Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Kasdi Merbah Ouargla
Faculté Des Nouvelles Technologies de L'Information et La communication
Département Informatique

Mémoire de master professionnel

Spécialité : Administration et Sécurité des Réseaux

Présenté par : Mr BAYAT Mohammed Elaymene Mr BAYAT Mohammed Elamine

Encadré par Mlle Toumi Chahrazad

Thème

Un système sécurisé pour les maisons intelligentes

Soutenu le : 19/06/2022

Devant le jury composé de:

Mlle	Toumi Chahrazad	Encadreur	*M.A.A*
Mr	Harouz Abdelhakim	Président	M.A.A
Mr	Euschi Salah	Membre de jury	M.A.A

Année universitaire 2021/2022

Remerciements

*Nous remercions **ALLAH** le tout puissant d'avoir nous donner le courage, la force et la patience de mener à terme le présent travail.*

La réalisation de ce mémoire a été possible grâce au concours de plusieurs personnes à qui nous voulons témoigner toutes nos reconnaissances.

*Nous offrons nos sincères et chaleureux remerciements à notre encadreur de recherche, Mlle. **Toumi Chahrazad**. Nous la remercions de nous avoir encadré, orienté, aidé et conseillé,son ouverture d'esprit, et ses qualités scientifiques exceptionnelles.*

Nous remercions également nos parents pour leur soutien et leurs encouragements.

Nous remercions aussi les membres du jury d'avoir accepté de présider le jury de notre soutenance, et pour avoir accepté de juger ce modest travail.

Nous désirons aussi remercier les professeurs de notre faculté, qui nous avons fourni les outils nécessaires à la réussite de nos études universitaires.

Dédicaces

Au meilleur des pères

À notre très chère maman

Qu'ils trouvent en moi la source de leur fierté

À qui je dois tout

À mes Amis

À tous ceux qui me sont chers

Résumé

L’évolution technologique nous permet aujourd’hui de prédire les espaces et d’améliorer l’adaptation du mode de vie dans les maisons.

L'internet a des utilisations qui ne se limite pas à la gestion des réseaux, mais s'étend à la gestion des objets (IOT). L'IoT est un domaine qui a connu les jours grâce ces évolutions. Cette technique fleurissante, faite de plus en plus partie de notre vie quotidienne. La domotique, où smart home, est une application populaire de l'IOT dans la vie réelle.

Cependant, comme tout autre domaine, l'IoT pose certains problèmes qui nécessitent une résolution. Principalement, la sécurité des informations et du réseau, le degré de la latence et la bande passante, et la consommation de l'énergie. L'objectif principal de notre travail est de concevoir et réaliser un système "AMINE" pour la sécurisation des smart homes. Aussi, nous avons développé une application multiplateforme, accessible via page web ou application mobile Android ou IOS, pour l'accès au système. De plus, nous avons réalisé un prototype d’une maison intelligente qui utilise l’énergie solaire pour y implémenter et tester le système. Nous avons aussi, étudier et comparer les temps d'envoi des données en utilisant différents algorithmes et mécanismes de sécurité.

Mots clés : Smart home, domotique, IoT, sécurité, JWT, RestApi.

ملخص

يسمح لنا التطور التكنولوجي اليوم بالتنبؤ بالمساحات وتحسين تكييف نمط الحياة في المنازل

إنترنت الأشياء هو مجال معروف الأيام .(IOT) للإنترنت استخدامات لا تقتصر على إدارة الشبكة، ولكنها تمتد إلى إدارة الكائنات بفضل هذه التطورات. هذه التقنية المزدهرة ، جعلت المزيد والمزيد من حياتنا اليومية. أتمتة المنزل ، أو المنزل الذكي ، هو تطبيق في الحياة الحقيقية IOT شائع

ومع ذلك ، مثل أي مجال آخر ، يطرح إنترنت الأشياء بعض المشاكل التي تحتاج إلى معالجة. في المقام الأول ، أمن المعلومات والشبكات ، ودرجة الكمون وعرض النطاق الترددي ، واستهلاك الطاقة

الهدف الرئيسي لعملنا هو تصميم وتنفيذ نظام «أمين» لتأمين المنازل الذكية. أيضًا، قمنا بتطوير تطبيق عبر النظام الأساسي، يمكن للهاتف المحمول، للوصول إلى النظام. بالإضافة إلى ذلك، قمنا IOS أو Android الوصول إليه عبر صفحة الويب أو تطبيق بتطوير نموذج أولي لمنزل ذكي يستخدم الطاقة الشمسية لتنفيذ واختبار النظام. نقوم أيضًا بدراسة ومقارنة أوقات تحميل البيانات باستخدام خوارزميات وآليات أمان مختلف

الكلمات المفتاحية: المنزل الذكي، إنترنت الأشياء، الأمن

Abstract

Technological change allows us today to predict spaces and improve the adaptation of the way of life in homes.

The Internet has uses that are not limited to network management, but extend to object management (IOT). IoT is a field that has known the days thanks to these evolutions. This flowering technique, made more and more part of our daily life. Home automation, where smart home, is a popular application of IOT in real life.

However, like any other domain, IoT poses certain problems that require resolution. Mainly, information and network security, the degree of latency and bandwidth, and energy consumption. The main objective of our work is to design and implement an "AMINE" system for securing smart homes. Also, we have developed a cross-platform application, accessible via web page or Android or IOS mobile application, for access to the system. In addition, we have developed a prototype of a smart home that uses solar energy to implement and test the system. We also study and compare data upload times using different algorithms and security mechanisms.

Key words : Smart home, domotique, IoT, sécurity, JWT, RestApi.

Table des matières

Liste des figures

Liste des tableaux

Liste des abréviations

IOT	Internet of things
IDO	Iternet des objets
CC	Cloud computing

Introduction Générale

La technologie et l'internet occupent une grande partie de notre vie de tous les jours. Aujourd'hui, l'utilisation de l'internet n'est pas limité à la gestion des réseaux, mais aussi s'est étendu à la gestion des objets, et s'appelle l'Internet des objets.. L'internet des Objets, ou Internet of things (IoT), se définit comme un réseau mondial de services inter-connectés, et d'objets intelligents de toutes natures. Parmi les domaines les plus en vue de l'utilisation de cette nouvelle technologie est le domaine de domotique ce qui est actuellement appelé les maisons intelligentes. La maison intelligente, qui est très connus sous le nom de Smart home, est un éspace dans lequel les appareils intelligentes sont connectés via des passerelles résidentielles. Ce qui constituent un réseau domestique local pour aider les gens dans leurs activités de la vie quotidienne. L'IoT converge avec d'autre technologies, le plus souvent le cloud computing, le fog ou le edge computing, et le Big data. Le cloud computing est un modèle informatique qui permet un accès facile et à la demande par le réseau à un ensemble partagé de ressources informatiques configurables (serveurs, stockage, applications et services). Avec l'évolution croissante de l'internet et des technologiques l'IoT a aussi émergé. Ceci nous amène à vivre dans une époque digitale, ou la sécurité est devenue de plus en plus demander.

Cependant, comme toute autre domaine, l'IoT pose certains des problèmes qui attendent d'être résolus. Principalement, la sécurité des informations et du réseau. En plus de la sécurisation de la smart home, répond a certains problématique existantes dans le domaine. Notre système répond a certains des questions posés dans le domaine : (1) comment faite sécuriser les données qui circulent ? ; (2) comment faire pour diminuer la latence

de transmission des données ; (3) comment faire si il y'a une coupure de l'internet ?

L'objectif de ce mémoire est de concevoir et réaliser un système "AMINE" pour la sécurisation des smart home. Aussi, nous visons mettre au point un prototype d'une maison intelligente pour y implémenter et tester notre système. Dans notre travail, en plus de la sécurisation de la smart home, on cherche à résoudre quelques problèmes existants dans le domaine. Notre système, vise à diminuer la latence, augmenter la bande passante, et sécuriser les informations transmises. Aussi, nous voulons réaliser une application pour l'accès au système sous forme d'une page web ou application Android. De plus, nous allons étudier et comparer les temps d'envoie des données en utilisant différentes algorithmes et mécanismes de sécurité.

Notre mémoire sera décliné en **trois** chapitres :

Chapitre 1 : dans ce chapitre nous allons présenter globalement l'Internet of Things (définition, architecture, caractéristiques, cas d'applications, ...). Après, nous allons exposer le cloud computing (définition, les types et les services du cloud). Nous terminerons ce chapitre par la notion de fog computing.

Chapitre 2 : dans le deuxième chapitre nous allons exposer la sécurité de l'information et leurs notions de bases. Dans ce chapitre nous allons présenter les mécanismes de sécurité tel que la cryptographie, la signature et les fonction de hachage. Après, on va citer quelque protocole de sécurité, les services de sécurité et nous détaillerons le service d'authentification avec ses techniques.

Chapitre 3 : dans le troisième chapitre nous allons présenter la motivation et objectifs de ce travail, la conception de notre système et une description de toutes les étapes de la réalisation d'un prototype de la maison intelligente, le système de sécurité de la maison intelligente et les différents outils que nous avons utilisés dans notre projet et les résultats obtenus.

Nous terminerons ce mémoire par une conclusion générale et quelques perspectives pour des travaux futures.

Chapitre 1

Généralités sur l'internet des objets (IDO)

1.1 Introduction

L'internet des objets donne une vision sur le monde connecté par des milliards d'objets intelligents qui sont capables de communiquer, connecter, détecter, actionner, grâce à une interconnexion avec les gens, les données et tous les objets où se crée une fusion entre le monde réel (physique) et le monde numérique (virtuel).

Dans le premier, nous allons donner les grands axes de l'Internet of Things (IoT) : sa définition, l'historique, son fonctionnement, ensuite ses principaux techniques, ses protocoles de communications. Après, les cas d'applications d'IOT et ses avantages et inconvénients. Après, nous allons présenter le cloud computing : sa définition, ses services, les types du nuage et les avantages et les inconvénients de ce dernier. Enfin, nous allons terminer ce chapitre avec la notion de fog computing.

1.2 Notion d'internet des objets

À l'origine, le concept Internet des objets où IoT pour l'Internet of Things débute en 1999 par Kevin Ashton, pionnier de la technique RFID (radiofrequency identification technologie d'identification automatique).

La même année, aux États-Unis et plus particulièrement au Massachusetts Institut of Technologie. Un groupe de laboratoire travaillait pour la création d'objets connectés à l'aide de l'identification par radiofréquence et les réseaux de capteurs sans fil.[16]

L'Internet des objets (IoT), également appelé Internet of Everything ou L'Internet industriel est un nouveau paradigme technologique envisagé comme un réseau mondial de machines et appareils capables d'interagir entre eux. L'IoT est reconnu comme l'un des domaines les plus importants de la technologie future et attire une grande attention d'un large éventail d'industries.[17]

FIGURE 1.1 – Internet des objets [1]

1.2.1 Définitions

Tel que définit par l'Union Internationale des Télécommunications (UIT), l'Internet des objets (IoT) est (une infrastructure mondiale pour la société de l'information, qui permet de disposer de services évolués en interconnectant des objets (physiques et virtuels) grâce aux technologies de l'information et de la communication interopérable existante ou en évolution).[18]

L'Internet des objets (Ido) est un nouveau paradigme à l'époque actuelle qui a donné

naissance à l'avènement de l'informatique omniprésente. Il aide un grand nombre d'équipements IoT à échanger des données détectées, à réagir aux événements et à interagir avec l'environnement en se connectant à Internet. Cette interaction entre des appareils hétérogènes permet des applications omniprésentes à l'IoT. Comme le nombre d'objets connectés à Internet est énorme et que l'IoT englobe une gamme extrêmement large d'applications, il est donc crucial d'avoir une architecture qui permet un contrôle flexible et la connectivité.[19]

1.2.2 Historique de l'IoT

La première idée de l'IoT est apparue il y a presque deux décennies, mais les technologies il avait déjà existé et étaient en développement depuis de nombreuses années. Regardons l'histoire de l'évolution de l'IoT et de ses technologies ordre chronologique :

- **En 1964** Internet, la principale technologie à l'origine de l'IoT Research Project Agency Network (ARPANET), principalement utilisé par de recherche pour partager les travaux de recherche, développer de nouveaux techniques d'interconnexion et de relier les ordinateurs à de nombreux centres informatiques du département de la Défense des États-Unis et aussi en public et secteur privé.[20]
- **En 1973** une autre technologie essentielle pour l'IoT est la RFID (radiofréquence Identification). Bien que les racines de l'IRF peut être retracée à la seconde Guerre mondiale et les progrès se sont poursuivis tout au long des années 1950 et 1960, mais le premier brevet américain pour la balise RFID avec mémoire réinscriptible a été reçu par Mario W. Cardullo en 1973. Cependant, un entrepreneur basé en Californie, Charles Walton a également reçu un brevet dans la même année pour le transpondeur passif pour déverrouiller la porte à distance.[21]
- **En 1991** Le concept d'informatique omniprésente a été proposé par Mark Weiser. L'informatique omniprésente a fait usage de l'informatique embarquée avancée comme un ordinateur d'être présent dans tout, mais invisible. Plus tard, il était connu comme l'informatique omniprésente. [22]
- **En 1999** La communication entre appareils a été présentée par Bill Joy dans son taxonomie d'Internet et le terme (Internet des objets) a été utilisé pour la première

fois par Ashton.[23]

— **En 2000** Grâce à la numérisation, la connectivité Internet est devenue norme pour de nombreuses applications et toutes les affaires et les produits devaient présence sur Internet et de fournir des informations en ligne. Toutefois, ces appareils sont encore principalement des choses sur Internet qui nécessitent plus d'humains interaction et surveillance au moyen d'applications et d'interfaces.[24]

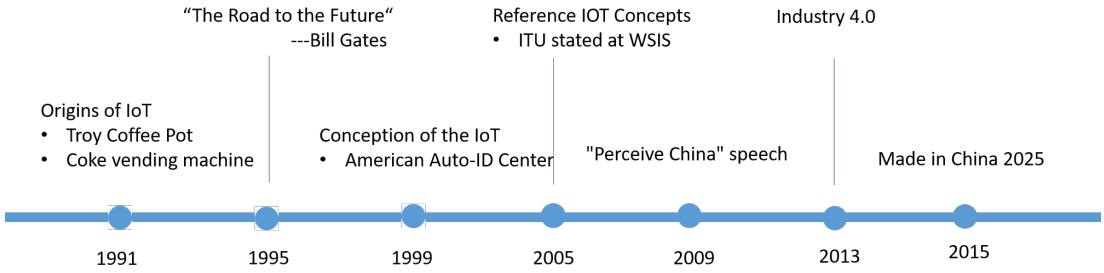


FIGURE 1.2 – Historique du développement de l'IoT [25]

1.3 Les composantes d'IoT

D'une façon pratique, un tel projet d'IoT s'articule sur cinq composantes, à savoir Les objets (les capteurs, les actionneurs), Le réseau, Les données, Les applications d'exploitation.

1.3.1 Les objets

Un objet dans l'Internet des objets peut être une personne avec un implant de moniteur cardiaque, un animal de ferme avec un transpondeur à biochimie, une automobile qui a des capteurs intégrés pour alerter le conducteur lorsque la pression des pneus est faible ou tout autre naturel ou humain...objet créé pouvant recevoir une adresse de protocole Internet (IP) et pouvant transférer des données sur un réseau.[26]

— **Les capteurs**

Les capteurs sont utilisés pour détecter des objets et des appareils, etc. Un appareil qui fournit une sortie utilisable en réponse à une mesure spécifiée. Le capteur at-

teint un paramètre physique et le convertit en un signal adapté au traitement (par exemple électrique, mécanique, optique) des caractéristiques de tout dispositif ou matériau pour détecter la présence d'une quantité physique particulière.[27]

— **Actionneur**

L'actionneur est un dispositif matériel pour transformer une information digitale en un phénomène physique ; d'où sa dénomination. Il peut moduler le comportement ou changer l'état d'un système. Il peut s'agir d'alarmes ou d'interrupteurs. [28]

— **Le micro contrôleur**

Le micro contrôleur Comprend des informations principales sur le produit telles que la portée, la segmentation et la perspective. De même, il comprend les statistiques de l'offre et de la demande, la faisabilité des investissements et les segments qui limitent la croissance d'une industrie. Il fournit spécifiquement Micro contrôleur IoT (MCU) la demande de produits, les procédures annuelles et la phase de croissance de l'industrie. [29]

Le micro contrôleur est équipé de plusieurs ports de connexion, de microprocesseur, ram, mémoire (vive et morte) comme un ordinateur, à la seule différence ses ressources sont très limitées accès pour faire que de petites opérations.[30]

— **Le micro-ordinateur**

Les micro-ordinateur sont souvent utilisés pour faire la passerelle (gateway) car ayant une puissance de calcul et une mémoire accès proches des ordinateurs. Certains même l'utilise comme ordinateur de bureau. On peut mettre un système d'exploitation dans ces micro-ordinateurs (linux la majeure partie) et ils sont équipés de bases de connectique pour accéder à internet et de se faire, on les utilise pour recevoir les données provenant des capteurs pour ensuite les envoyés au serveur cloud. [31][30]

IOT passerelle prend en charge une variété de communication. protocoles et types de données entre les différents capteurs.[32]

1.3.2 Le réseau

Le réseau IoT sert à doter un objet d'une connectivité à Internet pour permettre la remontée d'informations. Différents protocoles de communication sont disponibles sur le marché pour effectuer cela. Tous n'ont pas les mêmes caractéristiques. Pour les entreprises qui se lancent dans l'IoT, choisir le réseau de communication le plus adapté à leurs usages peut ainsi devenir un casse-tête chinois. Elles doivent prendre en compte la couverture du réseau, la durée de vie des objets sur batterie, la distance de communication ou encore le coût de service. [33]

1.3.3 Les données et les informations

Elles sont la source des valeurs, elles sont générées par les objets et stockées dans des BDDs pour une performante solution. [34]

Les informations sont les résultantes des données traitées, corrélées etanalysées, notre Diamant taillé. Ces informations doivent elles-aussi être stockées, archivées et sauvegardées dans des bases de données. [35]

1.3.4 Les applications d'exploitations

Les applications d'exploitation sont en principe les interfaces Homme-machine (IHM) dans lesquelles nous pouvons visualiser les données sous forme de tableau de bord. Ce sont les interfaces pour la visualisation des informations, on trouve les tableaux de bords, les Chartes, les graphes, ...etc.[35]

FIGURE 1.3 – Les 5 composantes de l'IoT[35]

1.4 L'architecture IOT

L'architecture existante d'Internet a été adoptée il y a environ quatre décennies dans la forme protocoles TCP/IP, mais aujourd'hui, il est incompatible de servir l'énorme réseau d'internet des objets.[36]

Les six couches de l'IoT sont décrites ci-dessous :

1.4.1 La couche de codage

La couche de codage est la base de l'IoT qui fournit l'identification aux objets d'intérêt. Dans ce calque, chaque objet est ID unique qui permet de discerner facilement les objets. [37]

1.4.2 La couche perception

C'est la couche perception de l'IoT qui donne un sens physique à chaque objet. Il se compose de capteurs de données sous différentes formes comme RFID étiquettes, capteurs IR ou autres réseaux de capteurs qui pourraient détecter la température, l'humidité, la vitesse et l'emplacement des objets, etc.

Cette couche rassemble les informations utiles des objets à partir de la capteurs liés à eux et convertit l'information en signaux numériques qui sont ensuite transmis à la couche réseau pour autres mesures.[38]

1.4.3 La couche réseau

Le but de cette couche est de recevoir les informations utiles dans le forme de signaux numériques de la couche de perception et de les transmettre à les systèmes de traitement dans la couche Middleware à travers les supports de transmission comme WiFi, Bluetooth, WiMaX, Zigbee, GSM, 3G etc avec des protocoles comme IPv4, IPv6, MQTT, DDS, etc.[39]

1.4.4 La couche Middlware

Cette couche traite les informations reçues des capteurs. Il comprend les technologies comme Cloud computing, Ubiquitous computing qui assure un accès direct à la base de données pour y stocker toutes les informations nécessaires. À l'aide d'un équipement de traitement intelligent, l'information est traitée et une action entièrement automatisée est prise en fonction des résultats de traitement de l'information.[40]

1.4.5 La couche Application

Cette couche fournit le service personnalisé en fonction des besoins de l'utilisateur, en utilisant le résultat des données traitées. Diverses applications intelligentes de haut niveau de l'IoT est réalisé pour toutes sortes d'industries. Les applications liées à l'IoT pourrait être un désastre surveillance, surveillance de la santé, maisons intelligentes, transport intelligent, planète intelligente etc. Ces applications encouragent l'expansion de l'IoT, et donc cette couche est essentiel au développement d'un réseau IoT à grande échelle.[41]

1.4.6 La couche Business

La couche Business est la couche supérieure de l'architecture IoT, où divers les modèles d'affaires sont générés pour les stratégies d'affaires efficaces. Les applications et les services

fournis par IoT sont gérés dans cette couche.[42]

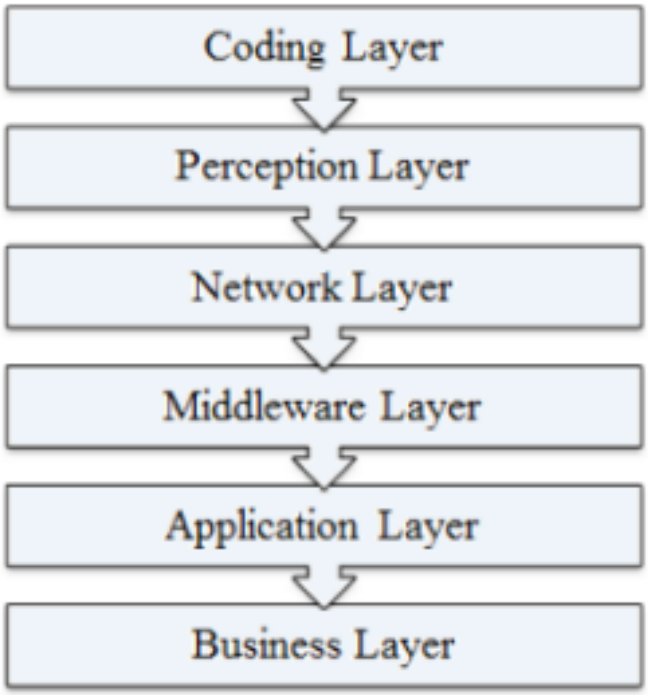


FIGURE 1.4 – Les six couches de l'IoT [2]

1.5 Les Technologies de communication IoT

Le développement d'un système informatique omniprésent où les objets numériques peuvent être identifiés de manière unique et peuvent penser et interagir avec d'autres objets pour collecter des données sur la base desquelles des actions automatisées sont prises, la nécessité d'une combinaison de technologies nouvelles et efficaces qui ne sont possibles que par intégration de différentes technologies qui peuvent rendre les objets à être identifiés et communiquer entre eux.[43]

1.5.1 Radio frequency identification (RFID)

Un système d'identification par radiofréquence (RFID) est un système de communication sans fil dans lequel la liaison radio entre la station de base et les transpondeurs est

fournie par les ondes rétrodiffusées modulées.[44] Une description plus complexe est une identification et des données électromagnétiques de proximité système de transaction. Utilisation des (étiquettes RFID) sur les objets ou les actifs, et des (lecteurs) pour recueillir l'étiquette information, l'IRF représente une amélioration par rapport aux codes à barres communication de proximité, densité de l'information et capacité de communication bidirectionnelle. Les systèmes opérationnels d'IRF impliquent des étiquettes et des lecteurs qui interagissent avec des objets (actifs) et systèmes de base de données pour fournir une fonction d'information et/ou opérationnelle.

RFID est utilisé pour une grande variété d'applications allant de l'accès familier au bâtiment, contrôler les cartes de proximité pour le suivi de la chaîne d'approvisionnement, collecter des péages, l'accès au stationnement des véhicules contrôle, gestion des stocks de détail, accès aux remontées mécaniques, suivi des livres de bibliothèque, prévention du vol, les systèmes d'immobilisation des véhicules et l'identification et le suivi du matériel roulant ferroviaire.[45]

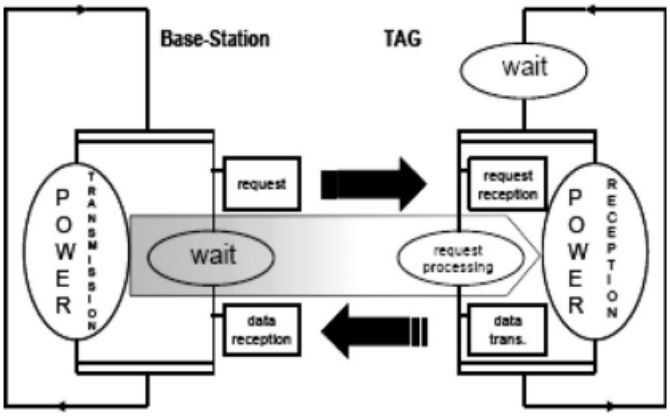


FIGURE 1.5 – Le system RFID[46]

1.5.2 Wi-Fi

Le réseau IEEE 802.11 est une spécification du réseau local sans fil (WLAN). En mode basse bande, IEEE 802.11(b, g, n) transmettre des données de 11 Mbps et jusqu'à 54M bps et va jusqu'à 32 mètres à l'intérieur et 95 mètres à l'extérieur.

La norme IEEE 802.11n utilise le double du spectre radio par rapport à 802.11a ou

802.11g. Cependant, IEEE 802.11a, c données de transmission est jusqu'à Gbps et peut dépasser la plage de plus de deux fois le b et g Le WiFi à basse bande transmet dans la bande ISM 2,4 GHz tandis que le WiFi à haute bande transmet dans la bande 5 GHz.[47]

1.5.3 Bluetooth

Bluetooth est une technologie sans fil d'une faible portée qui repose sur la proximité physique afin de gérer les connexions entre les appareils, sans avoir besoin d'un mot de passe. La norme Bluetooth utilise des ondes radio UHF (ultra haute fréquence) entre 2.400 et 2.485 GHz, qui peut étendre un maximum de 164 pieds entre deux appareils.[48]

Le Bluetooth permet d'obtenir des débits de l'ordre de 1 Mbps, correspondant à 1600 échanges par seconde en full-duplex, avec une portée d'une dizaine de mètres environ avec un émetteur de classe II et d'un peu moins d'une centaine de mètres avec un émetteur de classe I.

Les normes de Bluetooth

— IEEE 802.15.3 est un standard en cours de développement visant à proposer du haut débit (20 Mbit/s) avec la technologie Bluetooth ;

— IEEE 802.15.4 est un standard en cours de développement pour des applications Bluetooth à bas débit. [49]

1.5.4 ZigBee

ZigBee est développé par ZigBee alliance, qui compte des centaines de sociétés membres (Ember, Freescale, Chipcon, Invensys, Mitsubishi, CompXs, AMI Semi-conducteurs, ENQ Semi-conducteurs), à partir de semi-conducteurs et les développeurs de logiciels à l'original fabricants d'équipement. ZigBee et 802.15.4 ne sont pas la même chose. ZigBee est un protocole réseau basé sur des normes soutenu uniquement par l'alliance ZigBee qui utilise le services de transport de la spécification de réseau IEEE 802.15.4.[50]

Les caractéristiques de ZigBee sont : coût avantageux, débit : 10kbps-115.2kbps, portée radio : 10-75m, jusqu'à 65k noeuds par réseau, jusqu'à 100 réseaux co-localisés, jusqu'à 2 ans de durée de vie de batterie standards Alkaline. [51]

1.5.5 LORA

LoRa est la nouvelle technologie de communication Réseau étendu de faible puissance (LPWAN). Il s'agit d'un schéma de transmission sans fil ultra longue distance basé sur la technologie à spectre étalé. Il met l'accent sur la communication à longue portée avec la capacité de haute sensibilité de réception qui lui permet de travailler sous l'interférence sonore ou le plancher de bruit. Lora est une technologie de transmission WAN basse consommation. Il est principalement utilisé dans l'Internet des objets. Lora est l'abréviation du mot anglais longrange. La longue portée représente également un avantage essentiel de Lora, avec une longue distance de transmission. [52]

Caractéristiques de LoRa sont : longue distance de transmission : sensibilité -148dBm, distance de communication jusqu'à 15 kilomètres, faible consommation d'énergie de travail, de nombreux nœuds de réseau, forte capacité anti-interférence, et à petit prix.[53]

1.5.6 Technologies de réseau

Ces technologies jouent un rôle important dans le succès de l'IoT puisqu'ils sont responsables de la connexion entre les objets, donc nous avons besoin d'un réseau rapide et efficace pour gérer un grand nombre de dispositifs potentiels. Pour le réseau de transmission à large portée, nous utilisons couramment 3G, 4G, etc. mais, comme nous le savons, le trafic mobile est tellement prévisible car il n'a qu'à effectuer les tâches habituelles comme faire un appel, l'envoi d'un message texte, etc. afin que nous entrons dans cette ère moderne de l'informatique omniprésente, il ne sera plus prévisible qui demande un système super-rapide, super-efficace de cinquième génération sans fil qui pourrait offrir beaucoup plus de bande passante. [54]

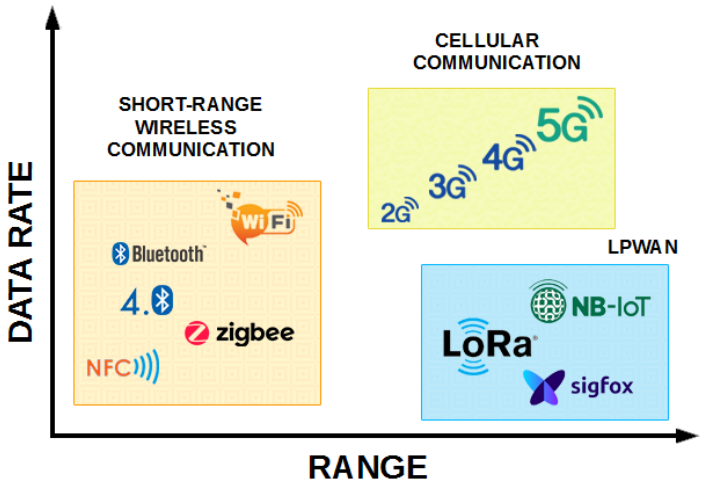


FIGURE 1.6 – Technologies IOT [3]

Le tableau suivant1.1 présente quelque caractéristiques des technologies de communication IOT :

Réseau	Bande passante	Distance	Energie	Coût
Bluetooth	0.27 mégabit/s	100 mètres	base	bas
Wifi	54 mégabit/s	50 mètres	moyenne	bas
4G	50 mégabit/s	1 kilomètre	haute	haute
5G	10 Gigabit/s	100 mètres	haute	haute
Zigbee	250 kbit/s	100 mètres	basse	bas
Lora	50 kbit/s	1 kilomètre	basse	moyenne

TABLE 1.1 – Les caractéristiques de technologies de communication iot

1.6 Les Protocoles de l'IoT

Il y a de nombreux protocoles parmi lesquels nous mentionnons :

1.6.1 HTTP

Le protocole de transfert hypertexte (HTTP) est un protocole au niveau de l'application pour les systèmes d'information hypermédias distribués, collaboratifs. C'est un protocole générique qui peut être utilisé pour de nombreuses tâches au-delà de son utilisation pour l'hypertexte, tels que les serveurs de noms et les systèmes de gestion d'objets distribués, transférer des données sur le Web, grâce à l'extension de ses méthodes de demande, codes d'erreur et en-têtes.[55]

Une caractéristique de HTTP est la saisie et la négociation de la représentation des données, permettant aux systèmes d'être construits indépendamment des données transférées. HTTP est utilisé par l'initiative mondiale d'information sur le World-Wide Web depuis 1990. Cette spécification définit le protocole appelé "HTTP/1.1", et est une mise à jour de la RFC 2068.[56]

HTTP basé sur l'architecture client/serveur. Un client, par exemple, peut être un ordinateur personnel, un ordinateur portable ou un périphérique mobile. Le serveur HTTP est généralement un hôte Web exécutant un logiciel de serveur Web, tel que Apache ou IIS. Lorsque vous accédez à un site Web, votre navigateur envoie une requête au serveur Web correspondant et il répond avec un code d'état HTTP. Si l'URL est valide et que la connexion est établie, le serveur enverra à votre navigateur la page Web et les fichiers associés. [4]

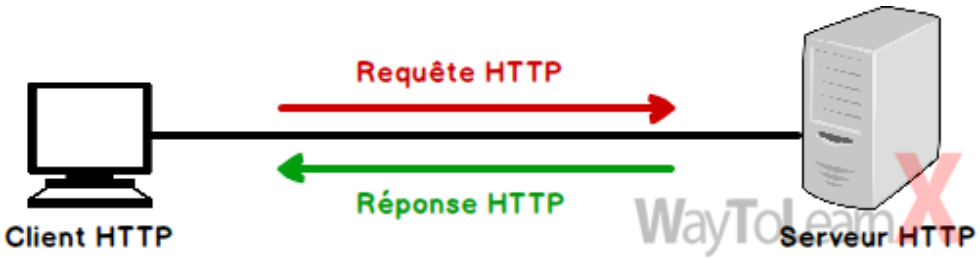


FIGURE 1.7 – Protocole HTTP [4]

HTTP définit également des commandes telles que GET et POST, utilisées pour gérer les soumissions de formulaires sur des sites Web. La commande CONNECT est utilisée pour faciliter une connexion sécurisée cryptée à l'aide de SSL. Les connexions HTTP

cryptées se font via HTTPS, une extension de HTTP conçue pour les transmissions de données sécurisées.[4][57]

1.6.2 MQTT

MQTT est un protocole de publication-abonnement basé sur TCP développé par IBM, puis open-sourcé pour la messagerie applications. Dans un format de publication-abonnement, les clients peuvent soit (publier) des données sur un sujet particulier sur le serveur ou (s'abonner) à un sujet où le serveur enverra automatiquement de nouvelles données sur le sujet à l'abonné une fois enregistré.

MQTT combine le coût relativement élevé et la qualité de vie élevée de TCP avec le un-à-un, un-à-plusieurs, et plusieurs-à-un capacités d'un format de publication-abonnement. En outre, ce protocole permet aux clients de préciser les sujets de télémétrie qui les intéressent et ne reçoivent que des données publiées sur ces sujets. [58]

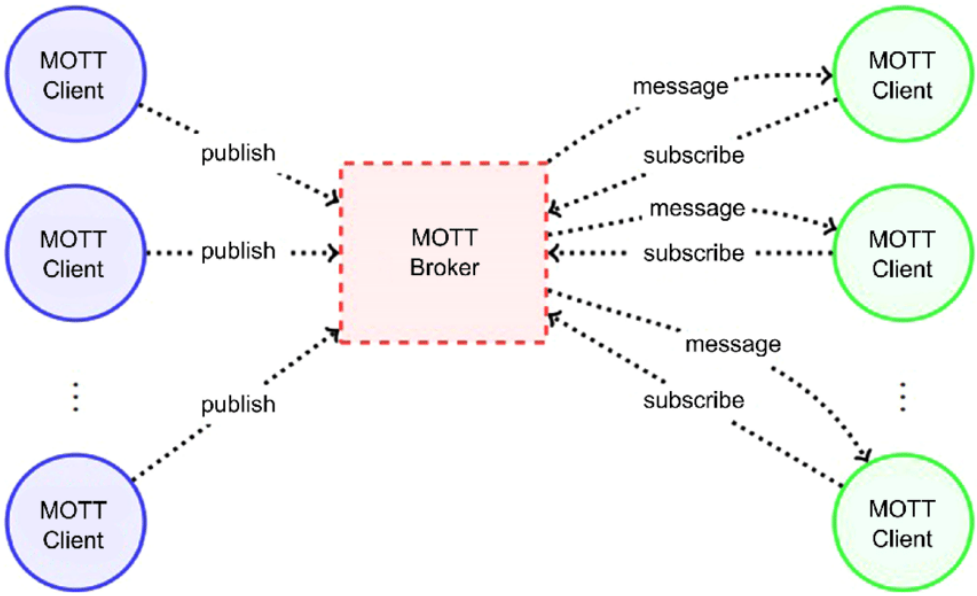


FIGURE 1.8 – Protocole MQTT [5]

1.6.3 CoAP

CoAP (Constrained Application Protocol) est un protocole apatride développé par l'IETF pour remplacer HTTP dans les périphériques limités par des ressources. Étant un

protocole RESTful UDPbased, il utilise une structure de requête/réponse et a de faibles frais généraux et un faible degré de QoS. Afin de recevoir de la télémétrie, un client doit constamment demander au serveur de envoyer l'information. CoAP soutient principalement un peer-to-peer style de communication, mais peut être étendu pour prendre en charge les fonctions uniques via l'utilisation du multicast IP. [59]

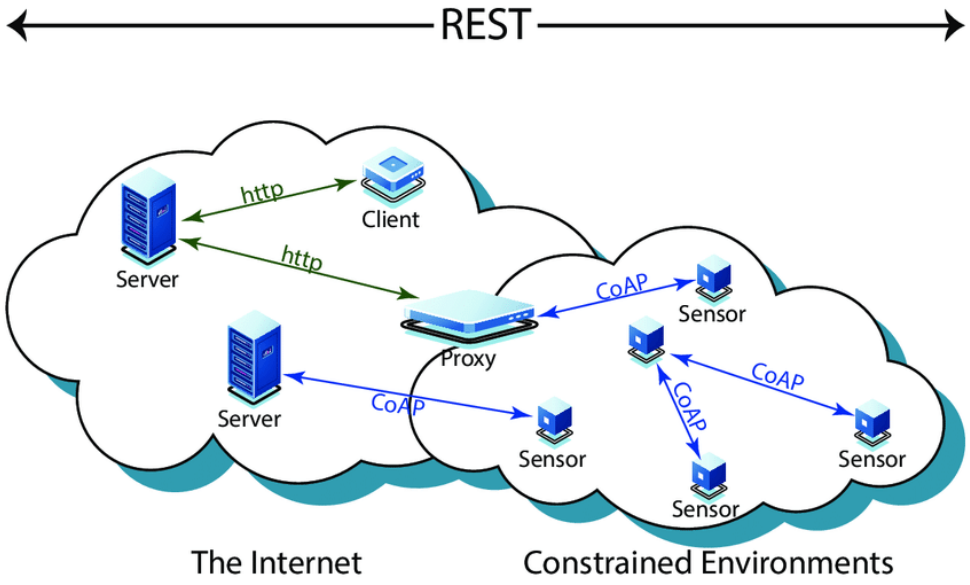


FIGURE 1.9 – Protocole coap [6]

1.6.4 REST API

Une API REST (également appelée API RESTful) est une interface de programmation d'application (API ou API web) qui respecte les contraintes du style d'architecture REST et permet d'interagir avec les services web RESTful. L'architecture REST (Representational State Transfer) a été créée par l'ingénieur des technologies de l'information Roy Fielding.[60]

Une API est un ensemble de définitions et de protocoles qui facilitent la création et l'intégration de logiciels d'application. Elle est parfois considérée comme un contrat entre un fournisseur d'information et un utilisateur d'information, ce qui permet de définir le contenu demandé au consommateur (l'appel) et le contenu demandé au producteur (la réponse). Par exemple, l'IPV conçu pour un service météorologique peut exiger que

l'utilisateur fournisse un code postal et que le producteur retourne une réponse en deux parties : la première pour la température maximale et la seconde pour la température minimale.[61]

REST est un ensemble de contraintes architecturales. Ce n'est ni un protocole ni un standard. Les développeurs d'API peuvent implémenter REST de nombreuses façons.

Lorsqu'un client émet une requête via une API RESTful, cette API transfère une représentation de l'état de la ressource au demandeur ou à l'endpoint. Cette information, ou représentation, est fournie via le protocole HTTP dans l'un des formats suivants : JSON (JavaScript Object Notation), HTML, XLT, Python, PHP ou texte brut. Le langage de programmation le plus couramment utilisé est JSON, parce que, contrairement à ce que son nom indique, il ne dépend pas d'un langage et peut être lu par les humains et les machines. Il travail avec les codes d'états (state = 400, 200,..) [60]

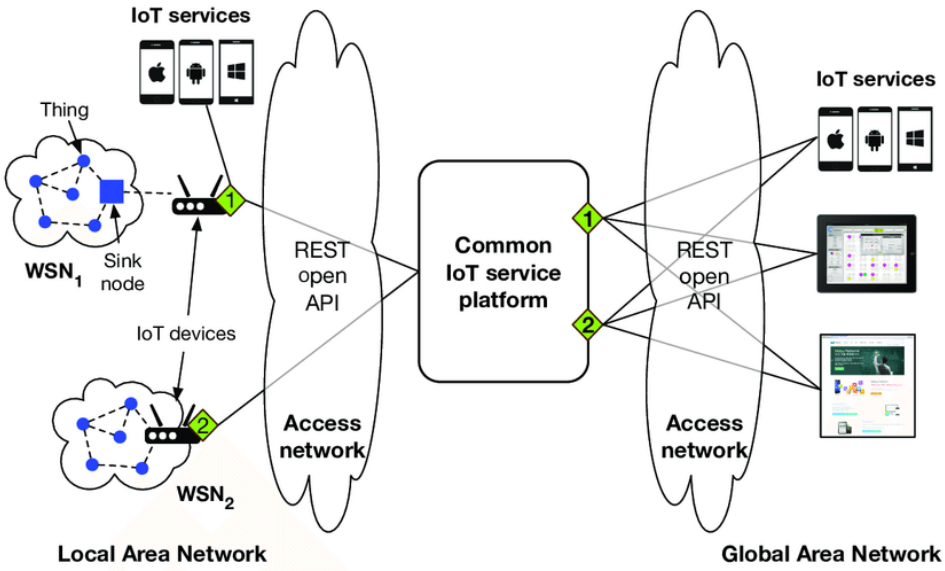


FIGURE 1.10 – Protocole Rest Api [7]

L'idée de base de toute API REST est la ressource qui est un objet ayant des types de données associées, une relation avec d'autres ressources, et un ensemble de fonctions qui fonctionnent dessus. Une ressource est dite être tout type d'information ou de données comme un document texte, une image, un service temporel ou un ensemble d'une autre ressource, etc. [8]

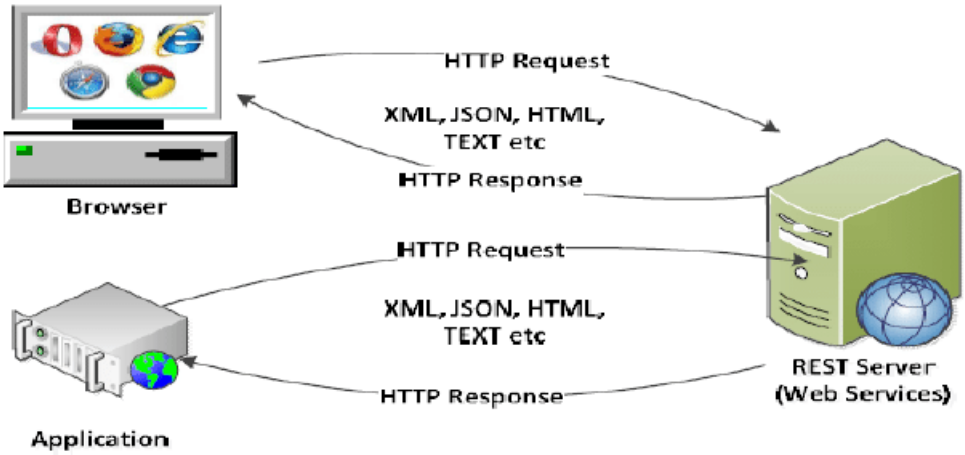


FIGURE 1.11 – Architecture de Rest Api [8]

Verbes de requête : Les verbes de requête sont généralement des méthodes HTTP qui décrivent ce qui doit être fait avec les données disponibles sur le serveur. Le navigateur utilise le verbe GET pour obtenir des données du serveur. Comme cela d'autres verbes de requête tels que PUT, POST, DELETE sont utilisés par le navigateur pour effectuer une opération sur le serveur. [8]

La méthode **Get** est utilisée pour récupérer ou obtenir les informations du serveur donné en utilisant une URL donnée. Dans REST API, elle effectue l'opération de lecture.

La méthode **Post** est utilisé pour envoyer des données au serveur comme le téléchargement d'un fichier ou le transfert de certaines données ou l'ajout d'une nouvelle ligne à la table de fin de page à tout type de formulaire web. En une phrase simple, on peut dire que la méthode post est utilisée pour insérer de nouveaux éléments dans le serveur principal. Dans l'opération REST API, il effectue l'opération de création.

La méthode **PUT** est le plus souvent utilisée pour mettre à jour une ressource existante.

La méthode **DELETE** est utilisée pour supprimer une ressource spécifiée par son URI.

1.7 Domaines d'applications de l'IoT

Aujourd'hui, l'IoT est partout, elle introduit plusieurs cas d'utilisation prometteurs et continue à se répandre touchant très nombreux secteurs notamment la santé, la domotique, l'industrie et d'autres.

1.7.1 Télémédecine

Hôpitaux intelligents : Les hôpitaux étaient équipés de flexible portable intégré avec des étiquettes RFID qui a donné aux patients, par lesquels non seulement les médecins mais aussi les infirmières sont également en mesure de surveiller la fréquence cardiaque, la pression artérielle, la température et autres conditions des patients à l'intérieur ou à l'extérieur des locaux de hôpital.

Il y a de nombreuses urgences médicales mais les ambulances prennent un certain temps pour atteindre le patient, Drone Ambulances sont déjà sur le marché qui peut voler à la scène avec la trousse d'urgence de sorte qu'en raison d'une surveillance appropriée, les médecins seront en mesure pour suivre les patients et peut envoyer le drone pour fournir soins médicaux jusqu'à l'arrivée de l'ambulance.[62]

Dans le secteur de la santé, des hôpitaux utilisent l'Internet des choses pour améliorer les soins aux patients et augmenter la productivité.[63]

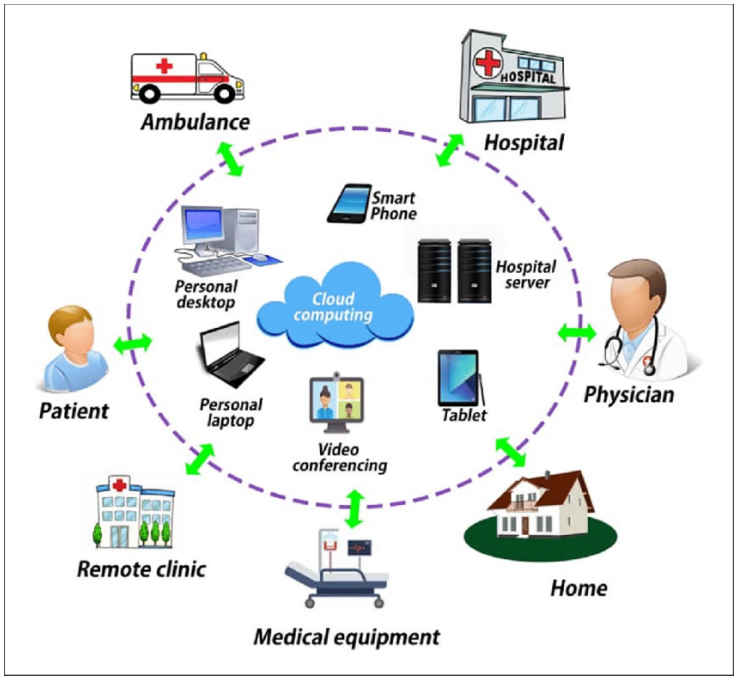


FIGURE 1.12 – Télémédecine [9]

1.7.2 Agriculture

Dans le secteur de l'agriculture, l'IoT et la technologie de manière globale permettent de transformer l'activité agricole de nombreuses manières. Tout d'abord, l'intégration de l'IoT dans le secteur agricole permettra de mieux faire face à l'accroissement de la population. L'IoT agriculture permet également de proposer des produits de meilleure qualité et plus soucieux de l'environnement. Le but étant de mieux répondre à la demande de consommateurs toujours plus exigeants et responsables. En outre, l'IoT agriculture permet aux agriculteurs de réduire les déchets. Ensuite, l'IoT agriculture permet une meilleure utilisation des ressources telles que l'eau ou encore l'électricité.[64]

Grâce à l'Internet des choses, l'agriculture est devenue plus confortable et plus facile à gérer dans son processus de production modeste.[63]

FIGURE 1.13 – L'agriculture [9]

1.7.3 La domotique

La domotique regroupe l'ensemble des technologies permettant l'automatisation des équipements d'un habitat. Elle vise à apporter des fonctions de 14 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. [65]

Elle est le plus secteur utilisé des objets connectés, une étude prévoit un 200 augmentation du nombre d'objets connectés à la fin d'offre plus de sécurité, de confort et de contrôle de la consommation d'énergie. [63]

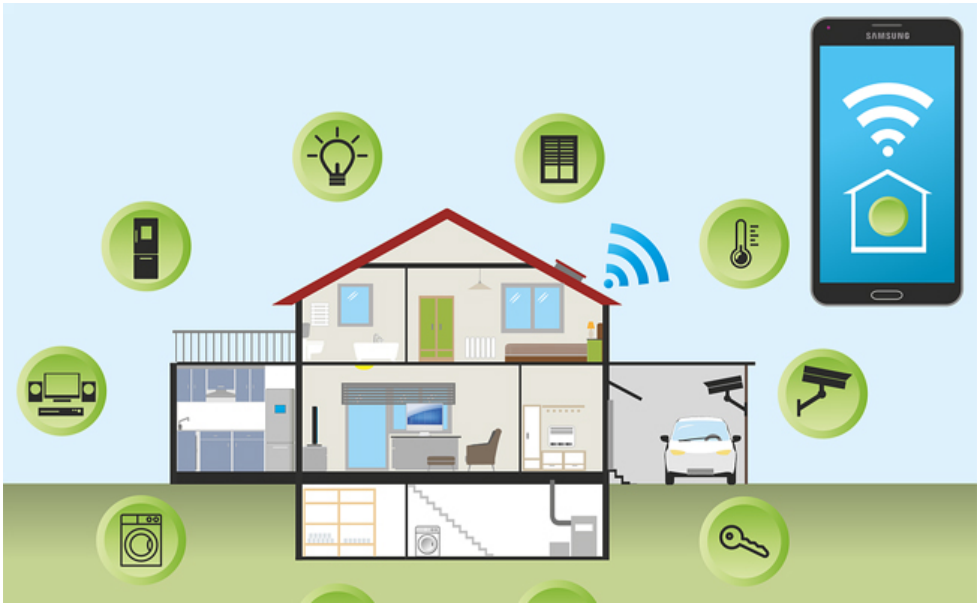


FIGURE 1.14 – La domotique [10]

1.7.4 Smart cities

Smart city est une ville utilisant les technologies de l'information et de la communication (TIC) pour améliorer la qualité des services urbains ou réduire leurs coûts. D'autres termes ont été utilisés pour des concepts similaires : ville connectée, cyber ville, ville numérique, communautés électroniques.

La ville intelligente repose souvent sur des outils numériques permettant une amélioration de la qualité de vie des citoyens. En effet, la technologie est utilisée au service d'un développement intelligent de la zone urbaine aussi bien au niveau de la mobilité que de l'environnement, de la participation citoyenne, etc. Il peut donc sembler évident que la ville intelligente découle souvent de la ville numérique pour une meilleure gestion urbaine.

[66]

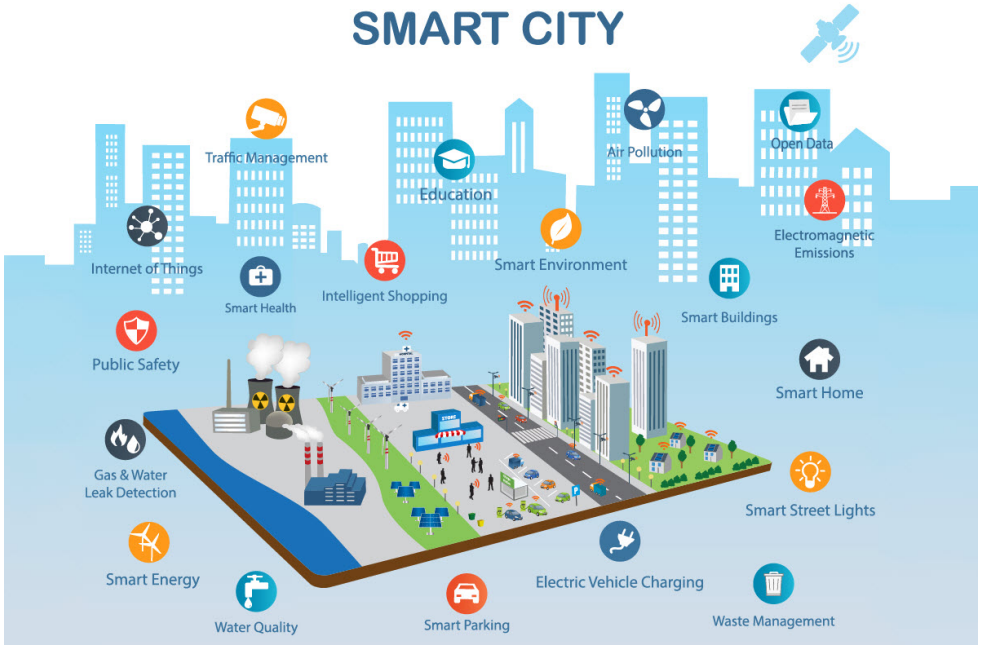


FIGURE 1.15 – Ville intelligente [11]

1.7.5 L'industrie

Système de fabrication intelligent est un système intégré et collaboratif qui répond aux demandes et aux conditions variantes de l'usine, du système d'approvisionnement et des besoins des clients, et cela en temps réel.

Cela est possible grâce aux capteurs et actionneurs sophistiqués et connectés en utilisant les technologies IoT dans le processus de fabrication mais également au pilotage des activités de production par l'intelligence artificielle. De plus, avec l'interconnexion aux systèmes extérieurs, les machines peuvent directement entamer la production dès qu'un bon de commande est reçu par l'usine.[12]

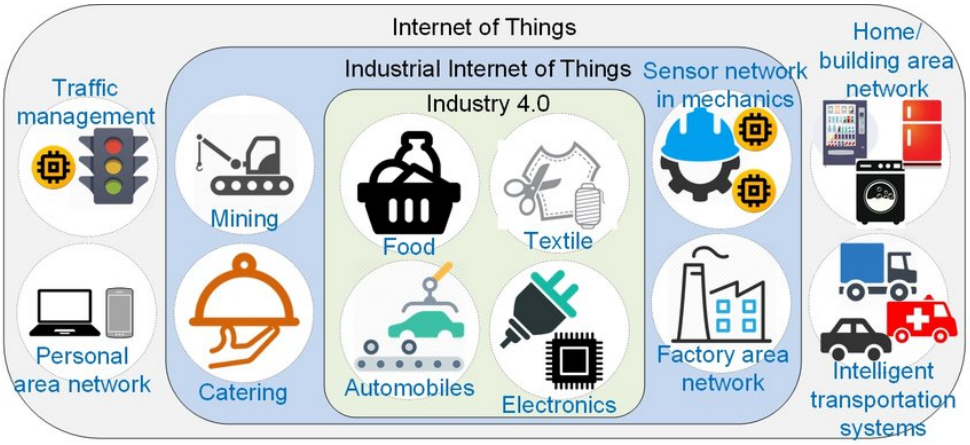


FIGURE 1.16 – L'industrie [12]

1.8 Les avantages et les inconvénients d'IOT

Les avantages et désavantages de l'IoT sont multiple, on peut citer les avantages et désavantages suivants :

1.8.1 Les avantages de l'IoT

Les avantages des IoT sont nombreux, nous pouvons citer les avantages suivants : [67] [68]

- Accès ubiquitaire à l'information pour un monde plus intelligent et un mode vie sophistiqué et confortable.
- Economiser du temps.
- L'éclairage intelligent.
- Possibilité d'exploitation des ressources géantes de l'Internet pour le stockage et le traitement des données écoulées de l'IoT.
- Il peut aider à contrôler plus intelligemment les maisons et les villes via les téléphones portables.
- Améliorer les services traditionnels généraux comme le transport et les parkings.
- Le gain du temps est un autre avantage de l'IoT.

— Les déplacements inutiles sont dès lors remplacés par une simple navigation sur le web pour commander des produits, contrôler l'état des objets et/ou endroits connectés.

1.8.2 Les inconvénients de l'IoT

On cite ci-dessous les inconvénients les plus marquants.

— **La sécurité :** La sécurité IoT peut être regroupée en deux grands classes. La première classe comprend les tâches de sécurité requises. Comme d'habitude, les principales difficultés potentielles sont liées, mais exigences contradictoires de différentes tâches. Par exemple, la force de l'authentification et la confiance sont en contradiction directe avec un critère de vie privée. La deuxième classe de la sécurité est liés aux paramètres de conception tels que le coût, la taille, la latence et, en besoins énergétiques. Comme d'habitude, l'impact ces exigences sont qu'ils limitent grandement acceptable solutions de sécurité.[69]

Les pirates peuvent accéder au système et voler des informations personnelles. Comme nous ajoutons un si grand nombre de dispositifs à Internet, il y a un risque que nos renseignements soient mal utilisés.

Application : pour la validation des chaînes d'entrée, mots de passe par défaut, etc.

Réseau : Pare-feu, cryptage incorrect des communications, manque de mises à jour automatiques.

Mobile : API non sécurisée, manque de cryptage, manque de sécurité de stockage.

Cloud – Authentification incorrecte, pas de cryptage pour le stockage et la communication.[2]

— **La complexité :** Avec la complexité des systèmes, il existe de nombreuses façons pour eux d'échouer.

— **L'hétérogénéité :** L'IoT est caractérisé par des objets et des systèmes basés sur des configurations matérielles, logicielles et réseaux très différentes. Ces objets peuvent interagir entre eux ou avec différentes plate-formes de services et cela via des réseaux hétérogènes.Les applications et les systèmes à implanter devront tenir compte de cette hétérogénéité. de l'IoT, c'est-à-dire permettre l'interopérabilité

et la modularité.[3]

— **La flexibilité :** Beaucoup s'inquiètent de la flexibilité d'un système IoT pour s'intégrer facilement à un autre. Ils s'inquiètent de se retrouver avec plusieurs systèmes conflictuels ou verrouillés.

— **L'énergie :** Mesurer la consommation d'énergie et surtout déceler une consommation anormale, gérer les installations, ou effectuer de la maintenance à distance.

— Il est très difficile de planifier, de créer, de gérer et d'activer une large technologie dans le cadre de l'IoT.

1.9 Le Cloud computing et le Fog computing

Aujourd'hui, la quantité des données générées chaque seconde est inimaginable. Face à ce grand volume de données, le Cloud Computing (CC) émerge comme la nouvelle forme de stockage et de gestion des données.

1.9.1 Définition du cloud computing

Un cloud "nuage" est un ensemble de matériels, de raccordements réseau et de logiciels fournissant des services qu'individus et collectivités peuvent exploiter depuis n'importe où dans le monde.[70]

Cloud computing devient l'une des prochaines industries des IT mots à la mode : les utilisateurs déplacent leurs données et applications vers le (Cloud) distant, puis y accéder dans un et omniprésente. Il s'agit encore une fois d'une utilisation de traitement central. Un scénario similaire s'est produit il y a environ 50 ans : un serveur informatique de partage de temps servi plusieurs utilisateurs. Jusqu'à il y a 20 ans quand les ordinateurs personnels sont venus à nous, les données et les programmes étaient principalement situés dans les ressources locales. Certainement actuellement, le paradigme Cloud computing n'est pas une récurrence de l'histoire. Il y a 50 ans, ils ont dû adopter les serveurs de partage de temps en raison des ressources informatiques limitées.[71]

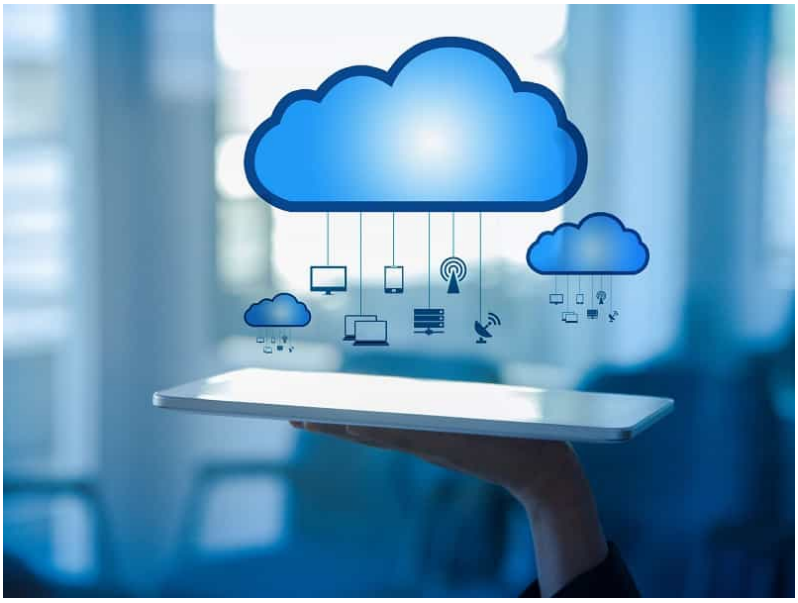


FIGURE 1.17 – Système du Cloud Computing[70]

1.9.2 Les services du Cloud Computing

Le CC a trois modèles de services : software as a service SaaS, plateforme as a service PaaS, infrastructure as a service IaaS comme les montre la figure suivante :

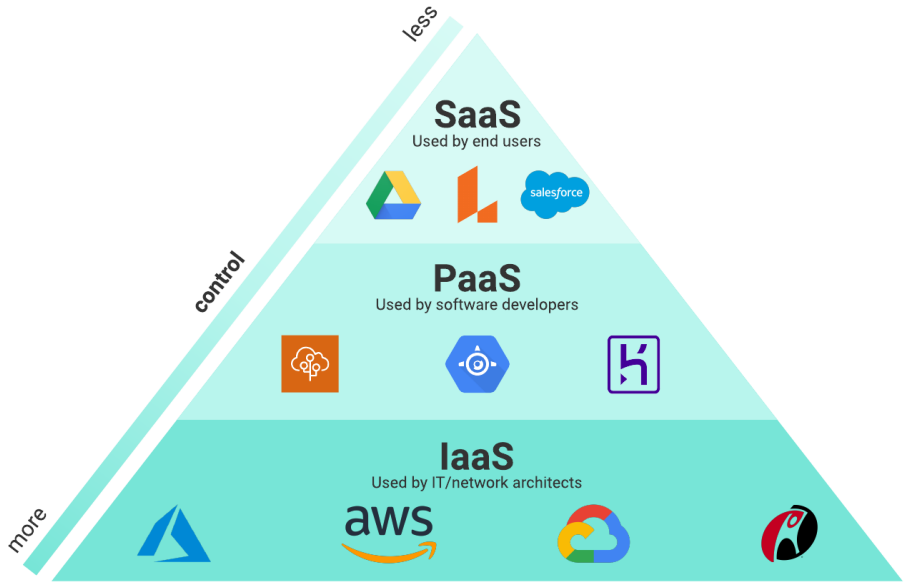


FIGURE 1.18 – Services du cloud computing

— **Saas ((Software as a Service)) :**

Le Software as a Service ou Logiciel en tant que Service en Français, est un modèle de distribution de logiciel au sein duquel un fournisseur tiers héberge les applications et les rend disponibles pour ses clients par l'intermédiaire d'internet. Grâce à un logiciel SaaS, les entreprises n'ont plus besoin d'installer et de lancer des applications sur leurs propres ordinateurs ou sur leurs Data Centers. Le coût d'acquisition de matériel est ainsi éliminé, au même titre que les coûts d'approvisionnement et de maintenance, de licence de logiciel, d'installation et de support. On compte également plusieurs autres avantages.[72]

— **Paas (plate-forme as a Service)**

La plate-forme as a Service est une couche inférieur au SaaS, dans le sens où elle consiste pour le fournisseur a offrir un environnement de développement et une infrastructure de déploiement : elle se compose donc d'un environnement d'exécution lié au langage de programmation (Ruby, Java, Python, .Net, etc.), de librairies de programmation, des outils de test et de monitoring, de bases de données, etc. Le consommateur développe ainsi ses applications en s'appuyant sur ces outils spécifiques du fournisseur. Il s'agit donc pour le fournisseur de mettre un middleware a disposition du consommateur. Ce middleware va assurer la gestion automatique de l'infrastructure et faciliter le déploiement des applications. Le principal désavantage du PaaS est que chaque fournisseur apporte sa propre technologie, avec un langage de développement de prédilection : Python et Java pour Google, .Net pour Windows Azure, Ruby pour Heroku, etc. Cette absence de standardisation des fonctions et API emprisonne le développeur dans la plateforme choisie et il devient donc dépendant du gestionnaire.[73]

— **IaaS (Infrastructure as a Service)**

Le service IaaS c'est la mise à disposition des ressources virtualisées, comme les machines virtuelles, des systèmes de stockage, des systèmes d'exploitation et d'autres ressources informatiques fondamentales. L'entreprise pourra par exemple louer des serveurs Linux, Windows ou autres systèmes, qui tourneront en fait dans une machine virtuelle chez le fournisseur de l'IaaS.[73]

1.9.3 Les types de cloud

Il existe 4 principaux types de nuages qu'une organisation peut utiliser.

Public Cloud

Les clouds publics sont gérés par des tiers qui fournissent des services cloud au public via Internet, ces services sont disponibles sous forme de modèles de facturation à la carte. Ils offrent des solutions pour minimiser les coûts de l'infrastructure informatique et deviennent une bonne option pour gérer les charges de pointe sur l'infrastructure locale. Les nuages publics sont l'option de prédilection pour les petites entreprises, qui sont en mesure de démarrer leur entreprise sans grands investissements initiaux en s'appuyant entièrement sur l'infrastructure publique pour leurs besoins informatiques. [74]

Private cloud

Tout comme le cloud public, les nuages privés permettent aux utilisateurs d'accéder, d'utiliser et de mettre en cache des données dans le nuage à distance. Cependant, l'infrastructure de cloud privé est généralement protégée par un pare-feu, qui est un système de sécurité qui suit et contrôle le trafic réseau. Cela signifie que seules les personnes autorisées peuvent utiliser ces ressources informatiques. Les entreprises qui ont des normes réglementaires strictes préféreront les clouds privés pour protéger leurs informations et données. [75]

Hybrid cloud

Les nuages hybrides sont une combinaison de nuages publics et privés. Ils sont conçus pour permettre aux gens d'utiliser et de stocker des données sur les deux plateformes en toute transparence.[76]

Community cloud

Un nuage communautaire est un nuage privé qui fonctionne de la même façon qu'un nuage public. Ils sont collaboratifs et permettent à divers organismes autorisés de partager

et de travailler sur les mêmes applications. En général, les entreprises qui appartiennent à la même industrie, mais qui ont des préoccupations communes en matière de sécurité ou de conformité, utiliseront les nuages communautaires. Par exemple, les fournisseurs de soins de santé et les organismes gouvernementaux mettent souvent en place des nuages communautaires dans leurs opérations.[77]

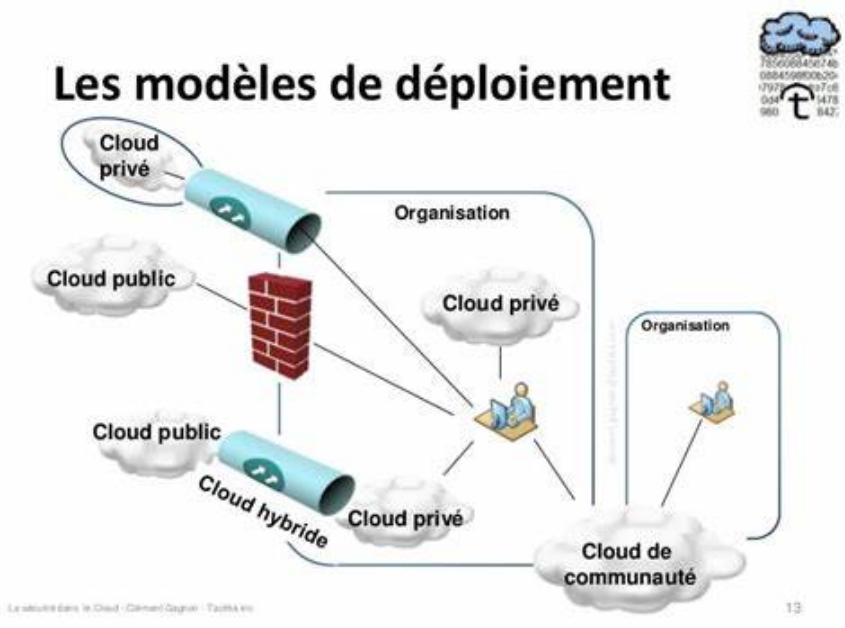


FIGURE 1.19 – Types du cloud computing [13]

1.9.4 Les avantages et les inconvénients du cloud

Les avantages et les inconvénients du cloud sont : [78]

Les avantages

— Souplesse.
— Recentrage sur le cœur de métier.
— L'avantage principal du cloud, c'est l'espace de stockage.
— Mobilité.

Les inconvénients

— Confidentialité et sécurité des données : les données sont hébergées en dehors de l'entreprise. Le fournisseur proposant le service héberge les données de l'entreprise utilisatrice.
— Le problème avec le cloud, c'est l'accès aux données.
— Dépendance : si l'entreprise souhaite des fonctionnalités très spécifiques, il peut être difficile de convaincre le fournisseur de proposer ces fonctionnalités.
— La nécessité d'avoir une connexion internet performante.

1.9.5 Qu'est-ce que le fog computing ?

Fog Computing est le terme inventé par Cisco qui se réfère à l'extension de l'informatique en nuage à un bord du réseau de l'entreprise. Ainsi, il est également connu comme Edge Computing ou Fogging. Il facilite le fonctionnement des services de calcul, de stockage et de mise en réseau entre les appareils finaux et les centres de données informatiques. [79]

— Les dispositifs qui composent l'infrastructure de brouillard sont appelés nœuds de brouillard.
— Dans le brouillard informatique, toutes les capacités de stockage, les capacités de calcul, les données ainsi que les applications sont placées entre le nuage et l'hôte physique.
— Toutes ces fonctionnalités sont placées plus vers l'hôte. Cela rend le traitement plus rapide car il est fait presque à l'endroit où les données sont créées.
— Il améliore l'efficacité du système et est également utilisé pour assurer une sécurité accrue.

1.9.6 Quels sont les avantages et les inconvénients du fog computing ?

Les principaux avantages du fog Computing se résument à l'augmentation de l'efficacité des ressources informatiques et de la structure informatique d'une organisation.[80]

— Le temps de latence.

— Conservation de la bande passante du réseau.

— Accès en ligne et hors ligne.

— Analyse en temps réel.

— Il offre une meilleure protection de la vie privée, car les industries peuvent analyser leurs données localement.

Les principaux inconvénients du fog Computing sont : [80]

— Emplacement physique dangereux.

— Complexité du réseau.

— Risques liés à la sécurité et à la vie privée

1.9.7 Différence entre le cloud computing et le Fog Computing

Le tableau suivant représente la différence entre le cloud computing et le Fog Computing :

Caractéristique	Cloud computing	Fog Computing
Latence	Le cloud computing a une latence élevée par rapport au fog computing	Le Fog Computing a une faible latence
Capacité	Le Cloud Computing n'offre aucune réduction des données lors de l'envoi ou de la transformation des données	Le Fog Computing réduit la quantité de données envoyées au cloud computing
Réactivité	Le temps de réponse du système est faible	Le temps de réponse du système est élevé
Sécurité	Le cloud computing a moins de sécurité que le Fog Computing	L'informatique de brouillard a une sécurité élevée
La vitesse	La vitesse d'accès est élevée en fonction de la connectivité de la VM	Élevé encore plus par rapport au Cloud Computing

Mode de communication	Réseau IP	Communication sans fil : WLAN, WiFi, 3G, 4G, ZigBee, etc. ou communication filaire (faisant partie des réseaux IP)
Environnement de travail	Bâtiment de datacenter spécifique avec systèmes de climatisation	Extérieur (rues, stations de base, etc.) ou intérieur (maisons, cafés, etc.)

TABLE 1.2 – Différence entre le cloud computing et le Fog Computing [15]

1.9.8 Examples du cloud

Pythonanywhere

Pythonanywhere est un cloud PAAS, ce que cela signifie est que vous pouvez simplement vous soucier de codage et laisser le mal de tête de l'hébergement, plate-forme, DB et PAAS considérations sur pythonanywhere.[81]

Dropbox

Dropbox est un service de stockage et de partage de copies de fichiers locaux en ligne proposé par Dropbox, Inc., entreprise localisée à San Francisco, en Californie. Dropbox fournit des logiciels client pour Microsoft [82]

1.10 Conclusion

L'internet des objets, le Cloud Computing et le fog computing sont des domaines technologiques complexes et sont largement utilisés au cours des dernières années en raison des solutions qu'ils offrent.

Ce chapitre comprend deux sections. La première portait sur la présentation de l'IoT, son fonctionnement, ses principaux technologies ainsi que les cas d'applications et ses

avantages, les inconvénients. Enfin, nous avons mis dans la deuxième partie l'accent sur les notions Cloud computing et le fog computing, ses définitions, ses services et types de déploiement ainsi que ses avantages et inconvénients.

Dans le chapitre suivant nous allons présenter la sécurité d'information.

Chapitre 2

Sécurité de l'information

2.1 Introduction

L'Internet des objets, le Cloud Computing et le fog computing ont été abordés dans le chapitre précèdent. À l'heure actuelle, ces déploiements sont de plus en plus remarquables en raison de leurs nombreux avantages et possibilités. Bien que ces progrès technologiques soient populaires, l'aspect sécuritaire demeure un sujet de préoccupation et peut ralentir les adoptions.

À travers ce chapitre, nous allons jeter un œil sur la sécurité de l'information. Nous allons présenter quelques notions de base de sécurité, les différents mécanismes de sécurité (la cryptographie symétrique et asymétrique), signature numérique, la fonction de hachage. Ensuite, nous allons parler sur quelques protocoles de sécurité. Enfin, nous présenterons les services de sécurité et nous allons terminer ce chapitre par le standard JWT.

2.2 Notions de base de la sécurité

La sécurité informatique comme étant le fait d'assurer le bon fonctionnement d'un système et de garantir les résultats attendus de sa conception.

Autrement dit, la sécurité représente l'ensemble des techniques et pratiques adoptées pour prévenir et surveiller l'accès non autorisé, diminuer la vulnérabilité d'un système contre les menaces de tous les formes (accidentelles ou intentionnelles), l'utilisation abusive,

la modification ou le refus d'une opération informatique.[83]

À partir de cette définition, on peut extraire les bases de la sécurité qui sont décrites dans ce qui suit :

— Menace : le système d'information doit être sécurisé par rapport violation potentielle d'une propriété de sécurité.[84]

— Risque : le risque est la possibilité qu'une chose critique apparaisse comme menace. Son évaluation permet d'établir des actions pour réduire et maintenir la menace à un niveau raisonnable et acceptable. [85]

— Attaque : elle consiste les moyens d'exploiter une vulnérabilité. Il peut y avoir un ou plusieurs attaques pour une même vulnérabilité mais pas toutes les vulnérabilités sont exploitables.

— Vulnérabilité : toutes les faiblesses système d'information qui peuvent être exploitées par des menaces.[86]

2.3 Les mécanismes de sécurité

Les mécanismes de sécurité ont pour but de protéger l'accès aux actifs d'un système (c.-à-d. les données et les ressources) contre les menaces à la sécurité. Ainsi, l'utilisation des mécanismes de sécurité permet de mettre en œuvre les les services de sécurité visant à empêcher la divulgation non autorisée et/ou la modification des données et/ou l'accès non autorisé aux ressources. Nous allons mentionner ci-dessous quelques mécanismes de sécurité : les différents types de fonctions de hachage et de cryptage (symétriques ou non) et les algorithmes de signature.

2.3.1 Cryptographie

la cryptographie est la science qui s'intéresse à l'étude des principes, des méthodes et des techniques qui s'attellent aux aspects de sécurité de l'information tels que la confidentialité, l'intégrité des données et l'authentification des intervenants, en utilisant des clés de cryptage.

La cryptographie permet de stocker les informations sensibles ou de les faire véhiculer

à travers un réseau peu sûr de façon qu'elles ne soient intelligibles qu'à un destinataire autorisé. Elle fournit des outils puissants minimisant les risques et susceptibles de rendre l'échange de l'information confidentielle, non falsifiable, authentique et non-altérable. [87]

Cryptographie symétrique

la cryptographie symétrique est aussi appelée chiffrement à clé privée ou à clé secrète, dans ce cas utilisé la même clé pour le chiffrement et le déchiffrement.

Les avantages des algorithmes symétriques sont plus rapides et facilement implémentés sur le matériel ainsi qu'ils sont des simples opérations de substitution et de transposition et mieux adaptés pour une utilisation sur le réseau internet filaire et en mobilité. Mais ces systèmes ne sont pas entièrement adéquats pour résoudre tous les problèmes de sécurité.[88] Ils sont basés sur deux grandes familles dans cette classe :

— Chiffrement par flux (Stream ciphers).

— Chiffrement par bloc (Block ciphers).

Divers algorithmes ont été développé jusqu'à présent pour décrire la cryptographie à clé symétrique. Ce sont AES, DES, 3DES, Blowfish, RC4.

— **DES** : DES a été approuvé en 1978 par le NBS. Le DES fut normalisé par l'ANSI (American National Standard Institute) sous le nom de ANSI X3.92, plus connu sous la dénomination DEA (Data Encryption Algorithm). L'algorithme de DES consiste à effectuer des combinaisons, des substitutions et des permutations entre le texte à chiffrer et la clé (KEY), en faisant en sorte que les opérations puissent se faire dans les deux sens (pour le déchiffrement). La combinaison entre substitutions et permutations est appelée code produit. La clé est codée sur 64 bits et formée de 16 blocs de 4 bits, généralement notés k1 à k16.[89]

— **AES** : Advanced Encryption Standard est un algorithme de cryptage symétrique qui a été choisi en 2001 par le National Institute of Standards and Technol ogy (NIST) comme norme fédérale de traitement de l'information FIPS-197. Elle fonctionne sur des blocs de données, l'État, qui ont une taille fixe de 128 bits. L'état est organisé comme une matrice de quatre lignes et quatre colonnes d'octets. Les longueurs de clés définies sont 128 bits, 192 bits ou 256 bits. [90] Comme la plupart

des chiffrements symétriques, AES crypte une entrée bloc en appliquant la même fonction ronde. Les itérations de la fonction de dix tours modifier l'état en appliquant des transformations non linéaires, linéaires et dépendantes de la clé. Chacune transforme l'état 128 bits en état 128 bits modifié.

— **Blowfish** : Bruce Schneier, l'un des plus grands cryptologues du monde, a conçu l'algorithme Blowfish et l'a rendu disponible en domaine public. Blowfish est une clé de longueur variable, 64 bits chiffrement de bloc. [91] Il se compose de deux parties : partie clé-expansion et une partie de cryptage de données. expansion de clé convertit une clé d'au plus 448 bits en plusieurs tableaux de sous-clés totalisant 4168 octets. Le chiffrement des (couramment) réseau. Chaque tour est composé d'un permutation, et une substitution dépendante de la clé et des données. Tous opérations sont des XORs et des ajouts sur des mots de 32 bits. Le seul opérations supplémentaires sont quatre recherches de données de tableaux indexées par rond. [92]

— **RC4** : il a été conçu en 1987 par Ron Rivest ; il fait partie des algorithmes dits de chiffrement en continu. Il est basé sur des permutations aléatoires, avec des opérations sur des octets. L'algorithme a une longueur de clé variable (de 1 à 256 octets).[93]

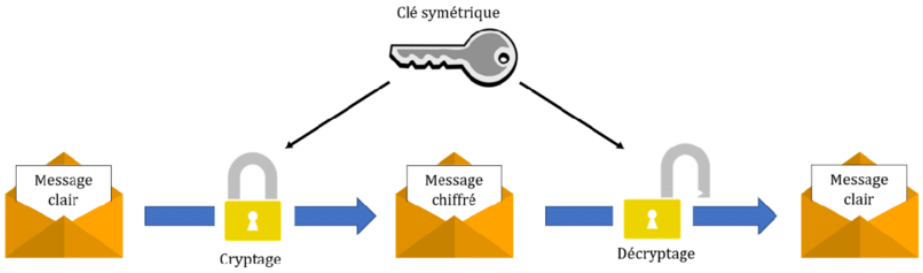


FIGURE 2.1 – Cryptographie symétrique[94]

Cryptographie asymétrique

La cryptographie à clé asymétrique est appelée cryptographie à clé publique. Dans cette technique, la l'expéditeur utilise une clé publique du destinataire pour cryptage et

le récepteur utilise sa clé privée pour décrypter le message. Le concept d'auto-certification est absent ici au lieu de signatures numériques sont utilisés pour certifier les clés. Cette méthode est plus pratique et fournit une meilleure authentification que la la vie privée reste intacte.[95]

Il y a divers algorithmes pour implémenter ce chiffrement. Ce sont RSA, Diffie-Hellman, ECC et algorithme de signature numérique.

— **RSA** Rivest-Shamir-Adleman (RSA) RSA est un algorithme à clé publique largement utilisé. RSA d'abord décrit en 1977. [96]

RSA est basé sur le principe que certains mathématiques opérations sont plus faciles à faire dans une direction, mais l'inverse est très difficile sans quelques informations supplémentaires. En cas de RSA, l'idée est qu'il est relativement facile de multiplier, mais beaucoup plus difficile à prendre en compte. Multiplication peut être calculé en temps polynôme où comme factoring temps peut croître exponentiellement proportionnel à la taille de le nombre. [97]

Principe de fonctionnement : Si Bob souhaite recevoir des messages en utilisant le RSA, il procède de la façon suivante :[98][99]

— **Création des clés** :

Bob crée 4 nombres p,q, e et d : p et q sont deux grands nombres premiers distincts. Leur génération se fait au hasard, en utilisant un algorithme de test de primalité probabiliste. e est un entier premier avec le produit (p-1)(q-1). d est tel que ed=1 modulo (p-1)(q-1). On peut fabriquer d à partir de e, p et q, en utilisant l'algorithme d'Euclide.

— **Distribution des clés** :

Le couple (n,e) constitue la clé publique de Bob. Il la rend disponible par exemple en la mettant dans un annuaire. Le couple (n,d) constitue sa clé privée. Il la garde secrète.

— **Envoi du message codé** :

Alice veut envoyer un message codé à Bob. Elle le représente sous la forme d'un ou plusieurs entiers M compris entre 0 et n-1. Alice possède la clé publique (n,e)

de Bob. Elle calcule

$$C = M^e mod n \tag{2.1}$$

C'est ce dernier nombre qu'elle envoie à Bob.

— **Réception du message codé** :

Bob reçoit C, et il calcule grâce à sa clé privée

$$D = C^d mod n \tag{2.2}$$

. D'après un théorème du mathématicien Euler,

$$D = (M^d)^e = M(mod n) \tag{2.3}$$

Il a donc reconstitué le message initial.

— **Diffie-Hellman** L'algorithme Diffie-hellman a été la première clé publique algorithme construit par Witfield Diffie et Martin Hellman en 1976. Diffie-hellman est utilisé pour la clé algorithme d'échange. Il est construit sous insécurité canal de connexion. Diffie-hellman se compose de deux clés : une clé privée et une clé secrète. Supposons que l'expéditeur veut définir établir une connexion avec le récepteur il crypte le message avec sa propre clé privée et la clé publique de l'expéditeur. Une fois récepteur reçoit le message il décrypte le message avec son propre clé privée et clé publique de l'expéditeur.[100]

— **ECC** (Elliptic curve cryptography) est un mécanisme notable dans le domaine de la clé publique cryptographie. ECC fait usage de courbes elliptiques dans lequel le variables et coefficients sont des éléments d'un champ fini. Par rapport aux autres algorithmes (comme RSA et DES), il fournit le même niveau de sécurité avec une taille de clé plus petite. Par conséquent, il aide à réduire le stockage et la transmission exigences.[101]

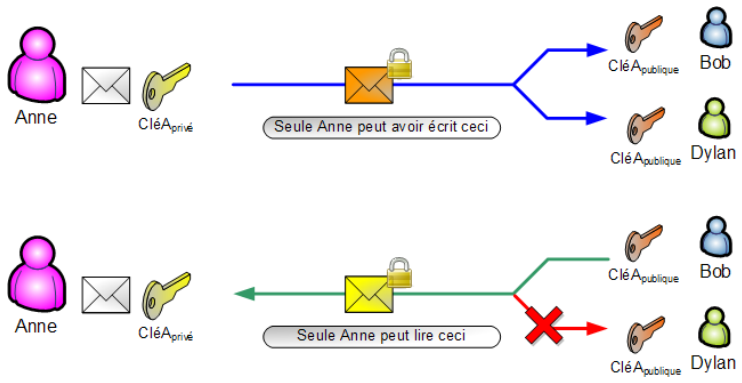


FIGURE 2.2 – Cryptographie asymétrique[95]

2.3.2 Fonction de hachage

La fonction de hachage devient un protocole incontournable dans le domaine de la cryptographie. Son objectif principal est de générer une valeur de hachage de taille fixe et réduite, à partir d'un message donnée de taille variable. Ce protocole permet offrent plusieurs fonctions cryptographiques, dont la génération rapide des signatures numériques.[102]

Les fonctions de hachage compressent une chaîne de longueur arbitraire à une longueur fixe. Ils fournissent un unique relation entre l'entrée et la valeur de hachage et donc remplacer l'authenticité d'une grande quantité de information (message) par l'authenticité d'un beaucoup valeur de hachage plus petite (authentifiant).[103]

L'un des buts d'une empreinte est de représenter des données de façon certaine tout en réduisant la taille utile qui sera réellement chiffrée. Cela empêche d'utiliser de vastes quantités de données.

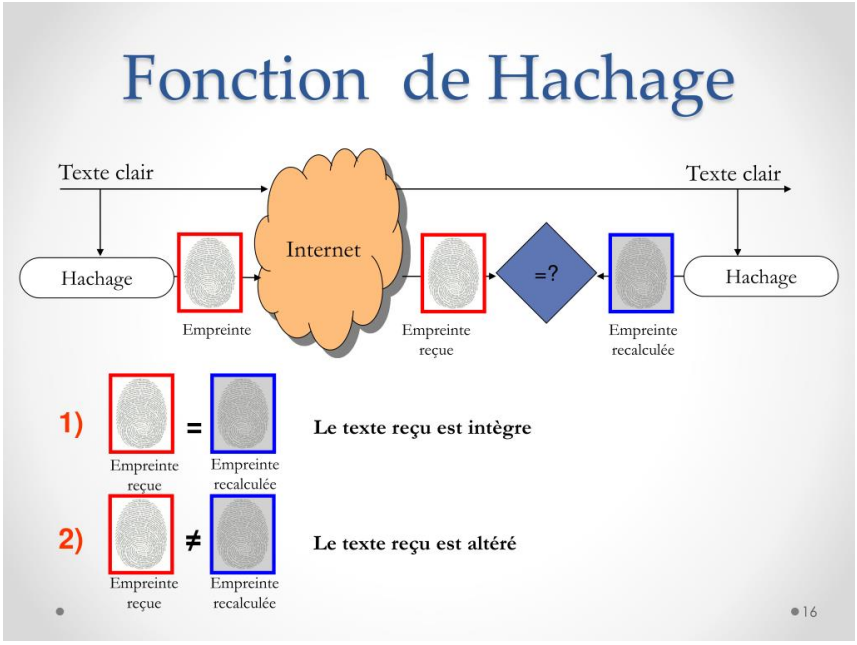


FIGURE 2.3 – fonctions de hachage[102]

Les fonctions de hachage cryptographique sont généralement classées en deux classes :

Hachage non codé fonctions également connues sous le nom de code de détection de manipulation (MDC) ou d'authentification de message Code (MAC) avec un seul paramètre et un message d'entrée.Fonctions de hachage avec deux entrées distinctes un message de saisie et une clé secrète. Généralement, le terme fonctions de hachage désigne les fonctions de hachage non clé.[104]

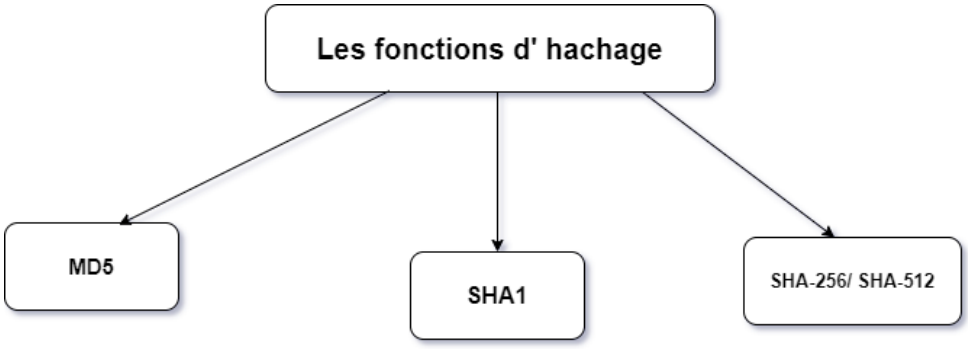


FIGURE 2.4 – fonctions de hachage[102]

— **MD5** est un algorithme de condensé de messages développé par Ron Rivest au MIT. C'est essentiellement une version sécurisée de son algorithme précédent, MD4

qui est un peu plus rapide que MD5. C'est le hash sécurisé le plus utilisé algorithme en particulier dans le message Internet-standard authentification.

L'algorithme prend comme entrée un message de longueur arbitraire et produit en sortie un 128 bits digérer le message de l'entrée. Ceci est principalement destiné à les demandes de signature numérique où un fichier volumineux doit être comprimé de manière sécurisée avant d'être chiffré avec une clé privée (secrète) sous une clé publique.[105]

— **SHA-1** est utilisé pour calculer un condensé de message pour un message ou un fichier de données qui est fourni comme entrée. Le message ou le fichier de données devrait être considéré comme une chaîne de bits. La longueur du message est le nombre de bits dans le message (le message vide a la longueur 0). Si le nombre de bits dans un message est un multiple de 8, nous pouvons représenter le message en hexadécimal. Le but du remplissage du message est de faire de la longueur totale d'un message rembourré un multiple de 512.[106]

— **SHA-256/ SHA-512** une nouvelle génération de SHA fonctions avec des tailles de message digest beaucoup plus grandes, à savoir 256 et 512 bits, appelés SHA-256 et SHA-512, a été introduit en 2000 et adopté comme norme FIPS en 2002. La principale motivation pour introduire ces nouvelles fonctions de hachage standard était de fournir des fonctions de hachage avec des niveaux de sécurité contre les attaques de recherche de collision compatibles avec les niveaux de sécurité attendus des trois tailles de clés standard pour la nouvelle sélection avancée Norme de chiffrement (128, 192 et 256 bits).

Beaucoup plus sûr mais encore moins rependus, ils créent respectivement des empreintes de 256 et 512 bits. Il est aujourd'hui fortement conseillé d'évoluer sur ces fonctions de hachage.[107]

Algorithme		Output [bit]	Input [bit]	Nombre de rondes	Collisions trouvées
MD5		128	512	64	Oui
SHA-1		160	512	80	tentatives
SHA-2	SHA-224	224	512	64	Non
	SHA-256	256	512	64	Non
	SHA-384	384	1024	80	Non
	SHA-512	512	1024	80	Non

FIGURE 2.5 – Les fonctions de hachage[108]

2.3.3 La signature

La signature est un mécanisme permettant d'authentifier l'auteur d'un document électronique, c'est-à-dire de garantir l'identité du signataire. Les signatures électroniques ont le même rôle qu'une signature classique sur papier : elles permettent d'authentifier l'auteur du document signé et de vérifier l'intégrité du message. De plus, contrairement à d'autres méthodes d'authentification offertes par la cryptographie symétrique comme les MAC, une autre propriété très importante, dans le domaine du commerce par exemple, est offerte avec ce type de schéma : la non-répudiation, c'est-à-dire l'impossibilité de remettre en cause la signature.[109]

Tous les algorithmes asymétriques peuvent l'utiliser non seulement pour le chiffrement, mais aussi pour signer un message. Les algorithmes de signature ne peuvent pas signer que des messages de taille fixe qui soient inférieurs à la taille de clé.

Le principe de la signature numérique est le suivant. Supposons qu'Obélix veut signer un document m. Il utilise sa clé privée pour créer la signature S, mais comme indiqué dans la section de hachage, il ne signe pas directement le document mais plutôt son empreinte, c'est-à-dire l'hach du document. En utilisant la clé publique correspondante, Astérix peut vérifier l'authentification de la signature. La signature numérique comprend trois algorithmes dont :[110]

— Un algorithme de génération de clé Kg = (pk, sk) à partir du paramètre de sécurité, il produit une paire de clés : clé privée et clé publique.

— Un algorithme de création de signature S(sk, m) = n

— Un algorithme de vérification de signature V (pk, n) = m

La figure ci dessus présente la signature numérique sans hachage :

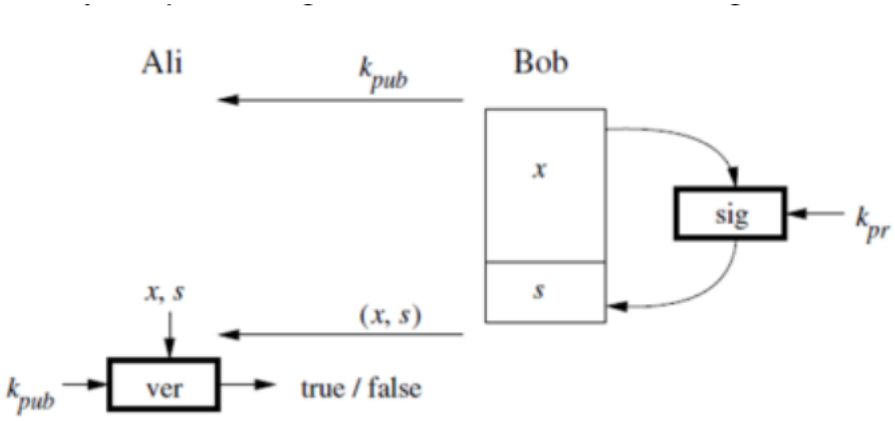


FIGURE 2.6 – La signature numérique sans hachage[108]

La figure ci dessus présente la signature numérique avec hachage :

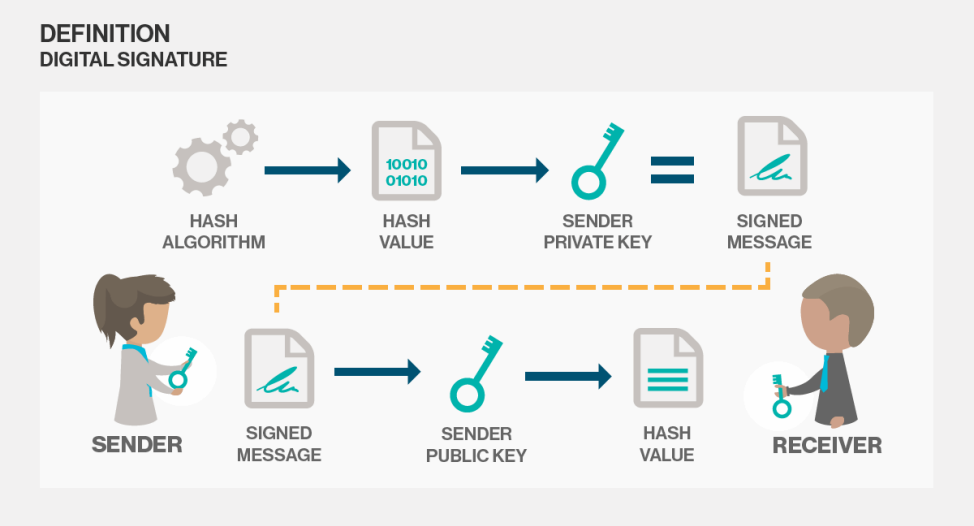


FIGURE 2.7 – La signature numérique avec hachage[110]

Signature RSA

Dans RSA, d est privé ; e et n sont publics Comme son nom l'indique, la clé publique est donnée à tout le monde et la clé privée reste privée, M est le message.

— Alice crée sa signature numérique en utilisant

$$S = M^d \mod n \tag{2.4}$$

.

— Alice envoie le message M et la signature S à Bob.

— Bob calcule

$$M1 = S^e \mod n. \tag{2.5}$$

— Si M1 = M alors Bob accepte les données envoyées par Alice.

Signature ELGAMAL

Le cryptosystème Elgamal nécessite une exponentiation modulaire opération. La force de sécurité du chiffre est une fonction des tailles du module ; il est basé sur le discret logarithme. [111] Ce processus de signature vise à signer un message pour assurer l'autorisation et l'intégrité du message. Cette section comporte deux processus : Processus de génération de signature Processus de vérification de la signature.[112]

L'algorithme de signature utilise une paire de clés comprenant une clé publique et une clé privée. La clé privée est utilisée pour générer une signature numérique pour un message, et une telle signature peut être vérifiée en utilisant la clé publique correspondante du signataire. La signature numérique permet l'authentification du message (le destinataire peut vérifier l'origine du message), l'intégrité (le destinataire peut vérifier que le message n'a pas été modifié depuis qu'il a été signé) et la non-répudiation. (l'expéditeur ne peut prétendre faussement qu'il n'a pas signé le message).[113]

Un message est signé comme suit :

— Choisissez un grand nombre entier P au hasard.

— alpha=primitive root

— a= un entier privée(1< a <= p-2)

— Calculer

$$beta = alpha^a \mod p. \tag{2.6}$$

— Public : P,alpha,beta

— Envoyer : (m=message,r,s), k est un nombre aléatoire, privée

— Calculer

$$R = alpha^k \mod (p) \tag{2.7}$$

.

— Calculer

$$s = k^{-1}(m - ar) \mod (p-1) \tag{2.8}$$

.

La vérification :

— Calculer

$$v1 = beta^r * r^s \mod (p) \tag{2.9}$$

.

— Calculer

$$v2 = alpha^m \mod (p) \tag{2.10}$$

.

— Si v1=v2 ,signature est valide sinon non valide.

2.3.4 Le codage

Le codage est le processus de transformation des données dans un format différent à l'aide d'une méthode accessible au public. Le but de cette transformation est d'accroître la convivialité des données, en particulier dans différents systèmes. [114]

Base binaire

Ce codage de l'information est nommé base binaire. C'est le codage de fonctionnement d'ordinateur. Il consiste à utiliser deux états (représentés par les chiffres 0 et 1) pour coder les informations.[115]

UTF-8

UTF-8 est un codage de caractères. Il attribue à chaque caractère Unicode existant une séquence de bits précise que l'on peut également lire comme un nombre binaire. Cela signifie qu'UTF-8 attribue un nombre binaire fixe à l'ensemble des lettres, chiffres et symboles d'une quantité toujours plus importante de langues.[116]

Base64

Le codage base 64 est un codage permettant de transformer toute donnée binaire en une donnée n'utilisant que 64 caractères ASCII disponibles sur la plupart des systèmes informatiques.

Le principe de ce codage consiste à découper la donnée binaire en tranches de six bits, que nous nommerons sextets, et d'associer à chaque sextet un caractère choisi parmi les 26 lettres majuscules (A, ..., Z), les 26 lettres minuscules (a, ..., z), les 10 chiffres décimaux (0, ..., 9), et les deux caractères + et /. L'alphabet cible de ce codage comprend donc 64 caractères.[115]

字符串	N								E															
ASCII(十进制)	78								69															
二进制	0	1	0	0	1	1	1	0	0	1	0	0	0	1	0	1	0	0						
索引值	(00 010011) 19								(00 100100) 36								(00 010100) 20							
对应编码	T								k								U							
Base64 结果	TkU=																							

FIGURE 2.8 – Codage base 64

编号	字符	编号	字符	编号	字符	编号	字符
0	A	16	Q	32	g	48	w
1	B	17	R	33	h	49	x
2	C	18	S	34	i	50	y
3	D	19	T	35	j	51	z
4	E	20	U	36	k	52	0
5	F	21	V	37	l	53	1
6	G	22	W	38	m	54	2
7	H	23	X	39	n	55	3
8	I	24	Y	40	o	56	4
9	J	25	Z	41	p	57	5
10	K	26	a	42	q	58	6
11	L	27	b	43	r	59	7
12	M	28	c	44	s	60	8
13	N	29	d	45	t	61	9
14	O	30	e	46	u	62	+
15	P	31	f	47	v	63	/

FIGURE 2.9 – Les codes base 64

2.4 Protocoles de sécurité

Différents protocoles de sécurité ont été analysés pour une base de données centralisée avec architecture distribuée. Le but est d'identifier des protocoles de sécurité adéquats pour atténuer la sécurité de l'information par l'identité d'un modèle sans s'appuyer sur des infrastructures technologiques.[117]

2.4.1 IPsec

Internet Protocol Security (IPsec) est une suite de protocole réseau sécurisé qui authentifie et crypte les paquets de données pour fournir une communication cryptée sécurisée entre deux ordinateurs sur un réseau Internet Protocol.

IPsec a initialement défini deux protocoles pour sécuriser les paquets IP : Authentication Header (AH) et Encapsulating Security Payload (ESP).[118]

Le protocole **ESP** assure la confidentialité des données (cryptage) et l'authentification (intégrité des données, authentification de l'origine des données et protection de la lecture).[119]

Le protocole **AH** est une partie de la suite de protocoles Internet Protocol Security (IPsec), qui authentifie l'origine des paquets IP (datagrammes) et garantit l'intégrité des données. [120]

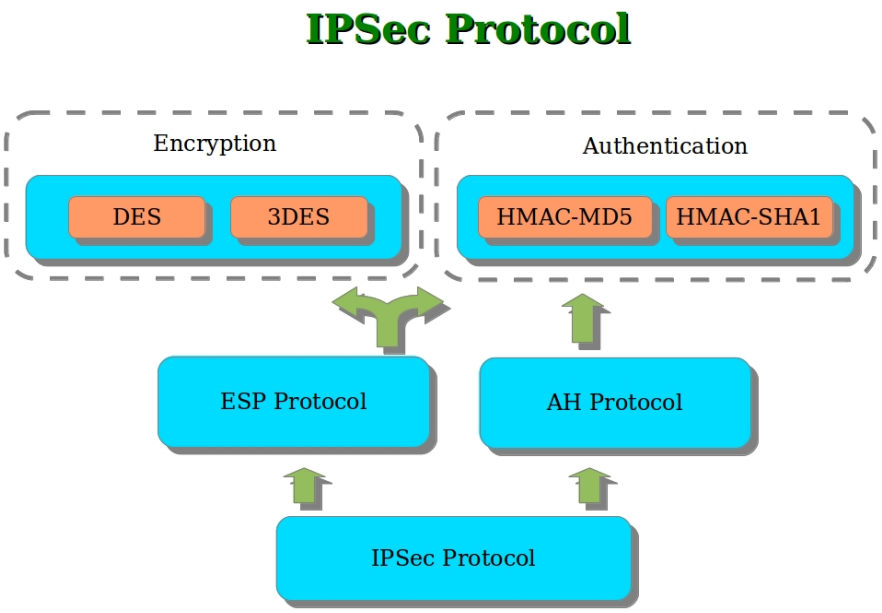


FIGURE 2.10 – IPsec [121]

2.4.2 SSL/TLS

SSL (Secure Socket Layer) et TLS (Transport Layer Security) sont des protocoles cryptographiques populaires qui sont utilisés pour imprégner les communications Web. Le protocole SSL/TLS utilise une paire de clés pour authentifier les identités et crypter les informations envoyées sur Internet. L'une d'elles (la clé publique) est destinée à une large diffusion, et l'autre (la clé privée) doit être conservée de la façon la plus sûre possible. Ces clés sont créées ensemble lorsque vous générez une demande de signature de certificat (CSR).[122]

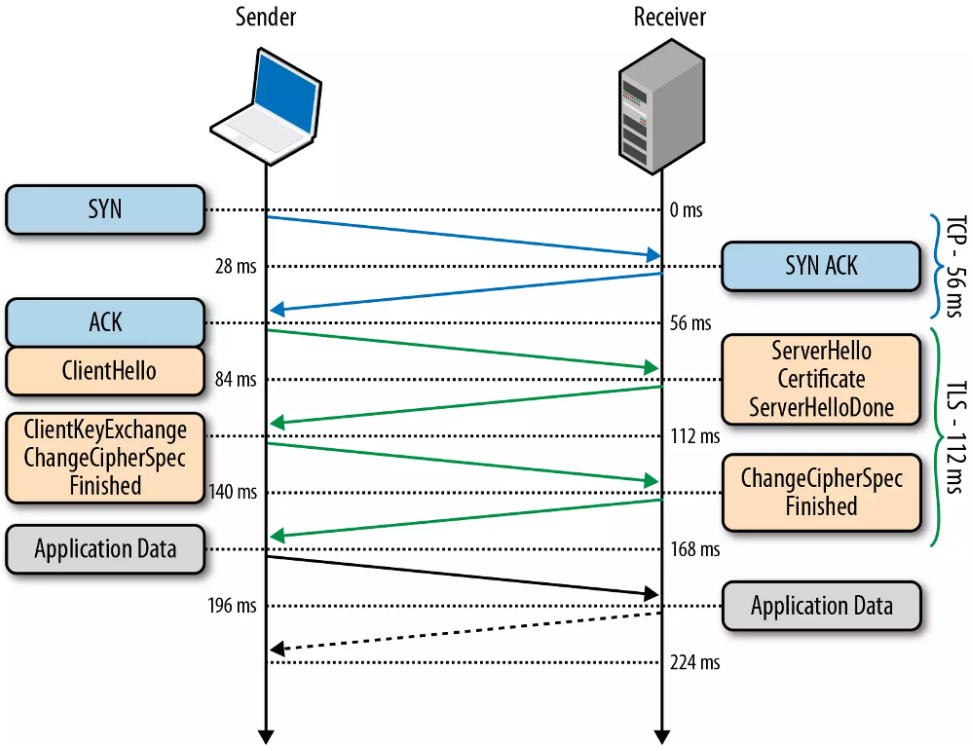


FIGURE 2.11 – SSL/TLS [123]

2.4.3 Kerberos

Kerberos est un protocole d'authentification ordinateur-réseau qui fonctionne sur la base de tickets pour permettre aux nœuds communiquant sur un réseau non sécurisé de prouver leur identité les uns aux autres de manière sécurisée. kerberos permet à un client de s'authentifier à plusieurs serveurs en supposant qu'il existe une clé secrète à long terme partagée entre le client et l'infrastructure Kerberos. La clé secrète à long terme du client a été générée à l'aide du mot de passe du client (décrit la technique de transformation du mot de passe présentée par la spécification standard). [124]

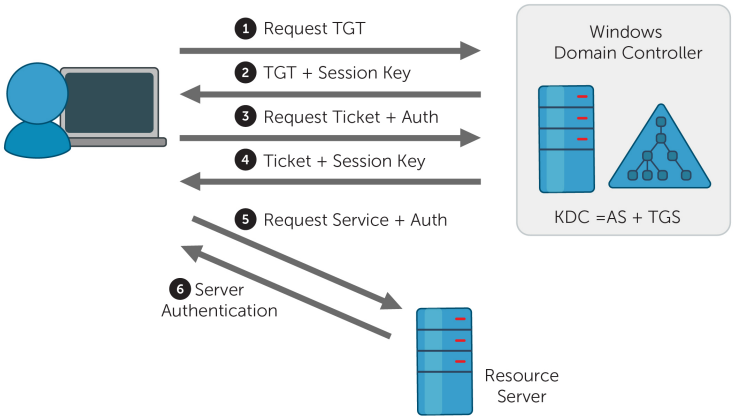


FIGURE 2.12 – Kerberos [124]

2.4.4 RADIUS

RADIUS (Remote Authentication Dial-In User Service), mis au point initialement par Livingston, est un protocole d'authentification standard, défini par un certain nombre de RFC. Le fonctionnement de RADIUS est basé sur un système client/serveur chargé de définir les accès d'utilisateurs distants à un réseau.[125]

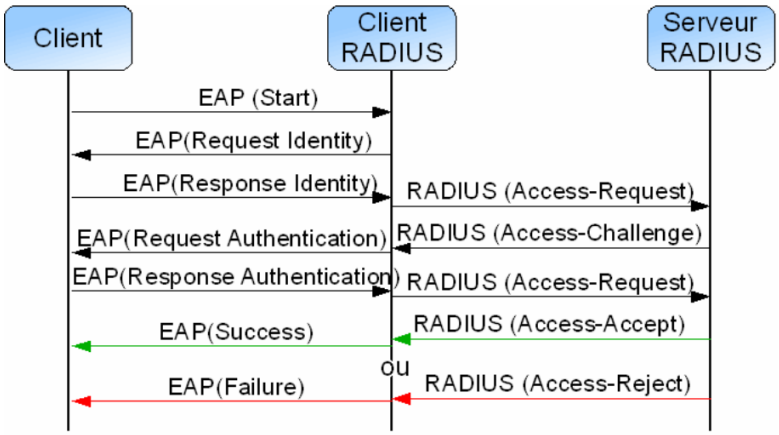


FIGURE 2.13 – RADIUS [126]

2.5 Les services de sécurité

La sécurité informatique vise généralement cinq principaux services :

2.5.1 Intégrité

Vérifier l'intégrité des données consiste à déterminer si les données, ressources, traitements ou services n'ont pas été altérées durant la communication de manière fortuite ou intentionnelle.[127]

2.5.2 Disponibilité

La disponibilité garantit que les utilisateurs autorisés d'un système ont un accès rapide et ininterrompu aux informations contenues dans ce système, ainsi que'au réseau.

Il est important de déployer des protections contre les interruptions de tous les systèmes qui doivent fonctionner en continu. Les différentes options incluent la redondance matérielle, la bascule, les sauvegardes de routine dans un espace géographiquement séparé.[128]

2.5.3 Confidentialité

La confidentialité est le maintien du secret des informations. Elle consiste à rendre l'information discrète ou inintelligible à d'autres personnes que les seuls acteurs de la transaction.[129]

2.5.4 Non répudiation

Elle consiste en l'assurance qu'une action sur la donnée réalisée au nom d'un utilisateur (après authentification) ne saurait être répudiée par ce dernier. À l'instar de la confidentialité et de l'intégrité, la non-répudiation fait appel à des mécanismes de chiffrement.[130]

2.5.5 Authentification

L'authentification est le processus qui permet aux gens, souvent aux employés, d'identifier qui ils sont afin qu'ils puissent accéder aux installations ou au réseau informatique de l'entreprise. Le processus est extrêmement important pour aider à prévenir l'accès non autorisé, ce qui peut mener à des atteintes catastrophiques aux données. Il existe une grande variété de méthodes d'authentification, allant de la simple (un mot de passe unique) à la complexe (authentification multi-facteurs qui peut inclure des mots de passe, des codes uniques et des données biométriques).[52]

Il y a trois types d'authentification de base que nous considérons habituellement :

La première est fondée sur la connaissance vous connaissez quelque chose comme un mot de passe ou un code PIN que seul vous, l'utilisateur identifié, connaissez.

La deuxième est fondée sur la propriété, c'est-à-dire que vous possédez quelque chose, comme une carte d'accès, une clé, un porte-clés ou un appareil autorisé, que vous seul devriez avoir.

Le troisième est biologique, c'est-à-dire qu'il s'agit d'une partie de votre corps physique ou d'un processus physiologique ou comportemental qui vous est propre, comme vos empreintes digitales ou le profil rétinien de votre œil.

L'authentification consiste à assurer l'identité des acteurs c.à.d. de garantir à chacun de correspondant que son partenaire est bien celui qu'il croit être. [131] Il y a quatre facteurs utilisés pour authentifier l'identité :[132]

- Ce que l'on connaît (facteur mémoriel) : une information que l'utilisateur a mémorisée et que lui seul connaît (exemple : un mot de passe, un nom).
- Ce que l'on est (facteur corporel) : une information qui caractérise l'utilisateur avec une empreinte qui lui est propre (exemple : voix, pupille, empreinte digitale)
- Ce que l'on possède (facteur matériel) : une information que seul l'utilisateur possède et enregistrée dans un support (exemple : une clé USB).
- Ce que l'on sait faire (facteur réactionnel) : une information ou un geste que seul l'utilisateur peut produire (exemple : une signature).

Les types de techniques d'authentification

L'authentification est l'un des processus les plus couramment utilisés dans le monde des applications mobiles et du développement Web. Il est nécessaire de connaître les différentes techniques d'authentification des utilisateurs et de leur permettre d'accéder à un logiciel particulier. Parmi les types les plus courants de techniques d'authentification, mentionnons :

— **Par session**

Dans l'authentification par session, les utilisateurs se connectent d'abord grâce à leurs identifiants. Ils sont alors authentifiés et commencent une session. Une session représente la période entre la connexion et la déconnexion de l'utilisateur. Le serveur enregistre les données de session de l'utilisateur, et envoie une copie à un petit fichier (un cookie), également enregistré dans le navigateur de l'utilisateur.[133]

— **Par cookies**

Les applications utilisent couramment des cookies de navigateur pour stocker les identifiants de session d'application Web lorsque les systèmes implémentent des méthodes de gestion de session.

Quelques défenses sont : [134]

- Seul un petit nombre de domaines devrait pouvoir accéder aux cookies. L'application doit baliser leurs chemins afin que les balises expirent à la fin, ou peu de temps après, la session perde de sa vitalité.
- Définir les balises (sécurisées) pour s'assurer que le système ne transfère que sur un canal sécurisé.
- Définir l'indicateur Http Only pour empêcher JavaScript d'accéder au cookie.
- Ajout de caractéristiques (même site) aux cookies, ce qui empêche certains navigateurs actuels d'envoyer des cookies ayant des demandes inter sites. Cela protège contre les attaques de type fuite d'informations et la contre façon à la demande entre sites.

— **Par jeton**

L'authentification par jeton solide repose sur un protocole qui permet à un utili-

sateur de recevoir un jeton d'accès unique après la confirmation de son identité. L'utilisateur peut alors accéder à l'application ou au site Web pour lequel le jeton a été accordé pour la durée de vie du jeton.[135]

Autrement dit, autoriser les utilisateurs à entrer leur nom d'utilisateur et leur mot de passe afin d'obtenir un jeton qui leur permet de récupérer une ressource spécifique - sans utiliser leur nom d'utilisateur et leur mot de passe. Une fois son jeton obtenu, l'utilisateur peut proposer le jeton - qui donne accès à une ressource spécifique pendant une période donnée au site distant.[136]

2.6 JWT

On va présenter la variante particulière des jetons d'authentification JSON Web Token(JWT)

2.6.1 Définition

JSON Web Token (JWT) est un standard ouvert (RFC 7519) qui définit une manière compacte et autonome de transmettre des informations entre les parties en tant qu'objet JSON. Ces informations peuvent être vérifiées et fiables car elles sont signées numériquement. Les JWT peuvent être signées à l'aide d'un secret (avec l'algorithme HMAC) ou d'une paire de clés publiques/privées utilisant RSA ou ECDSA.

Bien que les JWT puissent être chiffrées pour assurer le secret entre les parties, nous nous concentrerons sur les jetons signés. Les jetons signés peuvent vérifier l'intégrité des allégations qu'ils contiennent, tandis que les jetons chiffrés cachent ces allégations aux autres parties. Lorsque des jetons sont signés à l'aide de paires de clés publiques/privées, la signature atteste également que seule la partie qui détient la clé privée est celle qui l'a signée.[137]

Quand devriez-vous utiliser les jetons Web JSON ?

Voici quelques scénarios où les jetons Web JSON sont utiles :

— Autorisation : C'est le scénario le plus courant pour l'utilisation de JWT. Une fois que l'utilisateur est connecté, chaque demande subséquente comprendra le JWT,

permettant à l'utilisateur d'accéder aux routes, services et ressources qui sont autorisés avec ce jeton. Single Sign On est une fonctionnalité qui utilise largement JWT de nos jours, en raison de ses petits frais généraux et de sa capacité à être facilement utilisé dans différents domaines.

— Échange d'informations : Les jetons Web JSON sont un bon moyen de transmettre des informations en toute sécurité entre les parties. Comme les JWT peuvent être signées par exemple, en utilisant des paires de clés publiques et privées, vous pouvez être certain que les expéditeurs sont bien ceux qu'ils prétendent être. De plus, comme la signature est calculée à l'aide de l'en-tête et de la charge utile, vous pouvez également vérifier que le contenu n'a pas été altéré.[137]

Un jeton se compose de trois parties :

— Un en-tête (header), utilisé pour décrire le jeton. Il s'agit d'un objet JSON.
— Une charge utile (payload) qui représente les informations embarquées dans le jeton. Il s'agit également d'un objet JSON.
— Une signature numérique.

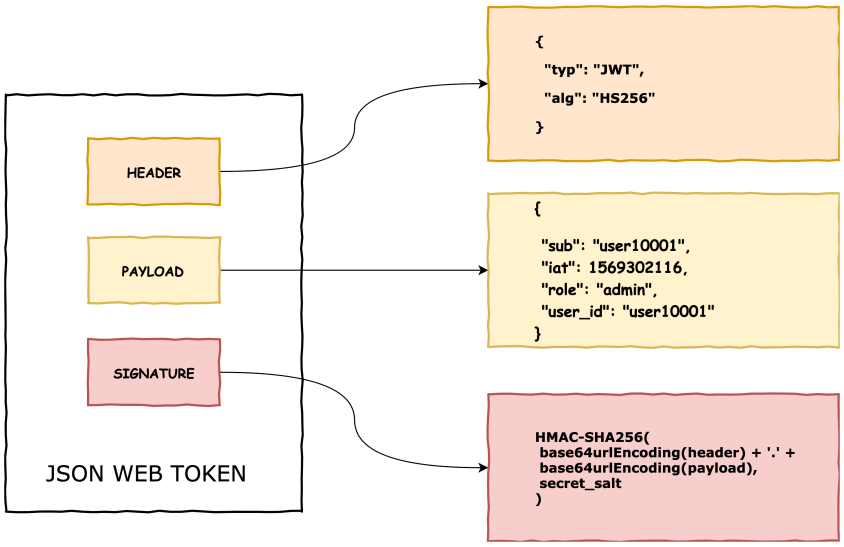


FIGURE 2.14 – Les composants de JWT [138]

La figure suivante illustre le travail avec **JWT**

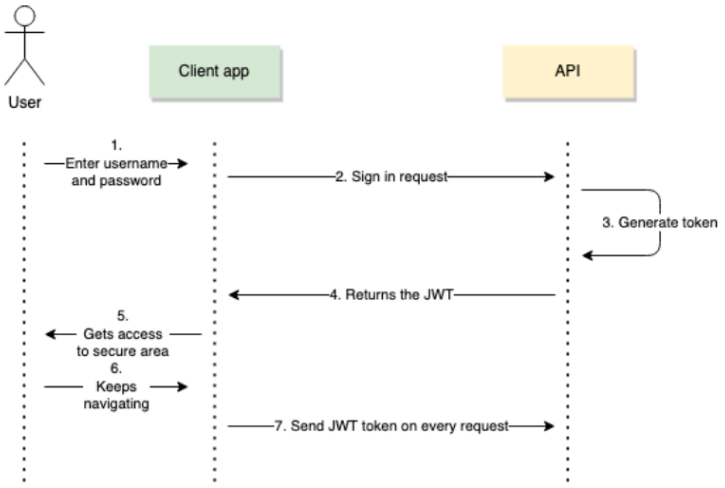


FIGURE 2.15 – JWT[139]

2.6.2 Le flux d'authentification

On va présenter les différents étapes d'authentification en utilisant JWT :[140]

— L'utilisateur soumet le nom d'utilisateur et le mot de passe au service d'authentification.

— Le service d'authentification valide les identifiants et génère un jeton JWT signé avec la chaîne secrète et une charge utile contenant l'identifiant utilisateur et l'horodatage d'expiration/durée

— Le client (navigateur) stocke le jeton dans un stockage local ou des témoins ou n'importe où il peut être récupéré à un moment ultérieur

— Lorsque l'utilisateur souhaite récupérer une ressource protégée (p. ex., naviguer vers une page protégée), le navigateur devra inclure le jeton dans chaque requête HTTP à notre serveur (habituellement via l'en-tête Authentification).

— Le serveur vérifie si la signature est valide – il génère une nouvelle signature à partir de l'en-tête et de la charge utile du jeton fourni, puis vérifie si elle correspond à la signature dans le jeton fournie.

— Le serveur récupère l'identifiant utilisateur du token JWT et les étoiles traitent la requête HTTP en conséquence

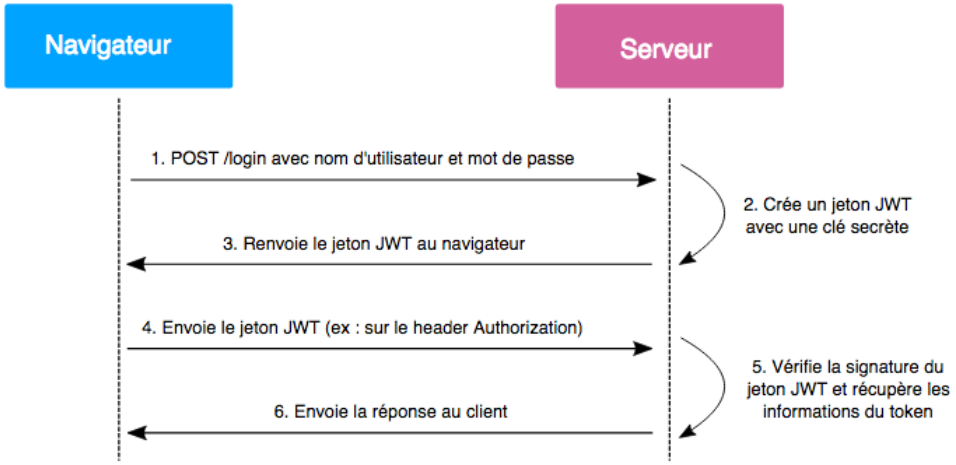


FIGURE 2.16 – Le flux d'authentification JWT [14]

2.6.3 Le fonctionnement de JWT avec HS-256

L'utilisateur POST une demande au serveur demandant de s'authentifier à l'aide de son nom d'utilisateur et de son mot de passe. Si ces identifiants existent dans la base de données, le serveur retournera un jeton JWT contenant les informations de l'utilisateur dans la charge utile. La prochaine fois que l'utilisateur fera une demande, il devra passer le JWT dans l'en-tête avec l'appel. Le serveur sera configuré pour vérifier l'authenticité du JWT entrant, de sorte qu'il n'acceptera que les jetons créés lui-même.

Il a utilisé l'algorithme symétrique par défaut HS-256 qui n'utilise qu'une seule clé secrète qui est partagée entre les deux parties pour la génération et la validation de la signature. Ainsi, Il faut veiller à ne pas compromettre la clé. [141]

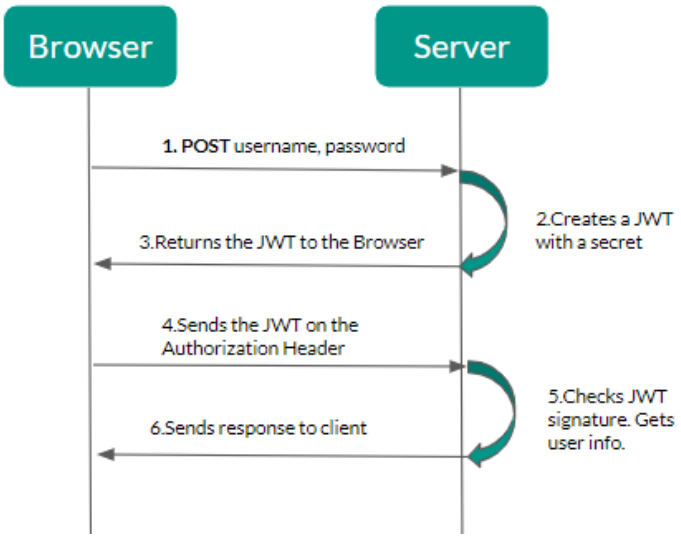


FIGURE 2.17 – Authentification JWT avec HS-256 [141]

2.6.4 Le fonctionnement de JWT avec RSA

Une autre approche serait d'utiliser un algorithme asymétrique qui utilise une clé publique/privée. Le serveur dispose d'une clé privée générer la signature et le consommateur du jeton obtient un public pour valider la signature. Il n'est pas nécessaire de conserver la clé publique. Sécurisé et ainsi il peut être facilement mis à la disposition des consommateurs pour utiliser.[141]

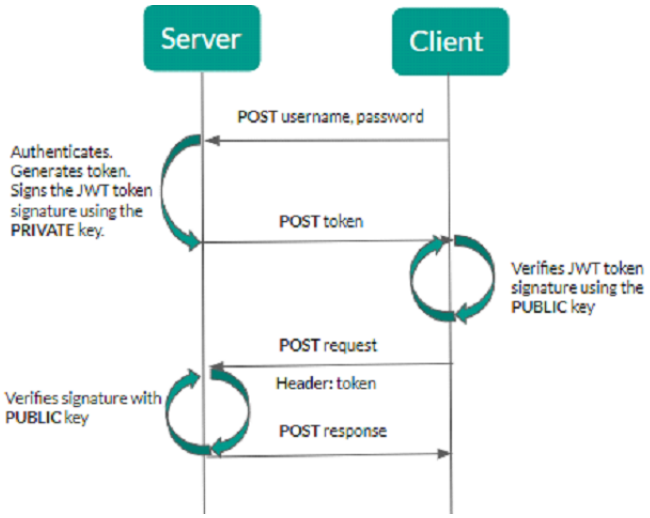


FIGURE 2.18 – Authentification JWT avec RSA [141]

2.7 Conclusion

Ce chapitre a introduit quelques notions sur la sécurité d'informatique, nous avons aussi présenté les mécanismes de sécurité, les protocoles de sécurité. Enfin, nous avons abordé les services de sécurité et la notion de JWT.

Dans le chapitre qui suit, nous présenterons notre système les différents outils utilisés pour le développement.

Chapitre 3

Conception et implémentation

3.1 Introduction

Après avoir présenté dans les chapitres précédents la notion d'IoT et la sécurité d'information.

Ce chapitre, est consacré à la conception et la description du processus de réalisation de notre système. Ceci en mettant en évidence l'ensemble des environnements (logiciels et matériels), de développement, de déploiement, et de fonctionnement, ainsi qu'un aperçu sur les application mobile permettant la gestion du système.

3.2 Problématique, motivation et objectif

L'évolution technologique nous permet aujourd'hui de prédire les espaces et d'améliorer l'adaptation du mode de vie dans les maisons. La maison intelligente, qui est très connus sous le nom de Smart home, est un éspace dans lequel les appareils intelligentes sont connectés via des passerelles résidentielles. Ce qui constituent un réseau domestique local pour aider les gens dans leurs activités de la vie quotidienne. Aussi, la majorité des individus, en particulier les personnes âgées surtout les femmes, passent la majeure partie de leur temps à la maison, d'où l'impact significatif du logement sur la qualité de vie. Ainsi, améliorer la sécurité et de confort à la maison semble être une tâche de grande importance sociale qui peuvent protéger les personnes et leurs biens contre divers dangers

tels que le vol, fuite de gaz, le feu, l'entrée d'étrangers à la maison...etc.

Cependant, comme toute autre domaine, l'IoT pose certains problèmes qui nécessite la résolution. Principalement, la sécurité des informations et du réseau, le degré de la latence et la bande passante, et la consommations de l'énergie. L'objectif principale de notre travail est de concevoir et réaliser un système "AMINE" pour la sécurisation des smart home. Aussi, nous voulons construit une application multiplatforme pour l'accès au système (page web ou application mobile Android ou IOS). De plus, nous avons réaliser un prototype d'une maison intelligente qui utilise l'énergie solaire pour y implémenter et tester le système. Notre système, "Amine", en plus de la sécurisation de la maison intelligente, répond a certains problématique existantes dans le domaine. Notre système répond a certains des questions posés dans le domaine : (1) comment faite sécuriser les données qui circulent ? ; (2) comment faire pour diminuer la latence de transmission des données ? ; (3) comment faire si il y'a une coupure de l'internet ?

3.3 Architecture de notre système

Notre système **AMINE** est conçue pour la sécurité des Smart home. **AMINE** permet à des utilisateurs de superviser leurs maisons intelligentes à distance et de contrôler certains aspects et capteurs. A travers une application, les utilisateurs accèdent au système via son smart phone, tablette ou ordinateur. Cela leurs permet d'envoyer des consignes et recevoir des informations à et de sa maison. Comme notre système est multiutilisateur, donc une stratégie sécuritaire est primordiale. Pour cela un système d'authentification à été mis en points. Cet échange des données et ce besoin en sécurité n'est possible qu'en présence d'une entité autoritaire.

La figure ci-dessus 3.1 représente l'architecture globale de notre système :

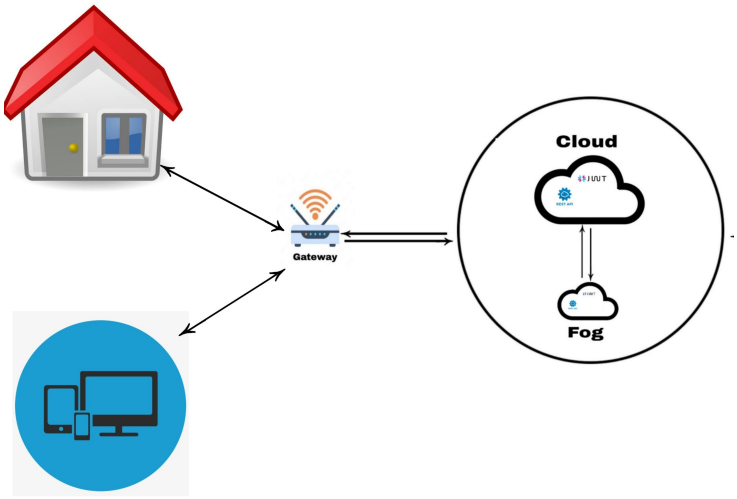


FIGURE 3.1 – Architecture générale de notre système "AMINE"

Notre système "AMINE" garanti :

— La sécurisation de la maison intelligente aux intrusion et au incidents probable.

— La sécurisation des données qui circulent dans le réseau. En appliquant le JWT dans le protocole d'application IoT Rest API pour l'authentification, ainsi le protocole SSL/TLS pour assurer la confidentialité.

— La bonne connectivité du système à tout moment surtout s'il y'a une coupure de l'internet, ainsi que la réduction de la latence de transmission. Cela en introduisant le fog computing avec le cloud.

— Augmenter la bande passante en et permettre la transmission d'une grande quantités de données en utilisant le Wi-Fi au lieu d'utiliser d'autres technologies de communication(zigbee, Lora,.....).

3.4 Les fonctionnalités du système

Nous aborderons l'ensemble des fonctions offertes par notre système.

1. **La notification**

   L'utilisateur reçoit une notification s'il y a un problème dans la maison (incendie, fuite de gaz, ...).

2. **Contrôle de température et l'humidité**

   Notre système aide à régler la température et l'humidité au seins de la smart home. Ces paramètres seront exploités pour réaliser les actions automatiques suivantes :

   — Envoyer une notification aux utilisateurs dès que la température dépasse les 50°C et l'humidité dépasse 95 pourcent.

   — Activation automatique de la ventilateur dès que la température dépasse 50°C ou si l'humidité dépasse 95 pourcent.

3. **Home monitoring**

   Un système de surveillance à domicile est fait d'appareils intelligents comme la caméra qui aident les propriétaires à gérer la maison.

4. **Détection d'incendie**

   Le capteur de flamme est utilisé pour détecter et répondre à la présence d'un feu. Afin de gérer la situation en cas d'incendie notre système.

   — Une notification aux utilisateurs est envoyé dés qu'incendie est détecter, indiquant qu'il y a une flamme dans la chambre.

   — Une alarme sonores est déclencher.

   — L'ouverture automatique de la fenêtre de la chambre.

   — Si l'utilisateur ne répond pas après 3 notifications, le système appellera les pompiers.

5. **Détection d'intrusion** Notre système utilise une caméra qui contient un système de reconnaissance. Ceci permet de reconnaître les intrus en analysant son visage de manière automatique. De plus, les membres de la famille sont tous connus par le système.

   — Détecter un visage et l'analyser.

   — Si la personne détecter est membres de familles, c'est à dire sa photo est enregistrée dans le serveur aucune action n'est effectuer.

   — Si la personne détectée n'est pas un membre de la famille, le système enverra une notification à l'utilisateur et s'il ne prend aucune mesure après 3 notifications, il appellera la police.

6. **Détection du gaz**

   Le capteur de gaz pour détecter et répondre à la présence d'une fuite de gaz.

   — Une notification aux utilisateurs est envoyé dés qu'une fuite de gaz est détecter, indiquant qu'il y a une fuite de gaz dans la chambre.

   — Une alarme sonores est déclencher.

   — L'ouverture automatique de la fenêtre de la chambre.

7. **La production d'électricité solaire**

   L'énergie solaire est captée par le biais de modules (de grands rectangles) recouverts de cellules de silicium, un matériau semi-conducteur. La captation et transformation de l'énergie solaire se déroulent ainsi : Les photons (la lumière du soleil) frappent les cellules.

8. **Contrôle de la lumières et lampes**

   Le contrôle de la lumière permette aux utilisateurs de faire la gestion des lumières. Nous contrôlons les lampes via un smartphone de n'importe où.

9. **Contrôle de la porte**

   Cette fonction assure l'ouverture de la porte d'une manière plus sécurisé en adoptant un système d'accès par l'application mobile.

10. **Contrôle de la fenêtre**

    L'utilisateur peut contrôler plus alaise l'ouverture des fenêtres d'une manière plus sécurisé en adoptant un système d'accès par l'application mobile.

3.5 Implémention

Après avoir présenté la conception de notre système, nous allons présenter notre application réalisée ainsi les différents outils nécessaires pour le développement.

3.5.1 Environnement matériel

Nous avons utilisé les matériels suivants pour réaliser notre travail :

— **Capteur de mouvement HC-SR501 :**

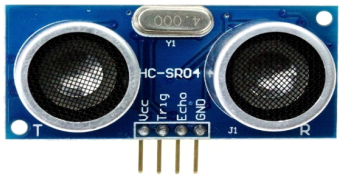
Le capteur de mouvement PIR (Passive Infrared Sensor) est un senseur électronique qui mesure la lumière infrarouge (IR) rayonnant à partir d'objets dans son champ de vision. Ils sont très souvent utilisés dans les systèmes d'alarmes ou de détection de présence pour leur faible coût et leur efficacité. Il permet de détecter de mouvement et activation d'une œuvre interactive l'ouvrage de lumière et l'envoi de notification, Commander d'une chatière automatisée. [142]

Les Caractéristiques :

- Dimensions : 32 x 24 x 27H mm
- Voltage : 5-12VDC
- Detection Distance : 3-7mt (approx, adjustable)
- Delay Time : 5-200s (adjustable)

— **Capteur ultrasonic HC-SRO4**

Le capteur HC-SR04 est un capteur à ultrason low cost. Ce capteur fonctionne avec une tension d'alimentation de 5 volts, dispose d'un angle de mesure de 15° environ et permet de faire des mesures de distance entre 2 centimètres et 4 mètres avec une précision de 3mm (en théorie, dans la pratique ce n'est pas tout à fait exact). [143]



— **Capteur d'humidité et de température DHT11**

Le DHT11 est composé de deux parties, un capteur d'humidité capacitif et un capteur de température à base de NTC. Il contient également un circuit électronique élémentaire qui effectue la conversion analogique vers numérique et qui débite un signal numérique proportionnel à la température et l'humidité mesurée par le capteur. [144] Caractéristiques du DHT11 :

- Plage de Mesure : - Humidité : 20-95 ; Température : 0-50°C ;
- Période de mesure : 2s.

— **Capteur MQ7**

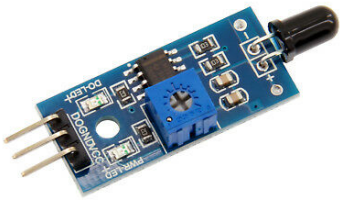
Module comportant un capteur MQ7 permettant de détecter la présence de monoxyde de carbone COde 300à 10000 ppm. Ce module est compatible Arduino et Raspberry Pi.[145]

Caractéristiques :

- Alimentation : 5 Vcc
- Plage de mesure : 300 à 10000 ppm
- Sortie analogique
- Sensibilité : 2 à 20 kΩ
- Faible temps de réponse

— **Capteur de flamme**

Le capteur de flamme KY-026 est un capteur qui permet de mesurer des longueurs d'onde sur une plage comprise entre 760 nm et 1100 nm. Ce capteur réagira donc en présence d'une flamme.[146]



Caractéristiques :

- Extrêmement sensible aux longueurs d'ondes entre 760-1100nm.
- Seuil de détection de flamme modifiable par un potentiomètre.
- Plage d'angle de détection : environ 60 degrés

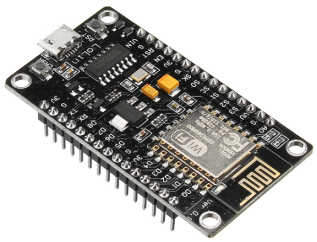
— **Buzzer active 5V**

Le buzzer est un moyen simple, rapide et pratique d'ajouter une sortie audio. Les alertes audio sont très efficaces pour attirer l'attention immédiatement, même à distance. Cela les rend très utiles pour les alarmes et les avertissements d'urgence. Bien sûr, vous pouvez également décider de les utiliser pour des applications musicales ou autres. [147] Leur caractéristiques sont :

- Tension de fonctionnement : 3.3 - 5 V
- Signal I/O : Numérique TTL (0 ou 1)

NodeMCU

NodeMCU est une plate-forme open source IoT, matérielle et logicielle, basée sur un SoC Wi-Fi ESP8266 ESP-12 fabriqué par Espressif Systems. Le terme (NodeMCU) se réfère par défaut au firmware plutôt qu'aux kits de développement.[148]



Caractéristiques :

- Alimentation : 5 Vcc via micro-USB - 5 à 9 Vcc via broche Vin (régulateur intégré).
- Microcontrôleur : ESP8266.
- Microprocesseur : Tensilica LX106 Fréquence : 80 MHz.
- Mémoire RAM : 64 kB.
- Interfaces : I2C, SPI, UART Interface Wifi 802.11 b/g/n.
- Dimensions : 58 x 31 x 12 mm.
- Référence fabricant : NodeMCU ESP8266.

— **ESP32-CAM CAMERA BLUETOOTH / WIFI OV2640**

L'ESP32-CAM est une carte de développement ESP-WROOM-32 du fabricant AI Thinker associé à une caméra couleur 2MP OV2640. Le module ESP32-CAM dispose également d'un lecteur de carte SD qui pourra servir à enregistrer des images lorsqu'un événement est détecté (détecteur de présence ou de mouvement par exemple). C'est vraiment une excellente base pour développer son propre système de vidéo surveillance IP sans avoir la crainte que le flux vidéo arrive sur des serveurs douteux.[149]

Caractéristiques :

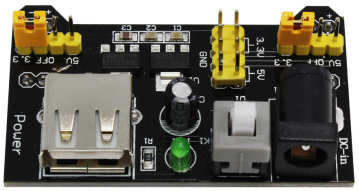
- Ultra petit 802.11b/g/n Wifi + module SoC BT / BLE.
- CPU 32 bits dual-core basse consommation jusqu'à 240 MHz, jusqu'à 600 DMIPS.
- Prise en charge des images téléchargées par WiFI.

— **Panneau solaire 5v**

Ce panneau solaire de 5V est léger et durable. Il est également imperméable, résistant aux UV et aux rayures. Ce petit panneau solaire de 5 V est idéal pour recharger vos batteries de 3,2 volts CC. Applications : capteurs sans fil, appareils IoT, suivi GPS, Lumières LED et autres petits appareils électroniques, etc.[150]

— **MB102 Module D'alimentation 3.3 V 5 V**

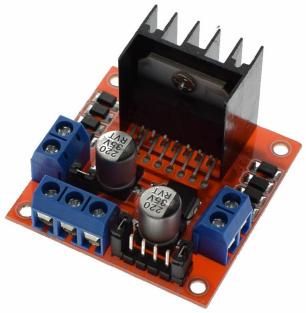
Modules D'alimentation MB102 de Plaque d'essai 3.3 V/5 V pour arduino.[151]



- Tout neuf et de haute qualité.
- Module d'alimentation, compatible avec 5 V, 3.3 V.
- Peu être utiliser avec une breadboard MB102.
- Tension d'entrée : 6.5-12 V (DC) ou USB alimentation.
- Tension de sortie : 3.3 V/5 V peut basculer.

— **L298 DUAL H-BRIDGE MOTOR DRIVER**

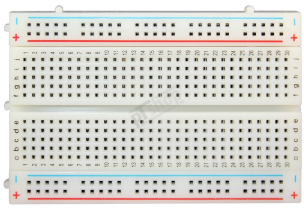
Ce module est basé sur le pilote L298 Double H-Bridge circuit intégré. Le circuit vous permettra de contrôler facilement et de manière indépendante deux moteurs jusqu'à 2A chacun dans les deux sens. Il est idéal et bien adapté pour la connexion à un micro contrôleur nécessitant seulement quelques lignes de commande par moteur. [152]



— **Ventilateur de refroidissement**

Le ventilateur de refroidissement est constitué d'un petit moteur électrique qui assure la rotation des pales, il produit le courant d'air qui va diminuer la température du liquide de refroidissement. Le fonctionnement du ventilateur de refroidissent est actionné par l' interrupteur de refroidissement. [153]

— **Plaque d'essai :**

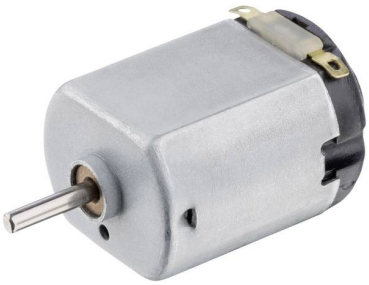


C'est une plaque en plastique isolant parsemé de plein de trous. Ces trous sont espacé de 2.54 mm qui est l'espacement standard des composants électroniques que nous utilisons dans nos montages. Vous l'aurez donc compris les trous permettent d'enfoncer des composants ce qui permettra de le relier entre eux afin de réaliser le montage à tester.[154]

— **Led**

Une diode électroluminescente ou LED (en anglais :Light-Emitting Diode,LED),est un composant opto-électronique capable d'émettre de la lumière lorsqu'il est parcouru par un courant électrique.[155]

— **MOTRAXX MOTEUR À COURANT CONTINU 1.5 V/DC 0.55 A FA10-130RAC-2280-RU**



Leur Caractéristiques : [156]

- Tension 1.5 V
- Tension d'alimentation (moteurs électriques) : 1,5 - 3 V/DC
- Courant nominal : 0.55 A.
- Vitesse de rotation : 8000 tr/min.
- Puissance nominale (W) : 0.37 W.
- Dimensions produit, hauteur : 15.1 mm.
- Dimensions produit, largeur : 19.5 mm.

— **Ordinateur**

Nous avons utilisé un ordinateur pour réaliser notre travail :

Un ordinateur ASUS caractérisé par :

- Processeur : i5-7200U CPU R5 M420.
- RAM : 8,00 GB.
- Système d'exploitation : Windows 11 64 bits.

3.5.2 Environnement logiciel

- **Langage de programmation** : nous avons conçu notre application sous les langages PYTHON, C, et Dart.

  Python Python est un langage de programmation de haut niveau conçu pour être facile à lire et simple implémenter. Python est open source, ce qui signifie qu'il est gratuit à utiliser, même pour les applications de commerce. Python peut fonctionner sur plusieurs plate-formes.[133] [157]

  C : C est un langage de programmation procédural. Il a été initialement développé par Dennis Ritchie en 1972. Il a été principalement développé comme un langage de programmation système pour écrire un système d'exploitation. [158]

  Dart : Dart est un langage de programmation optimisé pour les applications sur plusieurs plate-formes. Il est développé par Google et est utilisé pour créer des applications mobiles, de bureau, de serveur et web. Dart est un langage orienté objet à ramasse-miettes avec une syntaxe de type C++. Dart peut se compiler en code natif ou en JavaScript.

- **Éditeur :** Nous avons utilisé plusieurs éditeurs :

  Pycharm Pycharm est l'IDE le plus populaire pour Python, et il regorge des fonctionnalités telles que l'achèvement et l'inspection inégalée du code, un débogueur et prise en charge de la programmation Web et des cadres tels que Django et Flask.[159]

Android Studio : a été introduit en 2013 sur Google. Android Studio est une collaboration entre JetBrains et Google. Android studio est un nouvel IDE (Environnement de développement intégré) mis à disposition gratuitement Google à Développeurs Android. Il peut être téléchargé sous les systèmes d'exploitation Windows, macOS, Chrome OS et Linux.[160]

Arduino : Arduino est une plate-forme électronique open-source basée sur du matériel et des logiciels faciles à utiliser. Cartes Arduino sont capables de lire les entrées - lumière sur un capteur, un doigt sur un bouton, ou un message Twitter - et de le transformer en une sortie - activer un moteur, allumer une LED, publier quelque chose en ligne. Le logiciel open-source Arduino Software (IDE) permet d'écrire facilement du code et de le télécharger sur la carte. Ce logiciel peut être utilisé avec n'importe quelle carte Arduino. [161]

- **Plate-formes**

Flutter : Flutter est un framework open source de Google pour construire de belles applications multi-plate-formes nativement compilées à partir d'un seul codebase.
La première version de Flutter était connue sous le nom de code "Sky" et fonctionnait sur le système d'exploitation Android.[162]

Django : Django est un framework web basé sur Python qui vous permet de créer rapidement des applications web efficaces. Il est également appelé framework inclus dans les batteries parce que Django fournit des fonctionnalités intégrées pour tout, y compris l'interface d'administration Django, la base de données par défaut – SQLlite3, etc. Lorsque vous créez un site Web, vous avez toujours besoin d'un ensemble de composants similaires : une façon de gérer l'authentification des utilisateurs (inscription, connexion, déconnexion), un panneau de gestion pour votre site web, des formulaires. Le framework Django est basé sur l'architecture Model-View-Template (MVT).[163]

- **Les bibliothèques**

  Les bibliothèques utilisées sont :

  — **NumPy**

  NumPy est une bibliothèque pour langage de programmation Python, destinée à manipuler des matrices ou tableaux multidimensionnels ainsi que des fonctions mathématiques opérant sur ces tableaux.

  Plus précisément, cette bibliothèque libre et open source fournit de multiples fonctions permettant notamment de créer directement un tableau depuis un fichier ou au contraire de sauvegarder un tableau dans un fichier, et manipuler des vecteurs, matrices et polynôme.[164]

  — **Face Recognition**

  Reconnaissez et manipulez les visages de Python ou de la ligne de commande avec la bibliothèque de reconnaissance faciale la plus simple au monde. Construit en utilisant la reconnaissance faciale de pointe de **dlib** construit avec l'apprentissage profond.[165]

  — **OpenCV**

  OpenCV est une énorme bibliothèque open-source pour la vision par ordinateur, la machine learning et le traitement d'images. OpenCV prend en charge une grande variété de langages de programmation tels que Python, C ++, Java, etc.

Il peut traiter des images et des vidéos pour identifier des objets, des visages ou même l'écriture d'un humain. [166][167]

— **Rest Framework**

Le framework REST django est une extension pour Django qui facilite la création d'une interface REST pour les applications. Cette bibliothèque offre de nombreuses fonctionnalités différentes, tels que la sérialisation des données et les différentes vues et méthodes d'authentification. La sérialisation des données fait ici référence à la modification des formats de données de Django JSON, par exemple. Ces données sérialisées peuvent être facilement affichées sur une variété de plate-formes, telles qu'un site Web ou une application mobile.[168]

— **Urllib**

urllib bibliothèque est construite bibliothèque de demande HTTP python, comprenant quatre modules principaux : demande : la base, la requête HTTP est le module le plus important pour la demande de transmission analogique. Erreur : module de gestion des exceptions.[169]

— **Threading**

Le module threading est un module de la bibliothèque standard de Python, et donc toujours disponible pour Python Le module threading fournit des classes et des méthodes pour l'exécution simultanée de plusieurs méthodes, d'un même script.[170]

— **REST framework simplejwt**

Simple JWT est un plugin d'authentification JSON Web Token pour Django REST Framework.

Pour l'authentification, il existe une bibliothèque qui utilise des clés JWT rest framework simplejwt. Sur cette base, l'utilisateur envoie ses informations d'identification Adresse terminée par /api/token. Le serveur répond en entrant un clé d'accès et clé d'actualisation à plus long terme. Temps d'appels frontaux mesurer l'adresse /api/token/refresh avec une clé d'actualisation qui renvoie une nouvelle clé d'accès avec une nouvelle maturité.[168]

3.6 Présentation du système "Amine"

Nous avons implémenté le système "amine" que nous avons conçu dans la première partie, et pour mener à bien ce travail, nous avons utilisé plusieurs matériels (capteur, actionneur, ...), un panneau solaire pour la production d'électricité solaire, la caméra pour la vidéo de Surveillance, un microcontrôleur NodeMCU, la technologie de communication wifi, un smart phone comme un gatway, des protocoles d'application IoT RESTAPI et JWT, des platformes, cloud et fog computing.

Enfin, nous avons réalisé une application multiplate-forme pour gérer notre system en utilisant flutter.

La figure (3.2) ci-dessous illustre la présentation du système :

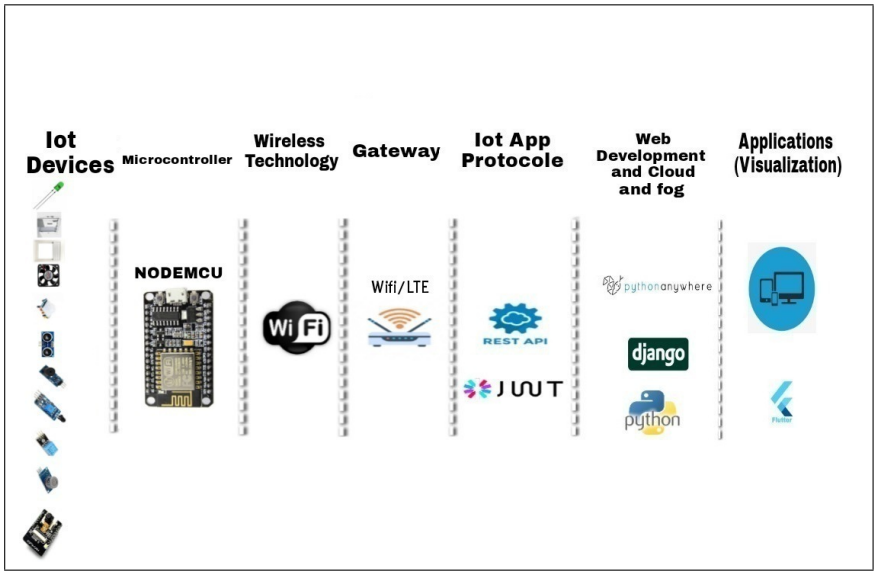


FIGURE 3.2 – Présentation du système

- **IOT devices**

  Notre système se compose de plusieurs devices tel que la caméra, les capteurs et les actionneurs. La camera pour prendre des photos numériques, elle peut même être utilisée comme vidéo de surveillance. On parle de capteurs qui permettent de faire des mesures de l'environnement physique (ex : température, humidité, bruit). Les actionneurs qui peuvent agir sur l'environnement (ex : des moteurs pour fermer ou ouvrir une porte).

- **Micro contrôleur**

  On utilise un microcontrôleur (NodeMcu) qui lit et collecte les données des capteurs et les envoie à la passerelle via le serveur, ou il active/désactive les actionneurs. S'il y a un problème avec la maison, il envoie une notification au utilisateur.

- **Technologies de communication IOT**

  Nous utilisons le wifi comme technologie de communication pour transférer de grandes masses de données, car d'autres technologies transfèrent de petites quantités de données. De plus, pour éviter le problème de nombre des passerelles matériaux.

- **Gateway**

  On utilise un smartphone comme un gateway (4G/WIFI) pour assurer la transition des données des objets connectés vers la plate-forme ou elles sont analysées et traitées.

- **IOT application protocols**

  Nous utilisons Rest API comme protocole d'application IoT et appliquons la norme JWT pour sécuriser le protocole. JWT nous permet d'ajouter la gestion de l'authentification à notre système.

- **Les plate-formes**

  On parle de (Plate-forme IoT) qu 'est souvent une solution cloud et fog capable de connecter plusieurs objets connectés, stocker leurs données, les traiter, les analyser et les exposer à travers différentes applications. Ces plate-formes IoT permettent aussi de faire communiquer de objets hétérogènes.

  On utilise la plate-forme django au niveau du local comme un fog et un service de cloud pythonanywhere (PaaS).

- **La visualisation**

  On crée une application multiplate-formes en utilisant Flutter qui permet d'exposer les services des objets connectés.

  Un utilisateur, à travers une application mobile ou web...etc., peut par exemple communiquer avec ses objets en consultant leurs données ou en envoyant des actions vers ses objets.

3.6.1 Le prototype de maison intelligente

Nous avons réalisé un prototype d'une maison intelligente. Cette maison se compose de capteurs, actionneurs, panneau solaire, une caméra, un microcontrôleur.

La figure suivante représente prototype de maison intelligente

3.6.2 Fonctionnement du système

Dans ce chapitre nous avons développés l'application **"Amine"** de commande permettant à l'utilisateur de contrôler son système domotique à distance d'une manière fiable et automatique. Aussi, l'application réalisée est multiplate-forme car elle nous permet d'accès au système sous forme page web ou application Android, Ios.

La figure 3.3 présente le logo de l'application.

FIGURE 3.3 – Le logo du système "Amine"

3.6.3 Authentification et accès au système

L'utilisateur se connecte au système tout en introduisant adresse émail ou son numéro téléphone et un mot de passe valides. L'inscription d'un nouveau utilisateur n'est pas un membre de la famille n'est pas disponible à tout le monde. Pour plus de sécurité. Cette tache est gérée par admin système "Amine".

La figure 3.4 présente l'interface d'authentification.

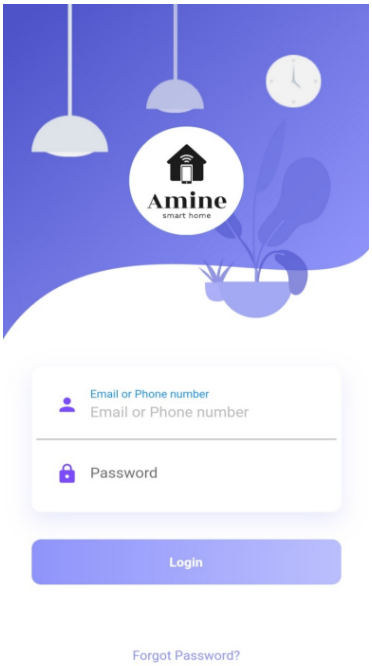


FIGURE 3.4 – Interface d'authentification du système "Amine"

3.6.4 Gestion de la maison

Après une authentification réussite le menu déroulant suivant sert de point de départ à toutes ces interfaces.Ce menu intuitif comprend toutes les fonctions de notre application. L'utilisateur choisit entre les fonctionnalités de l'application (contrôle les lumières de la chambre en clinquant sur le boutton **light**, pour la surveillance de la maison en clinquant sur le bouttonc**amera**, pour gérer les fenêtres en clinquant sur le boutton **window** et la porte **Door**, pour les notifications on a un boutton **cloche**, on ajoute une chambre on appuie sur le boutton+ ....).

La figure (3.5) ci-dessous présente l'accueil.

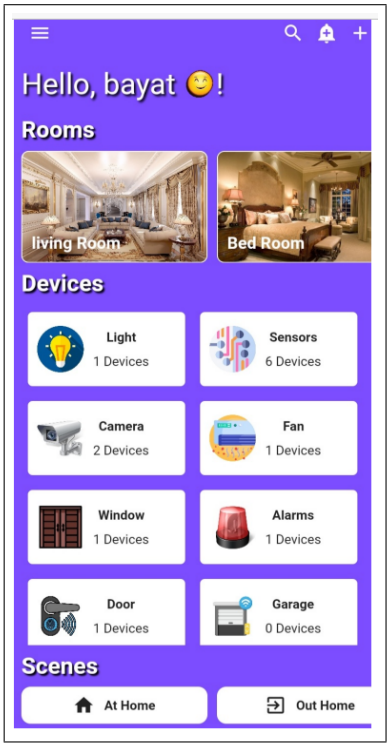


FIGURE 3.5 – Interface d'accueille de système Amine

— **Les informations de profil**

Cette fonctionnalité nous permet d'afficher les informations de profil.

La figure (3.6) ci-dessous présente les informations de profil.

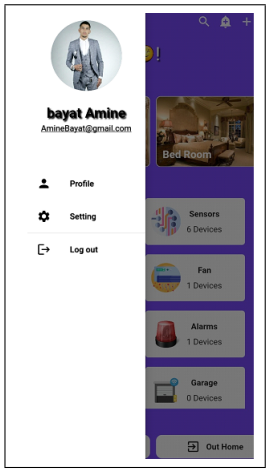


FIGURE 3.6 – Interface des informations de profil

— **Add room**

Cette fonctionnalité nous permet d'insérer une chambre en appuyant sur le bouton +. Ensuite, on saisissant le nom de la chambre et on ajoute sa photo.

La figure (3.7) ci-dessous présente l'ajout d'une chambre.

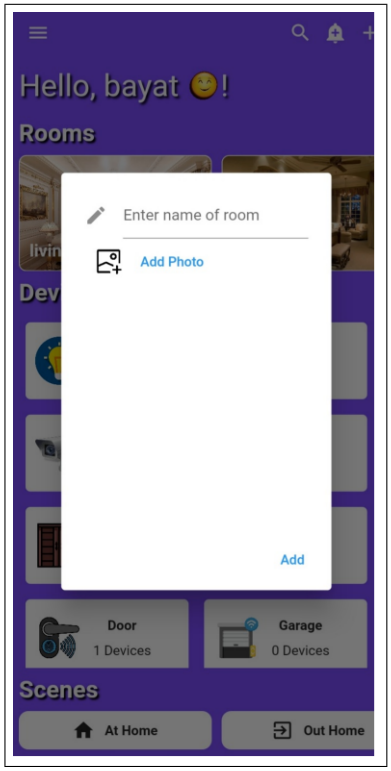


FIGURE 3.7 – Interface d'ajout d'une chambre

— **L'ajout des appareilles**

Lorsque la chambre est ajoutée, on ajoute des appareilles (capteurs, actionneurs) pour contrôler la chambre, y compris le capteur de flamme, les lampes, capteurs de gaz ...etc.

La figure (3.8) ci-dessous présente l'ajout des des appareils dans la chambre.

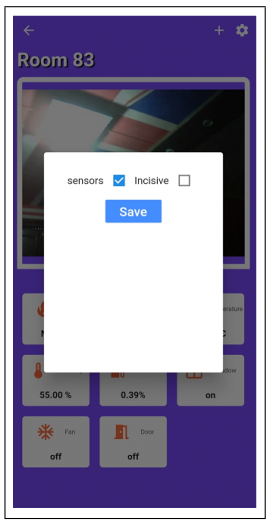


FIGURE 3.8 – Interface d'ajout des devices

Après, les détails de la chambre sont affichés, y compris le feu, l'éclairage (la lampe), la température et l'humidité, le pourcentage de gaz et les états de ventilateur, fenêtre et la porte.

La figure (3.9) ci-dessous présente l'état de la chambre.

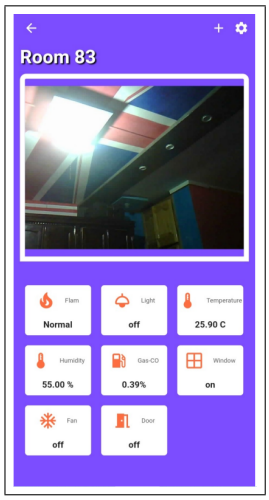


FIGURE 3.9 – Interface d'état de la chambre

— **Contrôle la lampe**

Cette fonctionnalité permet de contrôler les lampes d'une pièce de la maison intelligente allumée ou éteinte. Pour ce faire, on doit clique sur le bouton **light** (ON pour l'éclairage la chambre, OFF pour extinction).

La figure (3.10) ci-dessous présente le contrôle de lampe.

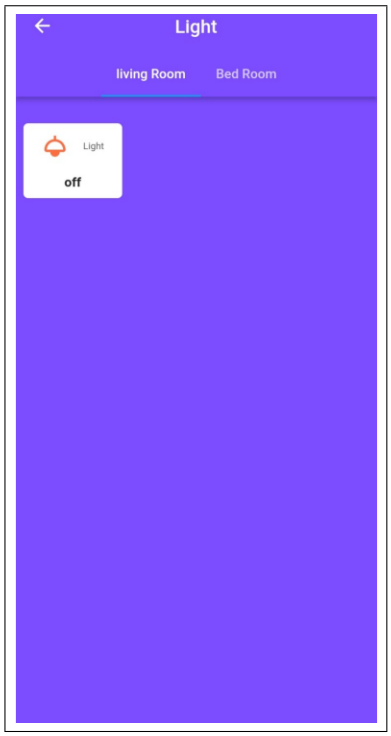


FIGURE 3.10 – Interface de contrôle la lampe

— **Contrôle ventilateur**

Ce interface permet de gérer ventilateur d'une pièce de la maison intelligente fonctionne ou non. Pour ce faire, on doit clique sur le bouton **fan** (ON pour allumer le ventilateur, OFF pour extinction).

La figure (3.11) ci-dessous présente le contrôle de ventilateur.

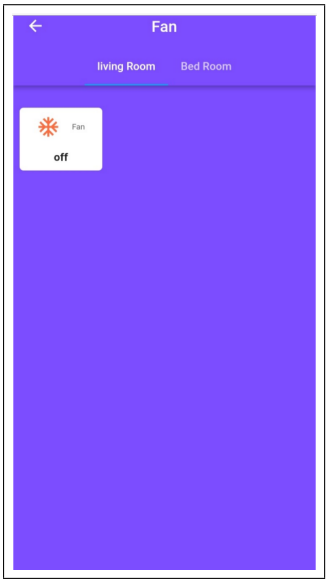


FIGURE 3.11 – Interface de contrôle ventilateur

— **Contrôle fenêtre**

Cette fonctionnalité permet de gérer les fenêtres d'une pièce de la maison intelligente (ouverte ou non). Pour ce faire, on doit clique sur le bouton **Window** (ON pour ouvrir la fenêtre, OFF pour fermer). La figure (3.12) ci-dessous présente le contrôle de la fenêtre d'une pièce.

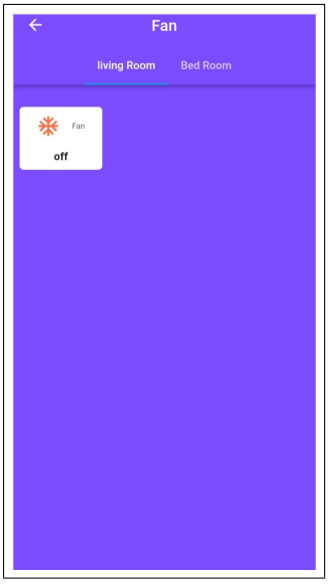


FIGURE 3.12 – Interface de contrôle fenêtre

— **Contrôle la porte**

Cette fonctionnalité permet de contrôler d'une pièce de la maison intelligente ouverte ou non. Pour ce faire, on doit clique sur le bouton **Door** (ON pour ouvrir la porte, OFF pour fermer la porte).

La figure (3.13) ci-dessous présente le contrôle de la porte d'une pièce.

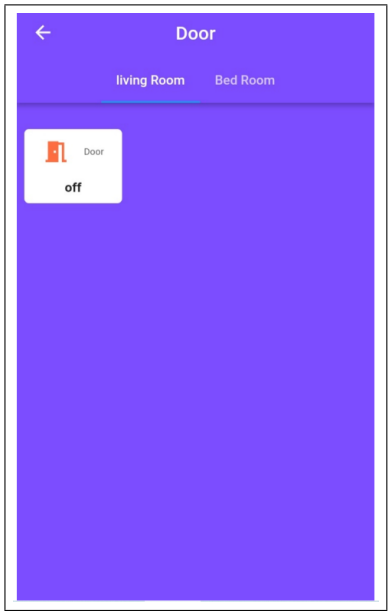


FIGURE 3.13 – Interface de contrôle porte

— **L'ajout de caméra**

On ajoute une caméra dans la chambre en introduisant adresse l'adresse ip de caméra et le nom de la chambre.

La figure (3.14) ci-dessous présente l'ajout de caméra.

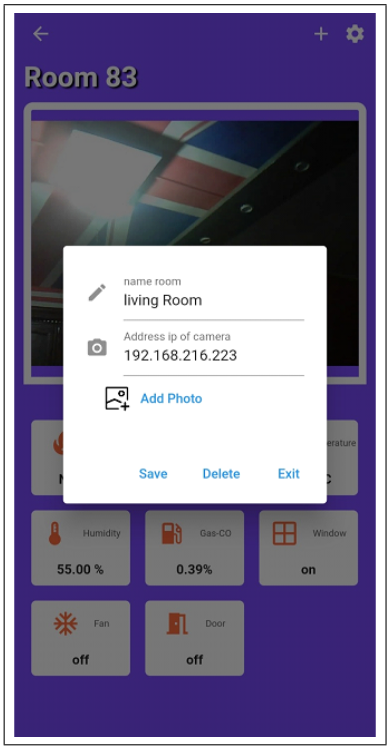


FIGURE 3.14 – Interface d' ajout une caméra

— **Monitoring**

Cette fonction déclenche automatiquement le fonctionnement de la caméra qui transmet les images en temps réel au Smartphone.

La figure (3.15) ci-dessous présente la caméra surveillance dans une pièce.

FIGURE 3.15 – Interface Monitoring

3.6.5 Les notifications

Les notifications sont des alertes qui s'affichent sur votre smartphone afin de vous avertir d'une nouvelle activité.

— **Détection du gaz**

Le système de détecte gaz avec l'envoi d'une notification et une action de déclenchement automatique de Ventilateur et ouvrir la fenêtre avec contrôle à distance du système.

La figure (3.16) ci-dessous présente la détection du gaz.

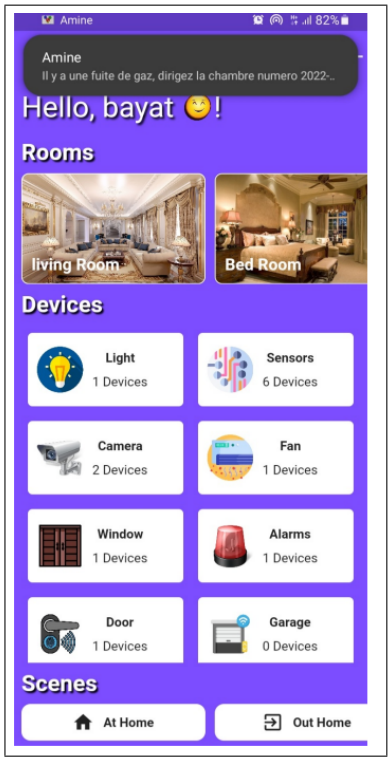


FIGURE 3.16 – Interface détection du gaz

— **Détection d'incendie** Le système de détecte d'incendie avec l'envoi d'une notification et l'ouverture de fenêtre avec contrôle à distance du système.

La figure (3.17) ci-dessous présente la détection d'incendie.

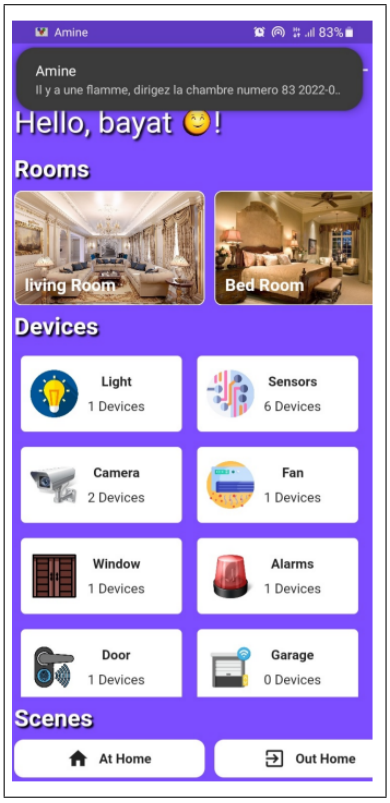
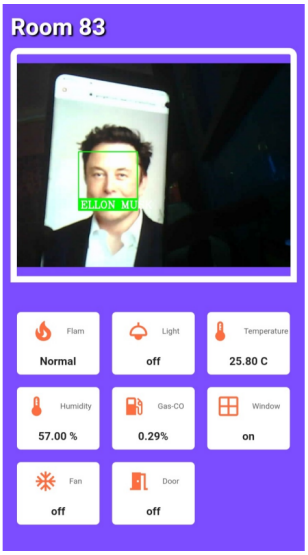
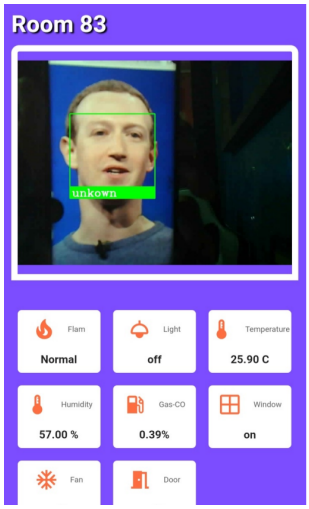


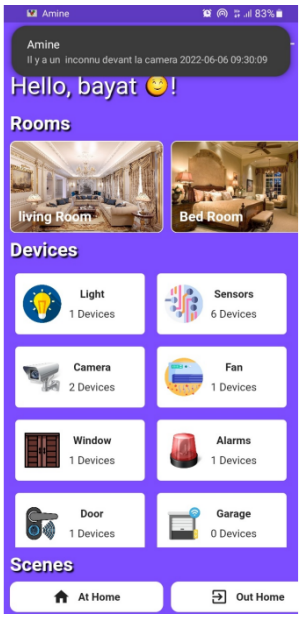
FIGURE 3.17 – Interface de détection d'incendie

— **Détection d'intrusion**

Le système de détecte d'intrusion si la photo de la personne existe dans la base il affiche son nom sinon il envoie un message de notification pour informer qu'il y a un inconnu dans la chambre.







— **Détection de mouvement**

Le capteur va détecte la présente d'une personne. La détection d'une présence va allumer la lampe et le système envoie un message de notification.

La figure (3.19) ci-dessous illustre la détection de mouvement.

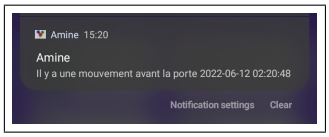


FIGURE 3.18 – Interface notification la détection de mouvement.

— **L'humidité et température**

Le capteur DHT11 capte les données d'humidité et de température de la chambre qui seront affichées dans un message de notification.

Si la l'humidité et la température dépassent les valeurs de seuil la ventilateur déclenche.

La figure (3.19) ci-dessous illustre l'humidité et la température de la chambre.

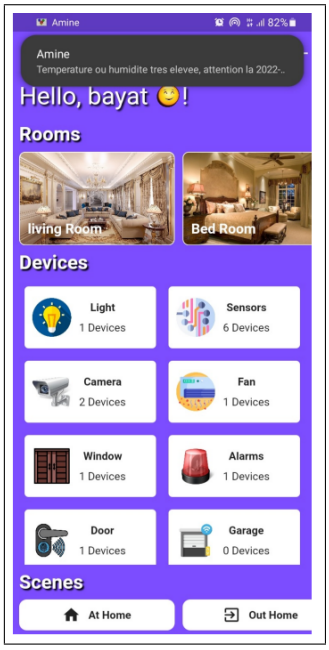


FIGURE 3.19 – Interface notification de l'humidité et la température de la chambre.

3.7 Résultats et discussions

Afin de valider nos objectifs concernant la latence des données transmises (Cloud, Fog) nous avons testé notre système par l'envoie des paquets cryptées, hachées vers le cloud/fog. Après, nous avons générer des tokens et on calcule le temps.

Le tableau suivant 3.1 présente la partie du fog :

Algorithme	Temps De Génération token au niveau de fog	Temps d' envoi de packet au niveau de fog
HS256	240-300 MS	7-12 MS
HS384	240-338 MS	8-16 MS
HS512	240-420 MS	8-18 MS
RS256	350-533 MS	10-18 MS
RS384	356-599 MS	11-20 MS
RS512	462-601 MS	13-36 MS

TABLE 3.1 – Temps d' envoi de packet et temps de génération token au niveau de fog en utilisant différents algorithmes

Le tableau suivant 3.2 présente la partie du cloud :

Algorithme	Temps De Génération token au niveau de cloud	Temps d' envoi de packet au niveau de cloud
HS256	520MS	250-288 MS
HS384	688 MS	290-315 MS
HS512	714 MS	320-366 MS
RS256	947 MS	415-978 MS
RS384	2000 MS	980-1057 MS
RS512	5001 MS	1060-1500 MS

TABLE 3.2 – Temps d' envoi de packet et temps de generation token au niveau de cloud en utilisant différents algorithmes

Nous notons que lors de l'envoi de paquets par le fog ou cloud à l'aide de l'algorithme **Hs-256**, le temps est très court comparé à d'autres algorithmes.Dans le cas de l'utilisation de la **RSA**, le délai est trop long. Nous remarquons la même remarque lors de générations des tokens.

Et à partir de là, nous concluons que l'algorithme HS-256 plus rapide que d'autres algorithmes lors de l'envoi de paquets par le fog ou cloud ou la génération des tokens.

Nous remarquons que lorsque vous envoyez des paquets par le fog, le temps est très court par rapport au Cloud. Dans le cas d'un usage en cloud, le temps est trop long. On constate la même remarque au cours des générations de jetons.

De là, on peut conclure que le fog est plus rapide que le cloud.

3.8 Conclusion

Dans ce chapitre, nous avons présenté la conception et la réalisation de notre système et les différents outils utilisés dans notre développement et la réalisation de ce travail. Enfin, nous avons présenté les résultats obtenus.

Conclusion générale

Au cours des dernières années, plusieurs projets de recherche ont porté sur la réalisation des maisons intelligentes et d'énormes progrès ont été accomplis grâce aux avancées en sécurité et au développement des réseaux de communication.

Néanmoins, il y a encore plusieurs défis à surmonter dans le domaine. Principalement, la sécurité des informations et du réseau, le degré de la latence et la bande passante, et la consommation de l'énergie.

Dans ce mémoire, nous avons conçu et réalisé un système "AMINE" pour la sécurisation des smart homes. Aussi, nous avons créé une application multiplateformes pour l'accès au système (page web ou application mobile Android ou IOS). Aussi, on a présenté la réalisation d'un prototype d'une maison intelligente en considérant divers paramètres : la température, l'humidité ainsi que la luminosité, monitoring, la sécurité ...etc.

Les problèmes que nous avons rencontrés dans notre travail sont : la compatibilité matérielle, difficulté d'électricité ; et d'acquisition du matériel. Aussi malheureusement nous avons perdu quelque de nos capteurs à cause d'un court-circuit.

Dans ce mémoire, nous avons parlé, en premier lieu, sur l'internet des objets (définition, ses domaines d'applications), ainsi qu'une bref présentation de le cloud et fog computing. Dans le deuxième chapitre nous avons présenté les mécanismes de sécurité. Nous avons terminé ce chapitre avec les services de sécurité en particulier l'authentification. Au troisième, nous avons expliqué la conception et la réalisation de notre système puis les résultats obtenus.

Dans la continuité directe de notre travail nous avons ensemble des perspectives.

Comme l'ajout d'un réfrigérateur, un climatiseur intelligent doté d'intelligences artificielles. Introduire l'IA dans le système pour avoir un système intelligent. Aussi, ajouter des règles et des dispositifs plus spécifiques pour la santé. Pour conclure, nous espérons sincèrement que ce mémoire pourra servir de base à de nouvelles études approfondies.

Bibliographie

[1] B. Alain, "futura-sciences." https ://www.futura-sciences.com/, Mis en ligne le 20 avril 2022.

[2] "techno-skills." https ://techno-skills.com/le-piratage-dobjets-iot/. Accessed : 2022-06-2.

[3] T. Wu, F. Wu, J.-M. Redoute, and M. R. Yuce, "An autonomous wireless body area network implementation towards iot connected healthcare applications," *IEEE access*, vol. 5, pp. 11413–11422, 2017.

[4] "waytolearnx." https ://waytolearnx.com/2019/06/protocole-http.html. Accessed : 2022-06-2.

[5] K. S. Aloufi and O. H. Alhazmi, "Performance analysis of the hybrid iot security model of mqtt and uma," *arXiv preprint arXiv :2005.06595*, 2020.

[6] M. A. Tariq, M. Khan, M. T. Raza Khan, and D. Kim, "Enhancements and challenges in coap—a survey," *Sensors*, vol. 20, no. 21, p. 6391, 2020.

[7] W. Jeng, S. DesAutels, D. He, and L. Li, "Information exchange on an academic social networking site : a multidiscipline comparison on researchgate q&a," *Journal of the Association for Information Science and Technology*, vol. 68, no. 3, pp. 638–652, 2017.

[8] R. Maurya, K. A. Nambiar, P. Babbe, J. P. Kalokhe, Y. Ingle, and N. Shaikh, "Application of restful apis in iot : A review," *Int. J. Res. Appl. Sci. Eng. Technol*, vol. 9, pp. 145–151, 2021.

[9] Z. Jin and Y. Chen, "Telemedicine in the cloud era : Prospects and challenges," *IEEE Pervasive Computing*, vol. 14, no. 1, pp. 54–61, 2015.

[10] "Lemagdeladomotique." `https://www.lemagdeladomotique.com/dossier-2-exemples-domaines-application-domotique-maison.html`. Consulté le 10/5/2022.

[11] "Smartgrids." `https://les-smartgrids.fr/smart-city-iot-choix-reseau-1-2/`. Accessed : 2022-05-17.

[12] T. Sylla, *Sécurité et vie privée centrées sur l'utilisateur dans l'IoT*. PhD thesis, Bordeaux, 2021.

[13] Z. Kartit, "Contribution á la sécurité du cloud computing : Application des algorithmes de chiffrement pour sécuriser les données dans le cloud storage," 2016.

[14] "vaadata." https ://www.vaadata.com/blog/fr/jetons-jwt-et-securite-principes-et-cas-dutilisation/. Accessed : 2022-06-6.

[15] "Acervo lima." https ://fr.acervolima.com/difference-entre-le-cloud-computing-et-le-fog-computing/. Accessed : 2022-06-2.

[16] K. Ashton *et al.*, "That 'internet of things' thing," *RFID journal*, vol. 22, no. 7, pp. 97–114, 2009.

[17] I. Lee and K. Lee, "The internet of things (iot) : Applications, investments, and challenges for enterprises," *Business horizons*, vol. 58, no. 4, pp. 431–440, 2015.

[18] L. Dalmasso, *De la vulnérabilité des nœuds capteurs à la certification des transactions sur le réseau, une approche de la sécurisation de l'Internet des Objets*. PhD thesis, Université Montpellier, 2020.

[19] M. Amiriyan, *Ordonnancement de données multiprotocoles des objets connectés dans la maison intelligente*. PhD thesis, école de technologie supérieure, 2021.

[20] R. Bolt, L. Beranek, and R. Newman, "A history of the arpanet : The first decade," *Arlington, VA*, vol. 1, 1981.

[21] N. Sharma, M. Shamkuwar, and I. Singh, "The history, present and future with iot," in *Internet of Things and Big Data Analytics for Smart Generation*, pp. 27–51, Springer, 2019.

[22] R. Want and T. Pering, "System challenges for ubiquitous & pervasive computing," in *Proceedings of the 27th international conference on Software engineering*, pp. 9–14, 2005.

[23] M. U. Farooq, M. Waseem, S. Mazhar, A. Khairi, and T. Kamal, "A review on internet of things (iot)," *International journal of computer applications*, vol. 113, no. 1, pp. 1–7, 2015.

[24] N. Sharma, M. Shamkuwar, and I. Singh, "The history, present and future with iot," in *Internet of Things and Big Data Analytics for Smart Generation*, pp. 27–51, Springer, 2019.

[25] H. Entreprise, "forum.huawei." https ://forum.huawei.com/enterprise/fr/tout-le-monde-apprend-l-iot-premiers-pas-avec-l-iot-1/thread/752173-100389, Mis en ligne le 20 avril 2022.

[26] S. Poolayi and A. H. Assadiyan, "Governance of iot in order to move ahead of the calendar & live the future today," *Journal of Management and Accounting Studies*, vol. 8, no. 3, 2020.

[27] "Acervolima." `https://fr.acervolima.com/capteurs-dans-l-internet-des-objets-iot/`. Accessed : 2022-05-17.

[28] "journaldunet, howpublished = `https://www.journaldunet.fr/web-tech/dictionnaire-de-l-iot/1489525-actionneur-definition-et-fonctionnement-de-l-apparei`l note = Accessed : 2022-05-17."

[29] "info-du-continent." https://info-du-continent.com/microcontroleur-iot-mcu-mises-a-jour-du-marche-2021/. Accessed : 2022-05-17.

[30] "Bloctechno." https://bloctechno.wordpress.com/2020/05/19/le-materiel-et-le-logiciel-pour-faire-de-iot/. Accessed : 2022-05-17.

[31] "matooma." https://www.matooma.com/fr/s-informer/actualites-iot-m2m/architecture. Accessed : 2022-05-17.

[32] C.-L. Zhong, Z. Zhu, and R.-G. Huang, "Study on the iot architecture and gateway technology," in *2015 14th International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES)*, pp. 196–199, IEEE, 2015.

[33] N. Gonzalez, "Architectures protocolaires interoperables pour le reseau de collecte de l'internet des objets," 2020.

[34] F. Souhayla, *l'internet des objets révolutionne notre vie quotidienne : application pour une maison intelligent*. PhD thesis, Universite laarbi tebessi tebessa, 2021.

[35] "Digora." https ://www.digora.com/fr/blog/quest-ce-que-liot-et-pourquoi-mener-une-strategie-diot. Accessed : 21 avril 2022.

[36] D. Zeng, S. Guo, and Z. Cheng, "The web of things : A survey," *J. Commun.*, vol. 6, no. 6, pp. 424–438, 2011.

[37] M. Zhang, F. Sun, and X. Cheng, "Architecture of internet of things and its key technology integration based-on rfid," in *2012 Fifth international symposium on computational intelligence and design*, vol. 1, pp. 294–297, IEEE, 2012.

[38] D. Bandyopadhyay and J. Sen, "Internet of things : Applications and challenges in technology and standardization," *Wireless personal communications*, vol. 58, no. 1, pp. 49–69, 2011.

[39] Y. Zhang, "Technology framework of the internet of things and its application," in *2011 International Conference on Electrical and Control Engineering*, pp. 4109–4112, IEEE, 2011.

[40] M. U. Farooq, M. Waseem, S. Mazhar, A. Khairi, and T. Kamal, "A review on internet of things (iot)," *International journal of computer applications*, vol. 113, no. 1, pp. 1–7, 2015.

[41] M. Wu, T.-J. Lu, F.-Y. Ling, J. Sun, and H.-Y. Du, "Research on the architecture of internet of things," in *2010 3rd international conference on advanced computer theory and engineering (ICACTE)*, vol. 5, pp. V5–484, IEEE, 2010.

[42] A. Cabrera, M. Abad, D. Jaramillo, J. Gómez, and J. C. Verdum, "Definition and implementation of the enterprise business layer through a business reference model, using the architecture development method adm-togaf," in *Trends and Applications in Software Engineering*, pp. 111–121, Springer, 2016.

[43] B. Khoo, "Rfid as an enabler of the internet of things : Issues of security and privacy," in *2011 International Conference on Internet of Things and 4th International Conference on Cyber, Physical and Social Computing*, pp. 709–712, IEEE, 2011.

[44] K. V. S. Rao, "An overview of backscattered radio frequency identification system (rfid)," in *1999 Asia Pacific Microwave Conference. APMC'99. Microwaves Enter the 21st Century. Conference Proceedings (Cat. No. 99TH8473)*, vol. 3, pp. 746–749, IEEE, 1999.

[45] C. M. Roberts, "Radio frequency identification (rfid)," *Computers & security*, vol. 25, no. 1, pp. 18–26, 2006.

[46] K. Domdouzis, B. Kumar, and C. Anumba, "Radio-frequency identification (rfid) applications : A brief introduction," *Advanced Engineering Informatics*, vol. 21, no. 4, pp. 350–355, 2007.

[47] R. Challoo, A. Oladeinde, N. Yilmazer, S. Ozcelik, and L. Challoo, "An overview

and assessment of wireless technologies and co-existence of zigbee, bluetooth and wi-fi devices," *Procedia Computer Science*, vol. 12, pp. 386–391, 2012.

[48] J. T. Vainio *et al.*, "Bluetooth security," in *Proceedings of Helsinki University of Technology, Telecommunications Software and Multimedia Laboratory, Seminar on Internetworking : Ad Hoc Networking, Spring*, vol. 5, 2000.

[49] "commentcamarche." https ://www.commentcamarche.net/contents/108-bluetooth-comment-ca-marche. Accessed : 2022-06-2.

[50] C. M. Ramya, M. Shanmugaraj, and R. Prabakaran, "Study on zigbee technology," in *2011 3rd International Conference on Electronics Computer Technology*, vol. 6, pp. 297–301, IEEE, 2011.

[51] "Réseaux de capteurs sans fils." https ://moodle.utc.fr/file.php/498/SupportWeb/co/Module$_R$CSF
$2022 - 06 - 2$.

[52] A. Zourmand, A. L. K. Hing, C. W. Hung, and M. AbdulRehman, "Internet of things (iot) using lora technology," in *2019 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS)*, pp. 324–330, IEEE, 2019.

[53] "mokolora." https ://www.mokolora.com/fr/what-is-lora-iot/. Accessed : 2022-06-2.

[54] M. U. Farooq, M. Waseem, S. Mazhar, A. Khairi, and T. Kamal, "A review on internet of things (iot)," *International journal of computer applications*, vol. 113, no. 1, pp. 1–7, 2015.

[55] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. Berners-Lee, "Rfc2616 : Hypertext transfer protocol–http/1.1," 1999.

[56] R. Fielding, J. Gettys, *et al.*, "Hypertext transfer protocol-http/1.1," 1998.

[57] P. Urien, "Collaboration of ssl smart cards within the web2 landscape," in *2009 International Symposium on Collaborative Technologies and Systems*, pp. 187–194, IEEE, 2009.

[58] Y. Chen and T. Kunz, "Performance evaluation of iot protocols under a constrained wireless access network," in *2016 International Conference on Selected Topics in Mobile Wireless Networking (MoWNeT)*, pp. 1–7, IEEE, 2016.

[59] S. Elhadi, A. Marzak, N. Sael, and S. Merzouk, "Comparative study of iot protocols," *Smart Application and Data Analysis for Smart Cities (SADASC'18)*, 2018.

[60] "redha." `https://www.redhat.com/fr/topics/api/what-is-a-rest-api`. consulté le 21/5/2022.

[61] D. Kreutz, F. M. Ramos, P. E. Verissimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking : A comprehensive survey," *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14–76, 2014.

[62] S. Housseman, *Modelisation et aide a la decision pour l'introduction de technologies communicantes en milieu hospitalier.* PhD thesis, Ecole Nationale Supérieure des Mines de Saint-Etienne, 2011.

[63] C. Bahhar, C. Baccouche, S. B. Othman, and H. Sakli, "Real-time intelligent monitoring system based on iot," in *2021 18th International Multi-Conference on Systems, Signals & Devices (SSD)*, pp. 93–96, IEEE, 2021.

[64] "Objetconnecte." "https ://www.objetconnecte.com/iot-agriculture-agriculture-4-0". consulté le 5 mai 2022.

[65] S. Benmakhlouf, M. Amarouche, and Y. Chiha, "Commande intelligente de l'éclairage d'une maison," 2020.

[66] V. Rialle, "Villes intelligentes sources d'inspiration," 2017.

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

[68] F. Souhayla, *l'internet des objets révolutionne notre vie quotidienne : application pour une maison intelligent.* PhD thesis, Universite laarbi tebessi tebessa, 2021.

[69] Z.-K. Zhang, M. C. Y. Cho, C.-W. Wang, C.-W. Hsu, C.-K. Chen, and S. Shieh, "Iot security : ongoing challenges and research opportunities," in *2014 IEEE 7th international conference on service-oriented computing and applications*, pp. 230–234, IEEE, 2014.

[70] F. E. Stiftung, "Le renouveau syndical, pilier de l'avenir du travail décent en tunisie," 2020.

[71] L. Wang, J. Tao, M. Kunze, A. C. Castellanos, D. Kramer, and W. Karl, "Scientific cloud computing : Early definition and experience," in *2008 10th ieee international conference on high performance computing and communications*, pp. 825–830, Ieee, 2008.

[72] Y. N. Diédhiou, "Conception et développement sous salesforce d'un module de gestion des dépenses et de la facturation d'un projet.," 2020.

[73] F. Robinet, "Thesis/thèse,"

[74] R. L. Grossman, "The case for cloud computing," *IT professional*, vol. 11, no. 2, pp. 23–27, 2009.

[75] "Anyconnector." `https://anyconnector.com/software-integration/types-of-cloud-computing.html`. Accessed : 2022-05-18.

[76] N. Leavitt, "Hybrid clouds move to the forefront," *Computer*, vol. 46, no. 05, pp. 15–18, 2013.

[77] S. Marston, Z. Li, S. Bandyopadhyay, J. Zhang, and A. Ghalsasi, "Cloud computing—the business perspective," *Decision support systems*, vol. 51, no. 1, pp. 176–189, 2011.

[78] B. LISAN, "Cloud computing," 2011.

[79] U. A. Deshmukh and S. A. More, "Fog computing : a new approach in the world of cloud computing," *Instr Technol*, vol. 49, 2016.

[80] "lebigdata." `https://www.lebigdata.fr/fog-computing-guide-complet`. Accessed : 2022-05-18.

[81] P. Barry, *Head first Python : A brain-friendly guide*. " O'Reilly Media, Inc.", 2016.

[82] J. Barker, *Cloud computing : étude & développement d'une application Serverless*. PhD thesis, Haute école de gestion de Genve, 2021.

[83] Y. Lescopy, "La sécurité informatique," *Post BTS R2i*, vol. 1, no. 6, 2002.

[84] F. HADJI, *Conception et réalisation d'un systme de cryptage pour les images médicales*. PhD thesis, UNIVERSITE MOHAMED BOUDIAF-M'SILA FACULTE DES MATHEMATIQUES ET DE L'INFORMATIQUE, 2018.

[85] D. Boukhlouf, *Une approche à base d'agents mobiles pour la sécurité des systmes d'informations sur le web*. PhD thesis, Université Mohamed Khider-Biskra, 2016.

[86] R. Yende, "Support de cours de sécurité informatique et crypto.," 2018.

[87] J. Fattahi, "Analyse des protocoles cryptographiques par les fonctions témoins," 2016.

[88] N. Mahammedi and H. Mahdadi, *Implémentation de benchmark d'opérations crypto baséesECC pour l'étude et comparaison de courbes elliptiques _ et _*. PhD thesis.

[89] B. Pandey, V. Thind, S. K. Sandhu, T. Walia, and S. Sharma, "Sstl based power efficient implementation of des security algorithm on 28nm fpga," *International Journal of Security and Its Application*, vol. 9, no. 7, pp. 267–274, 2015.

[90] M. Feldhofer, S. Dominikus, and J. Wolkerstorfer, "Strong authentication for rfid systems using the aes algorithm," in *International workshop on cryptographic hardware and embedded systems*, pp. 357–370, Springer, 2004.

[91] L. Christina and V. Joe Irudayaraj, "Optimized blowfish encryption technique," *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 2, no. 7, pp. 5009–5015, 2014.

[92] T. Nie and T. Zhang, "A study of des and blowfish encryption algorithm," in *Tencon 2009-2009 IEEE Region 10 Conference*, pp. 1–4, IEEE, 2009.

[93] A. Said and A. Kahina, *Cryptographie et sécurité des réseaux implémentation de L'AES sous Matlab*. PhD thesis, Université Mouloud Mammeri, 2008.

[94] M. Dumont, *Modélisation de l'injection de faute électromagnétique sur circuits intégrés sécurisés et contre-mesures*. PhD thesis, Université Montpellier, 2020.

[95] S. Chandra, S. Paira, S. S. Alam, and G. Sanyal, "A comparative survey of symmetric and asymmetric key cryptography," in *2014 international conference on electronics, communication and computational engineering (ICECCE)*, pp. 83–93, IEEE, 2014.

[96] P. Mahajan and A. Sachdeva, "A study of encryption algorithms aes, des and rsa for security," *Global Journal of Computer Science and Technology*, 2013.

[97] R. S. Dhakar, A. K. Gupta, and P. Sharma, "Modified rsa encryption algorithm (mrea)," in *2012 second international conference on advanced computing communication technologies*, pp. 426–429, IEEE, 2012.

[98] R. T. J. Caël, "Cryptosysteme hybride rsa-rijndael,"

[99] "Cryptographie et codes secrets." `https://bibmath.net/crypto/index.php?action=affiche&quoi=moderne/rsa`. Accessed : 2022-06-2.

[100] M. B. Yassein, S. Aljawarneh, E. Qawasmeh, W. Mardini, and Y. Khamayseh, "Comprehensive study of symmetric key and asymmetric key encryption algorithms," in *2017 international conference on engineering and technology (ICET)*, pp. 1–7, IEEE, 2017.

[101] M. Kumar, A. Iqbal, and P. Kumar, "A new rgb image encryption algorithm based on dna encoding and elliptic curve diffie–hellman cryptography," *Signal Processing*, vol. 125, pp. 187–202, 2016.

[102] S. Elmoumni, *Implémentation à haute performance de la fonction de hachage cryptographique SHA3 sur des circuits FPGA*. PhD thesis, Hassan II University-Morocco, 2020.

[103] I. B. Damgàard, "A design principle for hash functions," in *Conference on the Theory and Application of Cryptology*, pp. 416–427, Springer, 1989.

[104] K. W. Macharia, "Cryptographic hash functions,"

[105] J. Deepakumara, H. M. Heys, and R. Venkatesan, "Fpga implementation of md5 hash algorithm," in *Canadian Conference on Electrical and Computer Engineering 2001. Conference Proceedings (Cat. No. 01TH8555)*, vol. 2, pp. 919–924, IEEE, 2001.

[106] D. Eastlake 3rd and P. Jones, "Us secure hash algorithm 1 (sha1)," tech. rep., 2001.

[107] H. Gilbert and H. Handschuh, "Security analysis of sha-256 and sisters," in *International workshop on selected areas in cryptography*, pp. 175–193, Springer, 2003.

[108] "Ismag." `https://www.ismag.ma/wp-content/uploads/2020/03/Cryptograhpie-part2-elharfaoui.pdf/`. Accessed : 12-06-2022.

[109] J.-C. Zapalowicz, *Sécurité des générateurs pseudo-aléatoires et des implémentations de schémas de signature à clé publique.* PhD thesis, Rennes 1, 2014.

[110] J. Randimbiarison, "Signature numérique d'un document basée sur fido2," 2020.

[111] A. Okeyinka, "Computational speeds analysis of rsa and elgamal algorithms on text data," in *Proceedings of the world congress on engineering and computer science*, vol. 1, pp. 21–23, 2015.

[112] N. M. S. Iswari, "Key generation algorithm design combination of rsa and elgamal algorithm," in *2016 8th International Conference on Information Technology and Electrical Engineering (ICITEE)*, pp. 1–5, IEEE, 2016.

[113] C. Qi, S. Cui, and L. Hao, "A new threshold signature scheme based on ecc and factoring," in *2009 ISECS International Colloquium on Computing, Communication, Control, and Management*, vol. 4, pp. 550–553, IEEE, 2009.

[114] "securite | codage." `https://www.codage.biz/securite`. Accessed : 2022-05-18.

[115] "Le codage base 64." `https://www.fil.univ-lille1.fr/~wegrzyno/portail/Codage/Doc/TP/TP-Base64/tp-base64001.html`. Accessed : 2022-05-18.

[116] "ionos.fr."

[117] V. Shmatikov, "Security protocols,"

[118] A. Alshamsi and T. Saito, "A technical comparison of ipsec and ssl," in *19th International Conference on Advanced Information Networking and Applications (AINA'05) Volume 1 (AINA papers)*, vol. 2, pp. 395–398, IEEE, 2005.

[119] H. Alshamrani, "Internet protocol security (ipsec) mechanisms," *International Journal of Scientific Engineering Research*, vol. 5, no. 5, pp. 2229–5518, 2014.

[120] R. Malik and R. Syal, "Performance analysis of ip security vpn," *International Journal of Computer Applications*, vol. 8, no. 4, p. 0975, 2010.

[121] W. Qu and S. Srinivas, "Ipsec-based secure wireless virtual private network," in *MILCOM 2002. Proceedings*, vol. 2, pp. 1107–1112, IEEE, 2002.

[122] G. Kambourakis, A. Rouskas, G. Kormentzas, and S. Gritzalis, "Advanced ssl/tls-based authentication for secure wlan-3g interworking," *IEE Proceedings-Communications*, vol. 151, no. 5, pp. 501–506, 2004.

[123] C. J. D'Orazio and K.-K. R. Choo, "A technique to circumvent ssl/tls validations on ios devices," *Future Generation Computer Systems*, vol. 74, pp. 366–374, 2017.

[124] E. El-Emam, M. Koutb, H. Kelash, and O. F. Allah, "An optimized kerberos authentication protocol," in *2009 International Conference on Computer Engineering Systems*, pp. 508–513, IEEE, 2009.

[125] D. Nardjes and D. Nabila, *Mise en place d'un systme de sécurité basé sur le serveur d'authentification TACACS+*. PhD thesis, Université Mouloud Mammeri, 2015.

[126] "Igm.univ-mlv." `http://igm.univ-mlv.fr/~dr/XPOSE2007/jgauth02_RADIUS/802_1x.html`. Accessed : 12-06-2022.

[127] F. Ameur and F. R. Zerrouki, "Conception et realisation d'un systme hybride pour la compression et la sécurisation des documents," 2018.

[128] K. Karkouda, N. Harbi, J. Darmont, and G. Gavin, "Confidentialité et disponibilité des données entreposées dans les nuages," in *9me atelier Fouille de données complexes (EGC-FDC 2012)*, 2012.

[129] A. A. Soofi, I. Riaz, and U. Rasheed, "An enhanced vigenere cipher for data security," *Int. J. Sci. Technol. Res*, vol. 5, no. 3, pp. 141–145, 2016.

[130] "lesassisesdelacybersecurite." `https://www.lesassisesdelacybersecurite.com/Le-blog/Glossaire/Non-Repudiation`. Accessed : 2022-05-15.

[131] J. Petit, *Surcoût de l'authentification et du consensus dans la sécurité des réseaux sans fil véhiculaires*. PhD thesis, Université de Toulouse, Université Toulouse III-Paul Sabatier, 2011.

[132] "Syloe." `https://www.syloe.com/glossaire/authentification`. Accessed : 2022-05-15.

[133] "openclassrooms." `https://openclassrooms.com/fr/courses/`. Accessed : 2022-05-18.

[134] "parasoft." https ://fr.parasoft.com/blog/secure-coding-standards-enforcing-secure-coding-practices-with-sast/. Accessed : 2022-05-18.

[135] "Okta." `https://www.okta.com/fr/`.

[136] "Stack." https ://qastack.fr/programming/1592534/what-is-token-based-authentication. Accessed : 2022-06-6.

[137] "Jwt." `https://jwt.io/introduction`. Accessed : 12-06-2022.

[138] PANGAEA, "logrocket," 2019. Consulté le 20 MAI 2022.

[139] "Openreplay." `https://blog.openreplay.com/jwt-authentication-best-practices`. Accessed : 12-06-2022.

[140] "The startup."

[141] "Medium." `https://dianadarie.medium.com/jwt-authentication-with-sha-and-rsa-307e272f913f`. Accessed : 2022-05-18.

[142] "Tutoriel arduino." https ://arduino-tuto.blogspot.com/2018/08/utiliser-un-capteur-de-mouvement-pir-arduino.html. Accessed : 2022-05-22.

[143] "cours-gratuit."

[144] Z. S. Hanene and B. Lilia, *Réalisation d'une maison intelligente utilisant un module WIFI*. PhD thesis, Faculté des Sciences et Technologies, 2021.

[145] D. Koceila and C. Ramdane, *Conception et réalisation d'un systme de surveillance d'une serre agricole avec une carte Raspberry PI 2*. PhD thesis, Université Mouloud Mammeri, 2017.

[146] L. Boudjadja, M. Kaouane, and A. E. Soukkou, *Conception et simulation de fonctionnement d'une maison intelligente*. PhD thesis, Université de Jijel, 2020.

[147] N. Nicolas, "Etat de l'art et faisabilite technique d'une sonotheque urbaine, rurale et naturelle : Definition des interêts et besoins,"

[148] H. M Machet, *Réalisation et Commande D'un Drone Quadrirotor*. PhD thesis, université Ghardaia, 2021.

[149] "binarytech-dz." `https://binarytech-dz.com/.../cartes-diveres/esp32-cam-avec-ov2640-camera`. Accessed : 03-06-2022.

[150] F. ALAOUI, T. GHAITAOUI, *et al.*, *Etude et réalisation d'une poubelle intelligente alimenté par l'énergie solaire*. PhD thesis, universite Ahmed Draia-ADRAR, 2020.

[151] "mhtronic." `https://mhtronic.com/produit/mb102-module-dalimentation-de-plaque-dessai/`. Accessed : 03-06-2022.

[152] "dzduino." `https://www.dzduino.com/l298-dual-h-bridge-motor-driver-fr`. Accessed : 03-06-2022.

[153] "Automecanik." `https://www.automecanik.com/blog-pieces-auto/auto`. Accessed : 12-06-2022.

[154] "Le disrupteur dimensionnel." `https://www.robot-maker.com/forum/tutorials/article/39-utiliser-une-plaque-dessai-ou-breadboard/`. Accessed : 12-06-2022.

[155] "Microdombot." `https://microdombot.com/fr/led-diodes-electroluminescente/`. Accessed : 12-06-2022.

[156] "Manomano." https://www.manomano.fr/p/motraxx-moteur-a-courant-continu-15-v-dc-055-a-fa1 Accessed : 12-06-2022.

[157] "python definition." https ://techterms.com. Accessed : 19-05-2022.

[158] "developers.google." https ://developers.google.com/learn/topics/dart. Accessed : 2022-05-18.

[159] D. C. Escalera, "Predicción del impacto de regulaciones aéreas sobre los flujos de aeronaves utilizando técnicas de aprendizaje automático : un modelo a nivel de vuelos," 2020.

[160] H. R. Esmaeel, "Apply android studio (sdk) tools," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 5, no. 5, 2015.

[161] N. Zlatanov, "Arduino and open source computer hardware and software," *J. Water, Sanit. Hyg. Dev*, vol. 10, no. 11, pp. 1–8, 2016.

[162] M. Mikolaj, "Using flutter framework in multi-platform application implementation,"

[163] M. L. Hale, D. Ellis, R. Gamble, C. Waler, and J. Lin, "Secu wear : An open source, multi-component hardware/software platform for exploring wearable security," in *2015 IEEE International Conference on Mobile Services*, pp. 97–104, IEEE, 2015.

[164] N. HAOUES, "Correction des fautes d'orthographe par mesure de similarité sémantique entre les mots," 2021.

[165] A. Geitgey, "Face recognition documentation," *Release 1.2*, vol. 3, pp. 3–37, 2019.

[166] C. V. Toulouse, *Conception d'un systme d'acquisition de séquences d'images basé sur le Shape from Focus*. PhD thesis, Autres régions du monde. Université de Bourgogne (UB), FRA. ; Nicéphore Cité ..., 2013.

[167] T. H. Eddine, "La détection d'objet avec opencv et deep learning," 2020.

[168] T. Lambacka and A. Riikonen, "Full stack web-sovellus : React & django," 2021.

[169] M. Keita, "Data science sous python : Algorithme, statistique, dataviz, datamining et machine-learning," 2017.

[170] A. Martelli, *Python en concentré*. O'Reilly Media, Inc., 2004.