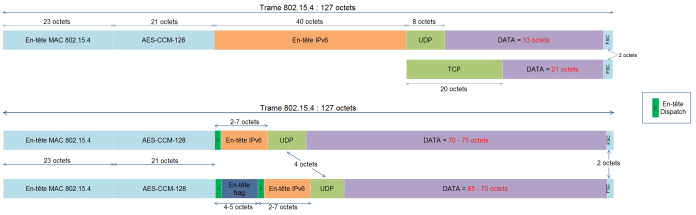
TABLE 1: WLAN PHYSICAL LAYER EVOLUTION

Year	Standard	Peak PHY rate (Mb/s) Including optional modes	Typical access point	Key technology
1997	802.11	2	2	Spread spectrum
1999	802.11b	11	11	CCK, spread spectrum
1999	802.11a	54	54	OFDM
2003	802.11g	54	54	OFDM
2009	802.11n	600	300-450	MIMO OFDM, wide bandwidth (40 MHz)
2013	802.11ac	6933	1300-3466	DL MU-MIMO, wide bandwidth (80 and 160 MHz)

D. *Diversité des outils ZigBee*



E. *Trame 802.15.4 non-compressée (dessus) et compressée (dessous)*



ACKNOWLEDGMENT

REFERENCES

[1] D. Miessler, "HP Study Reveals 70 Percent of Internet of Things...- Hewlett Packard Enterprise Community", Community.hpe.com, 2014. [Online]. Available : http://community.hpe.com/t5/Security-Products/HP-Study-Reveals-70-Percent-of-Internet-of-Things-Devices/ba-p/6556284#.Vnv3IRXhDIU. [Accessed : 24- Dec- 2015].

[2] Kuor-Hsin Chang, "Bluetooth : a viable solution for IoT ? [Industry Perspectives]", IEEE Wireless Commun., vol. 21, no. 6, pp. 6-7, 2014.

[3] P. McDermott-Wells, "What is Bluetooth ?", IEEE Potentials, vol. 23, no. 5, pp. 33-35, 2005.

[4] Miller, C. (2012). Exploring the NFC Attack Surface. [online] Media blackhat. Available at : https://media.blackhat.com/bh-us-12/Briefings/C_Miller/BH_US_12_Miller_NFC_attack_surface_WP.pdf[Accessed25Dec.2015]

[5] Mulliner, C. (2009). Vulnerability Analysis and Attacks on NFC-enabled Mobile Phones. [online] www.mulliner.org. Available at : https://www.mulliner.org/collin/academic/publications/vulnanalysisattacksnfcmobilephones_mulliner_2009.pdf [Accessed 25 Dec. 2015].

[6] Peyrard, F. and Conchon, E. (2014). Technologie sans contact NFC embarquée - Aspect protocoloraires. [online] Available at : http://www-public.tem-tsp.eu/~lauren_m/WEB-ActionSSO/2014-SSO-School-NFC-IRIT-ENSEEIHT-FabricePeyrard.pdf [Accessed 25 Dec. 2015].

[7] NFC Data Exchange Format. (2006). [online] Available at : http://www.eet-china.com/ARTICLES/2006AUG/PDF/NFCForum-TS-NDEF.pdff [Accessed 25 Dec. 2015].

[8] Cavoukian, A. (2012). Mobile Near Field Communications. [online] c.ymcdn.com. Available at : https://c.ymcdn.com/sites/www.issa.org/resource/resmgr/journalpdfs/feature0812.pdf [Accessed 25 Dec. 2015]

[9] Twinning, J. (2015). Behind The Numbers : Growth in the Internet of Things. [online] Ncta.com. Available at : https://www.ncta.com/platform/broadband-internet/behind-the-numbers-growth-in-the-internet-of-things-2/ [Accessed 25 Dec. 2015].

[10] Data protection working party. (2014). [online] Available at : http://ec.europa.eu/justice/data-protection/article-29/documentation/opinion-recommendation/files/2014/wp223_en.pdf [Accessed 25 Dec. 2015].

[11] Sheng, Z., Yang, S., Yu, Y., Vasilakos, A., Mccann, J., & Leung, K. (2013). A survey on the ietf protocol suite for the internet of things : Standards, challenges, and opportunities. Wireless Communications, IEEE, 20(6), 91-98.

[12] Daviet, C., Happy, M. and Zhao, M. (n.d.). Internet des Objets : problématiques de sécurité.

[13] Zanella, A., Bui, N., Castellani, A., Vangelista, L. and Zorzi, M. (2014). Internet of Things for Smart Cities. IEEE Internet of Things Journal, 1(1), pp.22-32.

[14] Gascon, D. (2015). Security in 802.15.4 and ZigBee networks | Libelium. [online] Libelium.com. Available at : http://www.libelium.com/security-802-15-4-zigbee/ [Accessed 25 Dec. 2015].

[15] Smith, M. (2015). Researchers exploit ZigBee security flaws that compromise security of smart homes. [online] Network World. Available at : http://www.networkworld.com/article/2969402/microsoft-subnet/researchers-exploit-zigbee-security-flaws-that-compromise-security-of-smart-homes.html [Accessed 25 Dec. 2015].

[16] Dini, G., & Tiloca, M. (2010, June). Considerations on security in zigbee networks. In Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC), 2010 IEEE International Conference on (pp. 58-65). IEEE.

[17] Cragie, R. (2009). ZigBee Security. 1st ed. [PDF] Available at : https://docs.zigbee.org/zigbee-docs/dcn/09-5378.pdf [Accessed 25 Dec. 2015].

[18] Zillner, T., & Strobl, S. (2015). ZigBee Exploited-The good, the bad and the ugly. Black Hat USA, 2015.

[19] J. Twining, "Behind The Numbers : Growth in the Internet of Things", Ncta.com, 2015. [Online]. Available : https://www.ncta.com/platform/broadband-internet/behind-the-numbers-growth-in-the-internet-of-things-2/. [Accessed : 24- Dec- 2015].

[20] Montenegro, G. and Kushalnagar, N. (2007). RFC 4944 - Transmission of IPv6 Packets over IEEE 802.15.4 Networks. [online] Tools.ietf.org. Available at : https://tools.ietf.org/html/rfc4944 [Accessed 25 Dec. 2015].

[21] Tools.ietf.org, (2011). IPv6 over Low Power WPAN Security Analysis. [online] Available at : https ://tools.ietf.org/html/draft-daniel-6lowpan-security-analysis-05 [Accessed 25 Dec. 2015].

[22] Abomhara, M. and Køien, G. (2015). Cyber Security and the Internet of Things : Vulnerabilities, Threats, Intruders and Attacks. Journal of Cyber Security and Mobility, 4(1), pp.65-88.

[23] Le Journal du Geek, (2014). Thread, le standard pour les objets connectés de Google et Samsung. [online] Available at : http://www.journaldugeek.com/2014/07/16/thread-standard-objets-connectes-google-samsung/ [Accessed 25 Dec. 2015].

[24] Fleury, G. (2015). Le protocole sans fil de Thread s'impose - Objetconnecte.com. [online] Objetconnecte.com. Available at : http://www.objetconnecte.com/protocole-sans-fil-thread-impose-1607/ [Accessed 25 Dec. 2015].

[25] Paganini, P. (2013). Near Field Communication (NFC) Technology, Vulnerabilities and Principal Attack Schema - InfoSec Resources. [online] InfoSec Resources. Available at : http://resources.infosecinstitute.com/near-field-communication-nfc-technology-vulnerabilities-and-principal-attack-schema/ [Accessed 25 Dec. 2015].

[26] Claure Cabrera, O. and Doyen-Le-Boulaire, M. (n.d.). Analyse du système de sécurité du drone : AR.Drone 2.0 Quad-Copter. [online] Available at : https://ensiwiki.ensimag.fr/images/a/ab/Drone.pdf [Accessed 25 Dec. 2015].

[27] Thesprawl.org, (2013). airoscapy | projects | sprawl. [online] Available at : http://www.thesprawl.org/projects/airoscapy/ [Accessed 25 Dec. 2015].

[28] Develle, K., Escande, L., Flamenbaum, R. and Wouters, T. (2012). ARDrone Parrot. [online] Available at : https://kadionik.vvv.enseirb-matmeca.fr/se/projets_avances/1213/rapport_sujet1_E3_1213.pdf [Accessed 25 Dec. 2015].

[29] . Lance and G. Kaleh, "A diversity scheme for a phase-coherent frequency-hopping spread-spectrum system", IEEE Transactions on Communications, vol. 45, no. 9, pp. 1123-1129, 1997.

[30] H. Doddala, "IoT end-to-end demo - Remote Monitoring and Service (Internet Of Things)", Blogs.oracle.com, 2015. [Online]. Available : https://blogs.oracle.com/IOT/entry/iot_end_to_end_demo. [Accessed : 24- Dec- 2015].

[31] C. Pichot, "Maintenance prédictive ?", Journée "Maintenance" - Club Automation, 2015. [Online]. Available : http://www.histoire-du-club-automation.org/journees2007/mars2007partieA.pdf. [Accessed : 24- Dec- 2015].

[32] V. Hermann, "Mise à jour forcée vers Windows 10 : Microsoft évoque une erreur", Nextinpact.com, 2015. [Online]. Available : http://www.nextinpact.com/news/96928-mise-a-jour-forcee-vers-windows-10-microsoft-evoque-erreur.htm. [Accessed : 24- Dec- 2015].

[33] . P. Philippe, "Boîte noire et assurance automobile", FFSA, Revue Risques, p. 57, 2003.

[34] J. Alfaiate and J. Fonseca, "Bluetooth security analysis for mobile phones", CISTI (Iberian Conference on Information Systems & Technologies), p. 169, 2012

[35] A. Ekbal, "Five Trends Shaping 802.11 WLANs", Mwrf.com, 2015. [Online]. Available : http://mwrf.com/systems/five-trends-shaping-80211-wlans. [Accessed : 20- Dec- 2015].

[36] 11 IoT protocols you need to know. (n.d.). [online] Available at : http://www.alliedelec.com/mkt/lp/aotb/pdf/aotb_article_iot_protocols.pdf [Accessed 25 Dec. 2015].