REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEURE ET DE LA RECHERCHE SCIENTIFIQUE

**UNIVERSITE IBN KHALDOUN - TIARET**

# MEMOIRE

Présenté à :

FACULTÉ DES MATHEMATIQUES ET DE l'INFORMATIQUE
DÉPARTEMENT D'INFORMATIQUE

Pour l'obtention du diplôme de :

MASTER

Spécialité : **Réseaux et Télécommunications**

**Par** :

REBAH Haroune Rachid et RAHMANI Nabil

**Sur le thème**

---

# Méthode de détection d'anomalies dans les réseaux de capteurs sans fil basée sur les séries temporelles

---

Soutenu publiquement à Tiaret devant le jury composé de :

| | | | |
|---|---|---|---|
| Mr **Bengheni Abdelmalek** | **M.C.B** | Université Ibn Khaldoun Tiaret | Président |
| Mr **BAKAR KHALED** | **M.A.A** | Université Ibn Khaldoun Tiaret | Encadrant |
| Mr **ALEM Abdelkader** | **M.A.A** | Université Ibn Khaldoun Tiaret | Examinateur |

2022-2023

# REMERCIEMENTS

*Nous tenons à remercier Dieu Tout-Puissant de nous avoir donné la volonté et la patience pour accomplir ce travail.*

*Tout d'abord, nous souhaitons remercier notre* encadreur *Monsieur BAkkAR.k pour son soutien, ses conseils éclairés et son accompagnement tout au long de ce projet.*

*Nous tenons également à exprimer notre gratitude envers les membres du jury qui ont accepté d'évaluer ce travail.*

*Nos remerciements vont également à tous les enseignants du département Informatique qui nous ont suivi au long de notre cycle d'études.*

*Nous exprimons nos profondes gratitudes à nos parents pour leurs encouragements et leur soutien.*

*Enfin, nous exprimons notre reconnaissance envers nos proches, nos amis pour leur soutien constant.*

# Résumé

Au cours des dernières années, l'utilisation des réseaux de capteurs sans fil (RCSF) s'est répandue et a suscité un intérêt croissant dans la communauté scientifique et industrielle. Cependant, ces réseaux sont confrontés à de nombreux défis en termes de sécurité. Les RCSF opèrent souvent dans des environnements non surveillés, ce qui les rend vulnérables à diverses attaques. La détection d'anomalies dans les RCSF revêt une importance capitale pour garantir leur bon fonctionnement et leur intégrité. Les chercheurs ont donc cherché à mettre en place des solutions de sécurité efficaces pour protéger ces réseaux. Dans cette étude, nous nous sommes concentrés sur la détection d'attaques par déni de service (DoS) dans les RCSF en utilisant des séries temporelles et des algorithmes d'apprentissage automatique. Pour ce faire, nous avons utilisé la bibliothèque PyCaret qui intègre des fonctionnalités avancées pour la détection d'anomalies, nous avons utilisé un ensemble de données spécialement conçu pour la détection d'attaques DoS dans les RCSF, appelé WSN-DS (Wireless Sensor Network Détection System). Cet ensemble de données comprend différentes attaques telles que Blackhole, Grayhole, Flooding et TDMA. En utilisant PyCaret, nous avons appliqué des algorithmes tels que l'IForest isolation, les K-plus proches voisins et le Local Outlier Factor pour identifier les comportements anormaux dans les séries temporelles générées par les capteurs.

Les résultats de notre étude peuvent être présentés sous forme de graphes de séries temporelles, permettant une visualisation claire des anomalies détectées. Ces résultats démontrent que la combinaison entre les séries temporelles et les algorithmes d'apprentissage automatiques peut être plus efficace dans le domaine de la détection d'anomalies dans les RCSF.

**Mots-clés :** réseaux de capteurs sans fil, les séries temporelles, La détection d'anomalies, la bibliothèque PyCaret, attaque DOS, WSN-DS, algorithme d'apprentissage automatique.

# Abstract

In recent years, the use of wireless sensor networks (WSNs) has become widespread and has attracted increasing interest in the scientific and industrial community. However, these networks face many security challenges. RCSFs often operate in unmonitored environments, which makes them vulnerable to various attacks. The detection of anomalies and attacks in the RCSF is of paramount importance to guarantee their proper functioning and their integrity. Researchers have therefore sought to implement effective security solutions to protect these networks. In this study, we focused on the detection of Denial of Service (DoS) attacks in RCSF using time series and machine learning algorithms. To do this, we used the PyCaret library which integrates advanced features for anomaly detection, we used a dataset specially designed for the detection of DoS attacks in RCSFs, called WSN-DS (Wireless Sensor Network System Detection). This data set includes different attacks such as Blackhole, Grayhole, Flooding and TDMA. Using PyCaret, we applied algorithms such as IForest isolation, K-nearest neighbors, and Local Outlier Factor to identify anomalous behaviors in sensor-generated time series.

The results of our study can be presented in the form of time series graphs, allowing a clear visualization of the anomalies detected. These results demonstrate that the combination between time series and machine learning algorithms can be more effective in the field of attack and anomaly detection in RCSF.

**Keyword:** wireless sensor networks, time series, anomaly detection, PyCaret library, DOS attack, WSN-DS, machine learning algorithm.

# ملخص

في السنوات الأخيرة ، انتشر استخدام شبكات الاستشعار اللاسلكية (WSNs) وجذب اهتمامًا متزايدًا في المجتمع العلمي والصناعي. ومع ذلك ، فإن هذه الشبكات تواجه العديد من التحديات الأمنية. غالبًا ما تعمل RCSF في بيئات غير خاضعة للرقابة ، مما يجعلها عرضة لهجمات مختلفة. يعد اكتشاف الانحرافات والهجمات في RCSF ذا أهمية قصوى لضمان حسن سيرها وسلامتها. لذلك سعى الباحثون إلى تنفيذ حلول أمنية فعالة لحماية هذه الشبكات. في هذه الدراسة ، ركزنا على اكتشاف هجمات رفض الخدمة (DoS) في RCSF باستخدام السلاسل الزمنية وخوارزميات التعلم الآلي. للقيام بذلك ، استخدمنا مكتبة PyCaret التي تدمج الميزات المتقدمة لاكتشاف الشذوذ ، واستخدمنا مجموعة بيانات مصممة خصيصًا لاكتشاف هجمات DoS في RCSFs ، تسمى WSN-DS (اكتشاف نظام شبكة الاستشعار اللاسلكي). تتضمن مجموعة البيانات هذه هجمات مختلفة مثل Blackhole و Grayhole و Flooding و TDMA. باستخدام PyCaret ، طبقنا خوارزميات مثل IForest isolation و Local Outlier Factor و K-nearest neighbors لتحديد السلوكيات الشاذة في السلاسل الزمنية التي يولدها المستشعر.

يمكن تقديم نتائج دراستنا في شكل رسوم بيانية متسلسلة زمنية ، مما يسمح بتصور واضح للحالات الشاذة المكتشفة. توضح هذه النتائج أن الدمج بين السلاسل الزمنية وخوارزميات التعلم الآلي يمكن أن يكون أكثر فاعلية في مجال اكتشاف الشذوذ فيRCSF .

**الكلمات المفتاحية:** شبكات الاستشعار اللاسلكية ، السلاسل الزمنية ، كشف الشذوذ ، مكتبة PyCaret ، هجوم DOS، WSN-DS ، خوارزمية التعلم الآلي.

# Sommaire

## Table des matières :

# Liste des figures

# Liste des tableaux

# Liste des acronymes

- **WSN**: Wireless Sensor Networks.
- **BS**: Base Station.
- **CH**: Cluster Head.
- **DOS** : Denial Of Service.
- **HIDS**: Host based Intrusion Detection System.
- **ICMP** : Internet Control Message Protocol.
- **IDS** : Intrusion Détection System.
- **IP** : Internet Protocol.
- **K-NN**: K- Nearest Neighbor.
- **LOF**: local outlier factor.
- **LEACH**: Low-Energy Adaptive Clustering Hierarchy.
- **ML**: Machine Learning.
- **NIDS**: Network based Intrusion Detection System.
- **RAM**: Random Access Memory.
- **RCSFs** : Réseaux de Capteurs Sans Fil.
- **TCP** : Transmission Control Protocol.
- **TDMA**: Time-Division Multiple Access

# Introduction générale

## Introduction générale :

Les réseaux de capteurs sans fil (Wireless Sensor Networks ; WSN) sont des réseaux constitués de nombreux capteurs de petite taille, déployés près des objets qu'ils surveillent. Ces capteurs sont autonomes dans la collecte, le traitement et l'acheminement des données environnementales vers des stations de collecte appelées nœuds puits (Sink) ou stations de base (BS). Ces nœuds peuvent être connectés à l'utilisateur via Internet ou un satellite, permettant ainsi la requête et la récupération des données environnementales spécifiques via les nœuds puits.

Cette technologie trouve des applications variées dans les domaines militaire, environnemental, de la santé, des bâtiments, du transport et médical. Cependant, Les RCSF sont vulnérables à diverses attaques qui menacent leur fonctionnement normal. La protection de ces réseaux est donc essentielle pour garantir leur intégrité et leur disponibilité. Cependant, les mécanismes de lutte contre les attaques dans les RCSF sont nombreux et complexes.

Dans ce contexte, nous avons utilisé une méthode une méthode de détection d'anomalies basée sur les séries temporelles pour renforcer la sécurité des RCSF. En utilisant des techniques avancées de Machine Learning, nous identifions les comportements anormaux à travers les informations collectées au sein du réseau à savoir la consommation d'énergie, les paquets envoyés, etc…, permettant ainsi une détection précoce des attaques.

Le reste du rapport est organisé comme suit :

Dans le chapitre 1 : nous présentons en détail les réseaux de capteurs sans fil, en expliquant leurs architectures, leurs caractéristiques ainsi que leurs domaines d'application. Nous abordons également les architectures de communication utilisées dans ces réseaux, ainsi que quelques systèmes d'exploitation spécifiques à cette technologie.

Dans le chapitre 2 : nous introduisons la sécurité des réseaux de capteurs sans fil, en discutant les objectifs de sécurité, les attaques courantes contre ces réseaux, les mécanismes de sécurité disponibles et les systèmes de détection d'intrusion.

Enfin, dans le dernier chapitre, nous avons utilisé une méthode de détection d'anomalies pour les réseaux de capteurs sans fil. Nous utilisons Pycaret une bibliothèque d'apprentissage automatique non supervisé qui effectue la tâche d'identifier les anomalies dans des séries temporelles. Nous présentons les résultats obtenus sur un ensemble de données spécifique aux attaques dans les RCSF. Ces résultats fournissent des informations précieuses sur les anomalies détectées, contribuant ainsi à renforcer la sécurité des réseaux de capteurs sans fil. Ce travail est terminé par une conclusion générale et des perspectives.

# I. Chapitre I : Généralités sur les Réseaux de Capteurs sans fil

# I.1 Introduction :

Dans le domaine des réseaux informatiques, de nombreuses nouvelles technologies ont vu le jour, notamment les réseaux de capteurs sans fil. Un réseau de capteurs sans fil (RCSF) est un réseau coopératif composé d'un ensemble de nœuds autonomes déployés dans la zone.

Le rôle de ces nœuds capteurs c'est la détection des mesures physiques, de les convertir en un signal numérique, et de les transmettre à une station de base pour un traitement plus approfondi, cette station est une interface entre le réseau (RCSF) et l'utilisateur. La collection de ces données à partir du réseau est relié aux contraintes de ressources des nœuds capteurs qui sont limitées (la capacité de calcule et de mémoire et l'énergie disponible sur la batterie de nœud capteur).

Les réseaux de capteurs sans fil RCSF ont un champ d'application vaste et diversifié. Ceci est rendu possible par leur cout faible, leur taille réduite, le support de communication sans fil utilisé et la large gamme des types de capteurs disponibles. Un autre avantage est la possibilité de s'auto-organiser et d'établir des communications entre eux sans aucune intervention humaine, notamment dans des zones inaccessibles ou hostiles, ce qui accroit davantage le nombre de domaines ciblés par leur application (bâtiments intelligents, l'industrie, médicale, militaire, transport, domestique…etc. [1]

# I.2 Historique des réseaux de capteurs sans fil :

Dans les années 1990, dans le monde de la recherche, est apparue une idée qui paraissait plutôt un rêve pour cette époque : imaginer un système nerveux central pour la Terre, capable de surveiller en temps réel les événements, ayant comme principaux bénéfices de pouvoir empêcher les accidents et d'économiser l'énergie. (Cette poussière intelligente a mis longtemps à apparaître) dit le professeur Pister, de l'Université de Californie à Berkeley. [2]

Aujourd'hui les réseaux de capteurs sont devenus des systèmes pouvant atteindre un très grand nombre de nœuds, avec une zone de couverture déterminée et déployés d'une manière plus ou moins dense dans un environnement hétérogène dont on mesure ainsi son état global. Les derniers progrès en termes de miniaturisation, ainsi que le remplacement du câblage classique par des technologies de communication radio, ont généré de nouvelles catégories d'applications qui visent de nombreux domaines : l'aéronautique, l'automobile, le médical,

l'environnement…, etc. De plus, les progrès des communications sans fil permettent aujourd'hui de répondre aux exigences peu envisageables auparavant. [3]

Dans ce qui suit, nous présentons une évolution au cours du temps des capteurs les plus utilisés ces dernières années.


***Figure I-1: Evolution des capteurs [4]***

Les réseaux de capteurs sans fil (RCSF) ont montré leur impact sur notre vie quotidienne. Les auteurs ont classé le processus de création de capteurs en trois générations. Le tableau I-1 illustre cette catégorisation des capteurs en des générations.[5]

**Tableau *I* -1: Génération des réseaux de capteurs**

| Génération | Période | Taille | Poids | Batterie |
|---|---|---|---|---|
| 1er | Les années 80 et 90 | Grande boîte à chaussures | Kilogrammes | Grosse |
| 2e | 2000-2003 | Boîte de cartes | Grammes | AA |
| 3e | 2010 | Particule de poussière | Négligeable | Solaire |

## I.3 Les réseaux de capteurs sans fil :

Les réseaux de capteurs sans-fil sont constitués d'un ensemble de nœuds capteurs, possédant des ressources particulièrement limitées, mais qui leur permettent néanmoins d'acquérir des données sur leur environnement immédiat, de les traiter et de les communiquer.

### I.3.1 Nœud Capteur sans fil :

Les nœuds capteurs sont des dispositifs de taille extrêmement réduite avec des ressources très limitées, autonomes, capable de traiter des informations et de les transmettre, via les ondes radio, à une autre entité (nœud capteurs, unité de traitements...) sur une distance limitée à quelques mètres.[6]

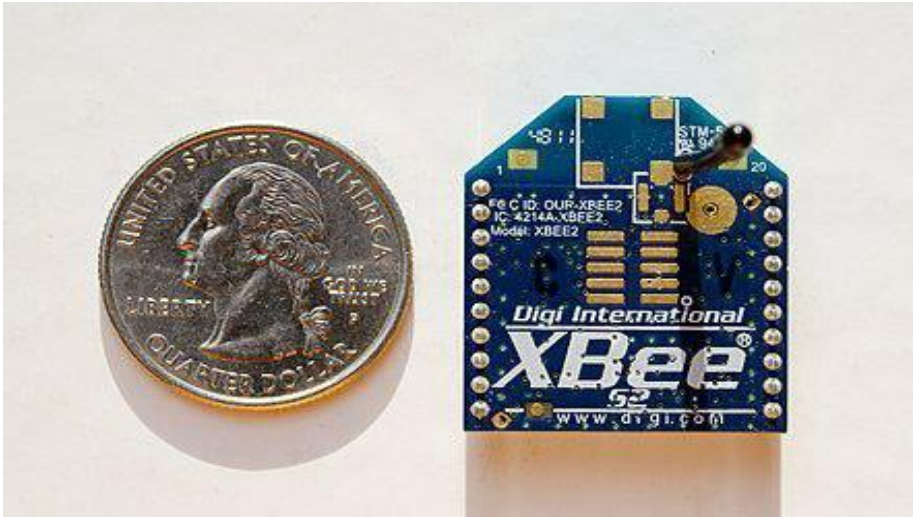


***Figure I-2*: exemple d'un capteur Zigbee**

### I.3.2 Type de capteurs :

Il existe plusieurs types de capteurs, avec des fonctionnalités diverses et variées. La plupart des capteurs dépendent de l'application pour laquelle ils ont été conçus (capteurs aquatiques, sous-terrain, etc..). La figure I.3 **[7]** illustre l'évolution des capteurs au cours de ces 20 dernières années. Cette

représentation met en avant l'importance des travaux de recherche de l'université de Berkeley dans l'essor des réseaux de capteurs, surtout sachant que l'entreprise Xbow2 (aussi appelé Crossbow) qui fait jusqu'à aujourd'hui office de référence dans la fabrication de capteurs.

Les capteurs fabriqués par Xbow au cours des dix dernières années (famille de capteurs Mica et Telos) sont sans aucun doute les plus utilisés dans les expériences et travaux de recherche. Ces capteurs sont capables de mesurer plusieurs métriques et utilisent Chipcon CC2420 qui est devenu le standard au niveau des modules de transmission utilisant le protocole de communication IEEE 802.15..4 **[8]**.


***Figure I -3*: Evolution des capteurs.**

## I.4 Architecture de capteur sans fil :

Un capteur est composé de quatre éléments de base : une unité de perception, de traitement, de communication et une unité de contrôle d'énergie (batterie) **[9]**.

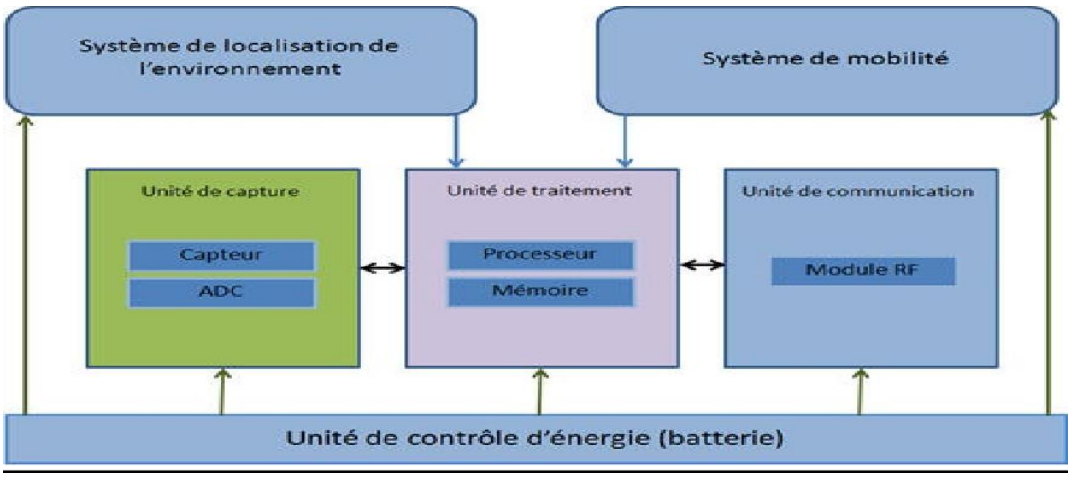


***Figure I -4*: Architecture d'un nœud capteur [10].**

### I.4.1 Le nœud capteur :

Un nœud capteur est composé de plusieurs modules dont chacun d'eux a une tâche particulière : acquisition, traitement, et transmission de données. Il comprend également une source d'énergie **[11]**.

### I.4.2 L'unité de traitement :

Elle dispose de deux interfaces, une interface pour l'unité d'acquisition et une interface pour l'unité de transmission. Cette unité comprend un processeur associé généralement à une petite unité de stockage et fonctionne à l'aide d'un système d'exploitation spécialement conçu pour les micro-capteurs (TinyOS par exemple) **[12]**.

### I.4.3 L'unité de transmission :

L'unité de transmission est responsable de toutes les émissions et réceptions de données via un support de communication radio **[13].**

### I.4.4 unité de contrôle d'énergie :

C'est le système le plus important au niveau du nœud, Le capteur doit disposer de sa propre source d'énergie qui alimente le reste des unités, cette unité se trouve généralement sous la forme de batterie de basse tension non renouvelable, de ce fait, la duréede vie du capteur dépend complètement de cette unité **[14].**

## I.5 Architecture d'un réseau de capteurs :

Un réseau de capteurs sans fil (RCSF), ou *Wireless Sensor Network (WSN*) est composé de centaines ou de milliers de capteurs. Ces appareils, appelés en anglais ***Motes***, sont alimentéspar des piles et sont typiquement déployés de façon plus ou moins aléatoire dans une zone géographique appelée zone de captage, ou zone d'intérêt. Généralement, ces capteurs font des mesures périodiques et utilisent une communication sans fil pour acheminer les données collectées à un dispositif plus puissant appelé nœud puits (Sink), ou station de base (base station), qui les traite en calculant par exemple leur maximum, moyenne ou médiane **[15]**. Ensuite, La station de base transmet ces données par Internet ou par satellite à l'ordinateur central pour analyser ces données et prendre des décisions. La figure suivante présente un RCSF.

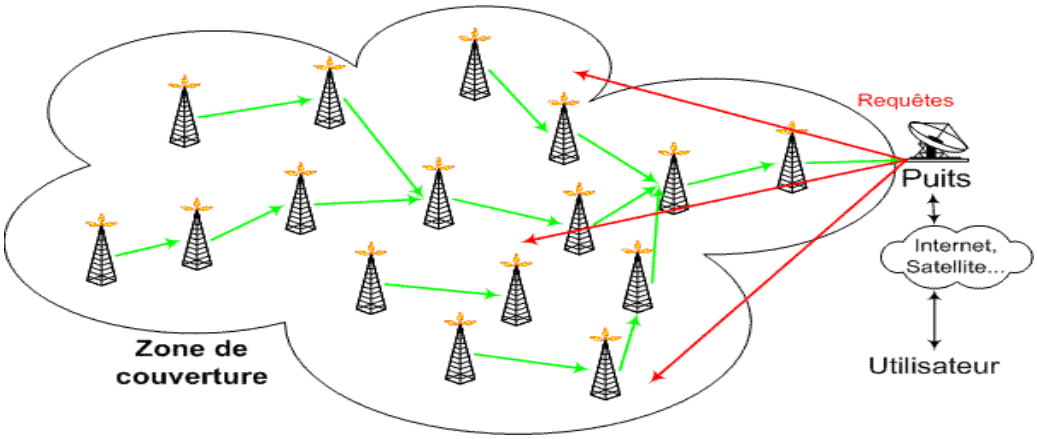


***Figure I -5: Réseaux de capteurs sans fil***

## I.6 Les principales caractéristiques et limites des RCSF :

Un capteur sans fil est doté des caractéristiques suivantes **[16]** :

- Capable de calculer.
- Capable de communiquer.
- Capte toujours.
- Préposition / déploiement aléatoire.
- Limitation de la durée de vie des batteries.
- Densité (petit / grand nombre).
- La rapidité : c'est le temps de réaction d'un capteur entre la variation de la grandeur physique qu'il mesure et l'instant ou l'information prise en compte par la partie commande.
- L'étendue de la mesure : c'est la différence entre le plus petit signal détecte et le plus grand perceptible sans risque de destruction pour le capteur.
- La sensibilité : c'est la plus petite variation d'une grandeur physique que peut détecter un capteur.

En analysant la gamme des composants disponibles sur le marché et les prototypes présents dans la littérature, il est évident que la principale caractéristique d'un nœud de capteurs sans fil est sa *petite taille*. Une deuxième caractéristique, évidente mais essentielle, est l'*autonomie* (pas seulement du point de vue de leur source d'énergie, maisaussi de leur fonctionnement). Ces deux premières particularités induisent plusieurs autres caractéristiques à considérer, en particulier la vitesse de calcul et la vitesse de transmission. Des performances élevées en termes de vitesse de traitement et de transmission impliquent une consommation d'énergie élevée.

De manière générale, il est souhaitable que la durée de vie de la batterie d'un capteur soit la plus longue possible, donc les différentes unités qui composent un capteur sont généralement très limitées en termes de ressources et de performance pour que leur consommation d'énergie soit extrêmement faible.

## I.7 Architecture protocolaire :

Le rôle de ce modèle consiste à standardiser la communication entre les composantsdu réseau afin que différents constructeurs puissent mettre au point des produits (logiciels ou matériels) compatibles. Ce modèle comprend 5 couches qui ont les mêmes fonctions que celles du modèle OSI ainsi que 3 couches pour la gestion de lapuissance d'énergie, la gestion de la mobilité ainsi que la gestion des tâches (interrogation du réseau de capteurs). Le but d'un système en couches est deséparer le problème en différentes parties (les couches) selon leur niveau d'abstraction. Chaque couche du modèle communique avec une couche adjacente (celle du dessus ou celle du dessous). Chaque couche utilise ainsi les services des couches inférieures et en fournit à celle de niveau supérieur. **[17].**

### I.7.1 La pile protocolaire dans un RCSF :

Afin d'améliorer la robustesse du réseau, l'architecture en couches est approuvée, en vuede la mise en place effective du RCSF.

La pile protocolaire dans les RCSF comprend cinq couches à utiliser par les nœuds du réseau telles que : la couche application, la couche transport, la couche réseau, la couche liaison de données et la couche physique ainsi que trois plans de gestion : le gestionnaire d'énergie, le gestionnaire de mobilité et le gestionnaire des tâches **[18]**.

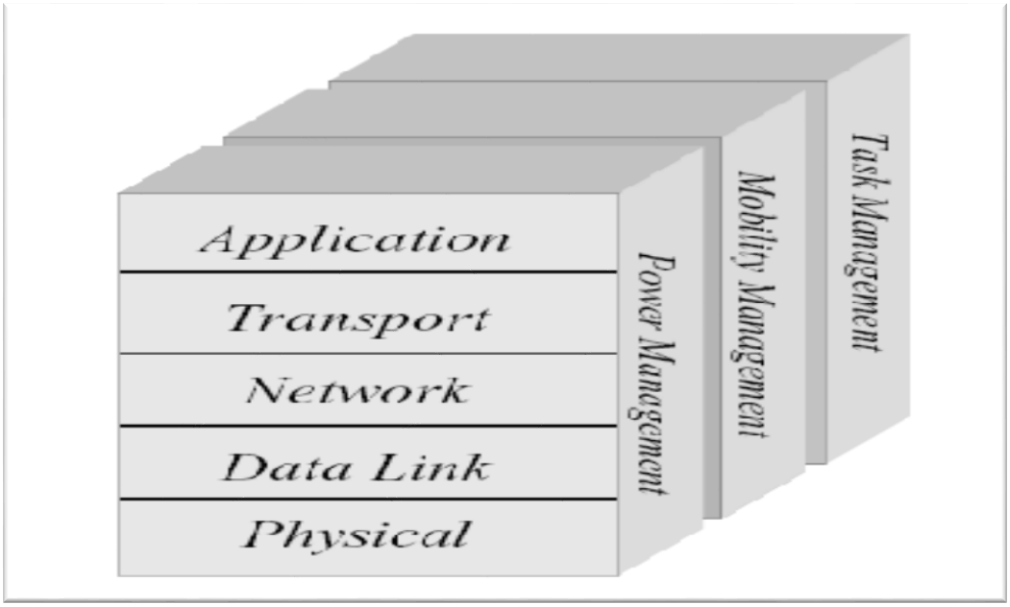


***Figure I -6*: Pile protocolaire d'une architecture de réseau de senseurs [19].**

### *I.7.2* Couche Physique :

- Responsable du support d'acheminant les données envoyées entre les nœuds.
- Il existe deux types de médias pouvant être utilisés pour les réseaux de capteurs : les ondes infrarouges et les ondes radiofréquences.**[20]**

### I.7.3 Couche liaison :

- Responsable du multiplexage des flux des données, du contrôle d'accès au media
- Du contrôle des erreurs
- Le protocole permet au nœud d'envoyer ses donnéesen étant assuré que ses voisins ont été correctement synchronisés et réduisant les délais d'accès au médium, et aussi permet également aunœud de ne pas saturer sa file d'attente. **[20]**

### I.7.4 Couche réseau :

Dans la couche réseau le but principal est de trouver une route et une transmission fiable des données, captées, des nœuds capteurs vers le puits "*Sink*" en optimisant l'utilisation de l'énergie des capteurs. Ce routage diffère de celui des réseaux de transmission ad hoc sans fils par les caractéristiques suivantes :

- Il n'est pas possible d'établir un système d'adressage global pour le grand nombre de nœuds.
- Les applications des réseaux de capteurs exigent l'écoulement des données mesurées de sources multiples à un puits particulier.
- Les multiples capteurs peuvent produire de mêmes données à proximité d'unphénomène (redondance).

Les nœuds capteur exigent ainsi une gestion soigneuse des ressources. En raison de ces différences, plusieurs nouveaux algorithmes ont été proposés pour le problème de routage dans les réseaux de capteurs **[21].**

### I.7.5 Couche transport :

Cette couche est chargée du transport des données, de leur découpage en paquets, du contrôle de flux, de la conservation de l'ordre des paquetset de la gestion des éventuelles erreurs de transmission. **[21]**

### I.7.6 La couche application :

Cette couche assure l'interface avec les applications. Il s'agit donc du niveau le plus proche des utilisateurs, géré directement par les logiciels. **[21]**

## I.8 Les niveaux de gestion dans les réseaux de capteurs sans fil :

Les niveaux de gestion propres aux réseaux de capteurs sans fil sont les suivants :

- **Le niveau de gestion d'énergie** : ce niveau qui gère l'énergie consommée par les capteurs.
- **Le niveau de gestion de la mobilité** : ce niveau détecte et enregistre le mouvement des nœuds capteurs et permet de maintenir l'itinéraire d'un capteur vers un utilisateur et garder la trace de l'emplacement de ses voisins.
- **Le niveau de gestion des taches** : Balance les tâches entre les nœuds afin d'économiserde l'énergie

## I.9 Quelques applications de RCSF

Nous présentons dans ce qui suit quelques applications récentes, plus évoluées et plus importantes :

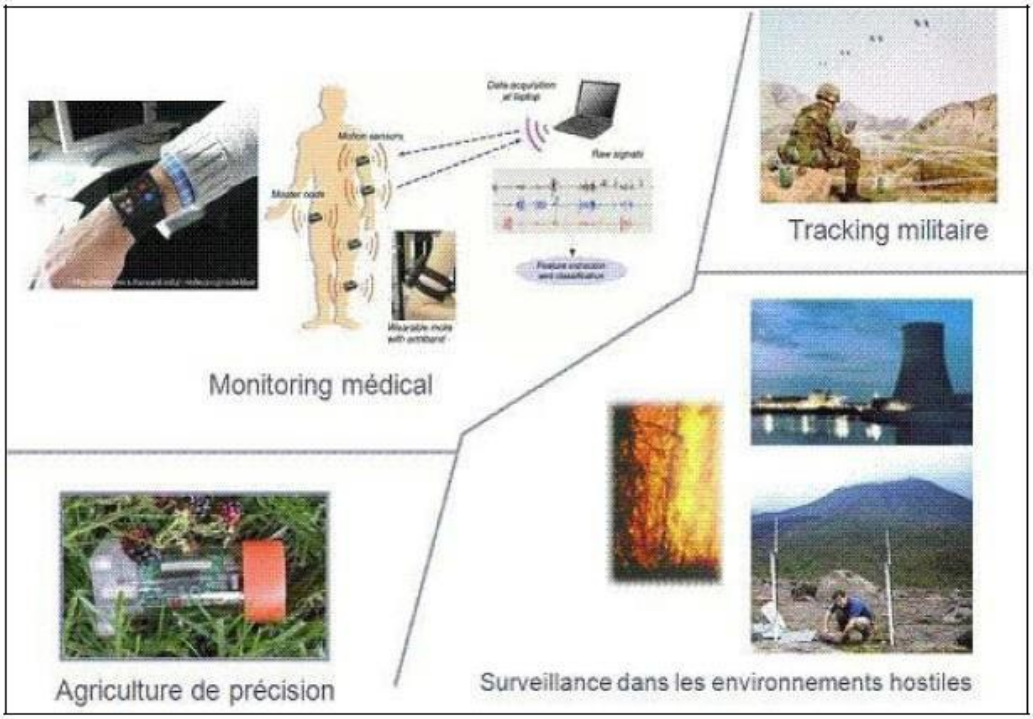


***Figure I -7*:Applications des réseaux de capteurs**

- **Le bâtiment :** L'installation d'un RCSF à l'intérieur d'un bâtiment permet de remédier efficacement au problème de perte d'énergie (mauvaise ventilation, mauvais usage d'air conditionné, etc.). Ils sont utilisés généralement pour un meilleur contrôle de la température et de l'humidité, ce qui augmente le niveau de confort des habitants.**[22].**
- **Surveillance des machines industrielles :** L'idée est de fixer des capteurs sur des endroitsdifficiles d'accès afin de détecter des événements qui indiquent le besoin de maintenance (vibration, fumée, bruits et nuisances etc.).**[22]**
- **Médicales :** Dans le domaine de la médecine, les RCSFs peuvent être utiliséspour assurer une surveillance permanente des organes vitaux de l'être humain grâce à des micro-capteurs qui pourront être avalés ou implantés sous la peau (surveillance de la glycémie, détection de cancers à l'étape précoce, etc.). **[22]**

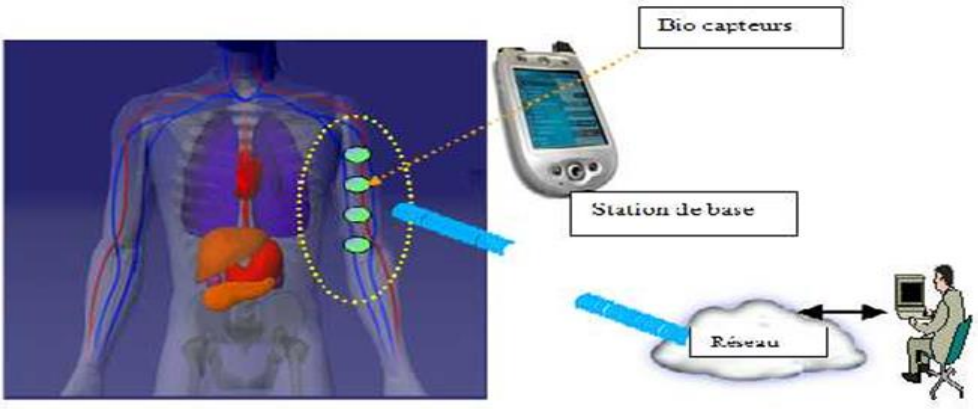


**Figure *I* -8: application médicale.**

- **Surveillance militaire :** Les premières applications potentielles des réseaux de capteurs sans fil ont concerné le domaine militaire. L'idée est de déployer des nœuds nanoscopiques, et donc invisibles, sur un champ de bataille pour surveiller les mouvements des ennemis.**[22]**

***Figure I -9*: Application Militaires. [23]**

- **Transport :** Il est possible d'intégrer des nœuds capteurs au processus de stockage et de livraison. Le réseau ainsi formé, pourra être utilisé pour connaître la position, l'état et la direction d'un paquet ou d'une cargaison. **[24]**
- **Domestiques :** En plaçant, sur le plafond ou dans le mur, des capteurs, on peut économiser l'énergie en gérant l'éclairage ou le chauffage en fonction de la localisation des personnes. **[24]**

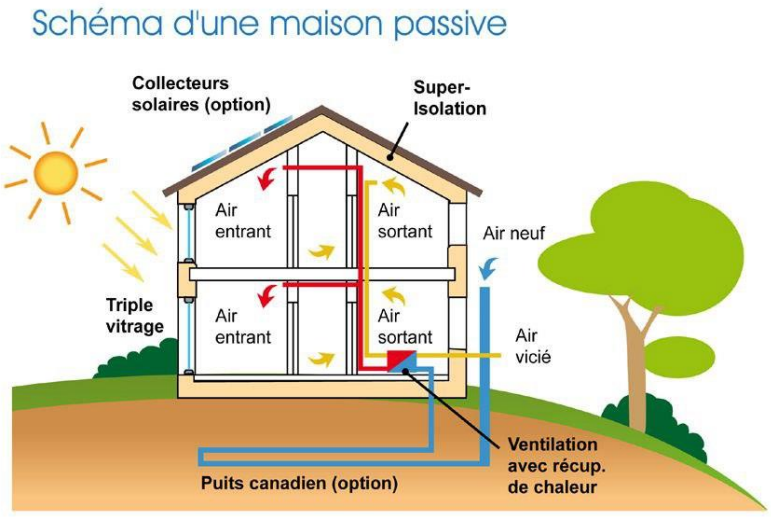


***Figure I -10*: Application domestique.[23]**

- **Environnementales :** Les réseaux de capteurs sont beaucoup appliqués dans ce domaine pour détecter des incendies, surveiller des catastrophes naturelles, détecter des pollutions et suivre des écosystèmes. **[24]**

***Figure I -11*: Application environmental. [25]**

## I.10 Topologies des réseaux de capteurs [26]

### a. Topologie en étoile

Dans cette topologie une station de base envoie ou reçoit un message

via un certain nombre de nœuds. Ces nœuds peuvent seulement envoyer ou recevoir un message de l'unique station de base, il ne leur est pas permis de s'échanger des messages.

**Avantage :** simplicité et faible consommation d'énergie des nœuds, moindre de communication entre les nœuds et la station de base.

**Inconvénient :** la station de base est vulnérable, car tout le réseau est géré par un seul nœud.

**b. Topologie « en toile » où « en grille »**

Dans ce cas (dit « communication multi-sauts »), tout nœud peut échanger avec n'importe quel autre nœud du réseau (s'il est à portée de transmission). Un nœud voulant transmettre un message à un autre nœud hors de sa portée de transmission, peut utiliser un nœud intermédiaire pour envoyer son message au nœud destinataire.

**Avantage :** Possibilité de passer à l'échelle du réseau, avec redondance et tolérance aux fautes.

**Inconvénient :** Une consommation d'énergie plus importante est induite par lacommunication multi-sauts. Une latence est créée par le passage des messages des nœuds par plusieurs autres avant d'arriver à la station de base.

C. **Topologie hybride**

Une topologie hybride entre celle en étoile et en grille fournit descommunications réseau robustes et diverses, en assurant la minimisation de laconsommation d'énergie dans les réseaux de capteurs. Dans ce type de topologie,les nœuds capteur autonome en énergie ne routent pas les messages, mais il y a d'autres nœuds qui ont la possibilité de

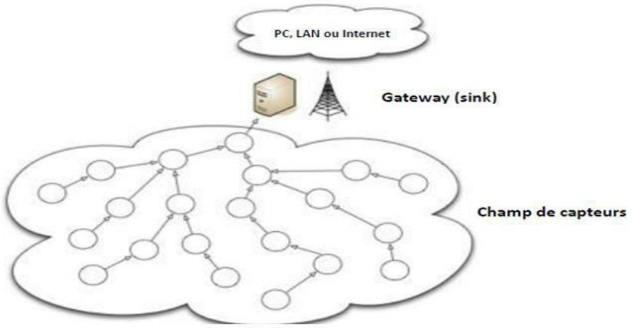


***Figure I -12*: Topologie hybride d'un RCSF [27]**

faire le routage des messages. En général, ces nœuds disposent d'une source d'énergie externe.

La norme indique que le réseau soit coordonné par un des FFDs, ce dernier peut routerdes données (contrairement au RFD). Dans cette norme, la topologie en étoile met l'accent surla durée de vie des batteries puisque chaque RFD est relié directement au coordonnateur. Par contre la topologie paire à paire s'intéresse à la fiabilité et à la scalabilité puisque tous les nœuds sont des FFDs et peuvent donc être reliés ensemble. La norme IEEE 802.15.4 peut supporter d'autres topologies, par exemple la topologie arbre de cellules "Cluster tree" qui combine les deux topologies précédentes (étoile et paire à paire ou maillé "Mesh"). Les différentes topologies du réseau supportées par IEEE 802.15.4 sont montrées dans la figure suivante :**[28]**

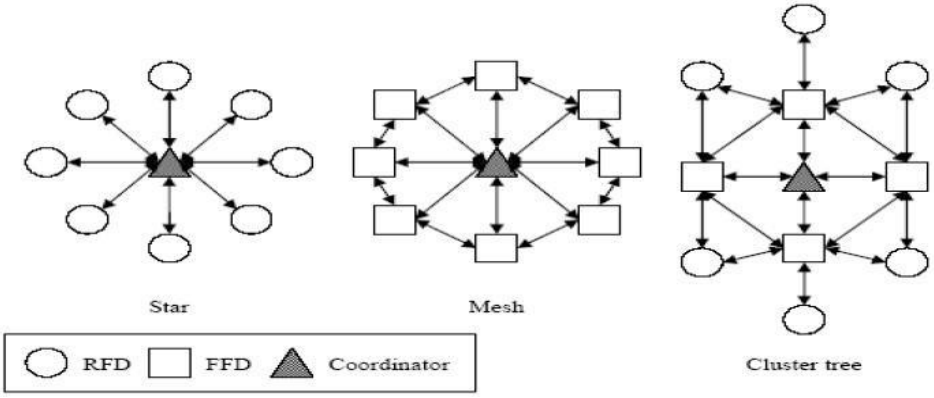


***Figure I -13*: Les topologies du réseau supportées par IEEE 802.15.4**

# I.11 Les systèmes d'exploitation pour les réseaux de capteurs :

## I.11.1 TinyOS :

TinyOS est un système exploitation open source conçu pour les réseaux de capteurs sans fil. Il est basé sur une architecture orientée composants qui favorise l'implémentation et l'innovation rapide. De plus, il génère un noyau de petite taille (quelques ko), comme l'exigent les contraintes de mémoire imposées par

les réseaux de capteurs. **[29]** Ce système d'exploitation est développé avec le langage NesC. Le développementet la maintenance de ce système est maintenant sous la responsabilité d'un consortium international, TinyOS Alliance. **[29]**

### I.11.2 Contiki :

Contiki est un système d'exploitation portable, open source et multitâche pour les réseaux de capteurs. Il supporte beaucoup de plateformes et il a un environnement de simulation Netsim.**[31]**

Ce système a été développe par l'équipe des systèmes embarques dans l'institut dessciences informatique suédois. Contiki supporte le multitâche et il implémente la pile protocolaire TCP/IP. Il ne consomme pas beaucoup de mémoire : quelques ko de code et quelques centaines d'octets dans la RAM.**[32]**

Au contraire de TinyOS qui se base sur la notion d'événements, celui-ci se base sur le multitâche et l'édition statique des liens.

## I.12 Conclusion :

Les réseaux de capteurs sans fil suscitent un intérêt considérable et marquent une nouvelle étape dans l'évolution des technologies de l'information et de la communication. Cette technologie émergente trouve des applications diverses dans de nombreux domaines.

Dans ce chapitre, nous avons examiné de manière approfondie les réseaux de capteurs sans fil, en mettant en évidence leurs caractéristiques et leur architecture. Nous avons également exploré les différentes applications et topologies des réseaux de capteurs sans fil.

Dans le prochain chapitre, nous aborderons la question de la sécurité des réseaux de capteurs sans fil.

# II. Chapitre II :

# La Sécurité dans les réseaux de Capteurs sans fil

# II.1 Introduction :

Parmi les caractéristiques majeures des nœuds de capteurs, nous distinguons la limitation de leurs ressources en termes de capacité de calcul, d'espace de stockage des données et la faible portée radio. Les ressources limitées de ces nœuds et les environnements hostiles dans lesquels ils pourraient être déployés, rendent ce type de réseaux très vulnérable aux attaques.

Dans ce contexte, une grande communauté de chercheurs tente de proposer des mécanismes de sécurité pour la prévention et la détection de tout type d'attaque, en tenant compte des contraintes de ce type de réseaux. **[33]**

Dans la sécurité, nous visons à atteindre certains objectifs dont nous parlerons dans ce chapitre, puis nous étudions la classification des attaques qui à leur tour présentent un grand risque contre le réseau, finalement nous parlerons des mécanismes de sécurité qui nous permettront à atteindre un bon niveau de sécurité et assuré le bon fonctionnement du réseau.

# II.2 Buts de sécurité [34] :

## II.2.1 L'authentification :

L'authentification permet de s'assurer que les données proviennent de la source appropriée, car un attaquant peut non seulement modifier les paquets de données, mais également en injecter de nouveaux. D'autre part, lors de la construction d'un réseau des nœuds capteurs, de nombreuses tâches (transfert des mesures vers les stations de base, synchronisation, etc.) nécessitent une authentification.

## II.2.2 L'intégrité :

L'intégrité des données est un service qui assure que les données n'ont pas été modifiées pendant leur transmission, On distingue les modifications accidentelles telles que celles causées par une mauvaise couverture du signal et les modifications volontaires causées par un attaquant, Cela concerne également la protection contre l'injection ou la modification de paquets de données.

## II.2.3 La confidentialité :

Consiste à garantir que les informations d'un nœud ne peuvent être consultées ou divulguées que par ses destinataires. Dans notre contexte, il faut s'assurer qu'aucun

capteur externe du système ne soit placé à proximité dans le but de surveiller les informations échangées.

### II.2.4 La disponibilité :

La disponibilité fournit une assurance sur la réactivité et le temps de réponse du système pour transmettre des informations de manière efficace de leur source à leur destination De plus, cela garantit que les services du réseau sont accessibles aux parties autorisées en tout temps et qu'ils fonctionnent même en présence d'attaques qui pourraient perturber n'importe quelle couche du réseau.

### II.2.5 La non-répudiation :

La non-répudiation est un mécanisme visant à empêcher la source ou la destination de nier leurs actions ou de prétendre qu'un échange n'a pas eu lieu.

### II.2.6 Le contrôle d'accès :

Le contrôle d'accès est un service crucial qui vise à restreindre l'accès au réseau uniquement aux éléments faisant partie du système. Les utilisateurs autorisés peuvent ainsi être en mesure de repérer et bloquer les messages provenant de sources externes au réseau.

## II.3 Les attaques contre les réseaux de capteurs sans fil :

Une attaque informatique est une combinaison de méthodes visant à perturber ou bloquer un service, empêchant ainsi les utilisateurs autorisés d'y accéder. **[35]**

### II.3.1 Classifications des attaquants :

#### II.3.1.1 Selon l'objectif recherché :

Les attaques contre les réseaux de capteurs sans fil peuvent être classées en fonction des objectifs de l'attaquant. Le premier type d'attaque vise à monopoliser les ressources pour les besoins de l'attaquant, mettant ainsi en danger les autres clients du réseau. La deuxième catégorie d'attaques a pour objectif de nuire aux exploitants du réseau en réduisant ou annihilant la capacité du réseau à fournir les services pour lesquels il a été déployé, ce qui peut causer des pertes financières ou priver les exploitants d'informations stratégiques. La troisième catégorie d'attaques, moins courante, cherche à tromper l'exploitant du réseau en falsifiant les résultats collectés ou transmis par les capteurs. **[36]**

### II.3.1.2 Selon la position de l'attaquant :

Les attaques peuvent être menées à partir de l'intérieur ou de l'extérieur du réseau. Les attaques internes peuvent être plus subtiles et difficiles à détecter si aucune méthode de détection d'intrusion n'a été mise en place. **[36]**

## II.3.2 Selon la capacité de l'attaquant :

La classification des attaques selon la capacité de l'attaquant se base sur les ressources dont il dispose par rapport aux nœuds présents dans le réseau. Si l'attaquant est équipé de ressources supplémentaires telles qu'un ordinateur portable avec un médium radio sophistiqué, on le considère comme un attaquant fort. En revanche, si l'attaquant possède les mêmes caractéristiques que les autres nœuds du réseau, il est considéré comme un attaquant ordinaire et n'a aucun avantage par rapport aux nœuds légitimes. Cela permet de distinguer les attaquants qui ont des avantages en termes de puissance, de bande passante ou d'énergie, et qui peuvent donc potentiellement causer plus de dommages au réseau. **[37]**

### II.3.2.1 Selon son nature :

Dans cette section, nous classons les attaques selon deux grandes catégories : les attaques passives et les attaques actives. **[37]**

#### II.3.2.1.1 Attaques passives : **[37]**

Dans ce type d'attaque, l'attaquant est généralement déguisé, c'est-à-dire caché, et se limite à écouter, collecter des données et analyser le trafic échangé, ce qui permet à l'attaquant d'intercepter, de contrôler et de surveiller les données entre les nœuds de communication. Ce type d'attaque est facile à mettre en œuvre et difficile à détecter. Ces attaques peuvent être classées selon les types suivants :

- Camouflage d'adversaires **:**

Après une attaque active, l'adversaire peut être compromis ou inséré dans le chemin de routage en tant que nœud légitime pour attirer des paquets afin d'analyser le trafic dans la zone.

- Écoute (Eavesdropping) **:**

L'écoute passive consiste à écouter attentivement les conversations privées. L'attaquant lui-même entre dans le chemin actif pour écouter passivement tout le trafic envoyé via le support de diffusion et extraire les données collectées sur le réseau

(données agrégées). Sans l'utilisation de mécanismes de cryptage pour protéger les messages, ce qui conduit à des facilités dans la compréhension du contenu de la conversation, et menace la confidentialité des données.

- Analyse du trafic **:**

Grâce à une analyse approfondie du trafic, les attaquants écoutent et analysent le trafic pour mener des attaques efficaces. Ainsi, il est possible d'obtenir des informations utiles sur la structure du réseau, de comprendre le rôle des nœuds ou de connaître les stations de base. Par exemple, nous pouvons identifier les contacts d'un nœud en filtrant le trafic réseau, l'analyse du trafic peut être utilisée pour localiser des informations confidentielles.

Par exemple, dans les communications tactiques, une augmentation du trafic rend l'adversaire conscient d'un événement imminent, ainsi que des silences qui peuvent indiquer une intention d'attaque, une infiltration ou un mouvement tactique. L'analyse du trafic peut être effectuée par l'une des méthodes suivantes : Analyse du trafic sur la couche physique, Analyse du trafic dans les couches MAC et supérieures, Analyse du trafic par corrélation d'événements.

### *II.3.2.2* Attaques actives :

Les attaques actives sont celles dans lesquelles l'attaquant cherche à altérer les informations ou à créer des messages falsifiés. **[34]**

Ces attaques ne portent pas seulement atteinte à la confidentialité des données, mais peuvent également avoir des conséquences sur la disponibilité, l'actualité et l'intégrité des données. Contrairement aux attaques passives, ces attaques peuvent être détectées en utilisant des mécanismes de sécurité plus avancés. Les attaques actives les plus connues sont classées dans les catégories suivantes : **[38]**

#### a. Attaques de la couche physique :

La fonction principale de cette couche est de moduler les données et de les transmettre via le support physique, en utilisant les fréquences appropriées, la détection de signaux et le chiffrement des données. Deux types d'attaques sont examinés dans cette couche. **[37]**

- **Attaque de brouillage (Jamming) :**

En raison de la vulnérabilité du canal sans fil, un attaquant peut utiliser un dispositif de brouillage puissant pour perturber la communication entre deux interlocuteurs. Si cette attaque cible des nœuds centraux tels que la station de base ou le *clusterhead*, elle peut isoler une région entière ou paralyser l'ensemble du réseau.

Afin de protéger les RCSF contre les attaques de brouillage, des techniques de transmission de signaux sont employées pour commuter rapidement les porteuses parmi de nombreux canaux de fréquence, comme l'étalement de code et le saut de fréquence.

Ces techniques empêchent l'attaquant de détecter le canal de fréquence utilisé entre l'émetteur et le récepteur. D'autres techniques sont également utilisées pour bloquer ces attaques, mais elles sont complexes et coûteuses en termes d'énergie, ce qui nécessite d'autres approches de sécurité pour maintenir les exigences des nœuds capteurs tels que la faible consommation d'énergie et le faible coût des dispositifs de détection. Une défense logique consiste à mettre les capteurs en mode veille à long terme et à les réveiller périodiquement pour tester le canal. **[37] [39] [40]**

- **Attaque d'altération (Tampering) :**

En termes simples, les Réseaux de Capteurs Sans Fil (RCSF) sont souvent déployés dans des environnements non sécurisés sans surveillance adéquate. Dans ces conditions, la méthode la plus facile pour une attaque est de modifier ou endommager physiquement les capteurs afin d'interrompre ou altérer leurs fonctions. Si les stations de base ou les points d'agrégation sont ciblés, l'impact négatif sera encore plus important. Toutefois, ces attaques physiques sont limitées en efficacité en raison de la densité élevée et la redondance des nœuds dans la plupart des RCSF, sauf si un grand nombre de capteurs sont compromis. **[41]**

L'altération physique est également une méthode courante d'attaque où un attaquant peut capturer un nœud capteur pour accéder facilement à des informations sensibles telles que les clés cryptographiques, altérer les circuits électroniques ou reprogrammer le nœud, ce qui peut causer des dommages irréversibles. **[40]**

## b. Les attaques de la couche liaison :

Elle décrit la manière dont les données sont transférées entre deux nœuds sur une seule distance. Elle est chargée du multiplexage des données, du contrôle des erreurs, de la gestion de l'accès aux médias, et autres fonctions similaires.

### ➢ Collisions :

Lorsque deux nœuds essaient de transmettre simultanément sur la même fréquence, cela entraîne une collision. Les paquets impliqués dans la collision sont rejetés et doivent être renvoyés. Un adversaire peut provoquer des collisions stratégiques dans des paquets spécifiques, comme les messages de contrôle ACK, ou pendant des périodes critiques telles que les périodes de réveil ou actives. Les conséquences de ces collisions peuvent inclure l'épuisement des ressources, l'injustice dans l'allocation des ressources et des retards coûteux dans certains protocoles de contrôle d'accès aux médias, qui affectent négativement les applications en temps réel s'exécutant sur les autres nœuds, en interrompant leurs transmissions de trames. **[39]**

Plusieurs techniques ont été proposées pour prévenir les collisions. Pour les collisions causées par des erreurs environnementales, les codes de correction d'erreur sont souvent utilisés. Cependant, ces codes nécessitent des coûts supplémentaires de traitement et de communication pour surmonter les collisions. De plus, il n'est pas possible de corriger plus que ce qui a été corrompu. Bien que les collisions malveillantes puissent être détectées, jusqu'à présent, aucune technique de défense appropriée n'a été trouvée pour surmonter complètement ces attaques. **[42]**

### ➢ Épuisement :

Un attaquant peut utiliser des collisions à plusieurs reprises pour épuiser les ressources, ce qui oblige les nœuds à réémettre les messages même s'il n'y a pas de collision ou de collision tardive. **[43]**

Cette attaque peut épuiser la puissance des nœuds. Pour éviter la perte d'énergie causée par les transmissions répétées, une solution consiste à limiter le débit du contrôle d'admission MAC. Une autre solution consiste à utiliser le multiplexage temporel où chaque nœud est attribué un créneau temporel pour transmettre sans besoin d'arbitrage pour chaque trame. Cependant, cette technique peut toujours être sensible aux collisions. **[44]**

➢ **L'injustice :**

L'injustice peut être considérée comme une forme moins sévère d'une attaque par déni de service (DoS). Pour provoquer une injustice dans un réseau, un attaquant peut utiliser de manière intermittente des attaques de la couche liaison ci-dessus. **[44]**

Les protocoles MAC de la couche de liaison gèrent les communications dans les réseaux en contrôlant les schémas de priorité pour une communication fluide. Il est possible d'utiliser ces protocoles pour altérer les schémas de priorité, ce qui entraîne finalement une diminution du service. **[43]**

### c. Les attaques au niveau de la couche réseau :

La couche dont il est question ici s'occupe de la gestion de l'envoi et du routage des données, les protocoles les plus importants utilisés à ce niveau sont l'IP et l'ICMP.

➢ **Attaque de trou noir (Blackhole) :**

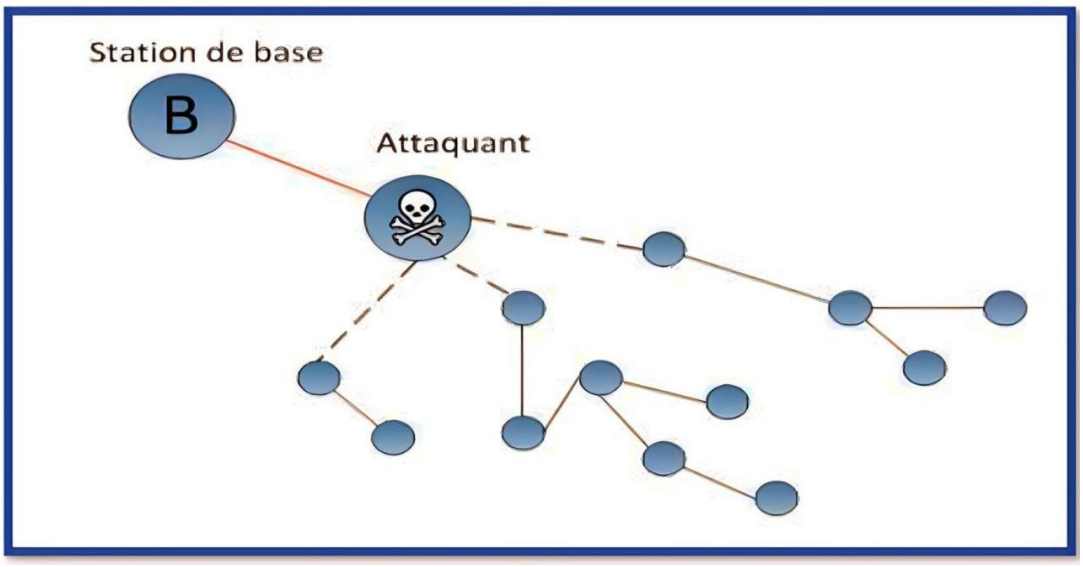


**Figure II-1: Attaque de trou noir (Blackhole)**

L'attaque par trou noir commence par l'insertion de nœuds malveillants dans le réseau de différentes manières. Le nœud altère la table de routage pour limiter le nombre maximal de nœuds adjacents par lesquels les informations sont transmises. Ainsi, toutes les informations qui transitent par ce nœud seront affectées.

### ➢ Les attaques Sybil :

Les attaques Sybil sont courantes dans les Réseaux de Capteurs sans Fil (RCSF) lorsqu'elles sont combinées avec d'autres types d'attaques. Dans ce type d'attaque, un nœud malveillant se comporte comme plusieurs nœuds légitimes en créant ou en volant les identités de nœuds légitimes. Cela permet à l'attaquant d'être présent à plusieurs endroits simultanément, ce qui peut causer des dommages importants aux protocoles de routage, en particulier ceux qui sont basés sur la localisation. Ces protocoles impliquent que les nœuds échangent des informations de coordonnées avec leurs voisins pour construire le réseau. Pour contrer ce type d'attaque, l'authentification qui permet de vérifier l'identité des nœuds est essentielle, mais les limites de calcul et de stockage des RCSF posent des défis pour mettre en œuvre cette solution. **[37]**

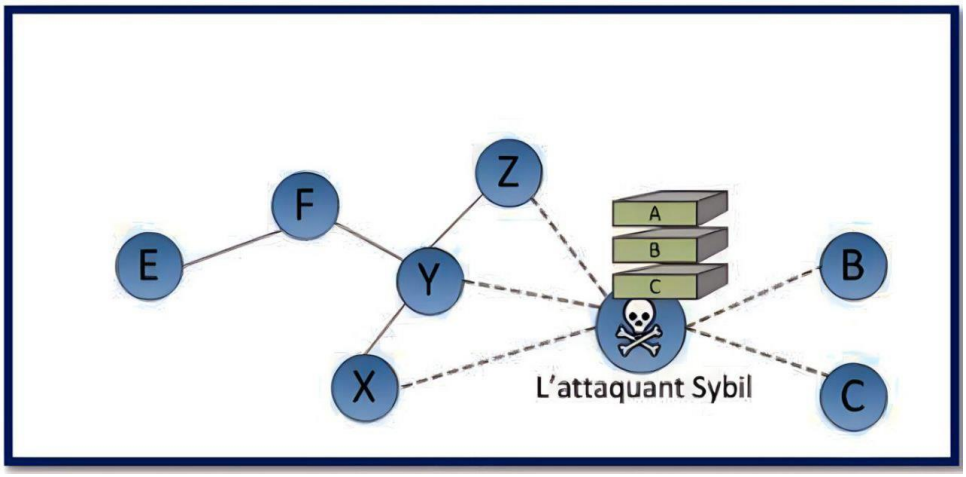


**Figure II-2: Attaque Sybil**

### ➢ L'attaque de trou de puits (Sinkhole Attack) :

Dans l'attaque de trou de puits, un nœud malveillant cherche à devenir le nœud relais sur la route des nœuds voisins pour attirer tous les flux de données dans une zone spécifique et empêcher les paquets de parvenir à leur destination. Cette attaque peut être combinée avec d'autres attaques, comme le routage sélectif, pour sélectionner les paquets qui sont transmis. Les protocoles de routage basés sur l'estimation de l'énergie ou de la fiabilité sont particulièrement vulnérables à cette attaque, et il est difficile de s'en défendre. **[37]**

### ➢ L'attaque de trou de ver (Wormhole Attack) :

Lorsqu'un attaquant compromet deux ou plusieurs agents dans un réseau, ils peuvent mener une attaque appelée "trou de ver". Cette attaque implique la capture de trafic en un point du réseau pour le réinjecter en un autre point en utilisant un canal auxiliaire distinct. Les nœuds attaquants peuvent réinjecter des informations de routage pour tromper les nœuds légitimes et leur faire croire qu'ils ont des voisins virtuels qui sont en réalité hors de leur portée. Cette attaque peut gravement perturber l'organisation du réseau. **[36]**

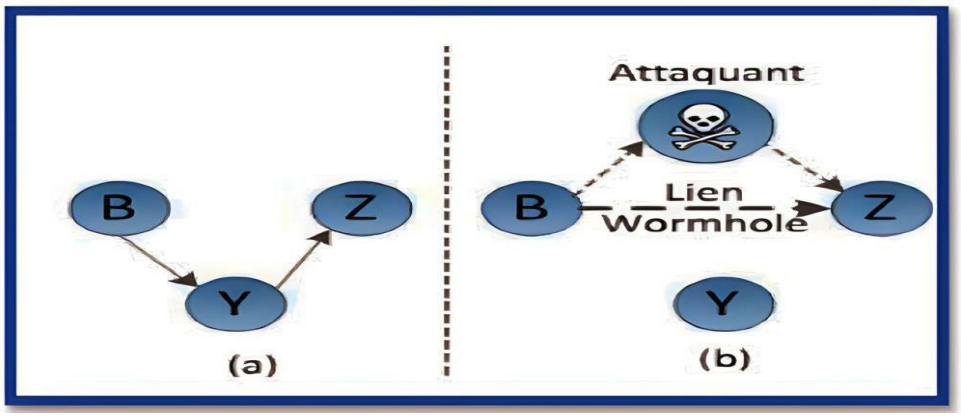


**Figure II-3:Attaque Wormhole [45]**

### ➢ L'attaque d'inondation par paquet Hello (Hello Flood Attack) :

Les réseaux de capteurs utilisent souvent des paquets "hello" pour découvrir les nœuds dans leur voisinage. Les nœuds voisins répondent à ces paquets avec des "hello-replay" pour indiquer qu'ils ont reçu le message. Cependant, un attaquant disposant d'une machine plus puissante peut forger et envoyer des paquets "hello-replay" avec une portée supérieure à celle des capteurs pour leur faire croire qu'ils ont des voisins virtuels qui sont en réalité bien au-delà de leur portée d'émission. Cette attaque de "déluge de paquets

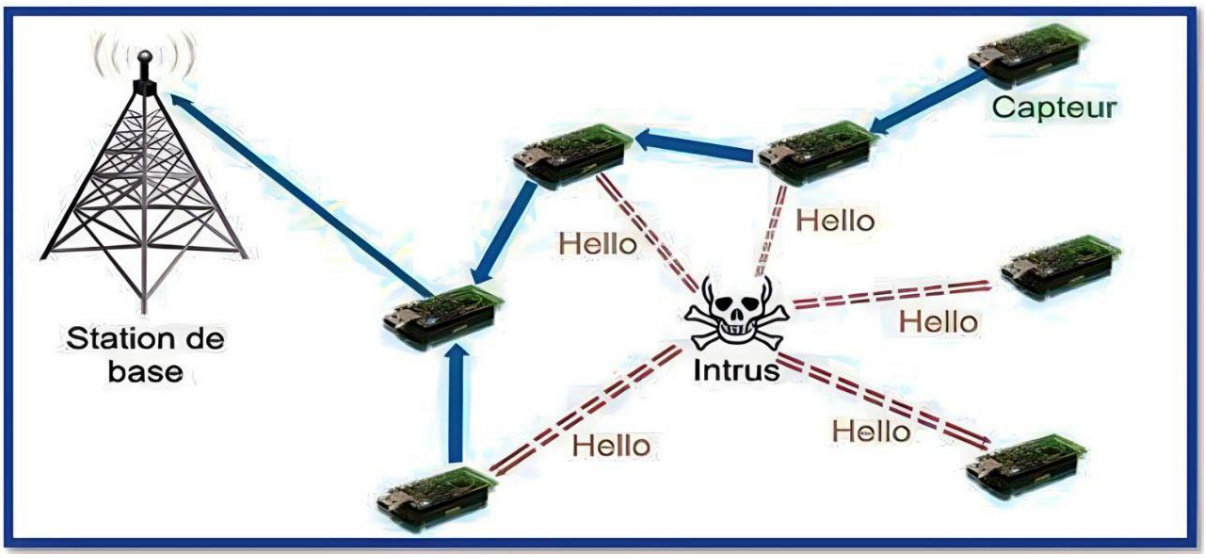


**Figure II-4: Attaque Hello Flood Attack. [46]**

- **Usurpation d'accusé de réception (Acknowledgement spoofing) :**

Cette technique peut être utilisée pour attaquer les algorithmes de routage qui se basent sur les accusés de réception de la couche liaison. Si des paquets sont transmis sur des liens de communication de mauvaise qualité et à plusieurs sauts, un attaquant peut facilement usurper les accusés de réception en incitant le nœud cible à envoyer des paquets sur ces liens. Cela permet à l'attaquant de lancer une attaque de routage sélective ou de provoquer une perte de paquets de manière efficace. **[37]**

### d. Les attaques au niveau de la couche transport :

La couche de transport assure une gestion globale des connexions entre l'expéditeur et le destinataire. Les protocoles qui stockent des informations de connexion peuvent être vulnérables aux attaques de saturation et de désynchronisation. Cependant, les protocoles de transport sans connexion offrent une protection contre ces types d'attaques. Nous allons maintenant examiner deux types d'attaques possibles sur cette couche. **[47]**

- **Inondation (Flooding) :**

L'objectif des attaques par inondation est d'épuiser les ressources des nœuds capteurs et de perturber la communication entre eux. L'attaquant peut envoyer plusieurs demandes de connexion à un nœud cible pour saturer son tampon de connexion, ce qui empêche la réception de demandes légitimes des autres nœuds du réseau.

Les mesures de défense contre ces attaques consistent à limiter le nombre de demandes de connexion de chaque nœud et à coder les paquets TCP SYN pour éviter de stocker l'état de la connexion sur le serveur. Cependant, ces techniques peuvent entraîner une augmentation des calculs et des coûts globaux dans les réseaux de capteurs sans fil (RCSF), ce qui les rend indésirables. **[37]**

- **Désynchronisation (De-synchronization) :**

L'attaque de désynchronisation consiste à perturber une connexion active entre deux nœuds en envoyant de faux messages avec des numéros de séquence ou des indicateurs de contrôle modifiés. Cette désynchronisation conduit à la retransmission de faux paquets entre les nœuds, ce qui gaspille leur énergie.

Pour contrer cette attaque, il est recommandé d'activer tous les champs de contrôle dans l'en-tête du paquet de transport. En outre, l'authentification des paquets échangés entre deux nœuds peut aider à vaincre cette attaque. **[37]**

### e. Les attaques au niveau de la couche application :

Les attaques au niveau de la couche application ont pour but de consommer la bande passante du réseau et drainer l'énergie des nœuds en submergeant les nœuds du réseau avec des stimuli de capteurs ou en injectant des paquets parasites ou rejoués dans le réseau. Les solutions proposées pour contrer ces attaques incluent la révocation des nœuds compromis, l'utilisation d'un mécanisme de clé efficace tel que le protocole LEAP, des techniques de limitation du débit et des algorithmes d'agrégation des données. En outre, la combinaison de l'authentification par paquets et la protection anti-rejoue peut empêcher ces attaques. **[47] [42]**

# II.4 Les mécanismes de sécurité dans les RCSFs :

## II.4.1 La cryptographie :

La cryptographie se réfère à l'analyse des méthodes mathématiques qui cherchent à garantir différents aspects de la sécurité de l'information, notamment la confidentialité, l'intégrité des données, l'authentification des entités et la vérification de l'origine des données. Les principaux buts de la cryptographie sont de préserver la confidentialité, d'assurer l'authentification, de maintenir l'intégrité des données et de prévenir la non-répudiation. **[48]**

L'utilisation d'un système cryptographique basé sur des clés sécurisées est une des premières mesures de sécurité mises en place dans les réseaux de capteurs sans fil (RCSF) pour permettre aux nœuds capteurs de chiffrer et d'authentifier les messages échangés entre eux. Les méthodes cryptographiques utilisées dans les RCSF doivent être adaptées aux contraintes des nœuds capteurs et des communications sans fil, et évaluées en fonction de divers critères tels que la taille du code, la taille des données, le temps de traitement et la consommation d'énergie. Toutefois, les nœuds capteurs ont des capacités de calcul et de mémoire limitées, ce qui rend difficile l'application des techniques cryptographiques traditionnelles. Pour répondre aux contraintes de sécurité dans les RCSF, il faut adapter les techniques existantes ou en développer de nouvelles. Les systèmes cryptographiques sont généralement classés en trois types : les techniques symétriques, asymétriques et hybrides. Dans les RCSF, le choix de la technique cryptographique appropriée est crucial en raison des limitations des nœuds capteurs. **[37]**

### II.4.1.1 Cryptographie symétrique :[49]

Le cryptage symétrique implique l'utilisation d'une clé partagée entre l'expéditeur et le destinataire, qui est utilisée par l'expéditeur pour coder le message et par le destinataire pour le décoder.

Parmi les algorithmes utilisés dans le cryptage symétrique : AES, DES, 3DES, IDEA et Blowfish.

➢ **Avantages :**

- Facilité de mise en place et d'utilisation.
- Plus rapide que la cryptographie asymétrique.
- Moins exigeante en ressources.
- Efficace pour le traitement et le transfert de grandes quantités de données.

➢ **Inconvénients :**

- La perte de la clé compromet toutes les données chiffrées avec cette clé.
- Il est nécessaire de partager la clé en toute sécurité avec l'autre partie.

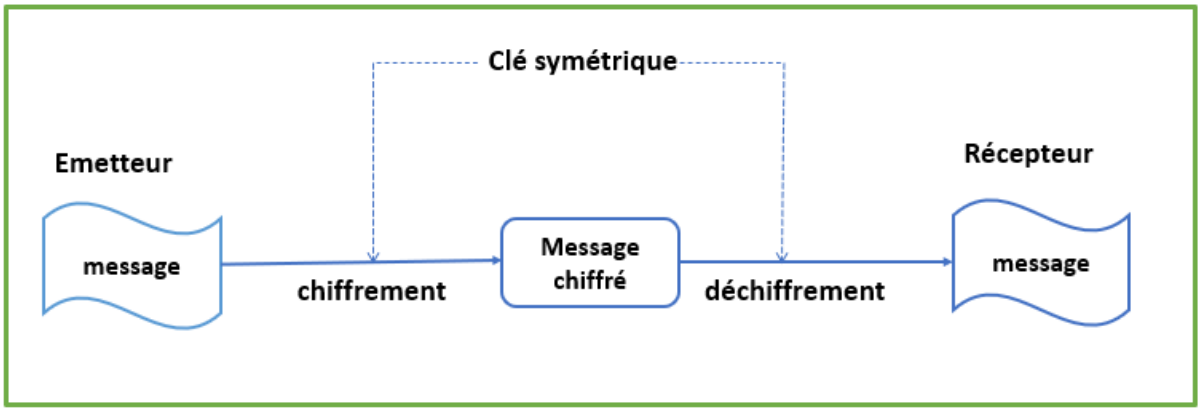


**Figure II-5: Cryptographie symétrique**

### II.4.1.2 Cryptographie asymétrique :

Les systèmes asymétriques reposent sur l'utilisation de deux clés distinctes et non reproductibles l'une à partir de l'autre pour créer des clés de chiffrement et de déchiffrement différentes. La première clé, gardée secrète, permet de déchiffrer et de signer des données, tandis que la deuxième clé, appelée clé publique, permet de chiffrer et de vérifier des données. **[37]**

Algorithmes utilisés dans ce type de chiffrement : RSA, ECC, DSA et El Gamal.

➢ **Avantages :**

- Les données ne peuvent être déchiffrées qu'avec la clé privée du propriétaire.
- En cas de perte ou de vol de la clé publique, les données ne sont pas compromises.
- Permet l'authentification, la non-répudiation et la confidentialité.

➢ **Inconvénients :**

- Il est plus lent que le chiffrement symétrique.
- Il utilise plus de ressources.
- En cas de perte de la clé privée, il n'y a aucun moyen de la récupérer. **[49]**

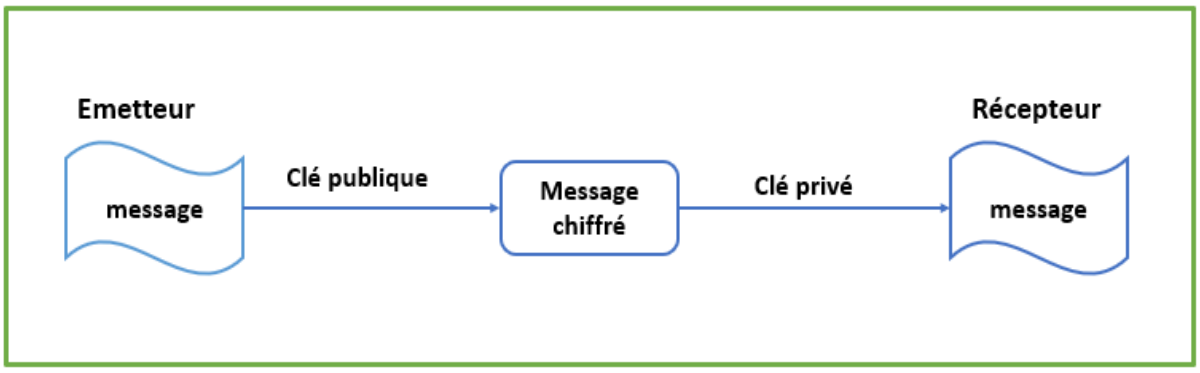


**Figure II-6: Cryptographie asymétrique**

### II.4.2 Fonctions de hachage :

La fonction de hachage est utilisée pour générer une empreinte, une chaîne de taille inférieure et généralement fixe, à partir d'une chaîne de longueur variable. Elle est considérée comme une fonction à sens unique, ce qui signifie que l'on peut facilement calculer l'empreinte d'une chaîne, mais il est impossible de retrouver la chaîne initiale à partir de l'empreinte. Cette fonction est souvent utilisée pour vérifier l'intégrité des messages transmis. L'émetteur utilise la fonction de hachage pour créer une empreinte du message et la transmet avec le message au récepteur. À la réception du message, le récepteur calcule l'empreinte du message reçu et la compare à l'empreinte initiale. Si les deux empreintes correspondent, cela signifie que le message n'a pas été altéré pendant la transmission. **[50]**

### II.4.3 Système de détection d'intrusion :

Un système appelé IDS (Système de détection d'intrusion) est utilisé pour écouter discrètement le trafic réseau et repérer les activités suspectes ou anormales, afin de prévenir les risques d'intrusion. Il existe Trois types d'IDS :

- Les NIDS, qui surveillent le réseau.
- Les HIDS, qui surveillent les hôtes.
- Les IDS hybrides, qui combinent les deux pour une détection plus précise.

Les IDS utilisent deux approches différentes pour détecter les intrusions : les IDS à signature, qui recherchent les empreintes des attaques connues, et les IDS à anomalie, qui détectent les divergences par rapport au fonctionnement normal des éléments surveillés.

Les IDS à anomalie sont capables de détecter de nouveaux types d'attaques, mais nécessitent des ajustements fréquents pour éviter les fausses alertes.

Les IDS dans le domaine des RCSFs ont atteint une certaine maturité et offrent des résultats fiables et pertinents. **[34]**

## II.5 Conclusion :

Dans ce chapitre, nous avons présenté l'importance de la sécurité dans les réseaux de capteurs sans fil (RCSF) et discuté des diverses attaques auxquelles ces réseaux peuvent être exposés. En outre, nous avons exploré les mécanismes de sécurité, qui reposent généralement sur des concepts de cryptage, afin de répondre aux problématiques de sécurité dans les RCSF.

Dans le chapitre suivant, notre intérêt se portera sur la méthode de détection d'anomalies pour les réseaux de capteurs sans fil, en se basant sur les séries temporelles.

# III. Méthode de détection d'anomalies dans les réseaux de capteurs sans fil basée sur les séries temporelles

# III.1 Introduction :

Les RCSF sont vulnérable à de nombreuses attaques et anomalies, Ce chapitre se concentre sur l'étude des attaques et des anomalies dans les séries temporelles à l'aide d'algorithmes d'apprentissage automatique inclus dans la bibliothèque pycaret. Pycaret est une bibliothèque populaire qui fournit des fonctionnalités avancées d'apprentissage automatique, notamment la détection d'anomalies. À l'aide de cette bibliothèque, nous explorons la base de données WSN-DS, une riche source d'attaques dans RCSF, pour approfondir notre compréhension des attaques potentielles et des anomalies trouvées et détectées dans ces données.

# III.2 Les séries temporelles :

Une série temporelle est une séquence de points de données indexés dans l'ordre du temps. Il s'agit d'une série de données à temps discret, le plus souvent prises à des points successifs équidistants dans le temps. Les séries temporelles sont utilisées dans divers domaines, notamment les statistiques, le traitement du signal, la reconnaissance des formes, l'économétrie, la finance mathématique, les prévisions météorologiques, la prévision des tremblements de terre, l'électroencéphalographie, l'ingénierie de contrôle, l'astronomie et l'ingénierie des communications. [51]

## III.3 Les séries temporelles dans les réseaux de capteurs sans fil [52] :

Les séries temporelles dans les réseaux de capteurs sans fil sont des données qui sont collectées et enregistrées à des moments spécifiques dans le temps. Ces données fournissent des informations sur l'évolution des phénomènes mesurés au fil du temps. Elles peuvent inclure des mesures régulières effectuées l'intervalles fixes ou des mesures sporadiques en réponse à des événements spécifiques.

Les séries temporelles de mesures dans un réseau de capteurs sans fil peuvent être utilisées à diverses fins, notamment :

### III.3.1 Détection des anomalies dans les séries temporelles :

L'analyse des séries temporelles peut aider à détecter les anomalies ou les comportements anormaux. Les modèles peuvent être formés pour identifier les schémas inhabituels ou les variations anormales dans les mesures, ce qui permet de détecter des situations telles que des pannes de capteurs, des intrusions ou des changements de comportement inhabituels.

### III.3.2 Prédiction des tendances futures :

Les modèles d'apprentissage automatique peuvent être utilisés pour analyser les séries temporelles et prédire les tendances futures. Par exemple, en utilisant les mesures passées de température, d'humidité et de pression, on peut prédire les conditions météorologiques futures.

### III.3.3 Optimisation de la consommation d'énergie :

Les séries temporelles de mesures peuvent être utilisées pour prédire et optimiser la consommation d'énergie des nœuds récepteurs dans un réseau de capteurs sans fil à énergie renouvelable. En utilisant des modèles de prédiction, on peut ajuster les cycles d'activité et de repos des nœuds pour maximiser l'utilisation de l'énergie disponible.

### III.3.4 Analyse rétrospective :

Les séries temporelles peuvent être stockées dans des bases de données pour une analyse ultérieure. Cela permet d'examiner les données historiques, d'identifier les tendances à long terme, de comparer différentes périodes de temps et de prendre des décisions éclairées basées sur les observations passées.

Il est important de noter que l'analyse des séries temporelles peut être complexe en raison de la nature des données temporelles et des défis spécifiques tels que la saisonnalité, les tendances et les dépendances temporelles. Des techniques d'apprentissage automatique spécifiques aux séries temporelles peuvent être utilisées pour modéliser et interpréter ces données de manière efficace.

## III.4 La détection d'anomalies :

La détection d'anomalies, également appelée détection de valeurs aberrantes ou détection d'événements inhabituels, fait référence au processus de recherche et d'identification de points de données ou de modèles qui se distinguent significativement du reste de l'ensemble de données. Une anomalie peut être définie comme un comportement ou une observation atypique, inattendu ou anormal par rapport aux schémas, aux tendances ou aux comportements normaux. [53

### III.4.1 La détection d'anomalies est ces domaines d'application [53] :

Différents domaines d'application ont été étudiés dans le domaine de la détection d'anomalies ces dernières années. Voici quelques-uns de ces domaines :

#### III.4.1.1 Réseaux de capteurs sans fil :

Les capteurs sont utilisés pour surveiller divers paramètres environnementaux et de localisation. Les anomalies dans les données de capteurs peuvent être causées par des défauts de capteurs ou des événements imprévus.

#### III.4.1.2 Systèmes de détection d'intrusion :

La détection d'intrusion vise à identifier les activités malveillantes dans les systèmes informatiques, telles que les attaques réseau ou les abus informatiques.

Cette détection est essentielle pour la sécurité informatique.

#### III.4.1.3 Détection de fraude :

Il s'agit de détecter les activités frauduleuses dans des organisations commerciales telles que les banques, les compagnies d'assurance, les opérateurs de téléphonie mobile, etc. Cela inclut la détection de fraudes par carte de crédit, les fraudes téléphoniques, les fraudes à l'assurance automobile, etc.

#### III.4.1.4 Diagnostic médical :

Dans le domaine médical, les données sont collectées à partir de divers appareils et peuvent présenter des anomalies en raison d'un état anormal du patient, d'erreurs d'instrumentation ou d'enregistrement.

#### III.4.1.5 Détection de dommages industriels :

Les anomalies dans ce domaine sont liées à des défauts de composants mécaniques tels que les moteurs, les turbines, les pipelines, etc.

#### III.4.1.6 Traitement d'images et de vidéos :

Les anomalies dans ce domaine concernent la détection de mouvements rares ou inconnus dans les applications de vidéosurveillance, ainsi que l'identification de régions anormales sur des images statiques, comme l'analyse d'imagerie par satellite. Ces domaines d'application peuvent impliquer des données de différentes natures, telles que des données temporelles, spatiales, binaires, discrètes, continues, audio, vidéo, etc…

### III.4.2 Type d'anomalies : [53]

Il existe une classification générale des anomalies qui s'applique dans plusieurs domaines d'application et qui peut être divisée en trois principaux types : ponctuelle, collective et contextuelle.

#### III.4.2.1 Anomalies de point :

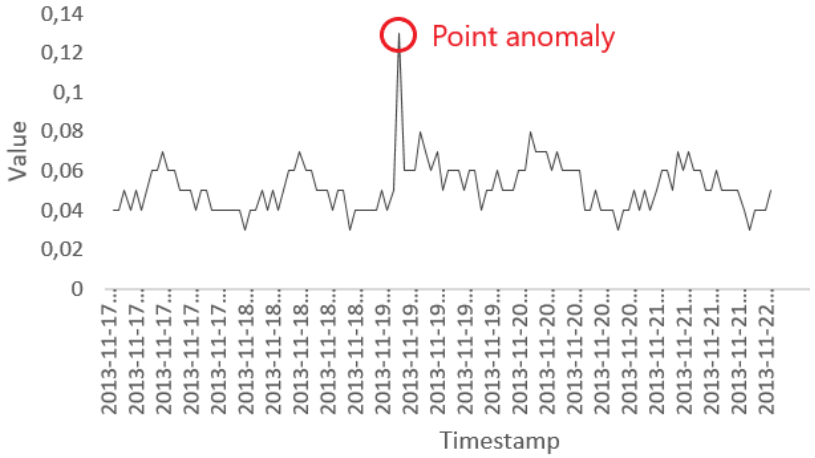


**Figure III-1: Anomalie ponctuelle dans une série temporelle de consommation énergétique** *[53]*

Les anomalies de point (ou globales) correspondent à un point de données considéré comme valeur aberrante car il est suffisamment différent ou éloigné de l'ensemble des données. l'exemple d'une série temporelle de consommation énergétique. Une observation qui a une valeur très élevée (sur consommation) par rapport la fourchette habituelle de consommation d'un bâtiment présente une anomalie de point ou ponctuelle.

#### III.4.2.2 Anomalies contextuelles :

Les anomalies contextuelles (ou locales) correspondent à un point de données (ou une séquence de points) différent ou éloigné des autres points de données mais

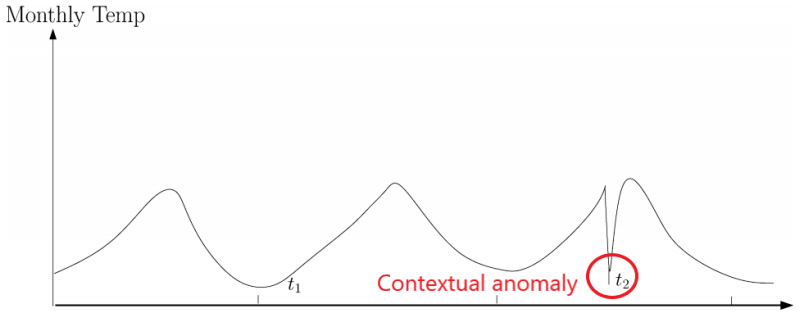


**Figure III-2: Anomalie contextuelle dans une série temporelle de température mensuelle** *[53]*

dans un contexte spécifique (spatial ou temporel). Par exemple, la figure 3 présente une anomalie contextuelle dans une série temporelle de température mensuelle. Une basse température en hiver à l'instant t1 est considérée normale, tandis que le même cas pourrait ne pas être normal en plein été à l'instant t2.

### III.4.2.3 Anomalies collectives :

Les anomalies collectives (ou séquentielles) correspondent à une collection d'observations qui est différente de l'ensemble des données. Par exemple, la figure 3 montre un exemple d'une série temporelle contenant une sous-série anormales parce qu'elle est différente par rapport à l'ensemble de sous-séquences de la série temporelle. Ceci peut correspondre à compteur en arrêt, qui échoue à remonter des données.

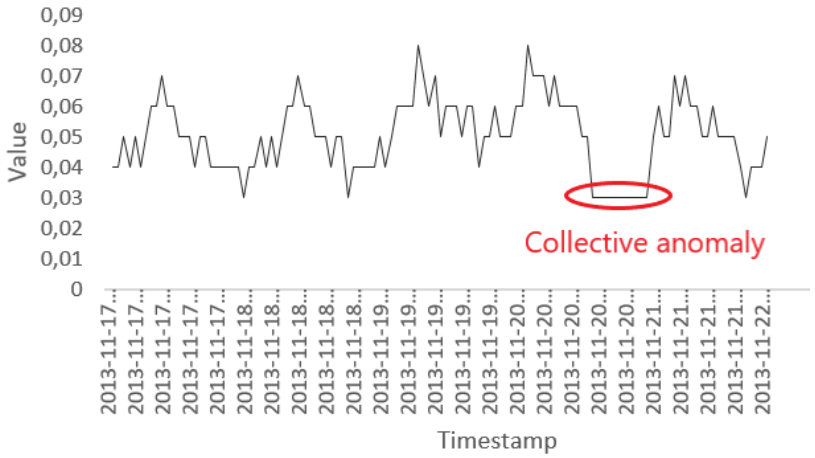


**Figure III-3: Anomalie collective correspondant à un arrêt de compteur**

## III.5Apprentissage automatique pour la détection d'anomalies :[53]

Les données disponibles influent sur les techniques de détection d'anomalies qui peuvent être appliquées. En effet, les instances de données peuvent être étiquetées (il existe une étiquette à chaque point de données donne des informations de retour des informations si la classe d'instance est normale ou anormale) ou non étiquetées. La détection peut alors se faire suivant les trois principes connus en apprentissage automatique à savoir : non supervisé, supervisé et semi-supervisé.

### III.5.1 Apprentissage automatique non supervisée :

L'apprentissage non supervisé est utilisé dans le cas où nous ne disposons pas de données étiquetées. Cette approche permet de déterminer les valeurs aberrantes sans connaissances préalables des données . Les techniques qui fonctionnent en mode non supervisé ne nécessitent pas de données d'entraînement mais supposent que le comportement normal est le plus fréquent ..L'avantage de cette méthode est qu'aucune donnée étiquetée n'est nécessaire, et elle est largement applicable dans différents domaines.

### III.5.2 Apprentissage automatique supervisée :

Détection d'anomalies supervisée nécessite un ensemble de données d'apprentissage qui contient des données étiquetées comme normales ou anormales.

Le défi de l'apprentissage supervisé est qu'il est généralement très long d'étiqueter les données et qu'il est normalement difficile d'inclure tous les types d'anomalies, ce qui est nécessaire pour que l'algorithme fonctionne correctement.

Son avantage est qu'il peut être utilisé lorsque les anomalies sont plus fréquentes que les instances normales. Contrairement aux méthodes non-supervisées, les méthodes supervisées sont conçues pour la détection d'anomalies spécifiques à l'application.

### III.5.3 Apprentissage automatique semi-supervisée :

Les données d'apprentissage contiennent des instances partiellement étiquetées, par exemple pour seulement la classe normale. Comme le mode supervisé, il peut être difficile de trouver des données qui couvrent toutes les instances normales. En règle générale, les méthodes non supervisées sont souvent utilisées dans un contexte exploratoire, où les valeurs aberrantes découvertes sont fournies à l'analyste pour un examen plus approfondi de leur importance spécifique à l'application.

## III.6 Méthode proposée :

Dans ce travail, nous avons exploré l'utilisation des séries temporelles et des algorithmes d'apprentissage automatique pour détecter les attaques dans les réseaux de capteurs sans fil (RCSF). Différentes bibliothèques ont été développées pour assurer la détection d'anomalies dans les séries temporelles, telles que ADTK [55], et PyCaret. Chaque bibliothèque propose une approche spécifique de détection, par exemple, ADTK permet à l'utilisateur de la paramétrer en fonction de la nature de la série. Pour notre étude, nous avons choisi d'utiliser la

bibliothèque PyCaret, qui combine à la fois les séries temporelles et les algorithmes d'apprentissage automatique.

Nous avons utilisé le jeu de données WSN-DS comme source de données pour effectuer la détection d'anomalies en utilisant la bibliothèque PyCaret. Les séries temporelles contenues dans le jeu de données WSN-DS sont spécifiques aux réseaux de capteurs sans fil (RCSF). La détection d'anomalies dans les RCSF revêt une importance cruciale pour garantir la sécurité du réseau en identifiant les attaques potentielles. L'approche que nous avons adoptée consiste à combiner les connaissances sur les séries temporelles avec les algorithmes d'apprentissage automatique de PyCaret pour améliorer la détection des attaques dans les RCSF. Dans le paragraphe suivant, nous donnons plus de détaille sur le contenu la base de données WSN-DS.

### III.6.1 Description du Data Set WSN-DS :

Almomani et all [56]. Ont construit un dataset pour les RCSFs, contenant quatre types d'attaques *DoS* plus un cas *normal* (non-attaque). L'ensemble de données recueillies est appelé WSN-DS [57] et il est destiné à aider les chercheurs à travailler sur les attaques Dos dans les RCSFs. Almomani et all, Ont utilisé ce dataset pour former un « Artificial Neural Network (ANN) » qui permet de détecter et de classer les types d'attaques Dos. Le protocole de routage LEACH a été utilisé pour recueillir l'ensemble de données, il est l'un des protocoles de routage les plus courants et les plus utilise dans les RCSFs.

Le dataset contient un total de 374661 enregistrements et 23 attributs, cependant dans les fichiers CSV, On trouve seulement 19 attributs décrits dans le « tableau 3.1 ». Quatre classes représentent quatre attaques Dos qui sont : Blackhole (10049), Grayhole (14596), Flooding (3312) et TDMA (6638), et normal avec les 340066 enregistrements restants. [56]

### III.6.2 Protocol LEACH :

Leach est un protocole de routage hiérarchique utilisé dans les WSN pour augmenter la durée de vie du réseau. Il suppose que la station de base est fixe et situé loin des nœuds de capteurs. Les nœuds sont homogènes et ont une énergie et une mémoire limitée

Les capteurs peuvent communiquer entre eux et ils peuvent communiquer directement avec la BS. Le protocole LEACH garantit l'organisation des nœuds en clusters pour distribuer l'énergie entre eux, et dans chaque cluster il y a un nœud appelé Cluster Head (CH) qui collecte les données reçues de son cluster des membres et les transmettre à la BS.

Chaque cycle du protocole LEACH se compose principalement de deux phases :

La phase de configuration : les clusters sont formés.

La phase d'état stable : les données collectées seront transférées au nœud à la BS. [59]

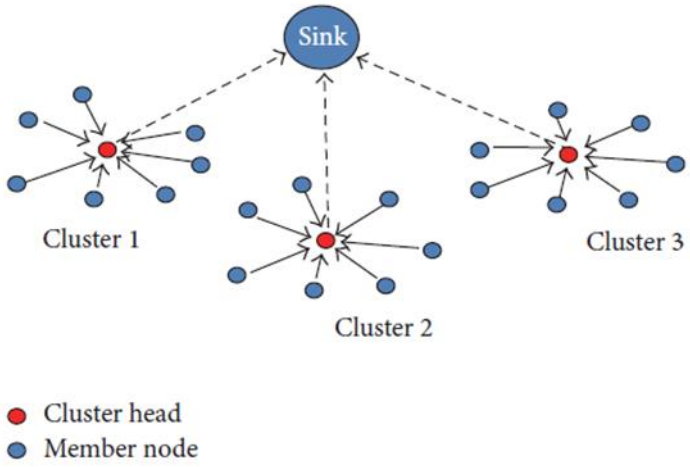


**Figure III-4: Structure des nœuds dans le protocole de routage LEACH [*59*]**

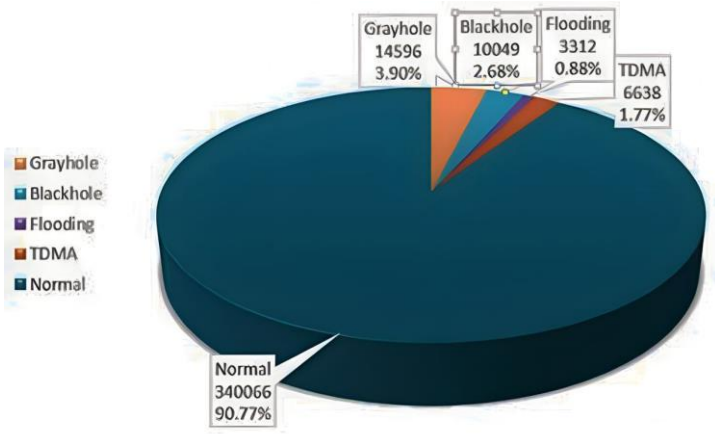


**Figure III-5: Statistiques d'attaque dans l'ensemble de données [58]**

**Tableau III-1Tableau WSN Simulation paramètres [59]**

| Paramètres | Les valeur |
|---|---|
| Nombre de nœuds: | 100 nœuds |
| Nombre de clusters: | 5 |
| Zone du réseau: | 100m × 100m |
| Emplacement du récepteur | (50, 175) |
| Taille de l'en-tête de paquet : | 25 octets |
| Taille du paquet de données : | 500 octets |
| Protocole de routage : | LEACH |
| Temps de simulation : | 3600 s |

**Tableau III-2: Nom d'attribut Description de l'attribut**

| # | Nom d'attribut | Description de l'attribut |
|---|---|---|
| **1** | **Id** | ID unique pour distinguer le nœud du capteur |
| **2** | **Time** | Temps de simulation actuel du nœud |
| **3** | **Is_CH** | Un indicateur permettant de distinguer si le nœud est CH avec la valeur 1 ou un nœud normal avec la valeur 0. |
| **4** | **Who CH** | L'ID de CH dans la ronde actuelle |
| **5** | **Dist_TO_CH** | La distance entre le nœud et son CH dans la ronde courante |
| **6** | **ADV_S** | Le nombre de messages publicitaires diffusés (broadcast) par CH aux nœuds. |
| **7** | **ADV_R** | Le nombre de messages publicitaires CH reçus des CHs |
| **8** | **JOIN_S** | Le nombre de messages de demande de jointure envoyés par les nœuds au CH |
| **9** | **JOIN_R** | Le nombre de messages de demande de jointure reçus par le CH des nœuds. |
| **10** | **SCH_S** | Le nombre de messages de diffusion de l'horaire de TDMA publicitaires envoyés aux nœuds. |
| **11** | **SCH_R** | Le nombre de messages d'horaire de TDMA reçus des CH |
| **12** | **Rank** | L'ordre de ce nœud dans le calendrier TDMA. |
| **13** | **DATA_S** | Le nombre de paquets de données envoyés d'un capteur à son CH. |
| **14** | **DATA_R** | Le nombre de paquets de données reçus de CH. |
| **15** | **DATA_Sent_TO_BS** | Le nombre de paquets de données envoyés au BS |
| **16** | **Dist_CH_To_BS** | La distance entre la CH et la BS. |
| **17** | **Send_code** | Le code d'envoi du cluster. |
| **18** | **Expaned Energy** | La quantité d'énergie consommée au cours de la ronde précédente. |
| **19** | **Attack type** | Type du nœud. C'est une classe de cinq valeurs possibles, Blackhole, Grayhole, Flood, et TDMA, et normal, si le nœud n'est pas un attaquant |

### III.6.3 Les attributs qui définissent chaque type d'attaque :

Dans notre travail, et d'après les études qui ont utilisé la même base de données, nous avons pu associer, pour chaque attaque, les attributs qui jouent un rôle crucial dans la détection et l'identification de ces attaques.

***Tableau III-3:Résumé des attributs de diverses attaques*** **[58]**

| Nom de l'attaque | Nom d'attribute |
|---|---|
| Grayhole | Time, Is_CH, who_CH, adv_r, join_r, Data_R data_sent_to_bs. |
| Blackhole | Time, Is_CH, who_CH, adv_r, join_r, Data_R, data_sent_to_bs. |
| Flooding | Is_CH, who_CH, adv_s, adv_r, data_sent_to_bs, Dist_CH_to_BS, consumed_energy. |
| TDMA | Time, Is_CH, join_r, sch_s, data_sent_to_bs. |

Le tableau 3 résume les attributs spécifiques à chaque type d'attaque étudié. Parmi ces attributs, nous avons identifié certaines caractéristiques communes. L'un de ces attributs est "data_sent_to_bs", qui indique la quantité de données envoyées au point d'accès (BS) par les nœuds capteurs. Cet attribut est pertinent dans toutes les attaques étudiées, car il permet de détecter les anomalies liées à l'envoi de données

Un autre attribut commun est consumed_energy qui représente la quantité d'énergie consommée par chaque nœud capteur pour fournir des informations au réseau. La consommation d'énergie est un indicateur important dans la détection d'anomalies, car les attaques peuvent souvent entraîner une utilisation excessive ou anormale de l'énergie par certains nœuds

. En se concentrant sur ces attributs clés, à savoir "data_sent_to_bs" et "consumed_energy", nous pouvons mieux comprendre les schémas de comportement et identifier les attaques potentielles dans les RCSF. Par exemple, une augmentation soudaine ou anormale de la quantité de données envoyées au BS ou une consommation d'énergie anormalement élevée peuvent indiquer la présence d'une attaque.

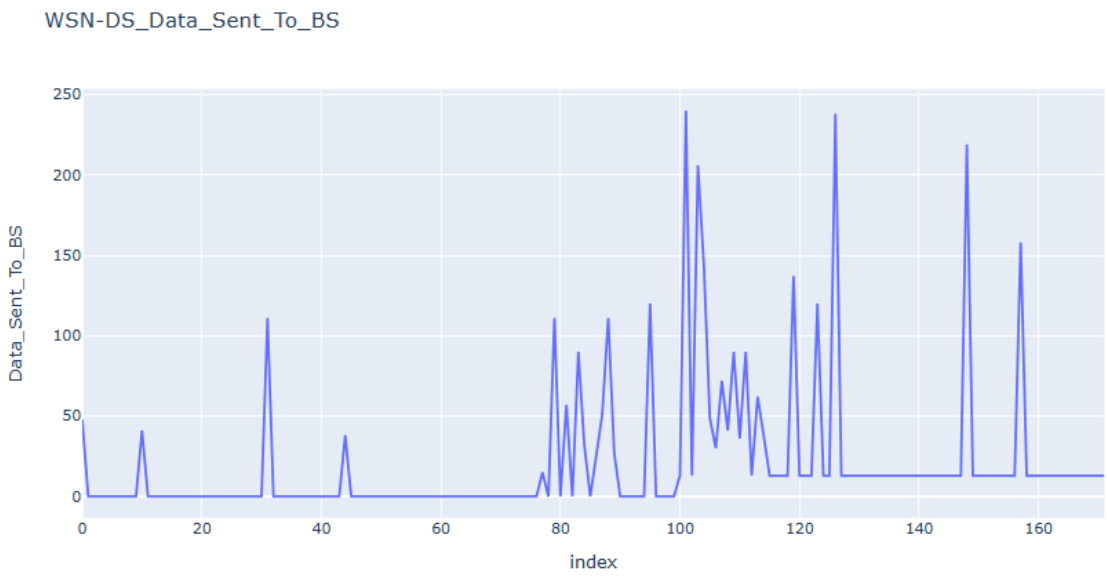


**Figure III-6 : graphique de data_sent_to_bs**

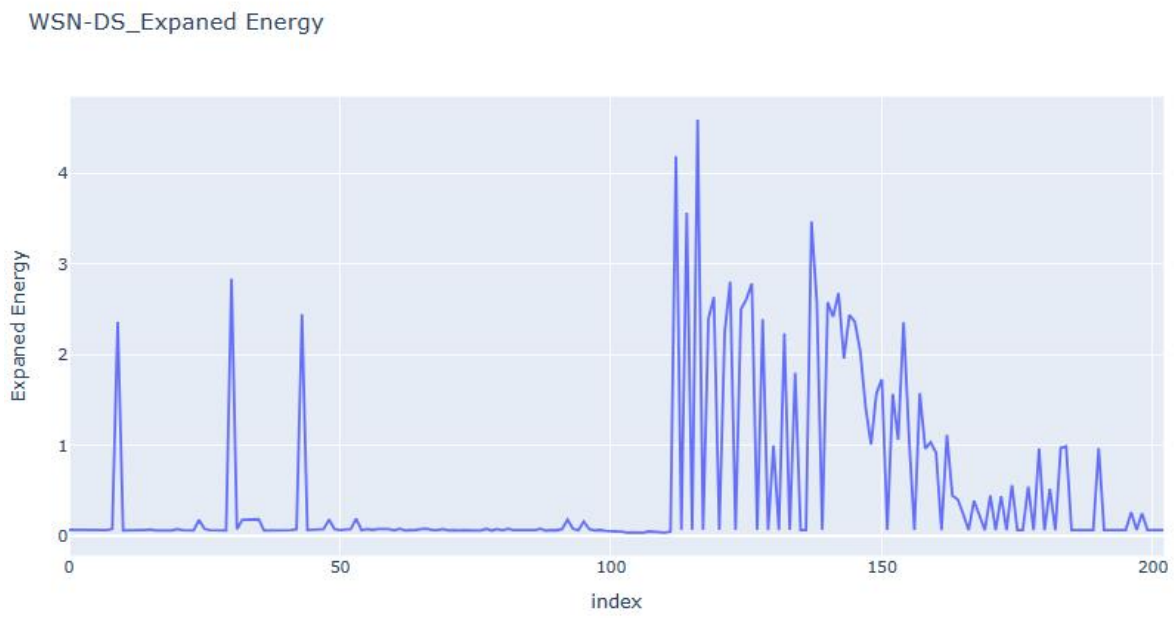


**Figure III-7 graphique de consumed_energy**

En utilisant des algorithmes de détection d'anomalies basés sur des séries temporelles et des techniques d'apprentissage automatique, nous pouvons analyser ces attributs pour identifier les modèles ou les écarts anormaux qui pourraient indiquer des attaques en cours ou imminentes. Cela permet de prendre des mesures proactives pour atténuer les menaces potentielles et maintenir l'intégrité et la sécurité du réseau de capteurs.

### III.6.4 Modèle des expérimentations :

Dans notre modèle d'expérience, nous avons commencés par accéder à la base de données WSN-DS et utilisons la bibliothèque PyCaret qui intègre des algorithmes d'apprentissage automatique. En utilisant ces algorithmes, nous extrayons les résultats de la détection d'anomalies et les représentons sous forme de séries temporelles. Cela nous permet d'analyser visuellement les anomalies détectées dans les données du RCSF, en fournissant des informations précieuses pour évaluer la sécurité et du réseau de capteurs sans fil

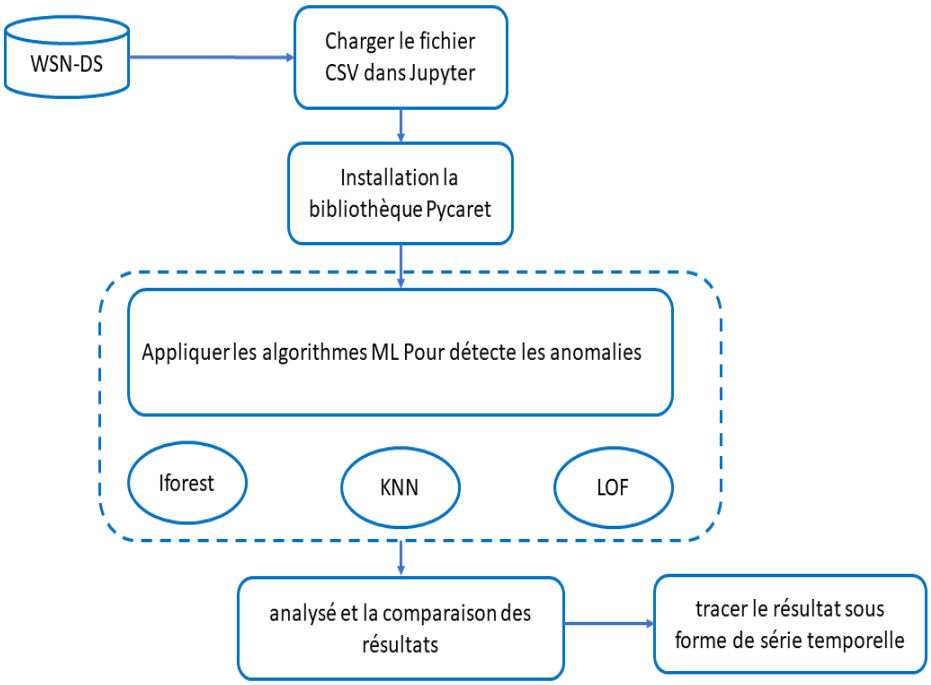


**Figure III-8: modèle des expérimentations**

# III.7 Implémentation :

## III.7.1 Langages et outils de développement :

Pour la réalisation de notre projet, nous avons utilisé les outils de développement suivants :

### III.7.1.1 Python : [60]

Python est le langage de programmation open source le plus employé par les informaticiens.

Ce langage s'est propulsé en tête de la gestion d'infrastructure, d'analyse de données ou dans le domaine du développement de logiciels. En effet, parmi ses qualités, Python permet notamment aux développeurs de se concentrer sur ce qu'ils font plutôt que sur la manière dont ils le font. Il a libéré les développeurs des contraintes de formes qui occupaient leur temps avec les langages plus anciens. Ainsi, développer du

code avec Python est plus rapide qu'avec d'autres langages. Les principales utilisations de Python par les développeurs sont :

• la programmation d'applications.

• la création de services web.

• la génération de code.

• la méta-programmation

### III.7.1.2 Anaconda Navigator :

Anaconda Navigator est une interface graphique de bureau (GUI) incluse dans la distribution Anaconda® qui vous permet de lancer des applications et de gérer facilement les paquets conda, les environnements et les canaux sans utiliser les commandes de la ligne de commande. Navigator peut rechercher des paquets sur Anaconda Cloud ou dans un dépôt local d'Anaconda. Il est disponible pour Windows, macOS et Linux **[61]**

### III.7.1.3 Jupyter notebook :

Python Notebook est maintenant connu sous le nom de Jupyter Notebook. Il s'agit d'une application Web open source qui vous permet de créer et de partager des documents contenant du code en direct, des équations, des visualisations et du texte narratif. Les utilisations incluent :

le nettoyage et la transformation des données, la simulation numérique, la modélisation statistique, la visualisation des données, l'apprentissage automatique et bien plus encore.

L'application Jupyter Notebook peut être exécutée sur un bureau local ne nécessitant aucun accès Internet ou peut être installée sur un serveur distant et accessible via l'Internet. **[62]**

### III.7.1.4 Les Bibliothèques utilisée

#### III.7.1.4.1 Pycaret

PyCaret est une bibliothèque d'apprentissage machine open-source à code bas et un outil de gestion de modèle de bout en bout intégré à Python pour automatiser les flux de travail d'apprentissage machine. Elle est incroyablement populaire pour sa

facilité d'utilisation, sa simplicité et sa capacité à construire et à déployer des prototypes d'apprentissage automatique de bout en bout rapidement et efficacement.

PyCaret est une autre bibliothèque à faible code qui peut être utilisée pour remplacer des centaines de lignes de code par quelques lignes seulement. Cela rend le cycle d'expérimentation exponentiellement rapide et efficace.

PyCaret est simple et facile à utiliser. Toutes les opérations effectuées dans PyCaret sont stockées séquentiellement dans un pipeline qui est entièrement automatisé pour le déploiement. Qu'il s'agisse de l'imputation de valeurs manquantes, de l'encodage à une touche, de la transformation de données catégorielles, de l'ingénierie des caractéristiques ou même de l'ajustement des hyperparamètres, PyCaret automatise tout cela. **[54]**

#### III.7.1.4.2 Pandas [63]

Pandas est une bibliothèque Python utilisée pour travailler avec des ensembles de données.

Il a des fonctions d'analyse, de nettoyage, d'exploration et de manipulation des données.

Le nom "Pandas" fait référence à la fois à "Panel Data" et à "Python Data Analyse" et a été créé par Wes McKinney en 2008.

Pandas nous permet d'analyser de grandes données et de tirer des conclusions basées sur des théories statistiques.

Pandas peuvent nettoyer des ensembles de données désordonnés et les rendre lisibles et pertinents. **[63]**

#### III.7.1.4.3 Plotly Express :

Plotly Express est un module haut niveau de la bibliothèque Plotly qui simplifie la création de graphiques interactifs en utilisant une syntaxe concise. Il permet de générer rapidement des visualisations courantes telles que les diagrammes en barres, les diagrammes circulaires et les diagrammes de dispersion, en automatisant de nombreux aspects du tracé. **[64]**

#### III.7.1.4.4 Plotly Graph Objects :

Plotly Graph Objects est un module de bas niveau de la bibliothèque Plotly qui offre un contrôle plus précis sur les graphiques. Il permet de créer des objets de graphiques individuels tels que des figures, des traces et des axes, en spécifiant les détails de chaque

composant. Il offre une flexibilité maximale pour personnaliser les graphiques et créer des visualisations complexes et personnalisées. **[64]**

## III.8 L'exécution du Scripts PyCaret :

La machine utilisée pour ce travail est un processeur Coré i5 avec une vitesse de 2.2 GHz et Jupyter Notebook est utilisé comme environnement de travail à l'aide du bibliothèque pycaret.

**Etape 1 :** Cela installera la bibliothèque pycaret. Une fois la bibliothèque installée, vous pouvez commencer à l'utiliser pour créer des modèles d'apprentissage automatique.

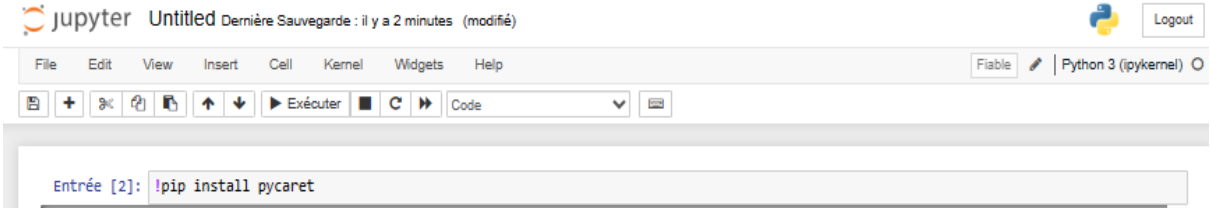


**Figure III-9: Installation pycaret**

**Etape 2 :** La bibliothèque Pandas est une bibliothèque Python populaire pour la manipulation et l'analyse de données. La bibliothèque Plotly Express est une interface de haut niveau pour créer des visualisations interactives avec Plotly. La bibliothèque Plotly Graph Objects fournit une API de niveau inférieur pour créer des visualisations

```
]: import pandas as pd
   import plotly.express as px
   import plotly.graph_objects as go
```

**Figure III-10: Importation de bibliothèques**

**Etape 3 :** Ce code va lire le fichier WSN-DS.csv et stocker les données dans un DataFrame Pandas.

La méthode head() sera ensuite utilisée pour afficher les 5 premières lignes du DataFrame

```
: data = pd.read_csv('WSN-DS.csv')
  data = pd.read_csv('data.csv')
  data = pd.read_csv('Energy.csv')

  data.head()
```

**Figure III-11: Importation des bases de données**

**Etape 4 :** L'instruction `from pycaret.anomaly import *` importe le module d'anomalie de la bibliothèque PyCaret. Le module d'anomalie contient des fonctions pour effectuer la détection d'anomalies.

L'instruction `setup(data, session_id=123)` initialise l'expérience de détection d'anomalies. Le paramètre `data` spécifie les données qui seront utilisées pour l'expérience, et le paramètre `session_id` spécifie un nombre aléatoire qui sera utilisé pour initialiser l'expérience de manière reproductible.

```
from pycaret.anomaly import *
s = setup(data , session_id = 123)
```

| | Description | Value |
|---|---|---|
| 0 | Session id | 123 |
| 1 | Original data shape | (374661, 19) |
| 2 | Transformed data shape | (374661, 24) |
| 3 | Numeric features | 18 |
| 4 | Categorical features | 1 |
| 5 | Preprocess | True |
| 6 | Imputation type | simple |
| 7 | Numeric imputation | mean |
| 8 | Categorical imputation | mode |
| 9 | Maximum one-hot encoding | -1 |
| 10 | Encoding method | None |
| 11 | CPU Jobs | -1 |
| 12 | Use GPU | False |
| 13 | Log Experiment | False |
| 14 | Experiment Name | anomaly-default-name |
| 15 | USI | e138 |

**Figure III-12: importe le module d'anomalie de la bibliothèque PyCaret**

**Etape 5 :** La fonction models() est un outil utile pour obtenir rapidement et facilement une liste des modèles de détection d'anomalies disponibles. Ces informations peuvent être utilisées pour sélectionner le meilleur modèle pour une tâche de détection d'anomalies particulière.

```
models()
```

| ID | Name | Reference |
|---|---|---|
| abod | Angle-base Outlier Detection | pyod.models.abod.ABOD |
| cluster | Clustering-Based Local Outlier | pyod.models.cblof.CBLOF |
| cof | Connectivity-Based Local Outlier | pyod.models.cof.COF |
| iforest | Isolation Forest | pyod.models.iforest.IForest |
| histogram | Histogram-based Outlier Detection | pyod.models.hbos.HBOS |
| knn | K-Nearest Neighbors Detector | pyod.models.knn.KNN |
| lof | Local Outlier Factor | pyod.models.lof.LOF |
| svm | One-class SVM detector | pyod.models.ocsvm.OCSVM |
| pca | Principal Component Analysis | pyod.models.pca.PCA |
| mcd | Minimum Covariance Determinant | pyod.models.mcd.MCD |
| sod | Subspace Outlier Detection | pyod.models.sod.SOD |
| sos | Stochastic Outlier Selection | pyod.models.sos.SOS |

***Figure III-13*: une liste des modèles de détection d'anomalies disponibles**

Nous avons vu dans de nombreux articles que les algorithmes Isolation Forest et *K-nearest neighbor*, Local Outlier Factor sont l'un des meilleurs algorithmes qui détecte les anomalies et donc nous avons expérimenté avec eux sur la base de données pour s'assurer que c'est vrai.

Isolation Forest (iForest) : L'isolation Forest (forêt d'isolation) est un algorithme d'apprentissage non supervisé de Machine Learning qui permet la détection d'anomalies dans un jeu de données (Data Set). Il isole les données atypiques, autrement dit celles qui sont trop différentes de la plupart des autres données. [65]

*K-nearest neighbor:* A supervised machine learning algorithm which is mostly used for classification. KNN classifies new cases based on similarity measures. KNN is effective for large training data. *[58]*

Local Outlier Factor (LOF) est un algorithme de détection d'anomalies non supervisé qui mesure la densité locale d'un point de données par rapport à ses voisins. Les points

de données avec une densité locale sensiblement inférieure à celle de leurs voisins sont considérés comme des valeurs aberrantes. [66]

**Etape 6 :** Le code crée d'abord un modèle de forêt d'isolement avec un taux de contamination de 0,2, ce qui signifie que 20 % des données sont susceptibles d'être des valeurs aberrantes. La function create_model() attribue ensuite le modèle à l'ensemble de données. La function assign_model() calcule les scores d'anomalie pour chaque ligne de l'ensemble de données et attribue une étiquette de 0 (valeur normale) ou de 1 (valeur aberrante) à chaque ligne. La fonction head() affiche les premières lignes des résultats, et nous appliquerons les d'autres algorithmes avec le même code.

```
iforest = create_model('iforest', fraction = 0.2)
iforest_results = assign_model(iforest)
iforest_results.head()
```

| | id | Time | Data_Sent_To_BS | Attack type | Anomaly | Anomaly_Score |
|---|---|---|---|---|---|---|
| 0 | 101000 | 50 | 48 | Normal | 1 | 0.073198 |
| 1 | 101001 | 50 | 0 | Normal | 0 | -0.072714 |
| 2 | 101002 | 50 | 0 | Normal | 0 | -0.072714 |
| 3 | 101003 | 50 | 0 | Normal | 0 | -0.072714 |
| 4 | 101004 | 50 | 0 | Normal | 0 | -0.072714 |

**Figure III-15: iforest résultat anomalies**

```
knn = create_model('knn', fraction = 0.2)
knn_results = assign_model(knn)
knn_results.head()
```

| | id | Time | Data_Sent_To_BS | Attack type | Anomaly | Anomaly_Score |
|---|---|---|---|---|---|---|
| 0 | 101000 | 50 | 48 | Normal | 1 | 10.0 |
| 1 | 101001 | 50 | 0 | Normal | 0 | 1.0 |
| 2 | 101002 | 50 | 0 | Normal | 0 | 1.0 |
| 3 | 101003 | 50 | 0 | Normal | 0 | 1.0 |
| 4 | 101004 | 50 | 0 | Normal | 0 | 1.0 |

*Figure III-14: KNN résultat t anomalies*

```
lof = create_model('lof', fraction = 0.2)
lof_results = assign_model(lof)
lof_results.head()
```

| | id | Time | Data_Sent_To_BS | Attack type | Anomaly | Anomaly_Score |
|---|---|---|---|---|---|---|
| 0 | 101000 | 50 | 48 | Normal | 1 | 1.117300e+00 |
| 1 | 101001 | 50 | 0 | Normal | 1 | 1.500000e+10 |
| 2 | 101002 | 50 | 0 | Normal | 1 | 1.050000e+10 |
| 3 | 101003 | 50 | 0 | Normal | 1 | 9.000000e+09 |
| 4 | 101004 | 50 | 0 | Normal | 1 | 9.000000e+09 |

**Figure III-16: lof résultat anomalies**

**Etape 7 :** Le code iforest_results.to_excel('iforest.xlsx') enregistrera le DataFrame iforest_results dans un fichier Excel nommé (iforest.xlsx).

```
iforest_results.to_excel("iforest.xlsx")
```

**Figure III-17: téléchargement de fichiers en Excel**

### III.8.1 Analyse des résultats :

Une fois que l'algorithme a détecté des anomalies dans la série de l'énergie consommée et les données, Pycaret permet l'exportation des résultats sous format de fichier Excel. Nous avons comparé les résultats de Pycaret qui se trouve dans la colonne « Anomaly » avec les données du Dataset. Comparons les attributs anormaux et attributs de type d'attaque. Les résultats sont présentés dans le tableau suivant :

**Tableau III-4: résultats des anomalies pour data_sent_to_bs**

| Type d'attaque | Nombres d'attaques | iForest | KNN | LOF |
|---|---|---|---|---|
| Blackhole | 10049 | 10049(100%) | 6168(61%) | 3272(33%) |
| Flooding | 3312 | 3312(100%) | 2670(80%) | 1246(38%) |
| Greyhole | 14596 | 14596(100%) | 12900(88%) | 3726(56%) |
| TDMA | 6638 | 6638(100%) | 2226(34%) | 4007(27%) |
| Normal | 340066 | 40336(12%) | 50968(15%) | 62682(18%) |

Le tableau montre les résultats (data_sent_to_bs anomalies) d'une expérience de détection d'anomalies utilisant trois algorithmes différents : iForest, KNN, Lof. L'expérience a été menée sur un ensemble de données d'attaques de trafic réseau, et le tableau indique le pourcentage de chaque type d'attaque qui a été correctement identifié par chaque algorithme.

Comme vous pouvez le voir, iForest était l'algorithme le plus performant, identifiant correctement 100 % des attaques de type Blackhole, 100 % des attaques de type Flooding,100 % des attaques de type Greyhole et 100 % des attaques de type TDMA. KNN était le deuxième algorithme le plus performant, identifiant correctement 61 % des attaques de type Blackhole, 80 % des attaques de type Flooding, 88 % des attaques de type Greyhole et 34 % des attaques de type TDMA. LOF était l'algorithme le moins performant, identifiant correctement 33 % des attaques de type Blackhole, 38 % des attaques de type Flooding, 56 % des attaques de type Greyhole et 27 % des attaques de type TDMA.

Dans l'ensemble, les résultats de cette expérience suggèrent que iForest est l'algorithme le plus efficace pour la détection d'anomalies dans les données de trafic réseau. Cependant, les trois algorithmes ont été capables d'identifier correctement un pourcentage significatif des attaques dans l'ensemble de données.

Voici une ventilation plus détaillée des résultats

Attaques de type Blackhole : iForest a identifié correctement 100 % des attaques de type Blackhole, tandis que kNN et LOF en ont identifié respectivement 61 % et 33 %.

Attaques de type Flooding : iForest a identifié correctement 100 % des attaques de type Flooding, tandis que kNN et LOF en ont identifié respectivement 80 % et 38 %.

Attaques de type Greyhole : iForest a identifié correctement 99 % des attaques de type Greyhole, tandis que kNN et LOF en ont identifié respectivement 88 % et 56 %.

Attaques de type TDMA : iForest a identifié correctement 98 % des attaques de type TDMA, tandis que kNN et LOF en ont identifié respectivement 34 % et 27 %.

Il est important de noter que ces résultats sont basés sur un seul ensemble de données et que les performances des algorithmes peuvent varier sur d'autres ensembles de données. Cependant, les résultats de cette expérience suggèrent que iForest est un algorithme prometteur pour la détection d'anomalies dans les données de trafic réseau.

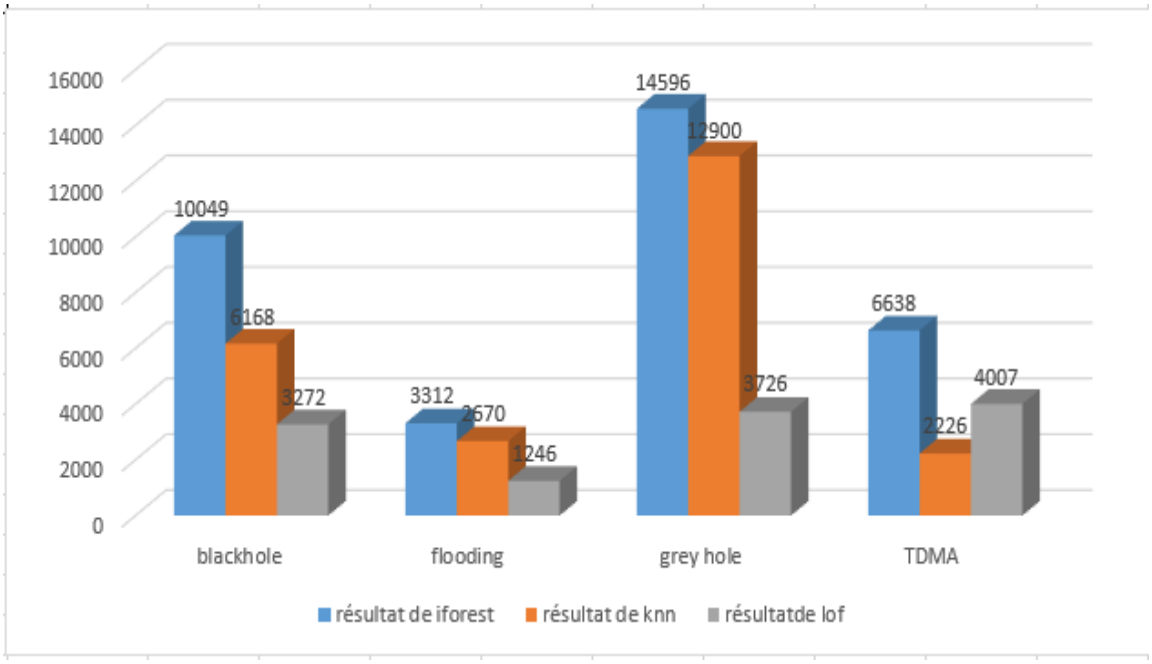


**Figure III-18: Histogramme Data_sent_to_bs.**

**Tableau III-5: résultats des anomalies pour Energy consumed.**

| Type d'attaque | Nombres d'attaques | IForest | KNN | LOF |
|---|---|---|---|---|
| Blackhole | 10049 | 10049(100%) | 7360(73%) | 2543(25%) |
| Flooding | 3312 | 3312(100%) | 2289(69%) | 743(22%) |
| Greyhole | 14596 | 14596(100%) | 12390(85%) | 1913(13%) |
| TDMA | 6638 | 6638(100%) | 2303(35%) | 4026(60%) |
| Normal | 340066 | 40336(12%) | 50590(15%) | 65708(19%) |

Le tableau (des anomalies pour Energy consumed) montre que iForest était l'algorithme le plus performant pour la détection des anomalies, iForest a correctement identifié 100 % des attaques de type Blackhole, 100 % des attaques de type Flooding, 100 % des attaques de type Greyhole et 100 % des attaques de type TDMA. KNN était le deuxième algorithme le plus performant, identifiant correctement 61 % des attaques de type Blackhole, 80 % des attaques de type Flooding, 88 % des attaques de type Greyhole et 34 % des attaques de type TDMA. LOF était l'algorithme le moins performant, identifiant correctement 33 % des attaques de type Blackhole, 38 % des attaques de type Flooding, 56 % des attaques de type Greyhole et 27 % des attaques de type TDMA.

Voici une ventilation plus détaillée des résultats :

Attaques de type Blackhole : iForest a identifié correctement 100 % des attaques de type Blackhole, tandis que kNN et LOF en ont identifié respectivement 61 % et 33 %.

Attaques de type Flooding : iForest a identifié correctement 100 % des attaques de type Flooding, tandis que kNN et LOF en ont identifié respectivement 80 % et 38 %.

Attaques de type Greyhole : iForest a identifié correctement 100 % des attaques de type Greyhole, tandis que kNN et LOF en ont identifié respectivement 88 % et 56 %.

Attaques de type TDMA : iForest a identifié correctement 98 % des attaques de type TDMA, tandis que kNN et LOF en ont identifié respectivement 34 % et 27 %.

Il est important de noter que ces résultats sont basés sur un seul ensemble de données et que les performances des algorithmes peuvent varier sur d'autres ensembles de données.

Cependant, les résultats de cette expérience suggèrent que iForest est un algorithme prometteur pour la détection des anomalies dans les données de trafic réseau.

Voici quelques réflexions supplémentaires sur les résultats :

iForest est l'algorithme le plus efficace pour détecter tous les types d'attaques.

KNN est un bon choix pour détecter les types d'attaques courants.

LOF est un bon choix pour détecter à la fois les types d'attaques rares et courants, mais il peut être moins efficace que iForest ou KNN.

Il est également important de noter que le pourcentage de trafic normal correctement identifié par chaque algorithme est relativement faible. Cela est dû au fait que le trafic normal est généralement plus difficile à distinguer du trafic anormal que le trafic d'attaque.

Dans l'ensemble, les résultats de cette expérience suggèrent que iForest est un algorithme prometteur pour la détection des anomalies dans les données de trafic réseau. iForest est l'algorithme le plus efficace pour détecter tous les types d'attaques, et il est également relativement efficace.

KNN est un bon choix pour détecter les types d'attaques courants et LOF est un bon choix pour détecter à la fois les types d'attaques

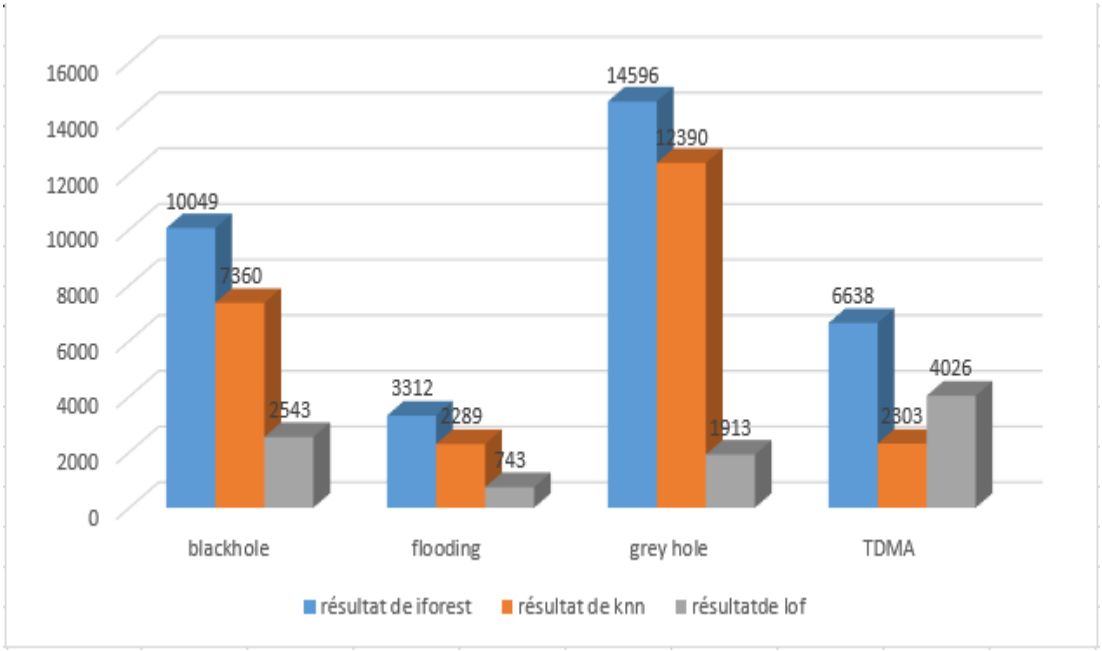


**Figure III-19: Histogramme Energy consumed.**

Attaque Normal : Concernant l'attribut normal est également considérée comme d'anomalies par les algorithmes de détection d'anomalies. Cela peut sembler contre-intuitif, car l'attribut normale devrait en réalité être classée comme normale et non comme une attaque.

Cependant, il est important de comprendre que les algorithmes de détection d'anomalies sont conçus pour identifier des schémas et des comportements anormaux dans les données, sans avoir de connaissance a priori des différentes classes ou types d'attaques.

Nombre d'attaques (data sent to bs) : 340 066

Iforest : 40336 (12%)

Knn :50 968 (15%)

Lof : 62 682 (18%)

Attaque Normale (Energy consumed) :

Nombre d'attaques: 340066

Iforest : 40336 (12%)

Knn :50 968 (15%)

Lof : 62 682 (19%)

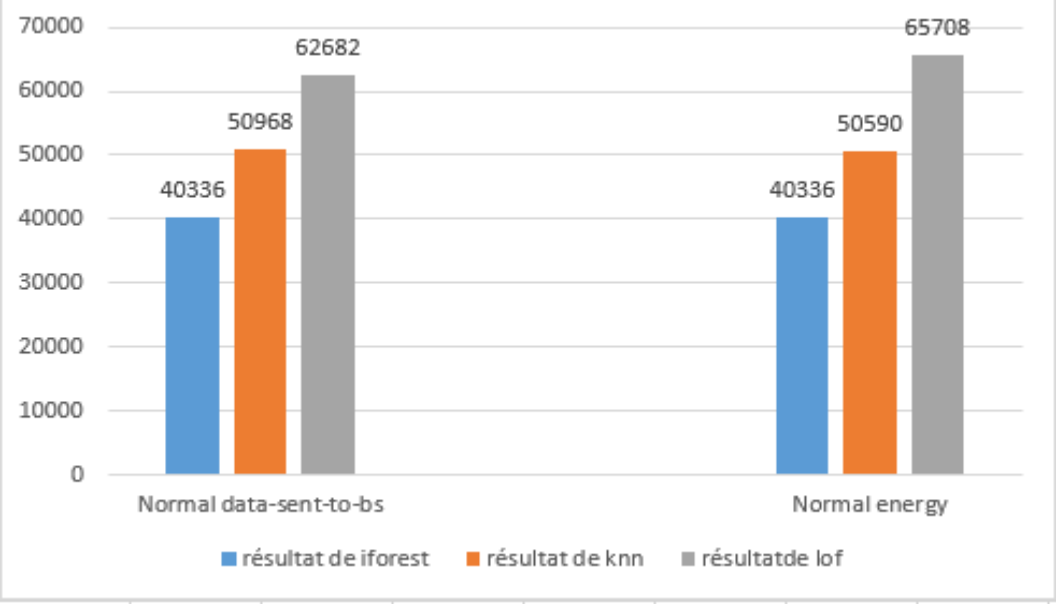


**Figure III-20: Histogramme normal Data_sent_to_bs et Energy consumed.**

**Etape 8:** Pour afficher l'anomalie dans un graphe série temporelle Data_sent_to_bs , nous avons utilisé le code suivant **:**

```
fig = px.line(iforest_results, x=iforest_results.index, y="Data_Sent_To_BS", title=' ANOMALY DETECTION Data_Sent_To_BS')


outlier_dates = iforest_results[iforest_results['Anomaly'] == 1].index
y_Data_Sent_To_BS = [iforest_results.loc[i]['Data_Sent_To_BS'] for i in outlier_dates]

fig.add_trace(go.Scatter(x=outlier_dates, y=y_Data_Sent_To_BS, mode = 'markers',
                name = 'Anomaly',
                marker=dict(color='red',size=10)))

fig.show()
```

**Figure III-21: Pour afficher l'anomalie dans un graphe série temporelle data_sent_to_bs**

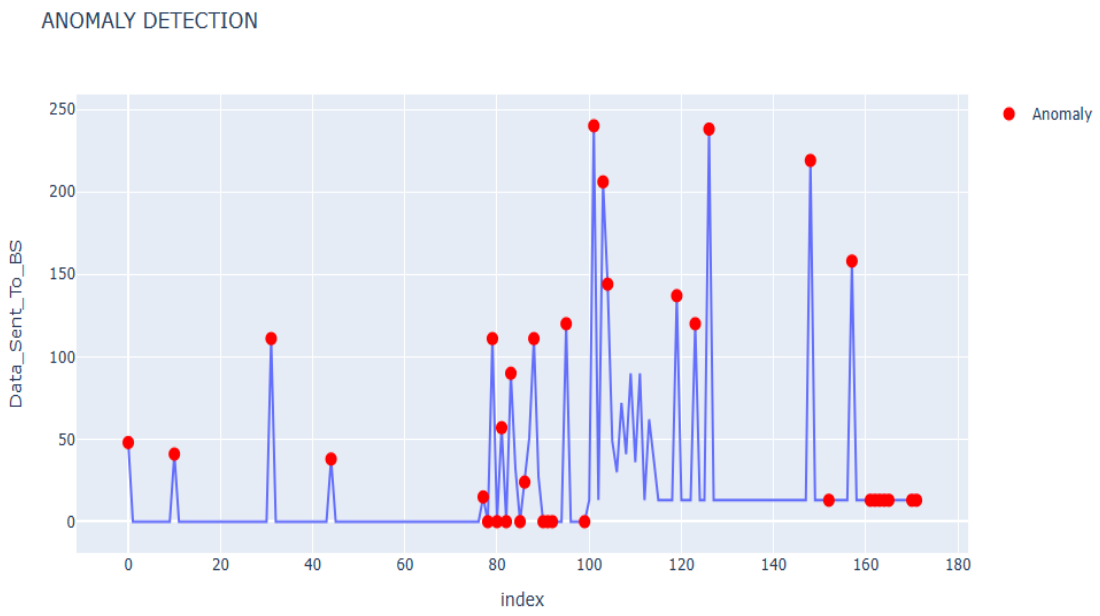


**Figure III-22: un graphe série temporelle data_sent_to_bs**

Pour afficher l'anomalie dans un graphe de série temporelle Expaned Energy, nous avons utilisé le code suivant **:**

```
fig = px.line(iforest_results, x=iforest_results.index, y="Expaned Energy", title=' ANOMALY DETECTION Expaned Energy')


outlier_dates = iforest_results[iforest_results['Anomaly'] == 1].index
y_ExpanedEnergy = [iforest_results.loc[i]['Expaned Energy'] for i in outlier_dates]

fig.add_trace(go.Scatter(x=outlier_dates, y=y_ExpanedEnergy, mode = 'markers',
                name = 'Anomaly',
                marker=dict(color='red',size=10)))

fig.show()
```

**Figure III-23: Pour afficher l'anomalie dans un graphe série temporelle Energy consumed**

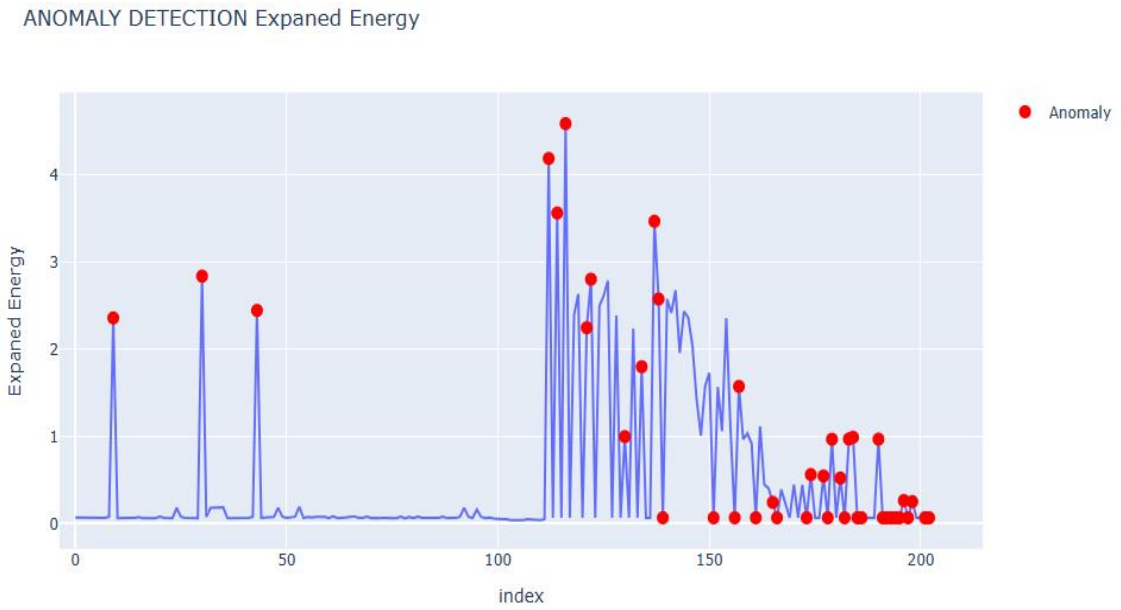


**Figure III-24: un graphe série temporelle Energy consumed**

Enfin, les premier et deuxième graphiques nous montrent une série temporelle avec des anomalies marquées par des points rouges. Cette anomalie indique une attaque ou un événement inhabituel. Par conséquent, ces graphiques fournissent une visualisation claire et concise des données, permettant aux analystes de détecter rapidement les anomalies et de prendre les mesures appropriées pour les résoudre.

## Conclusion générale :

Ce mémoire met en évidence les caractéristiques fondamentales des réseaux de capteurs sans fil ainsi que les besoins et les défis en matière de sécurité auxquels ils font face. Une attention particulière a été portée à la détection des attaques et des anomalies dans les séries temporelles générées par ces réseaux. Pour atteindre cet objectif, la bibliothèque PyCaret a été utilisée, intégrant des algorithmes d'apprentissage automatique et les séries temporelles adaptés à cette tâche.

La base de données WSN-DS, téléchargée à partir de Kaggle, a été utilisée dans cette étude. Cette base de données a été spécifiquement conçue pour les réseaux de capteurs sans fil et contient une variété d'attaques enregistrées telles que le trou noir, le trou gris, l'inondation et le TDMA. En se concentrant sur les champs de données "énergie consommée" et "données envoyées à la station", les algorithmes iForest, KNN et LOF de PyCaret ont été appliqués pour détecter les anomalies et les attaques présentes dans les séries temporelles.

L'utilisation de la bibliothèque PyCaret a permis de simplifier et d'accélérer le processus d'exploration des modèles, d'entraînement et d'évaluation. Les avantages des séries temporelles ont également été exploités pour représenter les résultats obtenus de manière visuelle et intuitive , les séries temporelles sont particulièrement adaptées à la représentation des données qui évoluent dans le temps, permettant ainsi d'identifier et d'analyser les anomalies et les comportements inhabituels dans les niveaux de puissance consommés et les données envoyées à la station.

En combinant les capacités de détection des algorithmes d'apprentissage automatique de PyCaret avec les avantages des séries temporelles, cette étude a contribué à une meilleure compréhension des attaques et des anomalies dans les réseaux de capteurs sans fil, offrant ainsi des perspectives pour renforcer la sécurité et la fiabilité de ces réseaux.

À l'avenir, on peut s'attendre à des améliorations dans l'utilisation de PyCaret pour les séries temporelles, notamment avec l'intégration de nouveaux algorithmes spécifiques aux RCSF, une meilleure visualisation des données temporelles, une validation croisée temporelle intégrée, et l'exploration de caractéristiques adaptées aux réseaux de capteurs sans fil. Ces avancées renforceraient la détection des attaques et des anomalies, améliorant ainsi la sécurité et la fiabilité des RCSF.

**Référence :**

[1] J.M. Kahn, R.H. Katz, and K.SJ Pister. Next century challenges : « mobile networking for smart dust. In Proceedings of the 5th annual ACM/IEEE international conference onMobile computing and networking», pages 271–278. AC 1999.

[2]Activityaware clustering algorithm for wireless sensor networ ksieeexplore.ieee.org, juillet 2014.

[3] SAHRAOUI Belkheyr, « Etude d'un protocole de routage basé sur les colonies de Fourmis dans les réseaux de capteurs sans fil », Mémoire de master, Université ABOU BAKR BELKAID, Tlemcen, 2013.

[4] Bendimerad Nawel , Système de surveillance d'infrastructures publiques à l'aide des réseaux decapteurs vidéo sans fil , These de doctorat, [année 2015].

[5] LEHSAINI Mohamed, «Diffusion et couverture basées sur le clustering dans les réseaux de capteurs : application à la domotique », Thèse de Doctorat, Université A.B Tlemcen et Université de Franche-Comté, Tlemcen, 2009.

[6] Mr. fares Abdelfatah, « Développement d'une bibliothèque de capteur sans fil », diplôme de master en informatique, université Montpellier 2, avril 2008

[7] Yacine CHALLAL, Hatem BETTAHAR, Abdelmadjid BOUABDALLAH, «Les Réseaux de capteurs (WSN: Wireless Sensor Networks) », Rapport interne, Université de Technologie de Compiègne, France, 2008.

[8] ulien BEAUDAUX,« Partitionnement logique dans les réseaux de capteurs sans fil», Mémoire de Master, Université de Strasbourg, Laboratoire des Sciences de l'Image de l'Informatique etde la Télédétection, 2010.

[9] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci. " A survey on sensor networks ". IEEE Communications Magazine, 40(8): 102-114, 2002.

[10] Mehdi Boullegue , Protocoles de communication et optimisationde l'énergie dans les réseaux de capteurssans fil , These de doctorat , Université Bretagne Loire [année 2016].

[11] DHIBEYA, «Routage avec QoS temps réel dans les réseaux de capteurs », Rapport de projet de fin d'études, École supérieure des communications, Tunis, 2007.

[12] TinyOs Community Forum. http ://www.tinyos.net/.

[13] http://www-igm.univ-mlv.fr/~dr/XPOSE2006/AurelieBunel/

[14] S. AKHENAK et R. ZEMOURI « Une architecture prédictible distribuée pour lagestion de l'énergie dans un réseau de capteurs sans fil »2012 /2013, Pages 26-30

[15] CASTELLUCCIA, Claude et FRANCILLON, Aurélien. Protéger les réseaux de capteurs sansfil. SSTIC08, 2008.

[16] Lamine, M. M. (2007). Securite dans les Reseaux de Capteurs Sans-Fil.*Memoire de Magistere en Informatique Ecole Doctorale d'Informatique de bejaia*, *2008*.

[17] Boukerche A. Werner Nelem Pazzia R.,Borges Araujo R. « Fault-tolerant wireless sensor network routing protocols for the supervision of contextaware physical environments> ;Journal of Parallel and distributed Computing,Avr 2006.-4 : Vol. 66.-pp.586-599

[18] Mémoire de magister thème : protocole de sécurité pour les Réseaux de capteurs Sans fil.

[19] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci. "Wireless sensor networks: a survey". *Computer Networks 38, Elsevier Science,* pp. 393–422, 2002.

[20] Zouinkhi, A. (2011). Contribution à la modélisation de produit actif communicant, Spécification et Evaluation d'un protocole de communication orienté sécurité des produits (Doctoral dissertation, Université Henri Poincaré-Nancy I).

[21] Boukerche A. Werner Nelem Pazzia R.,Borges Araujo R. « Fault-tolerant wireless sensor network routing protocols for the supervision of contextaware physical environments> ;Journal of Parallel and distributed Computing,Avr 2006.-4 : Vol. 66.-pp.586-599

[22] J. Haapola, Z. Shelby, C. Pomalaza-Raez, P. Mahonen, « Cross-layer energy analysis of multi-hop wireless sensor networks ». Proceeding of the 2nd European Workshop on Wireless Sensor Networks (EWSN'05), Istanbul, Turkey, January 31 - February 2, 2005. Pages 33 - 44.

[23]CHA 2008 Yacine CHALLAL, « Les réseaux de capteurs sans fil », Cours université Clermont, 2009.

[24]MAKHOUL Abdallah, « Réseaux de capteurs : localisation, couverture et fusion de données».Thèse de Doctorat, Université de Franche-Comté, 2008.

[25] Isabelle Guerin Lassous, « Réseaux Ad Hoc, réseaux de capteurs », Cours M2 Recherche RTS, RTS5, Page(s) : 05-10, Université de Lyon, 26 Septembre 2007.

[26] BENAMEUR Amina, « Mise en place d'un réseau de capteur sans fil pour l'irrigation intelligente, université Abou Bakr Belkaid Tlemcen 2012.

**[**27**]**Yacine CHALLAL, Hatem BETTAHAR, Abdelmadjid BOUABDALLAH, «Les Réseaux de capteurs (WSN: Wireless Sensor Networks) », Rapport interne, Université de Technologie de Compiègne, France, 2008.

**[**28**]**Guy Pujolle. "Les Reseaux". 5eme edition, 2006, ISBN : 2-212-11987-9.

[29] Tinyos. http:/www.tinyos.net/

[30] Berrachedi ,A , Diarbakirli, A, Sécurisation de protocole de routage hiérarchique LEACH dans les réseaux de capteurs sans fil. 2008/2009 disponible sur site : http://takeitesi.blogstop.com/

[31] Contiki. web site. http ://www.sics.se/contiki.

[32] Adam Dunkels, Bj• ornGr• onvall, and Thiemo Voigt. Contiki { a lightweight and exible operating system for tiny networked sensors. IEEE EmNetS-I, November2004. Swedish Institute of Computer Science.

[33] **:** S.A.H. SEDJELMACI, Mise en œuvre de mécanismes de sécurité basés sur les IDS pour les réseaux de capteurs sans fil, Thèse de doctorat, Février 2013.

[34] : Mémoire de Magistère en Informatique/Option : Réseaux et Systèmes

Distribués/Thème: Sécurité dans les Réseaux de Capteurs Sans-Fil/Présenté par/Messai

Mohamed Lamine/promo 2007/2008

[35] : ZERADNA, R., & CHORFI, I. (2022). L'utilisation de l'apprentissage automatique pour la détection des attaques Déni de Service (DOS) dans les réseaux de capteurs sans fil (Doctoral dissertation, Université Ibn Khaldoun-Tiaret-).

[36] : Monnet, Q. (2015). Modèles et mécanismes pour la protection contre les attaques par déni de service dans les réseaux de capteurs sans fil (Doctoral dissertation, Paris Est).

[37] : Mémoire de magistère dans Protocoles pour la Sécurité des Réseaux de Capteurs Sans Fil/ Université Hadj Lakhder-Batna/ réalise par Athmani samir /15-07-2018

[38] : Différence entre attaque active et attaque passive, consulter le : 07/04/2023

https://waytolearnx.com/2018/07/difference-entre-attaque-active-et- attaque-passive.html

[39]: Sen, J. (2010). A survey on wireless sensor network security. arXiv preprint arXiv:1011.1529.

[40] : D. E. BOUBICHE, « Une approche Inter-Couches (cross-layer) pour la Sécurité dans les RCSF. »,

Thèse de doctorat. Université de Batna 2., 2013.

[41]: S. C.-H. Huang, D. MacCallum, et D. Du, Éd., Network security. New York: Scott C.-H. Huang.

Springer, 2007.

[42] : Dhakne, A. R., & Chatur, P. N. (2017). Detailed Survey on attacks in wireless sensor network. In Proceedings of the International Conference on Data Engineering and Communication Technology: ICDECT 2016, Volume 2 (pp. 319-331). Springer Singapore.

[43]: Yadav, C., Raksha, K., Hegde, S. S., Anjana, N. C., & Sandeep, K. E. (2015). Security techniques in wireless sensor networks: a survey. International Journal of Advanced Research in Computer and Communication Engineering, 4(4), 2278-1021.

[44]: Wang, Y., Attebury, G., & Ramamurthy, B. (2006). A survey of security issues in wireless sensor networks.

[45] : Louiza, I. (2012). Système de détection d'intrusion hybride et hiérarchique pour les réseaux de capteur sans fil (Doctoral dissertation, Université Mouloud Mammeri). Figure

[46]: Alam, S., & De, D. (2014). Analysis of security threats in wireless sensor network. arXiv preprint arXiv:1406.0298.

[47]: Raymond, D. R., & Midkiff, S. F. (2008). Denial-of-service in wireless sensor networks: Attacks and defenses. IEEE Pervasive Computing, 7(1), 74-81.

[48]: K.-A. Shim, « A Survey of Public-Key Cryptographic Primitives in Wireless Sensor Networks », IEEE Commun. Surv. Tutor., vol. 18, no 1, p. 577-601, 2016.

[49]: https://blog.mailfence.com/fr/difference-chiffrement-symetrique-asymetrique/

[50] : Doumi, A. (2018). La Sécurité des Communications dans les Réseaux de Capteurs sans Fils (Doctoral dissertation, FACULTE DES MATHEMATIQUES ET DE L'INFORMATIQUE DEPARTEMENT D'INFORMATIQUE).

[51] : https://www.perplexity.ai/search/95cf456f-254a-491a-87e1-3f0687e47144?s=u

[52] : ZAZOUA, E. H. (2022). MODÈLES DE PRÉDICTION À APPRENTISSAGE AUTOMATIQUE POUR LES RÉSEAUX DE CAPTEURS SANS FIL À ÉNERGIE RENOUVELABLE.

[53] : Kraiem, I. B. (2021). *Détection d'anomalies multiples par apprentissage automatique de règles dans les séries temporelles* (Doctoral dissertation, Université de Toulouse-Jean Jaurès).

[54] : https://pycaret.gitbook.io/docs/learn-pycaret/official-blog/time-series-anomaly-detection-with-pycaret

[55] : https://adtk.readthedocs.io/en/stable/userguide.html

[56]: I. Almomani, B. Al-Kasasbeh, and M. Al-Akhras, "WSN-DS: A Dataset for Intrusion Detection Systems in Wireless Sensor Networks." Journal of Sensors, 2016. doi: 10.1155/2016/4731953.

[57]: https://www.kaggle.com/datasets/bassamkasasbeh1/wsnds

[58]: Rezvi, M. A., Moontaha, S., Trisha, K. A., Cynthia, S. T., & Ripon, S. (2021). Data mining approach to analyzing intrusion detection of wireless sensor network. *Indonesian J. Electric. Eng. Comput. Sci*, *21*(1), 516-523.

[59] : Ifzarne, S., Tabbaa, H., Hafidi, I., & Lamghari, N. (2021). Anomaly detection using machine learning techniques in wireless sensor networks. In *Journal of Physics: Conference Series* (Vol. 1743, No. 1, p. 012021). IOP Publishing.

**[60]** RÉ ; Daction, L. (2020, 31 mars). *Python : définition et utilisation de ce langage informatique*. https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1445304-python-definition-et-utilisation-de-ce-langage-informatique/

**[61] :** Aib, B. (2019). Vers une détermination des vrais influenceurs sur les réseaux sociaux les réseaux sociaux.

**[62]:** Project *Jupyter*. (2021). Home. https://jupyter.org/

**[63]:** https://Www.W3schools.Com/Python/Pandas/Pandas_intro.Asp

**[64]:** https://plotly.com/python/graph-objects/

**[65] :** https://metalblog.ctif.com/2022/10/03/la-detection-danomalies-en-machine-learning-nonsupervise/#:~:text=L'isolation%20Forest%20(for%C3%AAt%20d,la%20plupart%20des%20autres%20donn%C3%A9es.

**[66] :** Scikit-learn documentation : https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.LocalOutlierFactor.html