République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université A. Mira de Béjaïa
Faculté des Sciences Exactes
Département d'Informatique

Mémoire de fin d'étude

En vue d'obtenation du diplôme de Master Professionnel en Informatique

Option : Administration et Sécurité des Réseaux (ASR)

THÈME

Étude de l'impact de l'attaque par rang sur le protocole RPL dans l'IoT avec détection par apprentissage automatique.

Réalisé par :

M^{lle} MEZIANE Kenza.
M^{r} LAFKI Hichem.

Encadré par :

M^{me} OUYEHIA Samira.

Membres du jury :

Président :	M^{r} YAZID Mohand	Université A. Mira Béjaïa
Examinatrice :	M^{me} SABRI Salima	Université A. Mira Béjaïa
Examinateur :	M^{r}FARAH Zoubeyr	Université A. Mira Béjaïa
Examinatrice :	M^{me} BACHIRI Lina	Université A. Mira Béjaïa

Promotion : 2024/2025

Remerciements

Avant de présenter notre travail,

Nous tenons à remercier **ALLAH** de nous avoir aidé et donné le courage et la volonté pour réaliser ce travail et aboutir à son accomplissement.

Nos remerciements les plus sincères vont à **Madame OUYEHIA Samira**, notre encadrante, pour avoir accepté de nous accompagner dans ce travail. Nous la remercions pour sa disponibilité, ses conseils précieux et son aide tout au long de la réalisation de ce mémoire. Son soutien a été essentiel pour nous permettre d'avancer.

Nos vifs remerciements aux membres de jury pour l'honneur qu'ils nous font en acceptant d'examiner et d'évaluer ce travail.

Nous tenons enfin à remercier nos chers parents et toute personne pour l'encouragement et le soutien qui nous ont apportés durant ce travail et tous ceux qui ont contribué de près ou de loin à sa réussite.

Merci infiniment à tous.

Dédicace

Je Dédie Ce Travail,
À ma mère,l'amour de ma vie, celle qui m'a offert son amour et son soutien inconditionnels tout au long de ma vie et tout au long de mon parcours vers la réussite.

À mon père,la source de tout pour moi, celui qui s'est sacrifié pour notre bien-être, et qui m'a donné la force et le courage de devenir la personne que je suis aujourd'hui.

À ma grand-mère Imma Mamah, puisse Allah te donner une longue vie. Merci pour ton amour et tes prières constantes.

À ma sœur jumelle, Amel,ma moitié dans cette vie, celle avec qui j'ai tout partagé depuis toujours, merci d'être toujours là.

À mes oncles et tantes,en particulier mon oncle Abed. Karim, ma tante Karima aussi ma meilleur tante Nadia et son mari Ali,merci pour votre présence .

À tous mes proches, mes cousins et cousines, et tout spécialement Yacine, Lounis, Fazia, Nassima et Hannane, Naima, Werida, merci pour votre soutien et vos encouragements tout au long de ce chemin.

À mes amies Salwa, Kenza, Bina, Ayalar et Imane, merci pour votre amitié précieuse et votre soutien. Et tout particulièrement à mes meilleures amies, Cylia et Linda, avec qui j'ai partagé mon enfance et ma jeunesse,
Et à tous les autres amis avec qui j'ai vécu mes plus beaux moments pendant mon parcours universitaire, malheureusement la liste est longue, mais merci à chacun pour ces instants inoubliables.

Au final, je le dédie pour mon binôme Hichem, avec qui j'ai partagé ce parcours difficile mais accompli.

MEZIANE Kenza

Dédicace

Je dédie cet événement marquant dans ma vie à la mémoire de mon cher grand-père, que Dieu l'accueille dans Son vaste paradis.

À mes très chers parents, maman et papa, pour tous leurs sacrifices, leur amour, leur tendresse, leur soutien et leurs prières tout au long de mes études. Que Dieu leur accorde une longue et joyeuse vie, ainsi qu'à ma grand-mère.

À ma chere et unique sœure Hanane, à mes chères frères Walid et Faize pour leurs encouragements permanents.

À mes oncles, mes tantes, mes proches et tous mes cousins et cousines,

À tous mes professeurs tout au long de mon parcours scolaire et universitaire.

À mes amis et collègues qui m'ont toujours encouragé, et à qui je souhaite beaucoup de succès,

Au final, je le dédie à ma chère amie, ma binôme Kenza.Un grand merci d'avoir été une collègue formidable et pour sa patience, sa bonne humeur et sa précieuse contribution tout au long de cette aventure.C'est un bon souvenir partagé. Je souhaite que nous réalisions tous nos rêves.

Merci pour tout !
Hichem

TABLE DES MATIÈRES

TABLE DES FIGURES

LISTE DES TABLEAUX

Liste des abréviations

6LoWPAN IPv6 Low power Wireless Personal Area Network

ARM Advanced RISC Machine

CSV Comma-Separated Values

DAO Destination Advertisement Object

DAO-ACK Destination Advertisement Object Acknowledgement

DAE Deep Autoencoder

DAG Directed Acyclic Graph

DANN Deep Artificial Neural Network

DIO DODAG Information Object

DIS DODAG Information Solicitation

DODAG Destination Oriented Directed Acyclic Graph

DT Decision Tree

ECLAT Equivalence Class Clustering and bottom-up Lattice Traversal

ETX Expected Transmission Count

FN Faux Négatif

FP Faux Positif

IDS Intrusion Detection System

IEEE Institute of Electrical and Electronics Engineers

IETF Internet Engineering Task Force

IoT Internet of Things

IoTR-DS Internet of Things RPL-based Data Set

IPv6 Internet Protocol Version 6

KNN K-Nearest Neighbors

LLN Low Power and Lossy Network

MIT Massachusetts Institute of Technology

ML Machine Learning

MRHOF Minimum Rank with Hysteresis Objective Function

OF Objective Function

OF0 Objective Function Zero

PDR Packet Delivery Rate

RFC Request For Comments

RFID Radio Frequency Identification

RIA Increased Rank Attack

RIAI-DRPL Rank Increase Attack Identification – Detection in RPL

ROLL Routing Over Low-power and Lossy networks

RPL Routing Protocol for Low-Power and Lossy Networks

SBIDS Supervised-Based Intrusion Detection System

SVM Support Vector Machine

TP Vrai Positif

TN Vrai Négatif

UDP User Datagram Protocol

VeRA Version Number and Rank Authentication

WSN Wireless Sensor Networks

Introduction générale

Contexte

L'internet des objets représente l'un des axes les plus dynamiques de l'évolution technologique actuelle. Il connait une croissance exponentielle, avec des milliards d'objets connectés déployés dans des domaines variés.

Ces objets communiquent souvent dans des réseaux sans fil limités en énergie et en mémoire et en puissance. Ces réseaux sont appelés LLN (Low-power and Lossy Networks).

Dans ces réseaux contraints, le protocole RPL (Routing Protocol for Lower-Power and Lossy Networks) a été standardisé par l'IETF pour assurer un routage efficace et fiable.

Toutefois, en raison de leurs limitations inhérentes, les réseaux IdO restent particulièrement vulnérables aux attaques, notamment celles ciblant le protocole RPL. Parmi les attaques identifiées, les attaques par rang (ou Rank Attacks) sont particulièrement dangereuses, car elles exploitent les mécanismes de calcul du rang dans RPL pour perturber le routage, dégrader les performances réseau, et épuiser les ressources des nœuds.

Dans le cadre de notre projet, nous avons examiné l'effet de l'attaque par rang sur le protocole RPL en effectuant une série de simulations dans l'environnement Cooja avec Contiki OS. Nous avons mis en place un scénario où un nœud malintentionné manipule sa valeur de rang pour modifier la topologie du réseau. Pour évaluer l'impact de cette attaque, nous avons surveillé deux indicateurs principaux : la consommation énergétique moyenne et le volume des messages de contrôle échangés.

Ensuite, nous avons récupéré les données produites par la simulation pour les employer dans une étape de détection automatisée grâce à l'apprentissage automatique. Nous avons procédé à l'entraînement de divers modèles supervisés, tels que la régression logistique, la forêt aléatoire (Random Forest), le k-plus proches voisins (KNN) et les machines à vecteurs de support (SVM). Chaque modèle a été examiné en fonction de critères de performance tels que la précision, le rappel et l'indice F1, afin d'établir la méthode la plus performante pour identifier cette attaque au sein d'un réseau IdO.

Ce projet combine ainsi simulation réseau et intelligence artificielle, afin de proposer une méthode complète pour l'analyse et la sécurisation du protocole RPL face aux attaques par rang.

Problématique

La croissance exponentielle des dispositifs connectés a rendu les réseaux IdO séduisants pour les attaquants. Le protocole RPL, bien qu'optimisé pour les réseaux LLN, présente des vulnérabilités qui peuvent être exploitées à travers diverses at-

taques, dont les attaques par rang.

Ces attaques falsifient la valeur de leur rang d'un nœud afin d'attirer le trafic réseau. Cela compromet la topologie du réseau, augmente le nombre de messages de contrôle, réduit la durée de vie des nœuds, et affecte la fiabilité globale du système.

Il est donc important de comprendre comment ces attaques fonctionnent, de mesurer leur effet sur le protocole RPL, et de mettre en place des mécanismes de détection et d'atténuation efficaces.

Objectif

L'objectif principal de ce mémoire est de comprendre le protocole de routage RPL dans l'IdO, d'étudier comment les attaques par rang affectent ce protocole, de simuler ces attaques pour voir leur impact et de proposer une approche de détection intelligente de ces attaques à l'aide de techniques d'apprentissage automatique supervisé.

Organisation de la mémoire

Le présent travail est structuré en quatre chapitres, repartis entre une partie théorique et une partie pratique :

- **Chapitre 1 : Généralités et contexte sur l'Internet des Objets (IdO)**

  Ce chapitre présente les concepts de base associés à l'IdO : les définitions, le contexte historique, les technologies employées, les protocoles, les champs d'application, ainsi qu'une présentation des réseaux LLN (Low-power and Lossy Networks).

- **Chapitre 2 : Protocole RPL**

  Ce chapitre se dédie à l'étude détaillé du protocole RPL. Nous exposons le fonctionnement du protocole, l'architecture DAG/DODAG, les messages de contrôle (DIO, DAO, DIS), les fonctions objectives et les restrictions liées à la sécurité du protocole.

- **Chapitre 3 : Étude de l'attaque par rang et détection via machine learning**

  Ce chapitre aborde les différentes attaques visant le protocole RPL, notamment les attaques par rang (diminution, augmentation, WPS). Nous y présentons aussi les fondements du machine learning (apprentissage supervisé et non supervisé) et une revue de littérature résumant huit articles : quatre axes sur les attaques RPL, et quatre sur les techniques de détection.

- **Chapitre 4 : Simulation d'attaque, capture des données et détection avec des modèles d'apprentissage automatique**

Dans ce dernier chapitre, nous décrivons la simulation des attaques de rang (diminution et augmentation) à l'aide de Cooja. Deux métriques de performance sont utilisées : la consommation énergétique moyenne et le nombre de messages de contrôle, Nous générons un dataset à partir de la simulation de l'attaque par diminution de rang, les données extraites sont converties en format CSV, quatre algorithmes d'apprentissage supervisé sont appliqués pour détecter cette attaque. dont régression logistique et la forêt aléatoire et K plus proches voisins, arbre de décision. Ces modèles sont entraînés sur Google Colab, avec des métriques d'évaluation comme le taux de précision et de rappel et le F1-Score.

Enfin nous terminerons avec une conclusion générale, ainsi que quelques perspectives pour des travaux futurs.

.

CHAPITRE 1

GÉNÉRALITÉS ET CONTEXTE SUR L'IDO

1.1 Introduction

Le concept de l'Internet des Objets (IdO) se réfère à un réseau mondial d'appareils connectés, où le nombre d'équipements en ligne dépasse celui des utilisateurs humains. Chaque objet possède une adresse distincte, lui donnant la possibilité de communiquer des informations. Que l'on parle d'ordinateurs, de capteurs, de dispositifs RFID (Radio Frequency Identification) ou de téléphones portables, ces appareils ont la capacité non seulement d'émettre des informations, mais également de recevoir des instructions. Dans ce chapitre, destiné à présenter un aperçu général sur l'IdO, nous traiterons des bases essentielles de l'IdO comme la définition de cette technologie émergente et son évolution historique. Nous examinerons ensuite l'architecture des IdO, les divers composants et technologies associés, ainsi que les différents protocoles de l'IdO.

1.2 L'internet des Objets

Plusieurs définitions ont été données à l'internet des objets ou "Internet of Things (IoT)", en anglais, vu qu'il est assez difficile de capturer l'essentiel de l'IdO dans une seule définition unique. Dans ce qui suit, nous allons présenter quelques définitions données à l'IdO qui a été défini par différents auteurs de différentes manières.

Weill et Souissi caractérisent l'IdO en ces termes : « L'Internet des objets est une extension de l'Internet actuel à tous les objets pouvant communiquer, de manière directe ou indirecte, avec des équipements électroniques eux-mêmes connectés à l'Internet » **[01]**.

Dans **[02]**, les auteurs ont défini l'IdO comme : « un réseau de réseaux qui permet, via des systèmes d'identification électronique normalisés et unifiés, et des dispositifs

mobiles sans fil, d'identifier directement et sans ambiguïté des entités numériques et des objets physiques et ainsi de pouvoir récupérer, stocker, transférer et traiter, sans discontinuité entre les mondes physiques et virtuels, les données s'y rattachant. »

Les auteurs de la référence **[03]** définissent l'Internet des Objets (IdO) comme un ensemble d'entités physiques, telles que les appareils, les automobiles et les infrastructures, qui intègrent des capteurs, des logiciels et une liaison réseau. Ces dispositifs sont reliés et échangent des informations via Internet selon des protocoles de communication prédéfinis, assurant un transfert de données sûr et performant entre différents équipements.

1.3 Historique

En 1990, Le premier objet connecté était un grille-pain télécommandé et des cafetières, marquant le début de l'intégration des objets du quotidien dans le domaine numérique. En 1999, Kevin Ashton, gestionnaire de marque chez Procter and Gamble (directeur exécutif de Massachusetts Institute of Technology (MIT) Auto-ID Labs), a employé le terme « Internet des objets » pour la première fois pour décrire la connexion entre la technologie RFID et l'Internet. Il déclara lors d'une réunion du groupe : « Si nous parvenons à ajouter l'identification par fréquence radio et d'autres capteurs aux objets de la vie quotidienne, nous pourrons alors créer un Internet des objets et poser les fondations d'une nouvelle ère de la perception par les machines **[04]**. »
Ce concept a connu une évolution remarquable portée par l'intégration continue de technologies de pointe et la création d'objets toujours plus intelligents. Cela conduit à l'établissement progressif d'un réseau mondial d'objets interconnectés.

1.4 Les composants de l'IdO

L'objet connecté, doté d'une fonction mécanique et/ou électrique particulière, constitue le cœur de l'IdO. Il peut être conçu pour être connectable ou avoir une connectivité ajoutée ultérieurement. L'objet recueille et gère des informations provenant de capteurs, transmet ces données et reçoit des directives pour réaliser une tâche. Pour ces fonctionnalités, il faut généralement une source d'énergie, surtout si les données sont prétraitées dans l'objet **[05]**.
L'Internet des Objets se compose de cinq éléments :

1. **Capteur :** Les capteurs sont des dispositifs qui convertissent une information physique (comme la température, la pression, le débit, la luminosité, le mouvement, etc.) en un signal électronique. Ils sont installés sur les objets connectés. Ils possèdent une intelligence variable, selon leur capacité à intégrer eux-mêmes des algorithmes d'analyse de données et leur degré d'auto-adaptabilité. Il existe des capteurs simples : température, lumière, pression, fumée, énergie, etc ; et d'autres complexes : inertiels, magnétomètre, GPS (localisation), biométrique, etc **[05]**.

2. **Actionneurs :** L'actionneur est un appareil matériel qui convertit une donnée numérique en un phénomène physique. Il permet la supervision ou l'ajustement de l'état d'un objet matériel dans le monde réel, basé sur les informations obtenues de divers appareils IdO.
   Exemple d'actionneurs : haut-parleurs, interrupteurs, afficheurs, alarmes, vannes, ventilateurs, etc **[05]**.

3. **Connectivité :** Les objets IdO possèdent une antenne Radio Fréquence (RF) qui leur donne la capacité de se connecter aux réseaux et de transmettre des informations comme leur identification, leur état, des alertes et des données provenant de capteurs. De plus, ils ont la capacité de recevoir des données et des instructions en réponse. Le module de connectivité prend en charge la gestion du cycle de vie de l'objet ; en d'autres termes, cela concerne l'authentification, la mise à jour et l'effacement de l'élément du réseau, etc **[05]**.

4. **Énergie :** La contrainte principale à laquelle sont confrontés les capteurs est celle de l'énergie. On évalue l'autonomie des nœuds en nombre d'années.

5. **Réseaux de capteurs :** Les capteurs possèdent des mécanismes sans fil pour transmettre et recevoir des informations. Cependant, cela ne suffit pas pour rendre un groupe de capteurs accessible et interopérable. Pour ce faire, il est nécessaire que les capteurs se structurent en un réseau de capteurs, caractérisé par des composants très miniaturisés dotés de capacités de transmission sans fil **[06]**.

1.5 Architecture de l'IdO

Les chercheurs ont suggéré une multitude d'architectures distinctes. D'après certains spécialistes, l'architecture de l'IdO se compose de trois couches, alors que d'autres la conçoivent en quatre couches. Ils estiment que, grâce aux progrès dans l'IdO, l'architecture à trois niveaux n'est pas en mesure de répondre aux besoins des applications. Face à un enjeu lié à la sécurité et à la confidentialité dans l'IdO, une architecture en cinq niveaux a également été suggérée .

Les architectures à trois, quatre et cinq couches sont illustrées dans la Figure 1.2 afin de comparer leurs structures et les améliorations successives apportées **[07]**.

1.5.1 Architecture à trois couches

C'est une architecture très élémentaire qui adhère au concept fondamental de l'IdO. Elle est constituée de trois niveaux, nommés perception, réseau et application. La figure 1.1 présente cette Architecture à trois couches ainsi que des exemples illustratifs pour chaque niveau **[11]** .

- **La couche de perception :** Cette couche constitue la couche physique de l'architecture de l'Internet des Objets.Elle est composée de divers dispositifs tels que les capteurs (RFID, caméras, capteurs de température, etc.), la nano-technologie et d'autres technologies de marquage. Le but principal de la couche est l'identification des objets uniques et la collecte d'informations du monde physique à l'aide de ses capteurs **[08]**.

- **La couche réseau :** aussi appelée « couche de transmission », est au cœur de l'IdO . Cette couche a pour objectif de transmettre les informations recueillies depuis la couche de perception vers le système de traitement d'informations spécifique, via les réseaux de communication existants comme Internet , les réseaux mobiles, réseau de capteurs sans fil, ou tout autre réseau fiable. Il contient les instrumentations logicielles et matérielles du réseau internet en plus de la centres de gestion et d'information **[09]**.

- **La couche application :** Cette couche met en œuvre plusieurs applications pratiques de l'IdO selon les besoins des utilisateurs et les divers secteurs tels que la domotique, l'écologie intelligente, le transport intelligent ou encore l'hôpital intelligent. Elle peut être perçue comme un stade intermédiaire entre les technologies industrielles et leur maîtrise pour répondre aux exigences humaines **[10]**.

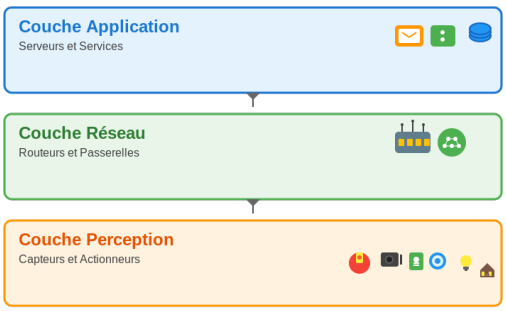

FIGURE 1.1 – L'architecture en trois couches de l'IdO.

1.5.2 Architecture à quatre couches

Du fait de l'évolution constante de l'IdO et des vulnérabilités inhérentes au système à trois couches, cette structure ne pouvait pas satisfaire toutes les demandes de l'IdO. Ainsi, les chercheurs ont suggéré une structure en quatre couches, reprenant les trois niveaux classiques (perception, réseau et application) et y ajoutant une couche de support. L'objectif de la création de cette couche est d'assurer la sécurité. La couche de support a deux fonctions principales. Elle assure que les données proviennent d'un utilisateur authentique et utilise des techniques de vérification

d'identité pour éviter les menaces. La deuxième tâche consiste à transmettre des informations vers la couche réseau **[07]**.

1.5.3 Architecture à cinq couches

L'architecture à quatre niveaux a également présenté des problèmes de sécurité et de stockage. Une architecture à cinq niveaux a été suggérée par les chercheurs. Elle a trois couches, comme les architectures précédentes, dont les noms sont la couche de perception, la couche de transport qui semble comme la couche réseau dans l'architecture à 3 couches et la couche d'application. Elle a en plus deux couches, la couche processing et la couche business **[07]**.

- **La chouche processing :** On la désigne souvent comme une couche middleware, qui a pour fonction de rassembler les informations issues de la couche de transport. Cette couche se charge de la gestion des services et dispose d'un lien vers la base de données. Elle réalise le traitement des données et les calculs omniprésents, faisant ainsi des décisions automatiques basées sur les résultats obtenus.

- **La couche business :** Cette couche est chargée de la gestion complète du système IdO. Il élabore des modèles commerciaux, des schémas, des structures d'organisation et bien plus encore, en se basant sur les données issues de la couche Application. En outre, il a la compétence de préciser comment créer, stocker et modifier les informations **[08]**.

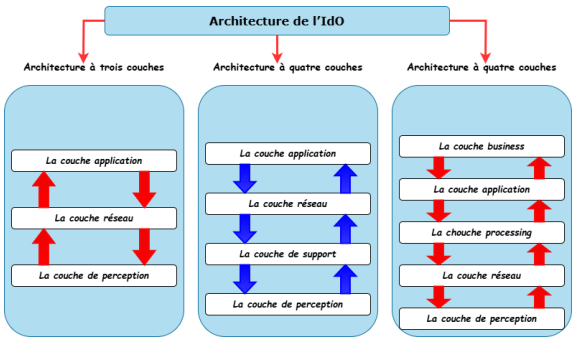

FIGURE 1.2 – Les architectures en couches de l'Ido (trois, quatre et cinq couches) .

1.6 Technologies de l'IdO

Un système IdO réunit de nombreux acteurs et composants technologiques qui assurent son bon fonctionnement . Celles-ci révolutionnent notre manière d'interagir avec le monde en connectant des objets physiques à internet, leur permettant de collecter et d'échanger des données en temps réel.

1. **RFID**

   La technologie RFID (Radio Frequency Identification) est une méthode de communication sans fil utilisée pour l'identification et le suivi d'objets.

Elle repose sur l'utilisation de puces électroniques, appelées tags RFID, qui émettent des informations lorsqu'elles sont interrogées par un lecteur RFID via des ondes radio. Deux types de puces RFID existent : les puces actives, capables de communiquer avec leur environnement en toute autonomie (grâce à une batterie) et les puces passives (qui ont besoin de recevoir ponctuellement de l'énergie électromagnétique pour pouvoir communiquer). Selon la bande de fréquence utilisée et la manière dont la puce a été intégrée à l'objet, leur portée peut varier de quelques centimètres à plusieurs mètres.
Plusieurs organismes de standardisation s'occupent de définir comment fonctionnent les RFID (étiquettes ou systèmes d'identification par radiofréquence) ; on peut citer notamment GS1/EPCglobal, pour l'industrie de la logistique, et l'ISO, avec ses nombreuses publications **[32]** .

2. **Le réseau de capteurs**

   Un réseau de capteurs (RCSF) ou WSN (Wireless Sensor Network) est un ensemble de nœuds qui communiquent sans fil et qui sont organisés en un réseau coopératif. Ces derniers sont capables de collecter, de traiter, d'analyser et de disséminer des informations et communiquent via des ondes radio afin de surveiller des phénomènes précis. Chaque nœud possède une capacité de traitement et peut contenir différents types de mémoires, un émetteur-récepteur RF et une source d'alimentation, comme il peut aussi tenir compte des divers capteurs et des actionneurs. Comme son nom l'indique, le WSN constitue alors un réseau de capteurs sans fil qui peut être une technologie nécessaire au fonctionnement de l'IdO **[34]** .

3. **M2M (Machine to Machine)**

   Le M2M (Machine-to-Machine) constitue un ensemble de technologies réseaux sans fil ou filaires rendant des systèmes communicants et leur permettant de s'échanger automatiquement des informations, sans intervention humaine. Il existe deux grandes familles de technologies, à savoir le sans-fil et les filaires. La plus connue des technologies sans fil est celle du réseau Internet mobile de type 2G, 3G, 4G, mais il en existe d'autres, basées sur des radiofréquences différentes allant de la plus courte portée avec le NFC, à la plus longue, avec le Wireless M-bus par exemple. Du côté des technologies M2M filaires, on trouve le courant porteur en ligne **[23]**.

1.7 Protocoles Utilisés dans IdO

L'Internet des Objets (IdO) repose sur plusieurs protocoles de communication qui permettent aux appareils connectés de communiquer entre eux et avec des systèmes centraux. Voici quelques-uns des protocoles clés utilisés dans l'IdO :

1. **Z-Wave**

   Z-Wave est un protocole de réseau sans fil développé par Zensys en 2001 et adopté par Sigma Designs en 2008. Il fonctionne sur les fréquences 868 MHz et 2,4 GHz, avec un débit binaire maximal de 200 kbit/s. La dernière version,

Z-Wave+, est entièrement compatible avec Z-Wave. Malgré ses fonctionnalités de sécurité, Z-Wave peut être contrôlé avec un équipement abordable **[36]**.

2. **ZigBee**

   ZigBee est un réseau technologique sans fil développé par la ZigBee Alliance, développé en 1998 et finalisé en 2004. Il est conçu pour les réseaux de capteurs à faible consommation d'énergie et à faible bande passante, souvent utilisés dans la domotique, notamment pour l'éclairage intelligent **[28]** .

3. **6LoWPAN (IPv6 over Low Power Wireless Personal Area Networks)**

   6 Lowpan est un protocole qui permet la communication des appareils à faible puissance. Pour cela, il utilise l'IPv6 sur des réseaux sans fil à faible consommation d'énergie en raison de l'optimisation de la transmission de données. **[29]**.

4. **LoRaWAN (Long Range Wide Area Network)**

   LoRaWAN est un protocole de communication longue portée et à faible coût conçu pour étendre les réseaux, en particulier pour les capteurs extérieurs, adapté à l'agriculture intelligente, à la gestion de la chaîne d'approvisionnement et à la surveillance à distance **[30]** .

5. **Bluetooth Low Energy (BLE)**

   Bluetooth Low Energy (BLE) est une version économique du Bluetooth, une technologie de communication mobile basée sur les réseaux cellulaires, adaptée aux réseaux longue distance et utilisée dans les villes intelligentes, la gestion des infrastructures et le suivi des actifs **[31]**.

6. **Sigfox**

   Sigfox est un réseau propriétaire dont la portée peut atteindre 10 kilomètres en ville et 30 à 50 kilomètres à la campagne. Il consomme peu d'énergie pour la transmission de données, mais nécessite des tests indépendants **[24]**.

1.8 Applications et domaines d'utilisation de l'IdO

Afin de résoudre des défis de longue date spécifiques à chaque secteur d'activité en utilisant l'Internet des objets (IdO), les entreprises sont aidées par Intel et son écosystème de partenaires. Ces derniers, dans le but de réduire les coûts, d'améliorer la productivité et la qualité de vie humaine et d'augmenter les recettes, permettent le développement rapide des objets connectés, la collecte de données et l'obtention d'informations, et ce, grâce au portefeuille de solutions ouvertes et évolutives qu'ils offrent. L'Internet des objets touchera plusieurs domaines d'application, et ceux-ci pourront être classifiés selon le type de disponibilité du réseau, l'échelle, la couverture, la répétabilité, l'hétérogénéité, l'impact et la participation des utilisateurs.

La figure 1.3 illustre les principaux domaines d'application de l'Internet des objets, permettant de mieux visualiser les secteurs concernés par cette technologie .

1. **Villes intelligentes**

   L'IdO joue un rôle clé dans la gestion des infrastructures urbaines, comme les systèmes de circulation, les éclairages publics et la gestion des déchets. Les capteurs peuvent surveiller la qualité de l'air, le niveau de bruit ou la consommation d'énergie, contribuant ainsi à améliorer la qualité de vie des citoyens **[25]**.

2. **La santé (Smart health)**

   L'Internet des objets (IdO) joue un rôle de plus en plus crucial dans le domaine de la santé en permettant une surveillance, un diagnostic et une gestion des soins plus efficaces, tout en améliorant l'expérience des patients **[34]**.

3. **Agriculture intelligente**

   L'Internet des objets (IdO) permet une gestion plus précise et efficace des ressources, favorise l'optimisation des rendements et contribue à la réduction des coûts. Grâce à l'usage de capteurs et d'appareils connectés, l'IoT transforme la manière dont les agriculteurs surveillent, gèrent et prennent des décisions concernant leurs exploitations agricoles.

4. **Le militaire**

   L'intégration des objets connectés permet aux forces armées de disposer d'une visibilité plus large et plus précise sur le terrain, de rendre leurs opérations plus efficaces et d'augmenter la sécurité des soldats **[27]**.

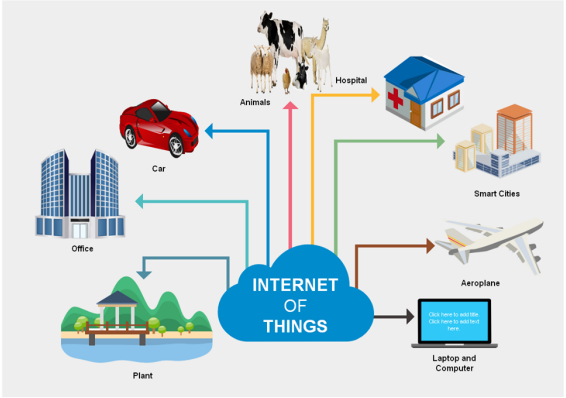

FIGURE 1.3 – Domaines d'application de l'IdO.

1.9 Sécurité dans l'IdO

Notre objectif en discutant des exigences et des mesures de sécurité pour Ido est de souligner à quel point il est crucial de protéger les données sensibles et de maintenir la confiance des utilisateurs. Il convient aussi de garantir que les systèmes sont résilients face aux menaces potentielles.

1.9.1 Exigence de la sécurité

En examinant les paramètres traditionnels de la demande de sécurité, il a besoin de construire un système Internet sûr des Objets, qui sont les suivants **[26]** :

- **Confidentialité des données :** Il est essentiel de protéger ces informations contre l'accès non autorisé. Cela peut être réalisé par des méthodes telles que le chiffrement des données, tant lors de leur transmission que lors de leur stockage.

- **Authentification :** En s'assurant que les utilisateurs et appareils autorisés puissent accéder aux dispositifs IdO, il est indispensable d'utiliser des mécanismes d'authentification robustes, comme des mots de passe forts, des clés publiques/privées, ou multi-facteurs.

- **Intégrité :** L'intégrité des données est cruciale pour garantir l'originalité des informations transmises à l'IdO, car elles ne doivent pas être réécrites, copiées ou remplacées par l'attaquant.

- **Disponibilité :** Un utilisateur autorisé peut utiliser divers services IdO sans interruption. Les attaques de type déni de service (DoS) représentent une menace majeure pour cette disponibilité.

1.9.2 Mesures de sécurité

Les exigences de la sécurité dans l'Internet des objets (IdO) sont cruciales pour garantir la confidentialité, l'intégrité, la disponibilité et l'authenticité des données échangées entre les appareils et les systèmes. L'IdO implique des millions d'objets connectés qui collectent, transmettent et traitent des informations sensibles, ce qui les rend vulnérables à diverses menaces.

1.10 Les avantages et les limites des réseaux connectés IdO

L'Internet des objets (IdO) offre de nombreuses possibilités, transformant notre quotidien et nos environnements professionnels. Cependant, cette révolution technologique comporte à la fois des avantages considérables et des défis importants. Il est donc essentiel d'examiner ses bénéfices tout en prenant en compte ses inconvénients potentiels.

1.10.1 Les avantages

- **Amélioration de la collecte de données :** La collecte de données moderne est confrontée à des limites et à une utilisation passive. L'IdO relie les espaces et les lieux que les humains souhaitent explorer pour une analyse précise du

monde et fournir des images précises **[33]**.

- **Automatisation et la Surveillance :** L'IdO permet l'automatisation de nombreuses tâches et la surveillance d'objets et d'environnements, améliorant la vie quotidienne et l'efficacité des installations industrielles, des réseaux de distribution et des stations météorologiques **[34]**.

- **Innovation et la Précision :** L'IdO a le potentiel d'améliorer la précision des processus et les possibilités d'innovation dans divers domaines, notamment dans la santé, l'agriculture, l'industrie, l'énergie et le transport.

1.10.2 Les limites

- **L'hétérogénéité de l'IdO :** L'Internet des objets (IdO) est sujet à l'hétérogénéité, car il n'existe pas de norme unique pour interconnecter tous les systèmes IdO hétérogènes. Il est donc essentiel de prendre en compte l'hétérogénéité lors de la création d'applications faciles à maintenir, et à intégrer à d'autres systèmes. Des normes communes sont nécessaires à la coopération et à la communication.

- **Les limitations de ressources et la croissance d'objets connectés :** Les limites de ressources pour l'Internet des objets (IdO) concernent principalement la capacité de traitement, la bande passante et la gestion de l'énergie. En raison du nombre élevé d'appareils connectés, la bande passante peut rapidement être saturée, affectant ainsi la communication entre les objets. De plus, de nombreux dispositifs IdO sont alimentés par des batteries limitées, ce qui rend leur autonomie un défi majeur, surtout dans des applications nécessitant un fonctionnement continu sur de longues périodes.

- **La fiabilité :** Les systèmes IdO sont vulnérables à des problèmes de fiabilité, ce qui rend nécessaire la mise en place solutions robustes et fiables et corrects dans les applications d'urgence et critiques telles que les soins de santé, la fabrication et les transports.

- **L'interopérabilité :** L'interopérabilité est la principale valeur d'un Internet des objets, en mettant en évidence que les systèmes connectés utilisent le même langage de protocoles et de codages. Les applications de l'IdO sont limitées par les industries utilisant leurs propres technologies et services, posant un problème d'interopérabilité. Une interface standardisée est cruciale pour gérer cette interopérabilité **[35]**.

1.11 Des réseaux IdO aux LLN : Une architecture optimisée pour les objets connectés

Les réseaux LLN (Low-Power and Lossy Networks) jouent un rôle essentiel dans l'implémentation de l'Internet des Objets (IdO). Ces derniers permettent aux dispo-

sitifs connectés, généralement limités en termes d'énergie et de puissance de calcul, de communiquer de manière efficace malgré des conditions environnementales instables. Ces réseaux sont conçus pour des applications à large déploiement, comme les capteurs environnementaux ou les systèmes de maison intelligente, ces réseaux nécessitent des protocoles spécifiques pour assurer la communication. C'est dans ce contexte qu'a été introduit le protocole de routage pour les réseaux à faible puissance et pertes élevées (RPL) **[17]**.

1.12 Conclusion

En conclusion, L'Internet des objets (IdO) représente une avancée significative en matière de connectivité et de technologie, offrant un immense potentiel pour améliorer la vie quotidienne. Au cours de ce chapitre, nous avons examiné les généralités de l'IdO et ses diverses, ses diverses applications . Dans ce qui suit, nous allons approfondir notre compréhension sur le protocole RPL, son architecture ainsi que ces différentes composants.

CHAPITRE 2

PROTOCOLE RPL

2.1 Introduction

Protocole de routage pour les réseaux à faible puissance et pertes élevées (Routing Protocol for Low-power and Lossy Networks) RPL est un protocole de routage conçu pour les réseaux basse consommation et faible bande passante, également appelés réseaux basse consommation et à pertes (LLN), dans les environnements IdO. Développé par l'internet engineering task force, RPL offre une solution fiable pour les réseaux présentant des conditions de communication difficiles, telles que des taux de perte élevés et une couverture réseau instable. Dans ce chapitre, nous allons examiner les généralités sur le protocole RPL ainsi que son adaptation aux Réseaux LLN et ces principales composants .

2.2 Les réseaux LLN (Low Power and Lossy Networks)

Les réseaux de faible puissance et pertes (LLN) désignent des environnements où les nœuds et leurs interconnexions sont extrêmement contraints par les ressources. Les nœuds sont généralement limités en termes de traitement, batterie et mémoire, et leurs interconnexions sont caractérisées par des liaisons instables avec des pertes élevées et des débits faibles. Les modèles de trafic varient et s'expliquent par points à point (P2P), points à multipoint (P2MP), ou multipoint à point (MP2P) . Ces réseaux sont compatibles avec IPv6 et sont reliés entre eux grâce à différentes technologies de communication comme IEEE 802.15.4 ou le Wi-Fi. Les LLN opèrent essentiellement sur des couches de liaison en employant des trames de taille réduite **[37]**.

Les réseaux LLN et les réseaux de capteurs sans fil sont distincts dans leur utilisation de l'internet et l'utilisation de IPv6, qui permet d'assurer un espace d'adressage

plus large qu'IPv4, et d'améliorer les options et l'étiquetage des colis relatifs à des "flux" particuliers **[21]**.

2.3 RPL et son adaptation aux réseaux LLN

Comme on le sait bien, le protocole RPL est principalement utilisé pour les bases de données LLN. Dans ce contexte, nous nous concentrerons sur l'adaptation de ce protocole dans les réseaux LLN :

2.3.1 Le routage dans les réseaux LLNs

Les ressources des réseaux LLN sont sévèrement limitées . Par exemple, les nœuds ont souvent une capacité de traitement, de batterie et de mémoire limitée. Les restrictions peuvent interférer avec le routage, entraînant des connexions instables, des pertes de paquets élevées et des débits faibles. Les protocoles développés pour le routage, qui étaient auparavant utilisés par les réseaux ad hoc et les réseaux de caméras non filtrés, ne s'adaptent pas à ces derniers. Ainsi, pour réussir le routage dans les réseaux LLN, il est nécessaire de choisir ou de développer des protocoles de routage spéciaux **[39]**.

2.3.2 Adaptation de RPL aux réseaux LLNs

L'Internet Engineering Task Force (IETF) a testé les protocoles de routage actuels dans les RFC et a constaté qu'aucun n'était suffisant pour répondre aux normes des réseaux à faible débit (LLN). Cela a conduit à la création du protocole de routage RPL (Routing Protocol), qui vise à optimiser les protocoles de communication pour les réseaux à faible débit (LLN) disposant de ressources énergétiques limitées, d'une faible bande passante et de taux de perte de paquets élevés. Le groupe RPL de l'IETF a établi une norme pour ces réseaux afin de garantir que leurs protocoles de communication soient optimisés pour ces limitations **[13]**.

2.4 Le protocole de routage RPL

Le protocole de routage destiné aux réseaux sans fil à basse énergie et à faible bande passante (RPL) — Routing Protocol for Low Power and Lossy Networks — est un protocole vectoriel distant conçu pour les appareils IPv6 à faible consommation, développé par ROLL (Routing Over Low power and Lossy networks, ou « Routage sur les réseaux à faible puissance et à pertes élevées ») dans la RFC 6550 pour répondre aux limites des réseaux LLN (la faible puissance de traitement, de batterie et de mémoire) et aux exigences d'énergie et de bande passante des réseaux sans fil. RPL adopte une approche proactive en construisant une structure de réseau appelée graphe orienté acyclique dirigé vers la destination (DODAG) **[40]**.

RPL assure avec succès et de manière efficace le routage des informations pour les nœuds disposant de ressources limitées, en offrant un cadre opérationnel qui

garantit une connectivité dans les deux sens, ainsi que la fiabilité et l'adaptabilité **[15]**.

La mission de RPL consiste à identifier la méthode optimale pour assurer le transfert des données entre les divers dispositifs du réseau **[14]**.

2.4.1 Architecture et composants du RPL

Pour comprendre comment le protocole RPL aborde des défis particuliers du réseau LLN et garantit une communication fiable et efficace dans l'internet des objets. IL est essentiel de connaitre son architecture et ses composants.

2.4.1.1 Graphe orienté acyclique (DAG)

Un graphe orienté possède la caractéristique que toutes ses arêtes sont orientées de manière à ce qu'aucun cycle ne soit présent. Chaque arête se trouve dans des chemins dirigées vers et finissant à un ou plusieurs nœuds racine **[12]**.

2.4.1.2 DODAG(Destination-Oriented DAG)

DODAG est un DAG à une seule destination, à la racine. C'est-à-dire à une seule racine DAG **[12]**.

Les définitions des graphes orientés acycliques (DAG) et des graphes orientés acycliques orientés vers la destination (DODAG) sont illustrées dans la figure 2.1, qui permet de visualiser la différence fondamentale entre un DAG général et un DODAG avec une racine unique.

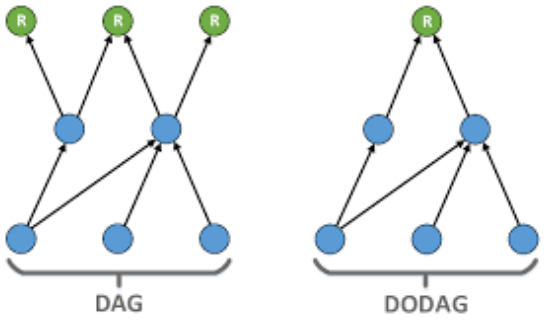

FIGURE 2.1 – Les graphes DAG et DODAG.

2.4.1.3 Fonction Objective (OF)

Dans le DODAG RPL, une fonction objective (OF) sert à déterminer des paramètres fondamentaux tels que les métriques de routage, les buts d'optimisation et la méthode de calcul du rang. Elle définit aussi la manière dont ces facteurs sont utilisés pour déterminer le rang et choisir les parents, ce qui influence la structure

et l'arrangement du DODAG **[13]**. L'IETF a défini deux fonctions d'objectifs pour RPL **[48]** :

- **Fonction Objective Zéro (OF0) :** Cette fonctionnalité utilise le nombre de sauts comme indice d'évaluation. Le rang d'un nœud, déterminé selon OF0, indique principalement le nombre de transitions entre ce nœud et la racine. La stratégie OF0, qui est la stratégie par défaut, vise à garantir l'interopérabilité entre différentes implémentations du protocole RPL.
  OF0 utilise simplement le nombre de sauts pour calculer le rang. La formule peut être exprimée comme :

$$\text{Rank}_n = \text{Rank}_p + \text{MinHopRankIncrease}$$

  où :
  — Rank_n : rang du nœud courant,
  — Rank_p : rang du parent,
  — MinHopRankIncrease : constante définissant l'augmentation minimale de rang (par défaut, 256).

  — **Fonction objective Minimum Rank with Hysteresis (MRHOF) :** Elle est mise en œuvre pour opérer avec les indicateurs additifs. Selon la définition de l'IETF, le nombre de transmissions attendu L'ETX (Expected Transmission Count) est utilisé comme critère à améliorer, cependant toute autre mesure additive (telle que le nombre de sauts ou le délai) pourrait également être appliquée. MRHOF utilise une hystérésis afin de réduire les fluctuations associées à l'utilisation d'une métrique dynamique.
  MRHOF choisit le parent avec le coût cumulatif le plus faible, en utilisant une métrique additive comme l'ETX (Expected Transmission Count). La formule est :

$$\text{Rank}_n = \text{Rank}_p + \text{ETX}_{(p,n)} \times \text{MinHopRankIncrease}$$

  où :
  — $\text{ETX}_{(p,n)}$: coût de liaison entre le parent p et le nœud n,
  — MinHopRankIncrease : constante (souvent 256).

2.4.1.4 Rang du Nœud (Rank)

Le rang d'un nœud détermine sa position individuelle par rapport aux autres nœuds en relation avec une racine DODAG. Le rang augmente strictement dans le sens descendant et décroît strictement dans le sens ascendant. Le calcul précis du rang est déterminé par la fonction objective (OF) de l'algorithme génétique différentiel. Par analogie, le rang peut suivre une distance topologique simple, être calculé selon les métriques de liaison et prendre en compte d'autres propriétés comme les contraintes **[13]**.

2.4.2 Les messages de contrôle dans RPL

Pour maintenir la topologie de routage, le protocole RPL emploie quatre nouveaux messages de contrôle ICMPv6.

2.4.2.1 Le DIO (Objet d'Information DODAG)

Il contient des données qui permettent à un nœud de détecter une instance RPL ou un parent préféré et de s'y joindre, d'assimiler ses paramètres de configuration, d'évaluer son rang et de sélectionner des parents qui réduisent le coût du chemin vers la racine du DODAG. Les DIO sont transmis à l'envers dans le réseau et sont généralement diffusés en multidiffusion. Toutefois, ils peuvent également être émis en monodiffusion si un nœud spécifique en fait la demande **[14]**.

2.4.2.2 Le DIS (Sollicitation Information DODAG)

Le message DIS (DODAG Information Solicitation) est un message de contrôle ascendant utilisé par un noeud pour solliciter des informations de configuration actualisées concernant la topologie DODAG, notamment lorsqu'il désire se connecter au réseau ou réinitialiser sa position au sein de celui-ci. Il est généralement envoyé en multidiffusion. En réponse, les nœuds voisins transmettent un message DIO en monodiffusion au nœud demandeur, facilitant ainsi son insertion dans la topologie **[15]**.

2.4.2.3 Le DAO (Destination Annonce Objet)

C'est un message de vérification envoyé par un nœud fils à son parent (ou à la racine du DODAG en fonction du mode opérationnel). Il contient des données qui signalent les destinations accessibles à travers ce nœud, indiquant ainsi à la racine ou aux parents que le nœud est présent et accessible. Ce message est essentiel pour la mise en place des itinéraires descendants dans les communications P2P. En mode Mode sans stockage, le DAO est transmis à la racine, tandis qu'en mode avec stockage, il est acheminé vers les parents. Il est toujours envoyé en monodiffusion, nécessite une confirmation et peut être désactivé pour optimiser l'utilisation des ressources **[15]**.

2.4.2.4 Le DAO-Ack Destination Annonce Objet - ACK)

Le message DAO-ACK est transmis en tant que paquet unicast par un destinataire DAO (Parent DAO ou racine DODAG), en réaction à un message DAO en mode unicast. Si le DAO-Ack n'est pas reçu, l'émetteur a la possibilité de renvoyer le DAO initial **[15]**.

La Figure 2.2 illustre les différents types de messages de contrôle utilisés par le protocole RPL pour construire et maintenir la topologie du réseau **[15]**.

2.4.3 La Construction du DODAG

Le DODAG est élaboré en se servant du protocole de découverte de voisinage (Neighbour Discovery) employé avec l'IPv6. La Constructiondu DODAG se fait de manière progressive **[18]** :

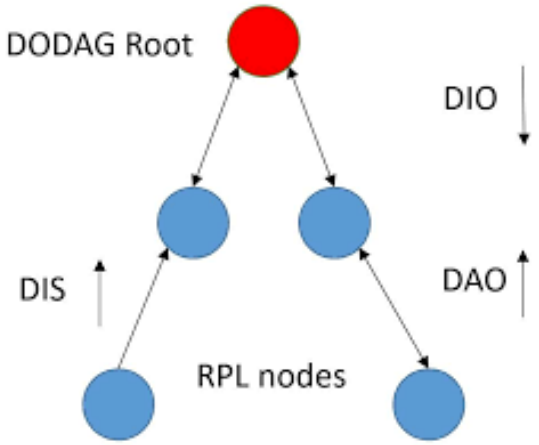

FIGURE 2.2 – Les messages de contrôle dans RPL.

- **Étape 1 :** Le processus de construction du DODAG commence à la racine du DODAG, qui diffuse régulièrement un message DIO à tous ses nœuds voisins. Ce message fournit des données essentielles telles que l'ID DODAG, sa fonction objective, ainsi que les informations nécessaires aux nœuds RPL pour découvrir une instance RPL, récupérer ses paramètres de configuration, choisir un ensemble parent et maintenir le graphe DODAG. L'emplacement d'un nœud dans le graphe est déterminé en fonction de sa position relative à la racine et doit constamment être supérieur au rang de ses parents.

- **Étape 2 :** Quand un nœud RPL reçoit un message DIO, Il doit d'abord prendre la décision de l'accepter ou pas. S'il ne satisfait pas à certains critères établis par RPL, il sera rejeté. Autrement, le nœud se charge du traitement du message DIO.

- **Étape 3 :** Suite à la réception d'un premier message DIO et à sa décision d'intégrer le DODAG, un nœud ajoute l'adresse de l'émetteur DIO à sa liste de parents et calcule son rang. Il envoie ensuite le message DIO contenant les détails de la mise à jour du rang à ses voisins. En se basant sur sa liste de parents, le nœud choisit un parent favori qui devient la passerelle par défaut à emprunter lorsque des données doivent être transmises à la racine du DODAG.

- **Étape 4 :** Quand un nœud est déjà associé à un DODAG et reçoit un autre message DIO, ensuite, le nœud met à jour le message DIO avec le nouveau rang et le diffuse à ses voisins. Par ailleurs, si le rang nouvellement calculé dépasse l'ancien, le nœud ne procédera pas à la mise à jour du rang et ne transmettra pas de message DIO. Il conserve le parent préféré précédent.

- **Étape 5 :** Un nœud qui n'a reçu aucun message DIO et qui n'est pas lié à un DODAG peut demander des informations sur le réseau en transmettant régulièrement des messages DIS à ses voisins.

- **Étape 6 :** À l'issue de ce processus, chaque nœud intégré au DODAG dispose d'un chemin par défaut à remonter jusqu'à la racine du DODAG.

2.4.4 Les modes d'opération du protocole RPL

Selon la capacité de mémoire des nœuds et la dimension potentielle du réseau, le protocole RPL propose deux modes opérationnels **[40]** :

- **Mode de stockage (Storing Mode) :** En ce mode, les nœuds intermédiaires ont la capacité de mémoriser des données de routage et d'acheminer les informations reçues vers leur destination appropriée en se référant aux informations de routage. Dans ce mode tous les messages DAO ne sont pas transmis à l'instance racine. Chaque nœud envoie son message à son parent immédiat (parent d'un saut) qui gère une table de routage à cette étape.

- **Mode sans stockage (Non-Storing Mode) :** Dans ce mode, uniquement la racine a la capacité de conserver des données de routage. Seuls les adresses de leur parent immédiat sont retenues par les autres nœuds du réseau. Dans les messages DAO, toutes les données concernant l'organisation du DODAG sont envoyées à la racine. Si des données doivent être dirigées vers une destination spécifique, les nœuds transfèrent ces données à la racine en les faisant passer par leur parent. La racine se chargera d'acheminer les données directement depuis la source jusqu'à la destination appropriée.

2.4.5 Modes de communication

RPL prend en charge trois types de communication : (MP2P), (P2MP) et (P2P).

- **Multipoint à point (MP2P) :** RPL a été développé principalement dans le but d'améliorer le flux de trafic multipoint à point (MP2P), Cette transmission MP2P a été réalisée grâce à la création de routes allant de chaque nœud vers la racine DODAG, en se basant sur le DIO du parent favori d'un nœud, ce qu'on appelle « Routes ascendantes » . Les points de destination pour les flux MP2P sont des nœuds désignés qui revêtent une certaine importance pour l'application, comme par exemple l'offre de connectivité à Internet ou au réseau IP privé principal **[12]** .

- **Point à multipoint (P2PM)** Il s'agit de routes descendantes (downward routes). RPL gère le trafic P2MP et emploie une méthode de diffusion de destination pour établir des chemins descendants depuis la racine vers d'autres nœuds. Cette fonction est employée pour acheminer des données vers un préfixe, une adresse ou un ensemble de diffusion multiple. Par exemple, les messages DIO, P2PM est le modèle de trafic requis par plusieurs applications LLN. En d'autres termes, ce service publicitaire facilite la distribution de routes descendantes depuis une source vers une ou plusieurs destinations en utilisant une adresse multicast **[19]**.

- **Point à point (P2P) :**On fait appel à ce modèle lorsque les nœuds établissent une communication en partageant des informations, sans que le root soit considéré comme l'expéditeur ou le destinataire du message. Pour le trafic P2P, l'établissement des chemins dépend du mode d'opération du protocole RPL. Si le paquet est acheminé vers une racine en mode sans stockage, celle-ci procèdera au routage vers la destination. Quand l'option STORING est en marche, le paquet progresse vers la racine jusqu'à ce qu'il rencontre un ancêtre qui connaît le chemin vers la destination. Il pourrait s'agir de l'ancêtre commun du DODAG. Dans certains scénarios, cela pourrait impliquer un nœud plus proche de la source ou de la destination **[16]** .
  La Figure 2.3 illustre les trois modèles de communication pris en charge par RPL, à savoir MP2P, P2MP et P2P, chacun correspondant à des besoins spécifiques dans les réseaux LLN **[20]**.

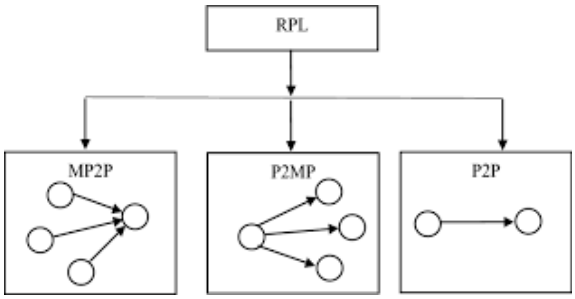

FIGURE 2.3 – Modèles de communication .

2.4.6 Les types de nœud dans RPL

Dans le cadre du protocole RPL (Routing Protocol for Low-Power and Lossy Networks), plusieurs types de nœuds peuvent être distingués en fonction de leur rôle et de leurs responsabilités dans le réseau :

- **Routeur de frontière LLN (LBR) :** Les Routeurs de bord pour réseaux à faible puissance et pertes (LBR), également appelés racines du DODAG, agit comme (Routeur de périphérie) entre l'internet et le LLN, qui peut avoir des racines multiples configurées dans le réseau, en mettant en place un point de collecte dans le réseau.Il se réfère à la racine d'un DODAG qui représente un point de collecte dans le réseau **[41]** .

- **Nœud Routeur :** Il se réfère à un périphérique qui peut transmet et génère du trafic des voisins, mais ne peuvent pas créer de nouveau DODAG. Il se connecte à un DODAG existant et serve de liens pour les paquets **[41]** .

- **Nœud hôte (stateless node) :** un nœud hôte (ou Host Node) fait référence à un nœud qui génère ou consomme des données, mais qui n'a pas de rôle direct dans le processus de routage. Les nœuds hôtes sont principalement des dispositifs finaux dans un réseau IoT (Internet des objets),

tels que des capteurs, des appareils connectés ou des objets intelligents **[42]** .

2.4.7 Gestion et maintien de la topologie

Dans un protocole de routage dynamique, l'emploi de mécanismes de réparation est essentiel pour préserver la topologie en cas de panne des nœuds et/ou des connexions. RPL supporte deux méthodes de réparation **[22]** :

- **Réparation globale :** Le mécanisme de réparation globale est initié par le nœud racine du DODAG lorsqu'une interruption de connectivité est détectée. Elle consiste à incrémenter le numéro de version du DODAG et à diffuser de nouveaux messages DIO dans le réseau. Cette démarche autorise tous les nœuds à établir une nouvelle topologie sans être limités par leur rang précédent. Les identifiants de séquence servent à différencier l'ancien DODAG du nouveau. Coûteux en termes de transfert de données et moins rapide comparé à une réparation locale.

- **Réparation locale :** Le processus de réparation locale s'active lorsqu'un nœud identifie une incohérence dans le réseau, telle qu'une boucle locale ou une rupture de liaison avec son parent direct. Dans ce contexte, le nœud affecté cherche à restaurer sa connexion en trouvant un nouveau parent sans reconstruire l'ensemble de la topologie du DODAG. Cette procédure comprend habituellement la transmission de messages de contrôle (tels que les DAO) et permet au réseau de converger rapidement, tout en minimisant l'impact sur la structure globale du réseau.

2.4.8 Trickle Timer

Étant donné que le protocole RPL est proactif, il envoie régulièrement des messages DIO pour établir et mettre à jour les itinéraires.

RPL utilise l'algorithme Trickle Timer afin de minimiser la charge des messages de contrôle en ne renvoyant que les mises à jour lorsqu'une incohérence est observée dans le réseau . Ainsi, l'algorithme Trickle offre la possibilité d'échanger des DIO de manière très robuste, économe en énergie et simple entre les nœuds du réseau LLN. L'envoi de DIO augmente lors de la détection d'une incohérence dans le réseau et se réduit quand le réseau est en état stable **[21]**.

2.4.8.1 Les paramètres et variables

Le Trickle Timer opère sur une période déterminée et inclut trois paramètres de configuration : l'intervalle minimum **Imin** , l'intervalle maximum **Imax** et une constante de redondance **k** :constante de redondance, k, est un entier positif (un nombre entier supérieur à zéro) **[21]** .

1. **Imin :** La taille minimale de l'intervalle de transmission est définie en unités de temps.
2. **Imax :** c'est la taille maximale de l'intervalle de transmission ,est décrit comme un nombre de doublements de la taille minimale de l'intervalle de transmission.
3. **k :** un facteur d'ajustement est un nombre naturel (un entier supérieur à zéro).

En plus de ces trois paramètres, Trickle gère également trois variables supplémentaires :

1. **I :** la taille de l'intervalle actuelle
2. **t :** un temps dans l'intervalle courant
3. **c :** le compteur.

2.4.8.2 Description de l'algorithme

L'algorithme Trickle Timer s'appuie sur les six règles ci-après **[21]** :

1. Initialiser un intervalle de temps I dans la plage [Imin, Imax]. Autrement dit, elle est supérieure ou équivalente à Imin et inférieure ou équivalente à Imax.

2. Lorsqu'un intervalle commence, l'algorithme réinitialise c à 0 et définit t à un point aléatoire dans l'intervalle du plage [I/2,I].

3. À chaque fois que Trickle détecte une transmission cohérente , il incrémente le compteur c.

4. À l'instant t, Trickle effectue la transmission uniquement si le compteur c est inférieur à la constante de redondance k.($k > 0$).

5. Lorsque l'intervalle I expire, Trickle double la longueur de l'intervalle. Si cette nouvelle durée d'intervalle dépasse le temps indiqué par Imax, Trickle établit que la durée d'intervalle I correspond au temps précisé par Imax.

6. Si Trickle reçoit une transmission incohérente et que I dépasse Imax, le minuteur de maintien est réinitialisé. Dans le cas contraire, si I est redéfini à Imin, un nouvel intervalle commence avec la réinitialisation du timer.

2.4.9 Sécurité du Protocole RPL

- **Non sécurisé :** Les signaux RPL sont envoyés sans mécanisme de sécurité, ce qui ne signifie pas nécessairement que le réseau RPL établi n'est pas sécurisé ; des méthodes de sécurité alternatives, comme la sécurité des liaisons de données, peuvent être utilisées.

- **Pré-installé :** Dans ce mode, les nœuds connectés à une instance RPL possèdent des clés préinstallées qui leur permettent de créer et de gérer des messages sécurisés.

- **Authentifié :** Dans ce mode, les nœuds possèdent des clés préinstallées à l'instar du mode « préinstallé » , cependant, ces dernières ne sont utilisées que pour permettre au nœud de se joindre à l'Instance RPL en tant que feuille (une feuille n'est capable d'envoyer du trafic sur le LLN qu'elle ne peut router). Pour se connecter à une instance RPL fonctionnant en mode « authentifié » , il est nécessaire d'obtenir une clé auprès d'une autorité d'authentification. RPL ne spécifie pas les procédures requises pour obtenir cette clé **[43]** .

2.4.10 Limite du protocole RPL

Plusieurs recherches ont démontré que le RPL se heurte à des contraintes qui restreignent son efficacité et son champ d'application : La réglementation RPL exige que chaque nœud employant un mode « stockage » conserve l'état de routage de tous les nœuds dans son sous-DODAG. Bien que RPL ait été spécifiquement conçu pour des nœuds ayant une mémoire restreinte, son but est de gérer des réseaux pouvant englober jusqu'à des milliers d'entités. Dans ces réseaux à haute densité, il est très probable que la capacité de stockage de ces appareils limités excède la capacité de routage requise. Par conséquent, un nœud saturé ne pourra pas traiter toutes les entrées de routage qui doivent être stockées dans sa table de routage, rendant ainsi plusieurs destinations au sein de son sous-DODAG inaccessibles pour la racine DODAG.

Quand on utilise le mode sans stockage, il est indispensable que la racine dispose d'un en-tête de routage source (la liste des divers nœuds relais vers la destination) pour chaque paquet de données envoyé dans la direction descendante. Néanmoins, le RPL est spécifiquement élaboré pour être employé sur des couches de liaison avec une unité de transmission maximale (MTU) limitée, autorisant ainsi un maximum de huit sauts du point d'origine au point de destination. Cela impose une restriction stricte sur la transmission par sauts multiples **[14]**.

2.5 Les métriques du RPL

Les métriques sont des paramètres de test du protocole de routage qui permettent de mesurer les performances de celui-ci. Dans notre étude, nous avons pris en compte les métriques suivantes **[67]** :

2.5.1 Taux de livraison des paquets (PDR)

Le PDR détermine le rapport entre le nombre de paquets de données parvenus à destination et le total des paquets transmis depuis la source, calculés à l'aide de cette formule :

$$PDR = \frac{pr \times 100}{\sum_{i=1}^{n} pgi}$$

Où

pr est le nombre total de paquets reçus par le nœud récepteur, **pg** est le nombre total de paquets générés par les nœuds sources et **n** est le nombre total de nœuds capteurs.

2.5.2 Délai de bout en bout (End to end delay)

L'évaluation du temps de bout en bout quantifie la moyenne des délais relatifs à chaque paquet de données reçu par le nœud destinataire ainsi qu'à chaque paquet de données transmis par les nœuds capteurs.

$$E2ED = \frac{\sum_{i=1}^{pr}(Tr_i - Tg_i)}{pr}$$

Cette équation est utilisée pour calculer le délai de bout en bout (E2ED), où **Tr** représente l'heure à laquelle le nœud récepteur a reçu des paquets de données et **Tg** indique le moment où chaque nœud source a produit des paquets de données.

2.5.3 Les messages de contrôle

Le trafic aérien correspond au quantité de messages de contrôle RPL transmis par les nœuds, comme les messages DIO, DAO et DIS. L'optimisation du processus de contrôle est essentielle, car le nombre de messages de contrôle influence directement la consommation d'énergie dans un réseau.

2.5.4 Consommation d'énergie

La consommation d'énergie fait référence à la quantité d'énergie consommée par les nœuds d'un réseau pendant toute sa durée de fonctionnement. Elle représente le total de l'énergie utilisée par chaque nœud. Cette consommation dépend directement du volume de messages envoyés et reçus, au temps de traitement, et au risque de surchauffe en cas d'inactivité.

2.6 Conclusion

RPL est un protocole de routage à vecteur de distance développé par le groupe de travail ROLL pour répondre aux contraintes très spécifiques des réseaux LLN. Le protocole a été élaboré dans le but d'être extrêmement adaptable aux diverses fluctuations des ressources du réseau. Dans cette partie, nous avons présenté brièvement les réseaux LLN. Ensuite nous avons détaillé le fonctionnement de son protocole de routage en décrivant les messages de contrôle, les mécanismes de construction et maintenance de la topologie, et les modèles de communication définis par le protocole en question.

Le chapitre suivant sera donc consacré à l'analyse de l'attaque par Rank, une menace spécifique au protocole RPL, et à l'étude de ses conséquences sur la stabilité et la sécurité du réseau.

CHAPITRE 3

ÉTUDE DE L'ATTAQUE PAR RANG ET DÉTECTION VIA MACHINE LEARNING

3.1 Introduction

Le protocole RPL (Routing Protocol for Low-Power and Lossy Networks) est largement utilisé dans les réseaux de capteurs sans fil et l'internet des objets (IdO), cependant, il comporte des failles pouvant être manipulées par des adversaires. L'attaque par Rang, qui altère le calcul du rang des nœuds dans RPL, représente un danger significatif en déstabilisant la structure du réseau et en nuisant à ses performances. Face à cette problématique, les techniques de l'apprentissage automatique (ML) offrent une solution prometteuse pour détecter et atténuer de telles attaques. Le ML, en se basant sur l'analyse des données du réseau et l'apprentissage des comportements normaux, favorise la détection des anomalies et la mise en œuvre de mesures correctives de façon autonome, renforçant par conséquent la sécurité et la résilience du réseau. Cette recherche examine l'attaque par rang dans RPL et analyse la façon dont l'apprentissage automatique peut servir à en réduire les impacts.

3.2 Les types d'attaques sur protocole RPL

Le protocole RPL est vulnérable à une large variété d'attaques de sécurité. Les propriétés des réseaux LLN — telles que les restrictions de ressources, l'absence d'infrastructure robuste, la sécurité physique limitée, la topologie dynamique et les connexions instables — les exposent particulièrement aux vulnérabilités et compliquent leur protection contre les attaques. Ces règles peuvent être propres au protocole RPL, mais elles sont également applicables aux réseaux de capteurs sans fil (WSN) ou aux réseaux filaires **[45]**.

La Figure 3.1 présente une classification de ces attaques selon leur impact sur la topologie, les ressources ou le trafic réseau **[44]**.

3.2.1 Attaques visant le trafic

Les attaques visant le trafic dans RPL (Routing Protocol for Low Power and Lossy Networks) sont des tentatives d'analyse et de manipulation des informations circulant au sein du réseau. Dans le cadre de RPL, ces assauts peuvent engendrer des répercussions considérables sur la sécurité, la confidentialité et l'intégrité des échanges. On peut mentionner parmi les diverses formes d'attaques ciblant le trafic dans RPL **[47]** :

- **Écoute passive :** sont déployées pour voler des données ou des informations transmises entre des points de connexion en profitant de l'absence de sécurité dans les canaux de communication. Parmi ces attaques, on trouve le sniffing (quand un nœud malveillant capte ou identifie des paquets transmis sur le réseau afin de récupérer des renseignements sur le routage) et traffic analysis.

- **Attaques de détournement :** se réfèrent aux activités malicieuses qui impliquent l'interception, la redirection ou la manipulation non autorisée des flux de données à travers le réseau. Parmi ces attaques : l'attaque d'identité qui utilise l'identité d'un nœud légitime pour obtenir le contrôle d'accès à une plus grande zone du réseau.

3.2.2 Attaques visant les ressources

Les attaques contre les ressources visent principalement à épuiser les capacités limitées des nœuds dans un réseau LLN, telles que l'énergie, la mémoire ou la capacité de traitement. Ces attaques entraînent l'exécution de processus inutiles ou génèrent un flux de données trop important, ce qui entraîne une surcharge des liaisons disponibles. En conséquence, la disponibilité du réseau est compromise et sa durée de vie est fortement diminuée. On identifie habituellement deux types de ces attaques, selon les ressources ciblées **[45]** :

- **Attaques directes :** C'est l'attaquant qui est directement responsable de l'épuisement des ressources. Cela peut généralement se faire en réalisant des attaques par débordement **(flooding attacks)** ou des attaques de surcharge concernant les tables de routage, lorsque le mode de stockage est actif.

- **Attaques indirectes :** Les attaques indirectes se réfèrent à des situations où le nœud malveillant incite d'autres éléments à créer une surcharge pour le réseau. Par exemple : Une attaque de version de numéro (un nœud malveillant prétend annoncer un faux numéro de version, entraîne l'envoi d'un message de contrôle aux autres nœuds pour vérifier et actualiser la

version numérique), l'attaque de rang augmenté, des attaques par incohérence du DAG.

3.2.3 Attaques visant la topologie

Les attaques contre le protocole RPL peuvent également cibler la topologie du réseau. On identifie deux types majeurs d'attaques parmi celles-ci **[46]** :

- **Sous-optimisation :** Lors d'attaques de sous-optimisation, le réseau ne parviendra pas à atteindre la configuration optimale (c'est-à-dire des chemins optimaux), ce qui conduira à une performance insatisfaisante. Par exemple : une attaque sinkhole, Dans cette situation, un nœud attire le trafic réseau en annonçant qu'il a le chemin le plus court vers le nœud racine, ce qui lui permet d'intercepter et de rediriger ce trafic vers un destinataire non autorisé.aussi on trouve les attaques par le trou de ver .

- **Attaques d'isolement :** L'attaque d'isolation vise à couper plusieurs nœuds du reste du réseau, les privant ainsi de toute communication avec la racine ou leurs parents. Cette attaque a pour but précis de troubler la communication entre les nœuds, les rendant incapables de préserver des liaisons opérationnelles avec le reste du réseau RPL. L'attaque de sélection du pire parent est une forme d'attaque par isolement où le nœud victime choisit le nœud le plus défavorable parmi les nœuds environnants. Aussi les attaques par déni de service et l'attaque de diminuer le rang sont des attaques de ce type.

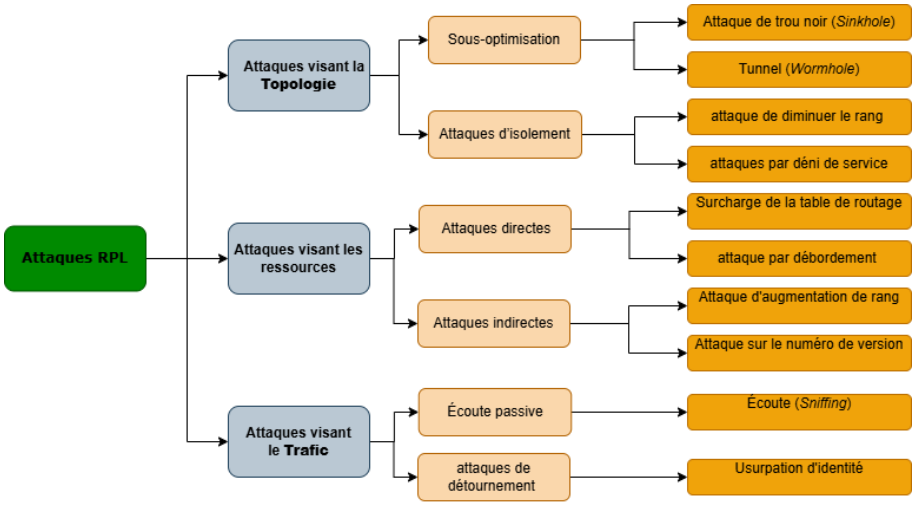

FIGURE 3.1 – Classification des attaques ciblant le protocole RPL.

3.3 Attaques par rang

Dans le RPL, l'attaque par rang est jugée parmi les plus destructrices, en raison de son potentiel à déstabiliser la totalité de la structure du réseau, tout

en ouvrant la voie à d'autres attaques comme le routage sélectif, l'attaque du trou noir, l'attaque par tunnel, et bien plus encore **[49]**.

Lors d'une attaque par rang, un nœud malveillant diffuse une valeur de rang erronée à l'aide de messages DIO à ses voisins, les poussant à le choisir comme parent favorisé via les messages DAO. Une fois identifié comme parent préféré (PPN) dans la zone ciblée, ce noeud nuit aux performances du réseau. Pour lancer une attaque par rang dans un réseau RPL, l'attaquant insère un nœud malveillant après la configuration de la topologie. Il existe trois types d'attaques par rang dans le RPL : l'attaque de diminution de rang et l'attaque de augmentation de rang , l'attaque du pire parent **[50]**.

3.3.1 Attaque de diminution de rang

L'attaque de diminution de rang est une forme d'assaut qui s'attaque à la topologie, où les attaquants se concentrent surtout sur la structure du réseau en contraignant le protocole à établir une topologie moins performante ou en coupant certains nœuds de communication avec l'ensemble du réseau. C'est l'une des attaques les plus sérieuses qui pourraient être menées contre le protocole RPL dans le cadre des communications IdO **[51]**.

Dans cette attaque, un nœud malveillant annonce délibérément un faux rang inférieur par le biais de messages de contrôle DIO pour tromper les nœuds voisins et attirer leur trafic.

Comme démontré dans la Figure 3.2, le nœud racine (nœud 1) initialise le DODAG en diffusant des messages DIO avec une valeur de rang de 1. Les nœuds voisins 2, 3 et 4 reçoivent ce message et utilisent l'équation de rang pour déterminer leur position, en leur assignant un rang de 2. Ils sélectionnent le nœud racine comme parent préféré et diffusent leurs messages DIO aux nœuds aval via multicast **[52]** .

Dans des conditions d'exploitation normales. La valeur de l'intervalle augmente progressivement à mesure que l'on s'éloigne du racine. Les nombres 5, 7, 8, 9 et ainsi de suite établiraient logiquement leurs rangs sur la base de celles de leurs parents, leur donnant des valeurs plus élevées (3 ou 4 comme on le voit dans le graphique).

Cependant, dans cette situation, le nœud 6 agit de manière malicieuse. Bien qu'il occupe réellement un rang 3 ou 4 dans la topologie , il annonce faussement un rang inférieur (rang = 2) aux nœuds environnants . Les flèches rouges en pointillé dans l'image mettent en évidence cette publicité trompeuse. Par conséquent, les nœuds adjacents tels que 5, 7, 8 et 9 se méprennent en pensant que le nœud 6 offre une voie plus directe vers la racine et optent donc pour le nœud 6 en tant que parent privilégié.

Cette action affecte la structure de routage standard basée sur un arbre (DODAG) et peut causer des dysfonctionnements réseau comme le routage incor-

rect du trafic, la surcharge ou même une attaque de point d'accès si le nœud 6 décide d'éliminer ou de changer le trafic qu'il reçoit. Par conséquent, l'attaque de diminution de rang met en péril l'intégrité, l'efficacité et la sécurité du réseau.

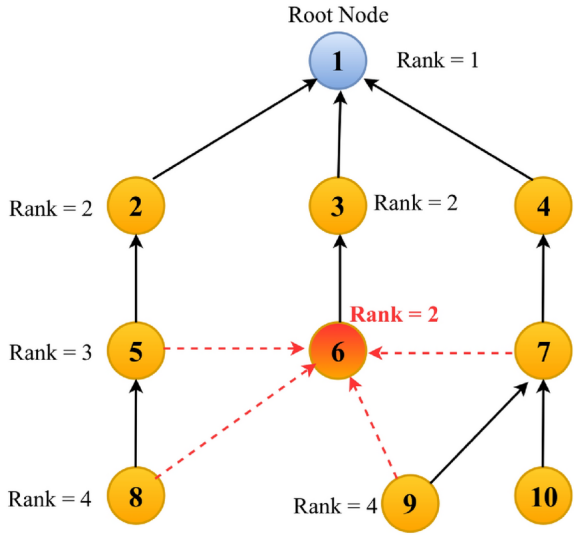

FIGURE 3.2 – Attaque de diminution de rang dans un réseau RPL.

3.3.2 Attaque de rang augmenté

L'attaque de rang augmenté est une menace pour les réseaux RPL, impliquant une manipulation malveillante du rang d'un nœud pour perturber la structure du routage et créer des boucles. Un nœud malveillant falsifie son message DIO pour signaler une valeur de rang supérieure à la normale, poussant ses voisins à opter pour un autre parent. Cela initie un processus de recherche de nouveaux parents dans les noeuds enfants du sous-DODAG.

Cette opération peut engendrer des boucles de routage, en particulier si le nœud malintentionné sélectionne un parent préféré situé dans son propre sous-DODAG, et s'il n'y a pas de mécanisme pour prévenir ces boucles **[53]**.

Comme démontré à la Figure 3.3, dans le premier cas de figure, le nœud 13 agit en tant qu'agresseur en élevant son rang à 3 et en choisissant le nœud 24 comme son nouveau parent favori. Bien que cela provoque momentanément une boucle entre les nœuds 13 et 24 (puisque le nœud 24 appartenait au précédent sous-DODAG du nœud 13), la situation se normalise rapidement par la liaison du nœud 24 à son parent (le nœud 12), permettant ainsi de restaurer le réseau.

Inversement, l'attaque est plus intense dans le deuxième cas : le nœud 31, qui n'a pas de parent de rechange, élève aussi son rang, contraignant ses descendants (nœuds 32 et 33) à se rattacher à un parent alternatif (nœud 22). L'impact de l'attaque sur la structure du DODAG est amplifié lorsque le nœud 31 sélectionne le nœud 32 comme nouveau parent, et que le nœud 21 augmente finalement son rang à 5 pour inclure le nœud 31 comme parent préféré. Nous

considérons cette attaque comme faisant partie des attaques de consommation de ressources, car ces renouvellements épuisent les batteries des nœuds et congestionnent le réseau RPL **[54]**.

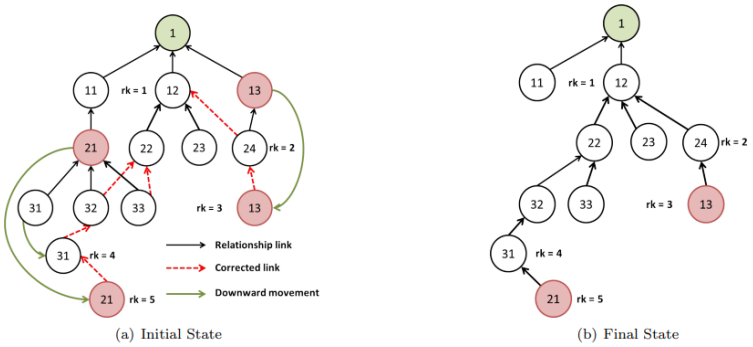

FIGURE 3.3 – Attaque de rang augmenté dans un réseau RPL.

3.3.3 L'attaque du pire parent

L'attaque du pire parent se produit lorsqu'un nœud malintentionné sélectionne délibérément le parent le moins performant, c'est-à-dire celui ayant le rang le plus élevé, parmi ses voisins pour gérer le trafic de ses descendants, ce qui compromet les performances du réseau. Bien que le nœud indique un rang approprié, cette sélection moins qu'idéale allonge les temps de transmission, augmente l'ETX et peut engendrer des boucles. Cette attaque, difficile à détecter, tire parti de la confiance que les nœuds enfants accordent à leur parent de préférence. Des solutions de sécurité qui reposent sur une analyse approfondie du graphe sont en mesure de détecter ces comportements atypiques **[52]**.
Par exemple, comme illustré dans la figure 3.4 , le nœud 4 est nuisible ; en raison de l'impact de l'attaque parentale la plus défavorable, le nœud 4 opte pour le nœud 8 comme son parent, ce qui ne représente pas le choix optimal d'un parent favorisé dans le contexte actuel **[52]**.

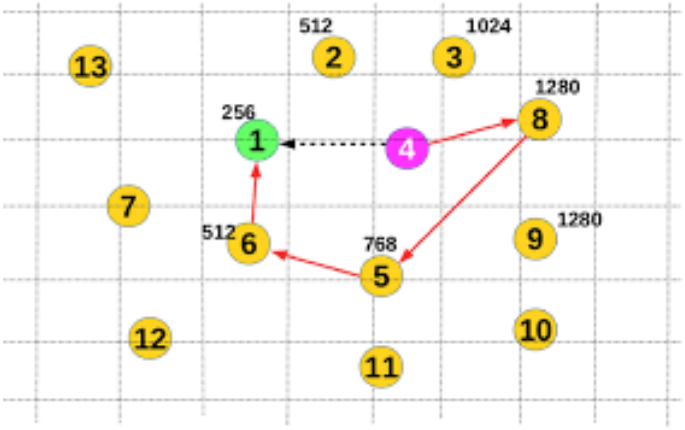

FIGURE 3.4 – L'attaque du pire parent en sélectionnant le pire parent de l'environnement.

3.4 Travaux associés à l'attaque par rang

Dans **[53]** les auteurs ont présenté l'identification des attaques par augmentation de rang (Rank Increased Attack - RIA) dans les réseaux RPL (Routing Protocol for Low power and Lossy networks) :

- **Type d'attaque abordé :** L'attaque par augmentation de rang (RIA) se produit quand un nœud malveillant augmente son rang, se plaçant ainsi plus éloigné du nœud principal. Cette opération altère la structure topologique du réseau, entraîne des cycles de routage et augmente le temps de transmission des paquets.

- **Méthode proposée :** L'article présente une méthode de détection (RIAI-DRPL) qui détecte les attaques par croissance de rang en surveillant les modifications inhabituelles du rang des nœuds. L'algorithme permet d'isoler les nœuds malintentionnés afin d'éviter les boucles dans le réseau RPL.

- **Fonction objective utilisée :** L'article propose une fonction objective pour identifier et prévenir les attaques RIA, qui examine la validité des rangs des nœuds. Cette fonctionnalité se base sur des signes de routage, comme le compte de transmission prévu (ETX), afin d'évaluer si un nœud dispose d'un rang valide.

- **Dataset utilisé :** Il n'existe pas de dataset public employé ni d'expérimentation pratique sur du matériel réel. C'est une étude par simulation, effectuée grâce à l'outil Cooja .

- **Résultats (Détection / Prévention) :** Les simulations effectuées démontrent que l'algorithme proposé identifie avec efficacité les attaques RIA. Il renforce la stabilité du réseau en minimisant les boucles de routage et en réduisant les temps de transmission des paquets. En outre, cela diminue la consommation d'énergie totale du réseau.

Dans **[55]** Les auteurs traitent des attaques éventuelles dans le cadre du protocole de routage RPL (Routing Protocol for Low-power and Lossy Networks) :

- **Type d'attaque abordé :** L'attaque de rang augmenté(RIA) indique dans article precedent et attaque par diminution de rang qui se réalise quand un nœud défavorable peut transmettre des messages avec des valeurs de rang intentionnellement inférieures, encourageant d'autres nœuds à s'y connecter, ce qui pourrait conduire à une interception illégale du trafic réseau.

- **Méthode proposée :** Les auteurs défendent une approche visant à sécuriser le protocole RPL (Routing Protocol for Low-power and Lossy Networks), en mettant l'accent sur l'authentification du numéro de version et du rang des nœuds. Cette technique vise à éviter les attaques internes où un nœud compromis pourrait illégalement altérer le numéro

de version ou le rang, ce qui perturberait la structure du réseau.

- **Fonction objective utilisée :** Les auteurs proposent un procédé de sécurité connu sous le nom de VeRA (Version Number and Rank Authentication) qui s'appuie sur l'usage de chaînes de hachage à sens unique. L'objectif de ce dispositif est de garantir que le numéro de version est correctement actualisé par le nœud racine du DODAG (Directed Acyclic Graph).

  Il est essentiel de garantir que les valeurs de rang transmises par les nœuds ne reçoivent pas de modifications incorrectes.

- **Dataset utilisé :** Validation effectuée uniquement par simulation, grâce à des outils tels que Cooja .

- **Résultats (Détection / Prévention) :** Les auteurs proposent une approche préventive, plutôt que de détection, en utilisant VeRA pour a sécuriser le numéro de version dans le protocole RPL ainsi que la valeur de rang annoncée par les nœuds.

Par contre dans **[47]** ils ont met l'accent sur l'analyse et la détection de trois catégories d'attaques dans les réseaux IdO qui emploient le protocole RPL :

- **Type d'attaque abordé :** L'attaque DIS (inondation) consiste a lenvoi massif de messages de controle DIS , épuisent ainsi les ressources des nœuds voisins. L'attaque par rang vise a altérer la structure du réseau en changeant illicitement le rang d'un nœud. Ainsi que l'attaque par trou de ver, elle etablit un passage malveillant entre deux nœuds, perturbant la communication standard du reseau .

- **Méthode proposée :** Un système de détection d'intrusion (IDS) hybride intégrant deux modèles d'apprentissage profond, le premier, DANN (réseau neuronal artificiel profond) est un modèle supervisé employé pour détecter les attaques identifiées en classifiant le trafic observé. Le deusiéme, DAE (autoencodeur profond), est un modèle semi-supervisé qui est formé exclusivement sur le trafic normale, permettent d'identifier les attaques non détectées en analysant les erreurs de reconstruction.

- **Fonction objective utilisée :**Optimiser la précision dans l'identification des attaques, qu'elles soient connues ou non, en minimisant les taux de faux positifs et faux négatifs.

- **Dataset utilisé :** IoTR-DS : un jeu de données spécialisé dédié à l'IdO, élaboré en reproduisant les attaques DIS, par rang et des attaques par tunnel, en plus du trafic standard. Ce jeu de données sert à l'entraînement et à l'évaluation du système IDS hybrid.

- **Résultats (Détection / Prévention) :** L'approche proposée (Hybrid DL-IDS) offre une détection performante des attaques Rank, DIS Floo-

ding et Wormhole au sein des réseaux RPL.

3.5 Classification des contre-mesures d'attaque de rang RPL

Plusieurs méthodes sont disponibles pour défendre les réseaux RPL contre les attaques de rang. On distingue deux principales catégories de mécanismes de défense contre ces attaques :

— Les solutions qui améliorent la sécurité du RPL en modifiant le protocole, en intégrant des techniques comme la cryptographie, la vérification de fiabilité, les seuils et autres.

— Les systèmes de détection d'intrusion (IDS) servent à identifier les anomalies dans le fonctionnement de RPL en se basant sur des spécifications prédéterminées et constatées.

3.5.1 Système de détection d'intrusion (SDI)

L'IDS examine les activités ou les processus sur un réseau ou un appareil, identifie les attaques, signale et/ou atténue l'impact négatif des attaques détectées. En raison de la variété des attaques et du comportement imprévisible de nouvelles attaques, les IDS sont susceptibles aux faux positifs et aux faux négatifs. Deux variantes d'IDS existent : l'IDS basé sur la signature (qui confronte les activités présentes aux schémas d'attaque préétablis) et l'IDS basé sur l'anomalie (qui identifie le comportement standard d'un réseau ou d'un équipement). Le système de détection d'intrusion fondé sur la signature requiert une compréhension détaillée de chaque attaque et offre des taux plus importants de faux positifs ainsi que de faux négatifs **[56]**.

Voici quelques illustrations de recherches qui ont suggéré des mesures IDS efficaces pour contrer les attaques de rang dans RPL **[57]** :

- **Le rang d'authentification et le métrique de routage (ARM)** qui est une identification basée sur une spécification hybride. Dans l'architecture ARM, le nœud de sink est perçu comme un module centralisé, alors que les autres nœuds sont considérés comme des modules répartis.

- **Le schéma de sélection sécurisé des nœuds parents :** qui utilise une valeur seuil pour choisir des nœuds légitimes comme parents des nœuds enfants. Le plan réduit efficacement la connexion entre les nœuds enfants et les nœuds malveillants, comme le montrent les résultats de l'évaluation.

- **Le système de détection d'intrusion basé sur le rang (SBIDS)** : qui repère avec efficacité les attaques de rang au sein du réseau RPL

en mettant en comparaison le rang actuel du nœud et le rang parent, assurant ainsi un écart minimal de rang entre les nœuds frères.

3.5.2 Modification du mécanisme de défense

L'objectif des techniques de classification est de renforcer le protocole RPL en incorporant des règles ou des algorithmes pour évaluer et valider dynamiquement les informations échangées entre les nœuds, contrecarrant ainsi les attaques de type RPL-cibling. Ces méthodes cherchent à identifier et à déstabiliser les comportements atypiques en relation avec la gestion des valeurs de distance (rang) en utilisant des critères distincts (tels que des limites, la concordance des messages DIO ou des modèles statistiques) avant la validation ou la diffusion des données de routage **[58]** . Voici quelques illustrations de recherches qui ont suggéré des mesures de modification du mécanisme de défense pour contrer les attaques de rang dans RPL **[57]** :

- **Le protocole Secure-RPL :** qui bloque les nœuds malveillants de se repositionner eux-mêmes dans l'arbre DODAG, réduisant ainsi l'impact des attaques. Il analyse les valeurs de rang et utilise une fonction de seuil pour protéger le réseau RPL.

- **SecTrust-RPL :** est un RPL conscient du temps et de la confiance conçu pour protéger contre les attaques de rang et Sybil, optimisant les performances du réseau et renforçant la confiance entre les nœuds voisins.

- **Un algorithme RIAIDRPL :** a été suggéré pour prévenir la création de boucles par un nœud malveillant disposant d'une valeur de rang élevée.

3.6 Machine Learning (ML)

L'apprentissage automatique , ou machine learning est une branche de l'intelligence artificielle (IA) qui donne la capacité aux ordinateurs et aux systèmes informatiques d'assimiler des informations à partir des données de manière autonome, sans avoir besoin d'être programmés spécifiquement pour chaque opération. Le concept fondamental est d'employer un algorithme ou un modèle dans le but de repérer des schémas ou des associations au sein des données, afin d'effectuer des prévisions ou des choix basés sur des informations inédites encore inexplorées. Le fondement de l'apprentissage automatique repose sur la construction de modèles statistiques capables d'évoluer avec le temps grâce à l'expérience acquise. Pour réduire l'écart entre la prédiction et les données réelles, les algorithmes modifient indépendamment leurs paramètres internes (tels que les poids dans un réseau de neurones) **[59]**.

L'apprentissage automatique peut être classé en plusieurs catégories principales selon la nature des données et des tâches d'apprentissage **[60]** :

3.6.1 Apprentissage supervisé

L'apprentissage supervisé est une technique d'apprentissage automatique où l'algorithme est établi sur des données labellisées, lui permettant de lier chaque exemple de formation à un résultat correspondant et de modifier ses paramètres internes pour généraliser les prédictions sur des entrées inconnues. Son objectif principal est d'établir une relation fonctionnelle entre les variables d'entrée et de sortie afin que des prédictions précises puissent être faites sur de nouvelles données. Il existe des applications importantes dans des domaines tels que la détection d'objets, la détection de spam et la reconnaissance vocale.

Les exemples suivants montrent les principaux algorithmes utilisés dans l'apprentissage supervisé, chacun étant adapté à une tâche particulière, comme la classification ou la régression.

- **K-VOISINS LES PLUS PROCHES (KNN) :** L'algorithme KNN classe les échantillons en mesurant la distance des caractéristiques, avec le principe que si la plupart des K échantillons les plus proches appartiennent à une catégorie, l'échantillon appartient également à la même catégorie. Il est facile à mettre en œuvre, sensible aux valeurs aberrantes et adapté aux classifications multiclasses.

- **SVM (Support Vector Machine) :** SVM est un classificateur linéaire d'apprentissage supervisé pour la classification binaire, minimisant à la fois les risques empiriques et structurels. Il est stable et s'applique aux tâches binaires, réduisant les multiples tâches en plusieurs problèmes binaires.

- **Arbre de décision :** Le DT (Decision Tree) est un modèle prédictif en exploration de données, illustrant la relation de correspondance entre les caractéristiques des objets et leurs valeurs, principalement utilisé pour la classification des paquets dans les réseaux.

3.6.2 Apprentissage non supervisé

Dans les techniques d'apprentissage non supervisé, les labels des échantillons d'entraînement ne sont pas identifiés. L'objectif est de mettre en évidence les propriétés inhérentes et les règles des données en examinant des séries d'échantillons non labellisés, fournissant ainsi une base supplémentaire pour l'analyse des données. La méthode la plus courante est le « clustering » , avec l'algorithme simple et largement utilisé K-means.

Les exemples suivants montrent les principaux algorithmes utilisés dans l'apprentissage non supervisé **[61]** :

- **Algorithme ECLAT (Clustering de classe d'équivalence et traversée de réseau ascendante) :** ECLAT (Equivalence Class Clustering and bottom-up Lattice Traversal) est une technique d'exploration de don-

nées verticale plus rapide qui utilise une structure horizontale, nécessitant moins d'analyses de base de données pour localiser les éléments fréquents.

- **Clustering à l'aide de K-Means :** Un algorithme de regroupement prisé qui classe les éléments en groupes selon leur similarité.

- **Algorithme Apriori :** L'algorithme apriori est conçu pour l'exploration de données, l'extraction de données à partir de grandes bases de données pour l'analyse des paniers d'achat et l'identification des achats courants et des effets indésirables des médicaments.

3.6.3 Apprentissage semi-supervisé

Il s'agit d'une technique d'apprentissage qui fusionne l'apprentissage supervisé et l'apprentissage non supervisé. Elle met l'accent essentiellement sur l'exploitation d'un faible nombre d'échantillons labellisés et d'une grande quantité d'échantillons non labellisés pour l'entraînement et la classification. Le semi-supervisé est employé pour les mêmes genres d'applications que le supervisé. Depuis son initiation, l'apprentissage semi-supervisé a été principalement appliqué à la manipulation de données synthétiques et n'a été éprouvé que dans un contexte d'expérimentation.

3.6.4 L'apprentissage par renforcement

L'apprentissage par Renforcement (AR) est une méthode d'apprentissage automatique où un agent reçoit des connaissances en faisant des erreurs et en découvrant des échecs, tout en étant satisfait pour ses actions vis-à-vis de son environnement. Le but d'un système de renforcement basé sur l'apprentissage (SAR) est de modifier de manière dynamique les paramètres pour obtenir le signal de renforcement optimal. Le signal de renforcement dans l'environnement ne sert pas à guider le système vers la bonne voie d'action, mais plutôt à évaluer positivement ou négativement l'action effectuée. L'apprentissage par renforcement fusionne des éléments de l'apprentissage supervisé et non supervisé.

3.6.5 Apprentissage hybride

Les approches hybrides combinent diverses méthodes d'apprentissage automatique pour améliorer les performances, réduire les contraintes ou renforcer la solidité du modèle. Par exemple, en combinant l'apprentissage supervisé avec l'apprentissage non supervisé.

Le tableau 3.1 suivant synthétise les principaux types d'apprentissage automatique évoqués ci-dessus, en présentant leurs caractéristiques, algorithmes associés et domaines d'application.

TABLE 3.1 – Résumé des types de modèles d'apprentissage automatique

Type	Mode d'apprentissage	Algorithmes	Applications
Supervisé	Données étiquetées	Régression linéaire, régression logistique,arbres de décision, forêt aléatoire, SVM, réseaux neuronaux, KNN,etc	Classification, régression, détection des spams, etc.
Non supervisé	Données non étiquetées	K-Means Clustering, Hierarchical Clustering, PCA, DBSCAN , Apriori , ECLAT	Clustering, détection d'anomalies, réduction de dimensionnalité, analyse du panier de consommation, etc.
Semi-supervisé	Étiquetées et non étiquetées	Algorithmes d'autoformation, modèles génératifs	Classification des images, reconnaissance vocale, etc.
Apprentissage par renforcement	Apprentissage par les agents	Q-Learning, Deep Q-Networks (DQN), méthodes de gradient des politiques	Robotique, jeux, véhicules autonomes, systèmes de recommandation
Hybride	Combinaison de méthodes	Méthodes d'ensemble (Random Forest, AdaBoost, XGBoost), Deep Learning + AR	Tâches nécessitant une grande précision ou combinant plusieurs approches d'apprentissage

3.7 Travaux associés au machine learning (ML)

Dans le but de détecter les attaques dans les réseaux RPL en utilisant des méthodes d'apprentissage automatique, plusieurs démarches ont été proposées, chacune se concentrant sur une attaque spécifique et une méthode machine learning différente.

Vikram Neerugatti et Rama Mohan Reddy **[62]** ont proposé une technique pour localiser les attaques par rang dans les réseaux RPL de l'IdO :

- **Technique ML utilisée :** Ils se sont basé sur MLTKNN, une technique de détection des attaques par rang dans les réseaux IdO de type d'apprentissage supervisé. Elle se repose sur l'algorithme KNN qui utilise le principe de la distance pour évaluer la similarité entre les points.

- **Type de données :** L'évaluation a été réalisée à l'aide de leur propre dataset simulé dans COOJA.
- **Résultats principaux :** la consommation d'énergie , le nombre de voi-

sins , le taux de livraison de paquet et le délai de bout en bout sont les métrique présentées dans les resultats.

Toutefois, dans **[63]**, les chercheurs convertissent les données en CSV (Comma-Separated Values), un format de fichier tabulaire, afin d'exécuter un modèle qui est surveillé par Google AutoML et Azure ML, ces plateformes évaluent automatiquement la performance du modèle face à des attaques complexes telles que la combinaison de l'attque par rang et de trou noir.

- **Technique ML utilisée et Type de données :** Les données (paquets réseau) sont étiquetées comme malveillantes ou bénignes, ce qui est une caractéristique de l'apprentissage supervisé. Pour entraîner les modèles, les auteurs utilisent les plateformes AutoML (Google) et Azure ML. Le modèle apprend à prédire si un paquet est malveillant en fonction de ses attributs (rang du champ, adresse IP, etc.).

- **Résultats principaux :** Les auteurs présentent les résultats de leur évaluation de deux algorithmes d'apprentissage automatique— SVM et Decision Forest (DF)— ainsi qu'un modèle généré automatiquement par la plateforme AutoML (Google). Ces modèles sont appliqués à un jeu de test pour identifier les attaques dans les réseaux RPL. Le modèle AutoML présente les meilleurs résultats, dépassant les 68,6 de DF et les faibles 3,1 de SVM. Ces résultats suggèrent que le modèle généré par AutoML est le plus adapté pour être intégré dans un système de détection d'intrusion (IDS).

Dans **[64]** Ils envisagent le développement futur d'un protocole hybride léger et sécurisé, et proposent un cadre de détection basé sur l'apprentissage automatique pour identifier ces attaques combinées.

- **Technique ML utilisée et Type de données :** ils ont utilise une technique d'apprentisage automatique supervise . Le processus repose sur trois modeles :le profilage des attaques en déterminant les paramètres caractéristiques basés sur la littérature et le comportement typique du protocole RPL ; l'intégrer du modèle d'apprentissage automatique avec divers algorithmes testés pour identifier les plus efficaces ; l'évaluation du modèle par simulation à l'aide de mesures d'efficacité et de précision. Pour choisir la méthode la plus adaptée, des approches d'apprentissage automatique et d'apprentissage profond sont comparées.

- **Résultats principaux :** Dans un environnement fortement interconnecté, il est possible de détecter simultanément en produisant une solution performante, proactive et efficace. C'est notament le cas pour les attaques par rang et les attaques par tunnel.

Au début de l'année 2020, les auteurs dans **[65]** ont proposé un protocole RPL hybride pour combattre les attaques par rang et par trou de ver grâce à l'apprentissage automatique.

- **Technique ML utilisée et Type de données :** Ils n'ont pas précisé le type de mode d'apprentissage utilisé, mais on peut raisonablement déduire qu'il s'agit d'un apprentissage supervisé , car le modéle est basé sur des exemples de nœuds autorisés et malveillants. La détection des attaques se fait grâce à un modèle entrainé sur les différences entre ces deux types de nœuds.

- **Résultats principaux :** Une analyse approfondie est prévue afin de sélectionner les meilleures méthodes d'apprentissage automatique, telles que SVM, pour obtenir des résultats efficaces. L'objectif est d'améliorer le protocole RPL en ajoutant une défense pratique et performante contre ces attaques.

3.8 Conclusion

En conclusion, ce chapitre a examiné les attaques par rang dans les réseaux RPL, en mettent en lumiére leur impact sur la stabilité et la sécurité du réseau. Nous avons présenté les mécanisemes par lesquelles ces attaques peuvent compromettre le protocole RPL, ainsi que les différentes stratégies de détection et de prévention proposées. Par ailleurs, l'utilisation du l'apprentissage automatique a été discutée comme outil prometteur pour détecter et prévoir ces attaques, en mettant l'accent sur sa capacité à améliorer la sécurité des réseaux LLN.

Dans le chapitre suivant, nous procéderons à une simulation du protocole RPL en cas d'attaque par rang, et sur l'efficacité des modèles d'apprentissage automatique à détecter ces attaques à partir des données issues du réseau simulé.

CHAPITRE 4

SIMULATION, CAPTURE DES DONNÉES ET DÉTECTION AUTOMATIQUE

4.1 Introduction

Dans ce chapitre, nous nous intéresserons à les attaques de rang qui ciblent le protocole RPL notamment l'attaque de diminution de rang et l'attaque de rang augmenté, puis analysé leurs effets en utilisant deux métriques principales : l'énergie et le nombre de messages de contrôle. Nous allons expliquer en détail la procédure de mise en place de notre simulation en utilisant Cooja. L'objectif est de simuler des scénarios réalistes afin d'approfondir la compréhension des processus, des vulnérabilités et des répercussions de ces attaques sur les réseaux IdO. Nous prévoyons d'employer VMware workstation player pour faire fonctionner une machine virtuelle appelée Instant ContikiOS. Les résultats des simulations (scénario normal et attaque) ont été consignés en format .csv et utilisés ensuite sur Google Colab. Par la suite, nous avons déployé quatre techniques de machine learning pour détecter l'attaque de diminution de rang, et nous avons comparé leurs performances respectives.

4.2 Métriques de la simulation

Nous avons simulé le réseau initial pendant 15 minutes, puis nous avons mesurer ses performances pour observer l'impact de l'attaque de diminution de rang. Dans notre étude, nous avons choisi d'évaluer le protocole RPL en termes de :

- **Consommation moyenne d'énergie :** C'est une mesure de performance essentielle lorsque nous visons les LLN, car leurs nœuds ont des

restrictions liées à la batterie. Il s'agit de la consommation moyenne de puissance en milliwatts (mW) par tous les nœuds du réseau. Dans notre situation, elle est obtenue grâce à la vue de collection (collect view).

- **Le nombre de messages de contrôle :** Il s'agit du nombre total de messages DIO, DAO et DIS qui ont été transmis à travers le réseau durant la simulation, afin de surveiller la surcharge du trafic. Un script Perl est mis en œuvre pour l'extraction et le calcul des messages.

Nous avons décidé d'examiner ces deux métriques, car elles sont sensibles aux attaques par rang. Les messages de contrôle traduisent directement les perturbations dans la topologie RPL dues aux fausses valeurs de rang, alors que l'évaluation de la consommation énergétique permet d'estimer l'effet de ces attaques sur la durée de vie des nœuds. En outre, ces deux métriques peuvent être facilement évalués grâce à l'outil Collect View de Cooja, ce qui les rend appropriés pour notre expérimentation.

4.3 Implémentation

L'objectif de notre expérimentation est de modéliser un réseau Low-Power and Lossy Network (LLN) en utilisant l'outil Cooja et la version 3.0 de Contiki et d'implémenter l'attaque de diminution de rang afin d'étudier ses impacts sur les performances du réseau, notamment en termes de la consommation d'énergie des noeuds, de surcharge du trafic et de la stabilité du réseau. Ce réseau sera composé de 14 nœuds normales et un noeud root.

Après avoir exécuté la simulation, nous avons utilisé l'outil Mote Output de Cooja pour sauvegarder les sorties dans un fichier texte (ex : coojaRPLOutput.log). Ce fichier est ensuite analysé automatiquement à l'aide d'un script Perl personnalisé , Ce processus nous a permis de comparer le comportement normal du réseau RPL avec celui observé lors d'attaque.

4.3.1 Réseau de référence

Nous avons mis en place un scénario de simulation d'un réseau LLN dans des conditions normales, en faisant varier 15 nouds. Tous les nœuds utilisent le code par défaut de Contiki et sont considérés comme légitimes. Ils transmettent leurs informations, telles que l'utilisation du CPU et la charge restante, vers le point de collecte, qui est le "sink" dans notre cas. Cette simulation nous permettra d'obtenir des données de référence sur le comportement normal du réseau, sans aucune attaque.

Les détails de la configuration de la simulation normale, sans aucune attaque, ainsi que les emplacements des nœuds sont présentés dans la figure 4.1 suivante.

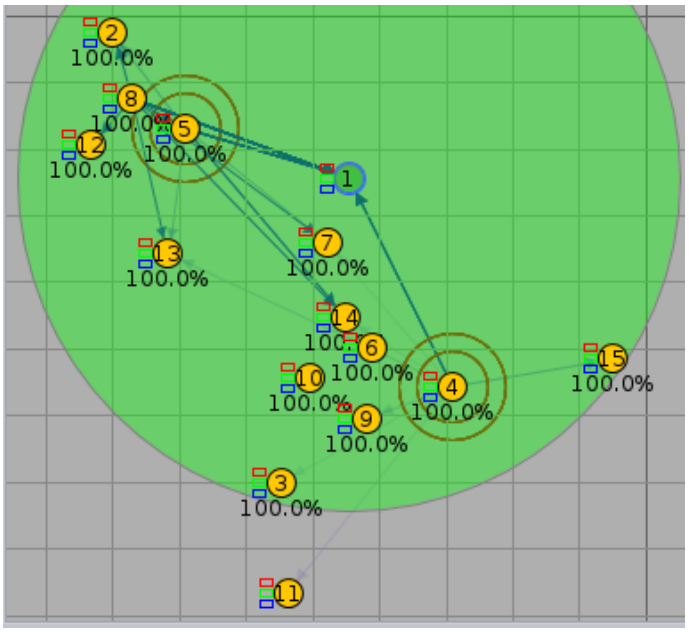

FIGURE 4.1 – La topologie de la simulation sans noeud malicieu.

- **Objectif :**

Pour obtenir une référence solide pour la comparaison, il est essentiel de créer un réseau de référence en suivant les paramètres indiqués dans le tableau 6.

Cette section se concentrera sur la méthodologie et les étapes nécessaires pour créer ce réseau de référence . Les résultats de ce scénario seront utilisés comme point de référence pour analyser et comparer les résultats issus de la simulation d'une attaque.

Pour ce faire, nous avons utilisé le simulateur Contiki/Cooja avec une portée de transmission Tx de 50 mètres et une portée d'interférence INT de 100 mètres, en plus d'une distribution aléatoire des nœuds.

- **Méthodologie :**

La construction d'un réseau de référence comprend les étapes suivantes :

1. Pour créer une nouvelle simulation, cliquez sur `Fichier-> Nouvelle simulation` dans la barre de menu. Ici, vous devrez définir le nom de la simulation et le type de support Cette étape est illustrée dans la figure 4.2 .

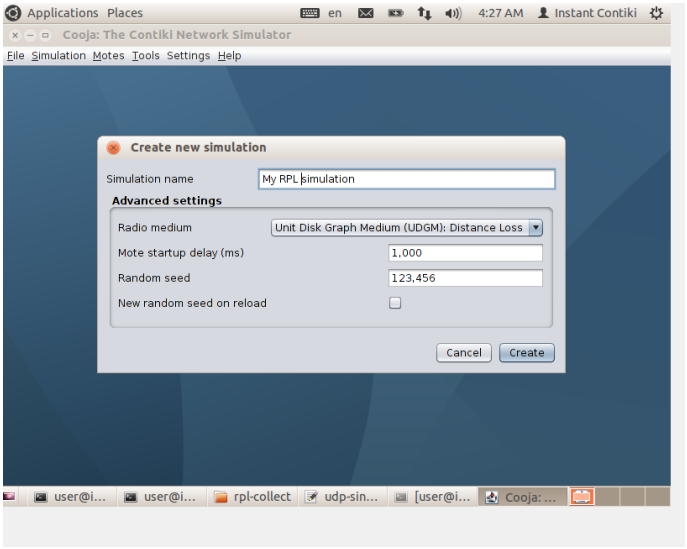

FIGURE 4.2 – Création d'une nouvelle simulation sur cooja.

2. La prochaine étape consistera à créer les types de nœuds qui constitueront le réseau. Le réseau de référence comprendra deux types de nœuds : un nœud de destination (sink node), qui fonctionnera en tant que routeur DODAG, et des nœuds feuille (leaf motes), qui fonctionneront simplement comme des nœuds clients. En cliquant sur `Motes-> Ajouter des nœuds-> Créer un nouveau type de nœud-> Sky mote` dans la barre de menu, une fenêtre apparaîtra. Ici, les firmwares des différents nœuds seront compilés pour les créer (voir Figure 4.3).

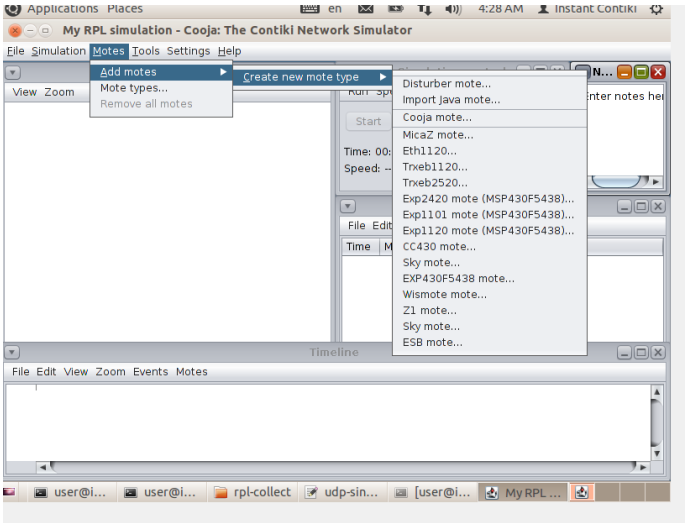

FIGURE 4.3 – L'ajout des noeuds à la simulation.

3. le nœud root sink est basé sur le fichier firmware suivant : `/Contiki/examples/ipv6/rpl-co sink` . Ensuite, appuyer sur le bouton "Compiler" puis "Créer". Cette étape est illustrée à la Figure4.4.

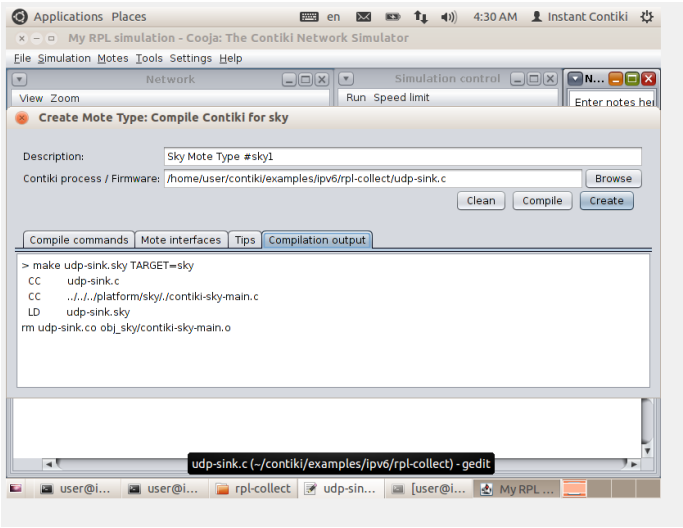

FIGURE 4.4 – téléchargement de code de sink mote.

4. Une fois que le fichier a été compilé avec succès le nœud sera crée (voir Figure 4.5).

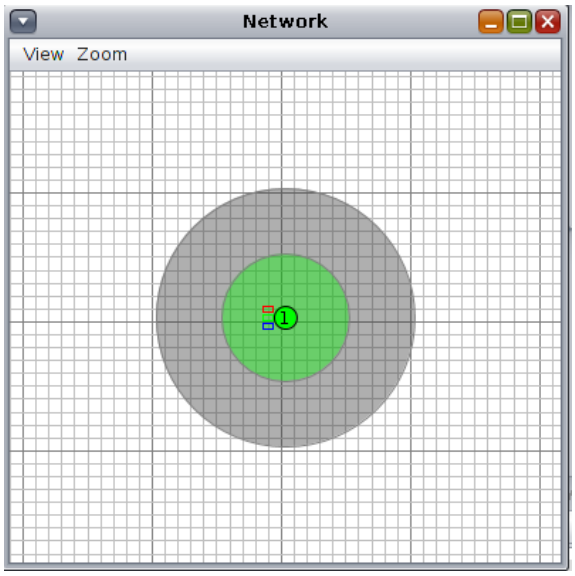

FIGURE 4.5 – Le noeud sink (racine) .

5. Pour ajouter les autres nœuds, on suit les mêmes étapes, mais avec le firmware suivant : `/Contiki/examples/ipv6/rpl-collect/udp-sender.c` — motes. Cette étape est illustrée à la Figure 4.6.

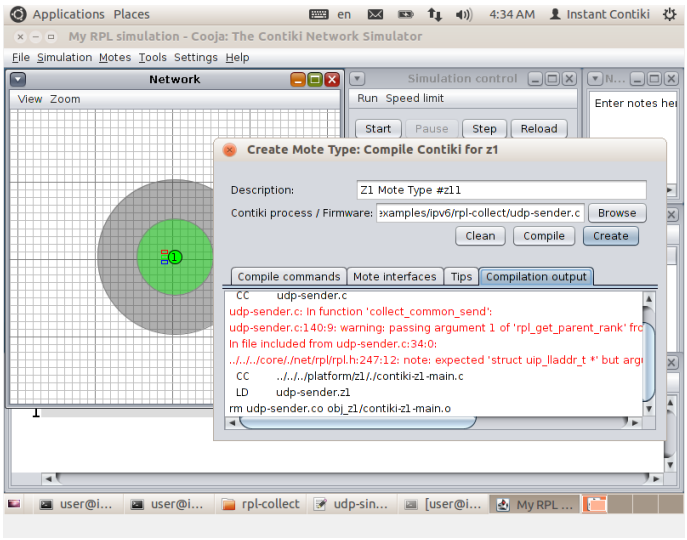

FIGURE 4.6 – L'ajout des noeuds clients.

6. L'environnement de simulation final est représenté dans la figure 4.7 . L'image montre également la portée de transmission et d'interférence du routeur DODAG.
7. **Démarrage de simulartion :** le démarrage s'effectue par un simple clic sur `Start`.

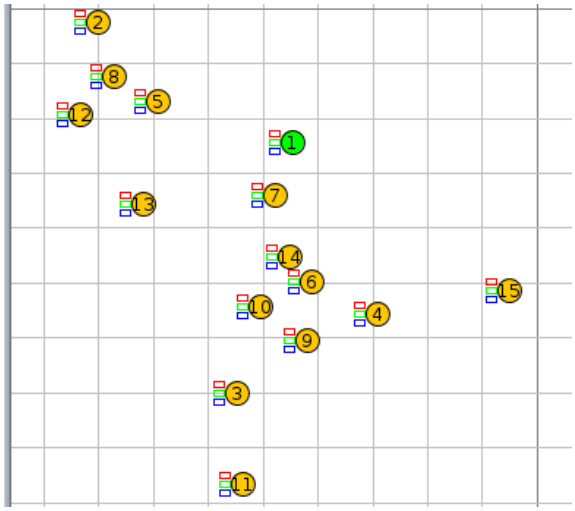

FIGURE 4.7 – La topologie créée avec l'emplacement des noeuds.

8. **Récupération des données de la simulation :** Dans notre étude, nous avons choisi d'évaluer le protocole RPL en termes de consommation moyenne d'énergie grâce à Collect View et en termes de nombre de messages de contrôle, pour surveiller ce dernier. Dans le protocole RPL, nous avons apporté les changements suivants :

   D'abord, dans le fichier `rpl-icmp6.c`, qui se trouve dans le répertoire `contiki/core/net/rpl/` du système d'exploitation Contiki OS. Nous avons introduit des commandes `printf()` dans les fonctions gérant les messages de contrôle afin de suivre leur envoi et leur réception. La figure 4.8

illustre l'ajout de la commande `printf("DAO sent\n");` dans la fonction `dao-output()`, tel qu'intégrée dans le code source.

```
/ ---------------------------------------------------------
void
dao_output(rpl_parent_t *parent, uint8_t lifetime)
{
printf("DAO sent\n"); // ajouter cette ligne.
```

FIGURE 4.8 – Ajout de ligne pour fonction dans code source de rpl-icmp6.c.

Comme pour d'autres modifications, les lignes suivantes ont été ajoutées au bon endroit dans le code source :

```
// Dans la fonction dio_output()
printf("DIO sent\n");

// Dans dio_input()
printf("DIO received\n");

// Dans dao_output()
printf("DAO sent\n");

// Dans dao_input()
printf("DAO received\n");

// Dans dis_ouput()
printf("DIS received\n");

// Dans dis_input()
printf("DIS received\n");
```

Ces instructions autorisent l'impression de chaque message de contrôle dans la fenêtre Mote Output de Cooja pendant la simulation. Ensuite, nous avons enregistré les résultats dans un fichier texte (coojaRPLOutput.log) à l'aide de l'option Save output to file. Nous avons employé un script Perl, pour effectuer l'analyse. Ce script parcourt le fichier ligne par ligne et compte le nombre d'occurrences des mots-clés "DIO", "DAO" et "DIS".

Le contenu du script Perl utilisé pour compter le nombre d'occurrences des mots-clés `DIO`, `DAO` et `DIS` est présenté dans la figure 4.9.

```perl
#!/usr/bin/perl
use strict;
my ($dio, $dao, $dis) = (0, 0, 0);

while (<>) {
    $dio++ if /DIO/;
    $dao++ if /DAO/;
    $dis++ if /DIS/;
}

print "DIO: $dio\nDAO: $dao\nDIS: $dis\n";
```

FIGURE 4.9 – Contenu de Script Perl.

- **Résultats et Discussion :**

1. **Topologie graphique du réseau :**
   La figure 4.10 présente une structure de routage plus complexe comportant plusieurs sauts intermédiaires et des chemins de routage réparties qui respectent la configuration hiérarchique standard RPL. Les nœuds conservent des valeurs de rang légitimes et déterminent des chemins de routage fondés sur des métriques véritables, créant une topologie de réseau solide et résiliente dotée de multiples voies redondantes pour le flux du trafic.

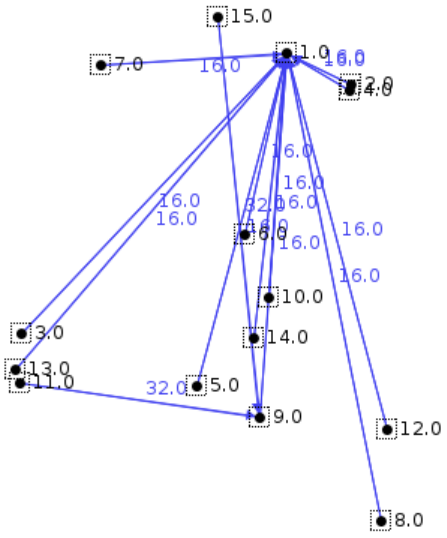

FIGURE 4.10 – Topologie graphique du réseau.

2. **Consommation d'énergie moyenne :**
   Après avoir procédé à la simulation à l'aide de Collect View, nous avons

obtenu les résultats suivants, présentés dans la figure 4.11 ci-dessous.

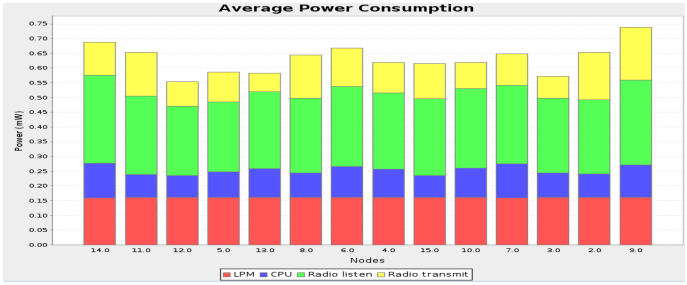

FIGURE 4.11 – Le graphe de consommation d'énergie moyenne du réseau.

Cette figure présente le graphe de consommation d'énergie moyenne du réseau, qui illustre une consommation moyenne par nœud entre 0,55 W et 0,75 W. Le LPM (environ 0,15 W) et le CPU (environ 0,10 W) restent stables, alors que l'usage de la radio, principalement en mode écoute (0,20–0,30 W), est dominant largement. Les transmissions (0,05–0,20 W) changent en fonction de l'activité des nœuds. Le nœud 9 consomme le plus d'énergie (0,75 W), possiblement en tant que coordonnateur ou relais, tandis que le nœud 12 est celui qui consomme le moins (0,55 W). La radio constitue entre 60 et 70 % de la consommation globale.

— **Transmission radio (Radio Transmit) :** indique la consommation d'énergie associée aux périodes où le nœud émet des transmissions radio (i.e. le nœud est actif).

— **Écoute radio (Radio Listen) :** représente la consommation d'énergie pendant les périodes où le nœud est en écoute de signaux provenant d'autres nœuds du réseau.

— **CPU :** représente la consommation d'énergie liée à l'activité du processeur du nœud.

— **LPM (Low Power Mode- Mode basse consommation) :** indique la consommation d'énergie lorsque le nœud est en mode basse consommation (mode inactif).

Comme on peut le constater dans les graphiques ci-dessus , à savoir la figure 4.11 dans la simulation RPL normale, l

Ces chiffres indiquent que le réseau fonctionne correctement, tous les nœuds participent normalement au routage et à la communication, sans aucune trace d'attaque ou de comportement anormal détecté.

3. **Messages de contrôle :**
La figure 4.1 présente les résultats du comptage des messages de contrôle effectué à l'aide du script Perl (inclus dans Documents), basé sur le fichier

`.log` enregistré.

Les résultats que nous avons obtenus grâce à la simulation standard sans attaques sont justes. Ainsi, ils peuvent servir de référence pour comparer les résultats d'autres simulations.

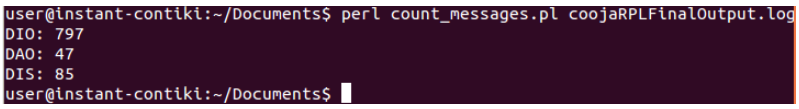

FIGURE 4.12 – Les message de contrôle dans le réseau sans attaque.

4.3.2 Implémentation des attaques de rang

Cette section détaillera la mise en place de nœuds malveillants au sein du réseau de référence et expliquera comment déclencher des attaques de rang. L'objectif de la simulation de l'attaque est de comprendre l'impact de la consommation d'énergie subi par les nœuds terminaux lorsqu'un nœud malveillant déclenche une attaque. Il sera également précisé que le nœud malveillant qui est dans notre cas le noeud 15 maintiendra une position précise pour chaque simulation afin de bien observer l'impact de chaque attaque.

- **Méthodologie :**

La méthode utilisée pour atteindre cet objectif peut être réalisée en suivant les étapes suivantes :

— Dupliquer le dossier Contiki pour créer une nouvelle instance du système d'ex ploitation Contiki. Nous avons renommé cette instance decreased-rank.

— Modifier les fichiers correspondants en fonction de l'attaque de diminution de rang.

— Créez un nouveau nœud (malveillant) en compilant le firmware du nœud dans la nouvelle instance de Contiki.

— Ajoutez le nœud au réseau de référence.

Pour la création d'un nœud malveillant se fait dans une autre instance de Contiki. Dans le cadre de l'étude, nous avons remplacé le nœud 15 par un nœud malicieux. Voici un exemple de la création du nœud malveillant dans la figure 4.13 :

FIGURE 4.13 – L'ajout du noeud malicieu.

Le réseau avec le nœud malveillant est représenté dans la figure 4.14.

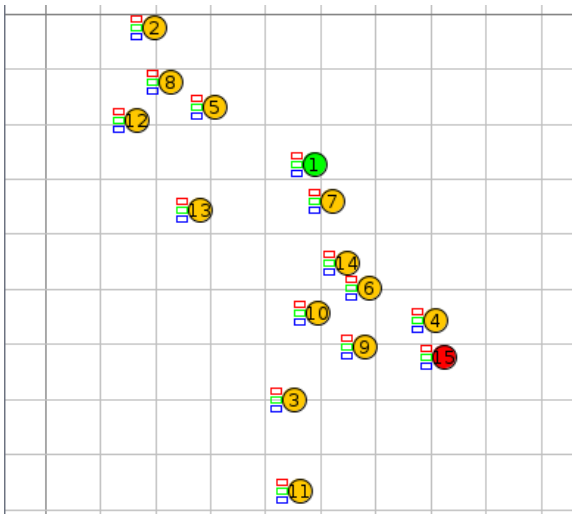

FIGURE 4.14 – Topologie du réseau avec un noeud malicieu.

4.3.2.1 Implémentation de l'attaque de diminution de rang

Dans ce cas précis, on a 15 noeuds dans la topologie. Parmi ces nœuds, il y a un nœud sink (récepteur) et 13 nœuds sender (émetteurs). Il y a également un unique nœud malicious (ID = 15). Ce dernier met en œuvre une attaque de diminution de rang en annonçant un rang artificiellement bas dans les messages DIO, visant à attirer le plus grand nombre possible de nœuds enfants. En manipulant le processus de sélection du parent de RPL, le nœud malveillant devient un point de passage privilégié pour le trafic réseau. Cette opération perturbe la topologie normale, engendre une surcharge sur certains nœuds. Après avoir simulé cette attaque pendant une durée de 15 minutes (temps réel), nous analyserons et interpréterons les conséquences qui en découlent.

- **Méthodologie :**

Pour réaliser cette attaque, il sera nécessaire de modifier du code dans les fichiers RPL qui sont situés dans : ‘’`contiki/core/net/rpl/`’’.

Rpl-private.h : Ce fichier contient les déclarations privées pour l'implémentation RPL Contiki, comme les valeurs par défaut des messages de contrôle, les temporisateurs, le mode de fonctionnement, les tables de routage DAG, et il contient diverses définitions liées au calcul du classement DAG.
L'implémentation de l'attaque diminution du rang peut être mise en œuvre en modifiant certaines de ses constantes. La figure 4.15 présente les modifications du code.

```
#define RPL_LIFETIME(instance, lifetime) \
        ((unsigned long)(instance)->lifetime_unit * (lifetime))

#ifndef RPL_CONF_MIN_HOPRANKINC
#define RPL_MIN_HOPRANKINC          0 // modifier set #define RPL_MIN_HOPRANKINC a 0
#define RPL_MIN_HOPRANKINC          256
#else
#define RPL_MIN_HOPRANKINC          RPL_CONF_MIN_HOPRANKINC
#endif
#define RPL_MAX_RANKINC             0  // changer (7 * RPL_MIN_HOPRANKINC) a 0

#define DAG_RANK(fixpt_rank, instance) \
  ((fixpt_rank) / (instance)->min_hoprankinc)

/* Rank of a virtual root node that coordinates DAG root nodes. */
#define BASE_RANK                   0

/* Rank of a root node. */
#define ROOT_RANK(instance)         (instance)->min_hoprankinc

#define INFINITE_RANK               256     // remplacer 0xffff par 256
```

FIGURE 4.15 – Les modifications ajoutées pour déclancher l'attaque Diminution du rang.

Rpl-timers.c : Il contient le code qui recalcule les rangs des nœuds en RPL. L'implémentation de l'attaque du rang nécessite également de désactiver ce recalcul, afin que les effets de diminution du rang ne soient pas annulés. Pour cela, on supprime la ligne concernée. La figure 4.16 présente la ligne à supprimer.

```
static void handle_dio_timer(void *ptr);

static uint16_t next_dis;

/* dio_send_ok is true if the node is ready to send DIOs */
static uint8_t dio_send_ok;

/*---------------------------------------------------------------------------*/
static void
handle_periodic_timer(void *ptr)
{
  rpl_purge_routes();
  rpl_recalculate_ranks();   // supprimer cette ligne

  /* handle DIS */
#ifdef RPL_DIS_SEND
  next_dis++;
  if(rpl_get_any_dag() == NULL && next_dis >= RPL_DIS_INTERVAL) {
    next_dis = 0;
    dis_output(NULL);
  }
#endif
  ctimer_reset(&periodic_timer);
```

FIGURE 4.16 – Supression de la ligne du code source.

La Figure 4.17 présente la topologie du réseau après l'attaque de diminution de rang.

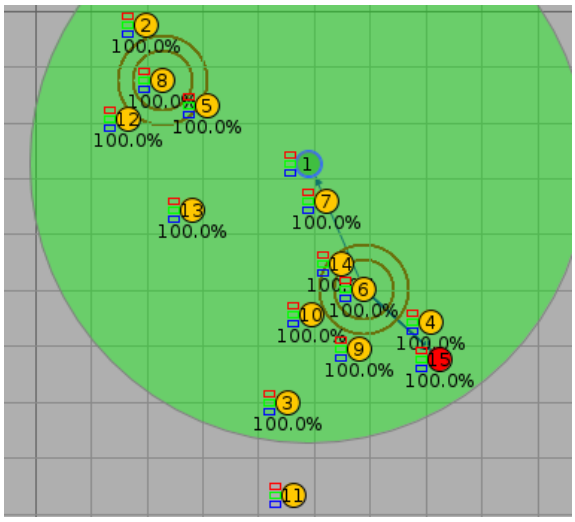

FIGURE 4.17 – La topologie de la simulation aprés l'attaque de diminution de rang.

- **Résultats et Discussion**

1. **Topologie graphique du réseau :**

   La figure 4.18 illustre une topologie de réseau qui connaît d'importants changements, démontrant l'efficacité de l'attaque par diminution de rang. En particulier, la structure de routage est simplifiée, plusieurs nœuds établissant des connexions plus directes vers les nœuds supérieurs, contournant efficacement les protocoles de routage hiérarchiques standard. Cette réduction indique que les nœuds malveillants ont réussi à annoncer des valeurs de rang délibérément inférieures, devenant ainsi plus attrayants en tant qu'options de routage pour les nœuds adjacents à la recherche de voies optimales vers la destination.

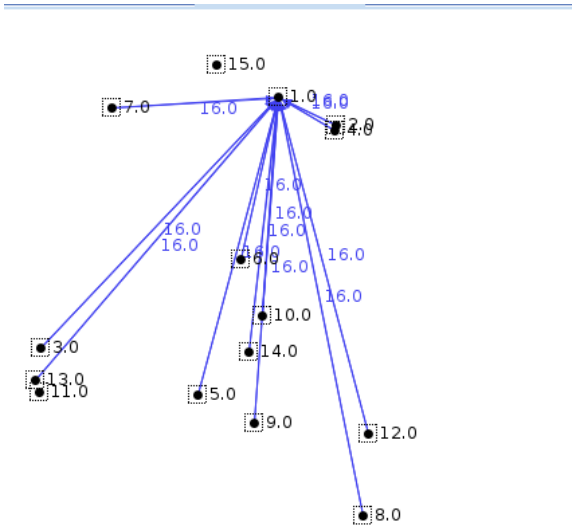

FIGURE 4.18 – Topologie graphique du réseau aprés l'attaque de diminution de rang.

2. **Consommation d'énergie moyenne :**

   Dans le cas de l'attaque par diminution de rang, la consommation d'énergie varie de 0,55 à 0,70 mW sur l'ensemble des nœuds. Le nœud 9 affiche la plus grande consommation avec 0,70 mW, tandis que les nœuds 5 et 8 consomment au minimum près de 0,55 mW. Cette répartition de puissance plutôt restreinte suggère que l'attaque a réussi à simplifier la structure hiérarchique du réseau, en supprimant les nœuds supérieurs distincts à forte puissance qui s'occupent généralement des tâches majeures de transmission de données dans les opérations RPL standard. L'utilisation du processeur assure une constance entre 0,08 et 0,10 mW et la consommation LPM reste stable autour de 0,15 mW. Cela prouve que l'attaque vise spécifiquement le comportement du protocole de routage sans affecter les opérations des nœuds essentiels.

   La Figure 4.19 illustre cette variation de consommation moyenne par nœud dans le cas de l'attaque de diminution de rang.

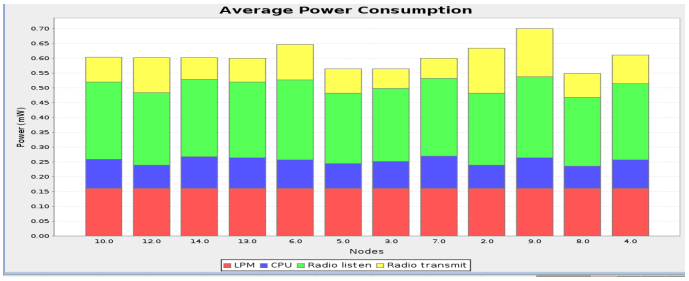

FIGURE 4.19 – Le graphe de consommation d'énergie moyenne aprés l'attaque de diminution de rang.

3. **Messages de contrôle :**
   Le tableau 4.1 compare le nombre de messages DIO/DAO/DIS délivrés par chaque nœud dans deux scénarios : sans attaque et avec l'attaque activée.

TABLE 4.1 – Les message de contrôle aprés l'attaque de diminution de rang

Type de Message	Réseau Normal	Attaque Diminution de Rang	Variation
DIO	797	1056	+259
DAO	47	93	+46
DIS	85	86	+1

   Le graphique 4.20 ci-dessous illustre une comparaison des messages de contrôle RPL entre le cas normale et celle d'une attaque diminution de rang. On remarque une forte augmentation des messages DIO (de 797 à 1056) et des messages DAO (de 47 à 93) dans le cas de l'attaque. Cela s'explique par la perturbation de la topologie : les nœuds modifient fréquemment leur parent en raison des informations erronées de rang, ce qui

entraîne une augmentation du nombre de messages nécessaires à la réorganisation du réseau. En revanche, le nombre de messages DIS reste presque inchangé (85 à 86), car les DIO sont déjà suffisamment nombreux pour transmettre l'information aux nœuds sans qu'ils aient à la demander eux-mêmes.

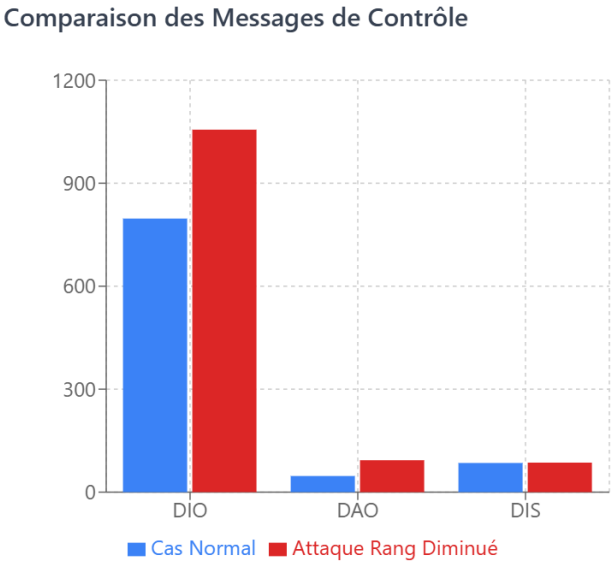

FIGURE 4.20 – La courbe des messages de controle pendant l'attaque diminution de rang et sans attaque.

4.3.2.2 Implémentation de l'attaque de rang augmenté

Dans ce cas précis, on a 15 nœuds dans la topologie. Parmi ces nœuds, il y a un nœud sink (récepteur) et 13 nœuds sender (émetteurs). Il y a également un unique nœud malicious (ID = 15). Ce dernier met en œuvre une attaque de type augmentation de rang. Ce dernier falsifie les messages DIO en déclarant un rang anormalement élevé, le rendant ainsi moins attractif en tant que parent dans l'algorithme de construction de la topologie RPL. L'objectif de cette tactique est de perturber la structure standard du réseau, en isolant des groupes de nœuds ou en les forçant à choisir des chemins de routage sous-optimaux. Cette attaque, simulée en temps réel pendant 15 minutes, sera par la suite examinée pour déterminer son effet sur la structure du réseau et l'efficacité globale.

- **Méthodologie :**

Pour réaliser cette attaque, il sera nécessaire de modifier du code dans le fichier RPL `rpl-icmp6.c` qui sont situés dans : `''contiki/core/net/rpl/''`.

Nous avons introduit les commandes suivantes dans le fonction gérant le message de controle messages de contrôl `static void dio-output(...)` :

```
    // Dans la fonction dio_output()
uint16_t fake_rank = dag->rank;
if(node_id == MALICIOUS_NODE_ID) {
```

```
  printf("Increased Rank Attack: Node %u sending fake high
      rank\n", node_id);
  fake_rank += 1024; // oubien autre grand valeur
}
set16(buffer, pos, fake_rank);
```

La Figure 4.22 présente la topologie du réseau après l'attaque de rang augmenté.

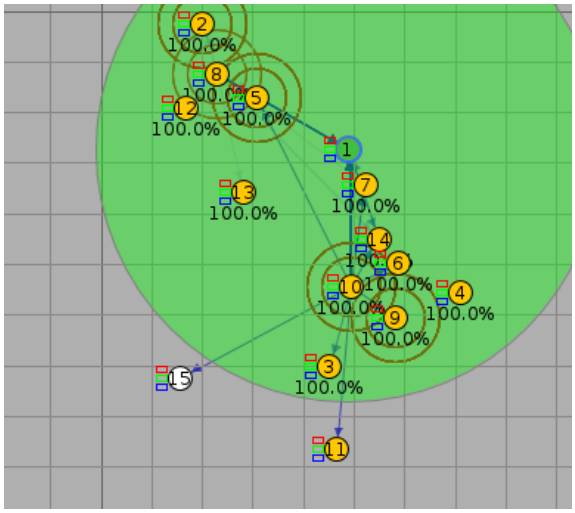

FIGURE 4.21 – La topologie de la simulation aprés l'attaque de rang augmenté.

- **Résultats et Discussion**

1. **Topologie graphique du réseau :**

   La figure 4.22 illustre une structure de routage qui devient plus complexe et fragmentée, certains nœuds établissant des chemins plus étendus et moins performants. Cette complexification indique que les nœuds malveillants ont réussi à communiquer des valeurs de rang artificiellement hautes, devenant moins séduisants pour le routage direct tout en provoquant des désordres dans la topologie générale du réseau qui contraint les nœuds légitimes à trouver des chemins alternatifs moins optimaux.

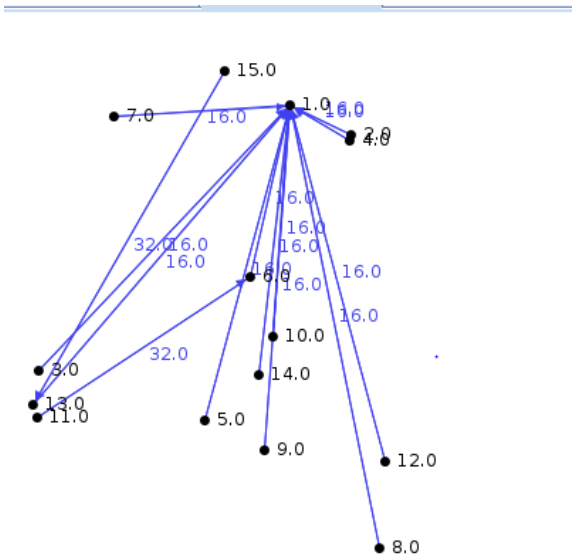

FIGURE 4.22 – Topologie graphique du réseau aprés l'attaque de rang augmenté.

2. **Consommation d'énergie moyenne :**

   Dans le cas de l'attaque de rang augmente, la consommation d'énergie varie de 0,52 à 0,67 mW, le nœud 9 ayant la valeur la plus élevée (0,67 mW) et le nœud 12 la plus basse (0,52 mW), générant une topologie déséquilibrée où certains nœuds sont fortement utilisés tandis que d'autres sont sous-utilisés. L'écoute radio domine la consommation d'énergie entre les nœuds, alors que la transmission radio varie considérablement entre les nœuds, l'utilisation du processeur restant stable à 0,08-0,10 mW et le LPM à 0,15-0,17 mW.

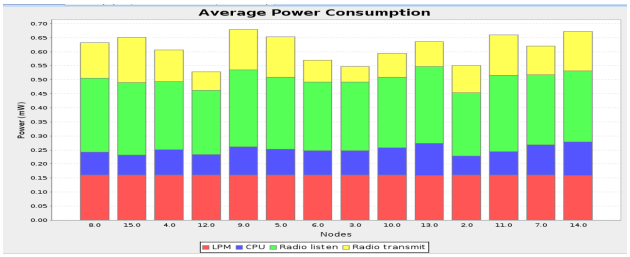

FIGURE 4.23 – Le graphe de consommation d'énergie moyenne aprés l'attaque de rang augmenté.

3. **Messages de contrôle :**
   Le tableau 4.2 compare le nombre de messages DIO/DAO/DIS délivrés par chaque nœud dans deux scénarios : sans attaque et avec l'attaque activée.

TABLE 4.2 – Les message de contrôle aprés l'attaque de rang augmenté.

Type de Message	Réseau Normal	attaque de rang augmenté	Variation
DIO	797	828	+31
DAO	47	59	+12
DIS	85	83	-2

Le graphique 4.24 suivant représente une comparaison des messages de contrôle RPL entre le cas normal et celle d'une attaque de rang augmenté. On observe une légère augmentation des messages DIO (de 797 à 828) et une augmentation plus significative des messages DAO (de 47 à 59). Cette variation s'explique par les effets perturbateurs de l'attaque sur les routes descendantes : un certain nombre de nœuds doivent réinitialiser ou modifier leurs routes vers la racine, ce qui conduit à la transmission de DAO supplémentaires. L'augmentation légère des DIO indique des modifications mineures dans la topologie, sans instabilité significative.

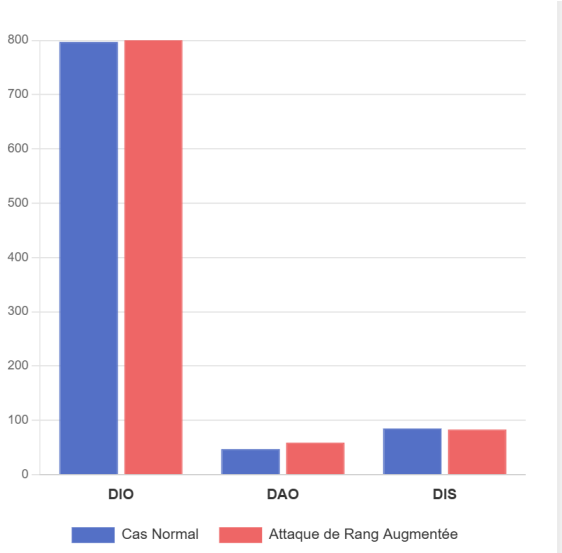

FIGURE 4.24 – La courbe des messages de controle pendant l'attaque de rang augmenté et sans attaque.

4.4 La détection de l'attaque de diminuation de rang avec l'algorithme d'apprentissage automatique

Cette partie explore l'application de l'apprentissage automatique pour identifier les attaques de diminution de rang. Le but est de détecter automatiquement les indices d'une attaque en se basant sur les informations collectées après la simulation de cas normal et de cas d'attaque.

4.4.1 Objectif

L'objectif est d'évaluer la capacité de divers algorithmes d'apprentissage automatique à détecter une attaque du type Decreased Rank dans un réseau RPL. Ainsi, un jeu de données équilibré (attaque /normal) a été employé pour former et évaluer les modèles.

Dans cette étude, quatre algorithmes supervisés de machine learning ont été mises en œuvre pour détecter l'attaque de diminuation de rang dans les réseaux RPL : Forêt Aléatoire (Random Forest), arbre de décision (Decision Tree), K plus proches voisins (KNN) et régression Logistique .

4.4.2 Méthodologie

L'environnement Google Colab a été employé pour mener à bien l'expérimentation. Cela offre l'opportunité d'utiliser un environnement cloud interactif avec les bibliothèques Python requises déjà installées. On suivi a les étapes suivantes :

1. **Importation des bibliothèques :** chargement des modules nécessaires (pandas, time, sklearn, etc.) qui permet l'analyse des données, l'entraînement des modèles et l'évaluation des performances.

2. **Ajout des fonctions :** pour récupérer la valeur temporelle en millisecondes et pour détermine le taux de précision d'une matrice de confusion.

3. **Chargement et préparation des données CSV simulées :**importation des fichiers contenant les donnees normales (DR-10N1R.CSV) et les donnees dattaque (DR-9N1M1R.CSV).

4. **Création des jeux de données (datasets) avec Pandas :** à l'aide de pandas nous avons transformé les fichiers CSV en dataset manipulable en phyton.

5. **Équilibrage des classes (même nombre de lignes dans chaque classe) :**afin d'éviter une disparité entre les classes (normal vs attaque), D'abord, on doit récupérer les numéros de rangées des ensembles de données. Ensuite, avec l'utilisation de boucles if, on récupère des numéros de rangées des ensembles de données.

6. **Fusion des des donnees :** nous avons concaténé les deux datasets (normal + attaque) pour créer un dataset complet à analyser.

7. **Extraction des variables :** nous avons séparé les variables de tel sorte que (X) represente les caracteristique du réseau (exemple : délais, messages, etc.), (y) correspond aux étiquettes qui indique si le comportement est normale ou malicioux .

8. **Séparation et normalisation des données :** nous avons découpé le dataset en deux sous-ensembles, un jeux d'entraînement (2/3) et un jeu de test (1/3). Ensuite, nous avos appliqué une normalisation à l'aide de la méthode StandardScaler afin que toutes les caractéristiques soient normalisées à la même échelle.

9. Entrainement des modèles : pour chaque modèle, on a utilisé un script python afin d'avoir des résultats pour les évaluer.

4.4.3 Résultats

L'évaluation de l'efficacité des modèles d'apprentissage supervisé a été effectuée en utilisant les indicateurs suivants :

- **Matrice de confusion :** Elle sert à juger les prédictions et démontre comment le modèle a catégorisé les situations normales et les attaques selon les critères de vrais positifs (TP), vrais négatifs (TN), faux positifs (FP) et faux négatifs (FN).

TABLE 4.3 – Matrice de confusion.

	Prédit : Normal	**Prédit : Attaque**
Réel : Normal	TN	FP
Réel : Attaque	FN	TP

- **Taux de précision (Accuracy) :** Évalue le pourcentage global de prédictions correctes en se basant sur la formule ci-dessous :

$$\text{Taux de précision} = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.1}$$

- **Precision :** Elle évalue la précision des prédictions qui sont positives (Parmi toutes les actions que le modèle a identifiées comme attaques , combien étaient réellement des attaques ?).

$$\text{Précision} = \frac{TP}{TP + FP} \tag{4.2}$$

- **Rappel :** Elle évalue la capacité du modèle à détecter les attaques (Sur l'ensemble des attaques effectives, combien ont été identifiées.)

$$\text{Rappe} = \frac{TP}{TP + FN} \tag{4.3}$$

- **F1-Score :** Il s'agit d'une moyenne harmonique entre la précision et le rappel.

$$\text{F1-Score} = 2 * \frac{Precision * Rappel}{Precision + Rappel} \tag{4.4}$$

- **Temps d'entraînement (ms) :**représente le temps qu'un algorithme met pour se former à partir des données d'apprentissage, exprimé en millisecondes (ms).

les resultats de la matrice de confusion pour chaque algorithme sont les suivants :

— **Régression Logistique :**le modele a bien detecte le comportement malicieux (34 vraie positifs) mais il a aussi genere un nombre eleve de faux positifs (35), ce qui diminue sa fiablite.

$$\begin{bmatrix} 20 & 35 \\ 11 & 34 \end{bmatrix}$$

— **Arbre de décision :** bonne balance entre détection et erreur, mais les 24 FP montrent une certaine conclusion entre comportement normal et malicieux.

$$\begin{bmatrix} 31 & 24 \\ 20 & 25 \end{bmatrix}$$

— **Forêt Aléatoire :**légère amélioration par rapport à l'arbre de décision, avec un meilleur équilibre entre TP et FP.

$$\begin{bmatrix} 31 & 24 \\ 19 & 26 \end{bmatrix}$$

— **K plus proches voisin :**résultat plus équilibré, mais moins de marques, nombre d'une performance moyenne.

$$\begin{bmatrix} 33 & 22 \\ 22 & 23 \end{bmatrix}$$

Les résultats tirés de l'environnement Google Colab sont illustrés dans le tableau 4.4 suivant.

TABLE 4.4 – Résultats des algorithmes de détection (attaque Decreased Rank)

Algorithme	Accuracy	Précision	Rappel	F1-score	Temps (ms)
Régression Logistique	0.54	0.49	0.76	0.60	7
Arbre de décision	0.56	0.51	0.56	0.53	5
Forêt Aléatoire	0.58	0.52	0.58	0.55	334
K plus proches voisins	0.56	0.51	0.51	0.51	11

4.4.4 Discussionet et analyse des résultats

Nous avons examiné les résultats de la détection de l'attaque de downgrade grâce à quatre algorithmes d'apprentissage automatique. Cette étude inclut des mesures de précision, de rappel, de F1-score, de matrice de confusion et la durée d'entraînement.

— **Régression Logistique :**Identifie efficacement les attaques (meilleur rappel : 0,76) donc utile si le but est de réduire les attaques non identifiées, même au risque de générer des faux positifs, cependant, il produit de nombreuses alertes erronées (précision : 0,49). Score F1 : 0,60 Un bon équilibre, cependant perturbé par les fausses alertes. Très rapide à former (7 ms).

— **Forêt Aléatoire :** présente la précision la plus élevée (0.58), offrant un bon compromis entre performance et détection, néanmoins avec un délai de formation important (334 ms).

— **K plus proches voisins et Arbre de décision :**Résultats stables mais moyens . Facile à implémenter, rapide .

Pour conclure, étant donné que notre objectif principal est de maximiser la détection des attaques, nous avons choisi de nous baser sur la métrique du rappel (recall), qui mesure la capacité du modèle à identifier les attaques sans en laisser passer. Sur ce critère, c'est la régression logistique qui s'avère être l'algorithme le plus efficace pour détecter l'attaque par diminution de rang, car il minimise le risque de faux négatifs. Cependant, si l'on considère l'ensemble des performances, le modèle Random Forest reste globalement le plus équilibré, offrant à la fois une bonne précision et une détection fiable.

4.5 Conclusion

Dans ce chapitre, nous avons simulé et étudié l'impact des attaques de rang sur le protocole RPL, notamment les attaques de diminuation de rang et les attaques de rang augmenté . Les simulations effectuées avec Cooja ont prouvé que l'attaque de diminution de rang a un impact plus significatif sur le fonctionnement du protocole RPL que celle par rang augmenté, en provoquant une surcharge notable de messages de contrôle et une consommation énergétique plus élevée, témoignant d'une perturbation plus profonde de la topologie du réseau. Sur la base des résultats de simulation, nous avons collecté un ensemble de données que nous avons utilisé sur Google Colab pour identifier l'attaque en employant quatre algorithmes d'apprentissage automatique (Régression Logistique, Arbre de Décision, Random Forest et KNN). L'étude comparative des résultats a évalué la précision et la durée d'entraînement de chaque technique, soulignant leur capacité à améliorer l'identification des menaces dans les réseaux IdO.

CONCLUSION GÉNÉRALE

Il est important de retenir que l'internet des objets (IdO) représente aujourd'hui une révolution technologique majeure, ce qui le rend avec ses diverses technologies et protocoles un élément crucial dans des domaines comme les villes intelligentes et le secteur de la santé, sans oublier le militaire. Toutefois, ce développement rapide s'accompagne de nouveaux défis, notamment en matière de sécurité. Parmi les protocoles conçus pour s'adapter à ces réseaux, on a trouvé le protocole de routage RPL qui s'impose comme une solution de référence pour le routage dans les environnements contraints LLN, tout en apprenant ses modes d'opération et de communication, et en donnant la topologie RPL structurée de divers types de nœuds. Cependant, il présente plusieurs vulnérabilités de cote de ressources de trafic et même de topologie, dont l'attaque de rang, qui est la plus critique en exploitant la structure hiérarchique de protocole pour perturber gravement le réseau.

Ce mémoire a permis d'étudier en profondeur cette attaque, en analysant ses types de diminution et d'augmentation de rang et du pire parent, et en classifiant ses contre-mesures d'attaque de rang RPL, que ce soit le système de détection d'intrusion (SDI) ou la modification de mécanisme de défense. Dans un second temps, nous avons exploré une approche innovante de détection basée sur les techniques de machine learning, qui s'adaptent face à des comportements malveillants dynamiques. Selon la nature des données et les tâches d'apprentissage, on a vu que le ML peut être classé en plusieurs catégories d'apprentissage : supervisé, non supervisé, semi-supervisé et hybride, ainsi que par renforcement, accompagné par leur mode et leur algorithme et le domaine d'application.

Concernant la partie pratique, on a mis en œuvre une approche expérimentale établie sur la simulation avec Cooja et la version 3.0 de Contiki en utilisant VMware Workstation Player comme machine virtuelle. Nous avons créé la topologie d'un réseau RPL composé de 15 nœuds, un nœud racine et 14 nœuds clients, puis nous avons simulé le réseau initial pendant

15 minutes et nous avons mesuré ses performances pour observer l'impact de l'attaque de diminution et d'augmentation de rang. Nous avons choisi d'évaluer le protocole RPL en termes des deux métriques, celle de la consommation moyenne d'énergie et celle du nombre de messages de contrôle. Puis nous avons présenté et analysé les résultats obtenus.

Cette étude a produit des résultats significatifs. Nous avons pu voir comment ces attaques ont affecté deux indicateurs importants : la quantité d'énergie utilisée et le nombre de messages de contrôle envoyés sur le réseau. En ce qui concerne la consommation d'énergie, nous avons observé une augmentation significative à la suite des attaques, ce qui entraîne une accumulation de la batterie du nœud. Cela pourrait avoir des effets immédiats sur la stabilité globale du réseau et la durée de vie des nœuds. En ce qui concerne la quantité de messages de contrôle, nous avons remarqué une augmentation notable de l'exécution des attaques. Cela pourrait entraîner une augmentation du trafic réseau, une utilisation inefficace des ressources et une baisse des performances globales.

Ces résultats mettent en évidence la vulnérabilité du protocole RPL face à ces attaques spécifiques et soulignent l'importance de renforcer la sécurité dans les réseaux RPL. Des mesures de prévention et de détection ont été mises en place pour préserver l'intégrité et la stabilité du réseau. On a utilisé quatre algorithmes d'apprentissage automatique pour la détection de l'attaque de diminution de rang, dont la régression logistique, la forêt aléatoire, K plus proches voisins et l'arbre de décision. Sur la base des résultats de la simulation, nous avons collecté un ensemble de données, un dataset est constitué pour une étude comparative sur l'évaluation de la précision et la durée d'entraînement de chaque technique. Ces modèles sont évalués à travers des métriques telles que le rappel et la précision, on a comparé la performance de ces algorithmes et on a trouvé que la régression logistique montre une bonne capacité à détecter les attaques et la forêt aléatoire atteint la meilleure accuracy, par contre le KNN est plus rapide que la forêt aléatoire.

La réalisation de ce projet de fin d'études nous a été très bénéfique, où on a pu :

- Approfondir nos connaissances sur un nouveau type de réseaux, L'IoT et les LLNs.
- Comprendre le fonctionnement du protocole de routage RPL.
- Comprendre le fonctionnement de l'attaque par rang et la connaissance des techniques de détection et d'atténuation basée sur le machine learning.
- Apprendre à programmer dans l'environnement de Contiki et maitriser un nouveaun simulateur Cooja.

Perspectives Dans le futur, ce travail pourrait être amélioré en incor-

porant des modèles d'apprentissage plus avancés tels que le Deep Learning,afin d'atteindre une détection plus précise. Une autre voie potentiellement efficace serait la mise en place d'une détection en temps réel et le développement d'un système intégral de détection d'intrusion (IDS) tirant parti de l'apprentissage automatique, en s'appuyant sur les connaissances accumulés dans ce mémoire. En définitive, cette méthode pourrait être étendue à d'autres formes d'attaques sur RPL et renforcée par des systèmes de détection distribuée afin d'améliorer la sécurité des réseaux IdO.

BIBLIOGRAPHIE

[1] Weill, M. et Souissi, M. *L'Internet des objets : concept ou réalité ?* Réalités industrielles, Les Annales des Mines. ESKA, 2010. Consulté le 4 mars 2025.

[2] Benghozi, P.-J., Bureau, S., Massit-Folléa, F., Waroquiers, C. et Davidson, S. *L'internet des objets : quels enjeux pour l'Europe* Éditions de la Maison des sciences de l'homme, 2009. Consulté le 4 mars 2025.

[3] Christophe, B., Boussard, M., Lu, M., Pastor, A. et Toubiana, V. *The web of things vision : Things as a service and interaction patterns* Bell Labs Technical Journal, 2011. Consulté le 6 mars 2025.

[4] Evans, D. *L'Internet des objets : Comment l'évolution actuelle d'Internet transforme-t-elle le monde* Livre blanc, Cisco, 2011. Consulté le 7 mars 2025.

[5] Belhadj, N. et Abbad, A. *La sécurité de l'Internet des Objets (IoT)* Mémoire de fin d'études, Université U-TIARET, 2022. http://dspace.univ-tiaret.dz:80/handle/123456789/2580. Consulté le 12 mars 2025.

[6] Yick, J., Mukherjee, B. et Ghosal, D. *Wireless sensor network survey* Computer Networks, vol. 52, no. 12, pp. 2292–2330, 2008. Consulté le 12 mars 2025.

[7] Burhan, M., Rehman, R. A., Khan, B. et Kim, B.-S. *IoT elements, layered architectures and security issues : A comprehensive survey* Sensors, vol. 18, no. 9, 2018. Consulté le 17 mars 2025.

[8] Farooq, M. U., Waseem, M., Khairi, A. et Mazhar, S. *A critical analysis on the security concerns of Internet of Things (IoT)* International Journal of Computer Applications, vol. 111, pp. 1–6, 2015. DOI : 10.5120/19547-1280. Consulté le 20 mars 2025.

[9] Said, O. et Masud, M. *Towards Internet of things : Survey and future vision* International Journal of Computer Networks, vol. 5, pp. 1–17, 2013. Consulté le 22 mars 2025.

[10] Shi, Y.-R. et Hou, T. *Internet of Things key technologies and architectures research in information processing* In : *Proceedings of the 2nd*

International Conference on Computer Science and Electronics Engineering (ICCSEE), 2013. Consulté le 23 mars 2025.

[11] Khan, R., Khan, S. U., Zaheer, R. et Khan, S. *Future Internet : The Internet of Things Architecture, Possible Applications and Key Challenges* 10th International Conference on Frontiers of Information Technology (FIT 2012), 2012. Consulté le 24 mars 2025.

[12] Khan, M. R. (2023). *Performance and route stability analysis of RPL protocol* [Master's Degree Project, Stockholm, Sweden]. Consulté le 25 mars 2025.

[13] Thubert, P., Winter, T., Brandt, A., Hui, J., Kelsey, R., Levis, P., Pister, K., Struik, R., Vasseur, J., & Alexander, R. (2012). RPL : IPv6 Routing Protocol for Low Power and Lossy Networks. *RFC 6550, IETF*. Consulté le 2 avril 2025.

[14] Mansour, M., & Boughebache, B. (2024). *L'impact des attaques de type topologie sur les réseaux RPL* [Mémoire de fin d'études, Université de Kasdi Merbah Ouargla]. Consulté le 2 avril 2025.

[15] Kharrufa, H., Al-Kashoash, H. A. A., & Kemp, A. H. (2019). RPL-based routing protocols in IoT applications : A review. *IEEE Sensors Journal, 19*(15), 5952–5967. Consulté le 2 avril 2025.

[16] Ali, M.-S. A. (2021). *Estimation de la Qualité de Lien dans RPL* [Mémoire de fin d'études, Université Abdelhamid Ibn Badis – Mostaganem]. Consulté le 3 avril 2025.

[17] Burhan, M., Rehman, R. A., Khan, B., & Kim, B.-S. (2018). IoT Elements, Layered Architectures and Security Issues : A Comprehensive Survey. *Sensors*. Consulté le 4 avril 2025.

[18] Boursas, I., & Djabrouhou, I. (2024). *Mitigation de l'attaque de numéro de version contre les réseaux IoT basés sur le protocole de routage RPL* [Mémoire de master, Université Saad Dahlab Blida 1, Algérie]. Consulté le 4 avril 2025.

[19] Mahmud, A., Hossain, F., Choity, T. A., & Juhin, F. (2025). Simulation and Comparison of RPL, 6LoWPAN, and CoAP Protocols Using Cooja Simulator. In Uddin, M., & Bansal, J. (Eds.). Consulté le 4 avril 2025.

[20] Sharma, S., & Verma, V. K. (2020). Security explorations for routing attacks in low power networks on the internet of things. *The Journal of Supercomputing, 77*, 4778–4781. https ://doi.org/10.1007/s11227-020-03471-z. Consulté le 4 avril 2025.

[21] Levis, P., Gnawali, O., Clausen, T. H., Ko, J., & Hui, J. W. (2011). RFC 6206 : The Trickle Algorithm. *Internet Engineering Task Force (IETF)*. Consulté le 5 avril 2025.

[22] Diakite, M. (2021). *Analyse et contre-mesure des attaques de topologie dans les réseaux LLN* [Mémoire de fin d'études, Université Abdelhamid Ibn Badis – Mostaganem]. Consulté le 5 avril 2025.

[23] Ramarosaona, M. (2017). *Conception et création d'une plateforme IoT avec le protocole MQTT* [Mémoire de fin d'étude, Université d'Antananarivo]. Consulté le 6 avril 2025.

[24] Journal du Net. *Les réseaux IoT*. [En ligne]. Disponible sur : https://www.journaldunet.fr/web-tech/dictionnaire-de-l-iot/1181267-les-reseaux-iot/. Consulté le 7 avril 2025.

[25] IHS Markit. *Rapid expansion projected for smart home devices*. IHS Markit Online Newsroom. Disponible sur : https://news.ihsmarkit.com/press-release/technology/rapid-expansion-projected-smarthomedevices-ihs-markit-says. Consulté le 7 avril 2025.

[26] Mayuri, A., & Sudhir, T. (2015). Internet of Things : Architecture, Security Issues and Countermeasures. *International Journal of Computer Applications, 125*(14). Consulté le 8 avril 2025.

[27] Haniche, M., & Tabrait, N. (2019). *Internet des objets dans le domaine de l'agriculture de demain* [Thèse de doctorat, Université Mouloud Mammeri]. Consulté le 8 avril 2025.

[28] Baker, N. (2005). « ZigBee and Bluetooth strengths and weaknesses for industrial applications ». *Computing & Control Engineering Journal, 16*(2), 20–25. DOI : 10.1049/cce :20050204. Consulté le 8 avril 2025.

[29] Montenegro, G., et al. (2007). « Transmission of IPv6 Packets over IEEE 802.15.4 Networks ». *RFC 4944, RFC Editor*. Disponible sur : https://www.rfc-editor.org/info/rfc4944. Consulté le 9 avril 2025.

[30] Augustin, A., et al. (2016). « A Study of LoRa : Long Range & Low Power Networks for the Internet of Things ». *Sensors, 16*(9). Disponible sur : https://www.mdpi.com/1424-8220/16/9/1466. Consulté le 9 avril 2025.

[31] Gomez, Carles, Oller, Joaquim et Paradells, Josep. « Overview and Evaluation of Bluetooth Low Energy : An Emerging Low-

Power Wireless Technology ». Sensors, vol. 12, no 9, 2012, p. 11734–11753. [Consulté le 10 avril 2025]. Disponible sur : https ://www.mdpi.com/1424-8220/12/9/11734.

[32] Bitaillou, Alexis, Parrein, Benoît et Andrieux, Guillaume. « Synthèse sur les protocoles de communication pour l'Internet des objets de l'industrie 4.0 ». Thèse de doctorat, Université de Nantes, 2019. [Consulté le 9 avril 2025].

[33] Houha, A., Mehah, Siham, Ouabba, Lamia et al. « Internet of Things, protocoles de communication et simulation d'un scénario [maison intelligente]». Thèse de doctorat, Université Abderrahmane Mira-Bejaia, 2021. [Consulté le 11 avril 2025].

[34] Khaldi, Tinhinane et Khelifi, Cylia. « Étude d'impacts des attaques sur le protocole de routage RPL dans l'IoT ». Mémoire de master, Université Abderrahmane Mira-Bejaïa, 2020. [Consulté le 11 avril 2025].

[35] Labiod, Yasmine. «Mécanisme de sécurité pour l'Internet des objets ». Thèse de doctorat, Université Badji Mokhtar - Annaba, 2022. [Consulté le 11 avril 2025].

[36] Bitaillou, Alexis, Parrein, Benoît, Andrieux, Guillaume. «Synthèse sur les protocoles de communication pour l'Internet des objets de l'industrie 4.0». Rapport technique, LS2N et IETR, Université de Nantes, 2019. Identifiant : ffhal-02365063f. [Consulté le 11 avril 2025].

[37] Olsson, Jonas. «LoWPAN demystified». Texas Instruments, 2014. [Consulté le 12 avril 2025].

[38] Osterlind, F. et al. « Cross-Level Sensor Network Simulation with COOJA » In : *Proceedings of the 2006 31st IEEE Conference on Local Computer Networks*, 2006, pp. 641-648. [Consulté le 7 mai 2025].

[39] Gaddour, Olfa et Koubâa, Anis. «RPL in a nutshell». Mémoire de stage de fin d'études, Master Informatique, Institut de la Francophonie, Université Lyon 1, 14 février 2013. [Consulté le 12 avril 2025].

[40] Larouci, Aicha et Ben Slimane, Afaf. « Simulation et comparaison de protocoles de communication pour l'Internet des objets » . Mémoire de fin d'études, Université KASDI Merbah – Ouargla, 2022–2023. [Consulté le 12 avril 2025].

[41] Parasuram, Aishwarya, Culler, David et Katz, Randy. «An Analysis of the RPL Routing Standard for Low Power and Lossy Networks». Université de Californie, Berkeley, 14 mai 2016. [Consulté le 12 avril 2025].

[42] Gaddour, Olfa et Koubâa, Anis. «RPL in a nutshell : A survey». Computer Networks, vol. 56, no 14, 2012, p. 3163–3178. [Consulté le 12 avril 2025].

[43] Levis, P., Clausen, T., Hui, J., Gnawali, O., Ko, J. «The Trickle Algorithm». IETF Request for Comments 6206, mars 2011. [Consulté le 9 avril 2025].

[44] Verma, Abhishek et Ranga, Virender. «Security of RPL based 6LoWPAN Networks in the Internet of Things : A Review». DOI : 10.1109/JSEN.2020.2973677. Disponible sur : https ://ieeexplore.ieee.org/document/899828. [Consulté le 9 avril 2025].

[45] Mayzaud, A., Badonnel, R. et Chrisment, I., "A Taxonomy of Attacks in RPL-based Internet of Things," *International Journal of Network Security*, vol. 18, no. 3, mai 2016, pp. 459–473. *Consulté le 09/04/2025.*

[46] Kiran, Usha, Maurya, Poonam et Sharma, Himanshu, "Investigating Routing Protocol Attack for Low Power and Lossy IoT Networks," 29 mai 2023. *Consulté le 09/04/2025.*

[47] Prajapati, A. K. et al., "A comprehensive survey on RPL routing-based attacks, defenses and future directions in Internet of Things," 2025. *Consulté le 10/04/2025.*

[48] Kamgueu, Patrick Olivier, "Configuration dynamique et routage pour l'internet des objets," *Thèse de doctorat, Université de Lorraine*, 2017. *Consulté le 02/04/2025.*

[49] Mouradian, C., Guyen, T. N. et Glitho, R. H., "LEADER : Low Overhead Rank Attack Detection for Securing RPL based IoT," *arXiv preprint arXiv :2011.12996*, 2020. *Consulté le 11/04/2025.* Disponible sur : https://arxiv.org/abs/2011.12996.

[50] Hafique, U., Khan, A., Rehman, A., Bashir, F. et Alam, M., "Detection of Rank Attack in Routing Protocol for Low Power and Lossy Networks," *Annals of Telecommunications*, vol. 73, 2018, pp. 429–438.

[51] Ghaleb, B., Al-Dubai, A., Hussain, A., Ahmad, J., Romdhani, I. et Jaroucheh, Z., "Resolving the Decreased Rank Attack in RPL's IoT Networks," *arXiv preprint arXiv :2305.10025*, 2023.

[52] Mayzaud, A., Badonnel, R. et Chrisment, I., "Investigating Routing Protocol Attacks on Low Power and Lossy IoT Networks," *2016 1st IEEE Conference on Network Softwarization (NetSoft)*, Séoul, Corée du Sud, 2016, pp. 422–428. DOI : 10.1109/NETSOFT.2016.7502464.

[53] Manoharan, S. et Dhanalakshmi, R., "RIAIDRPL : Rank Increased Attack (RIA) Identification Algorithm for Avoiding Loop in the RPL DODAG," *International Journal of Computer Applications*, vol. 179, no. 19, avril 2018, pp. 30–36.

[54] Rekha, S. N. et Karthikeyan, S., "Enhanced Rank Attack Detection Algorithm (E-RAD) for securing RPL-based IoT networks by early detection and isolation of rank attackers," *Wireless Personal Communications*, vol. 120, 2021, pp. 117–139. *Consulté le 02/05/2025.*

[55] Shreenivas, S. V., Papadimitratos, P. et Sirsikar, S., "VeRA : Version Number and Rank Authentication in RPL," in *Proceedings of the 14th ACM International Symposium on Mobility Management and Wireless Access (MobiWac '16)*, ACM, 2017, pp. 43–50. *Consulté le 20/04/2025.*

[56] Wallgren, L., Raza, S. et Voigt, T., "Routing Attacks and Countermeasures in the RPL-Based Internet of Things," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 794326. *Consulté le 03/05/2025.* Disponible sur : https://doi.org/10.1155/2013/794326.

[57] Amara, C., Boudia, O. K. et Tari, A., "Detection and Mitigation of RPL Rank and Version Number Attacks in the Internet of Things," *SRPL-RP, Wireless Communications and Mobile Computing*, 2021,

Article ID 5550312. *Consulté le 03/05/2025*. Disponible sur : https://doi.org/10.1155/2021/5550312.

[58] Saini, H. K. et Poriye, M., "Threats, Detection and Mitigation of Rank Attack : A Survey," *Materials Today : Proceedings*, vol. 56, part 4, 2022, pp. 2894–2901. *Consulté le 03/05/2025*. Disponible sur : https://doi.org/10.1016/j.matpr.2021.11.324.

[59] Mitchell, T. M., *Machine Learning*, New York : McGraw-Hill, 1997.

[60] Zhang, Y., Wang, Y., Wang, Y., Wang, Y. et Wang, Y., "A Survey of Networking Applications Applying the Software Defined Networking Concept Based on Machine Learning," *IEEE Access*, vol. 7, 2019, pp. 95379–95391. *Consulté le 03/05/2025*. Disponible sur : https://doi.org/10.1109/ACCESS.2019.2928564.

[61] Naeem, S., Ali, A., Anam, S. et Ahmed, M. M., "An Unsupervised Machine Learning Algorithms : Comprehensive Review," *International Journal of Computing and Digital Systems*, vol. 13, no. 1, 2023, pp. 911–921. *Consulté le 03/05/2025*. Disponible sur : https://doi.org/10.12785/ijcds/130172.

[62] Neerugatti, Vikram et Reddy, A. Rama Mohan, "Machine Learning Based Technique for Detection of Rank Attack in RPL based Internet of Things Networks," *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, vol. 8, no. 9S3. *Consulté le 20/04/2025*.

[63] Ioulianou, Philokypros P., Vassilakis, Vassilios G. et Shahandashti, Siamak F., "ML-based Detection of Rank and Blackhole Attacks in RPL Networks," Département d'informatique, Université de York, Royaume-Uni. *Consulté le 20/04/2025*.

[64] Fatima-tuz-Zahra, Jhanjhi, N. Z., Brohi, Sarfraz Nawaz et Malik, Nazir A., "Proposing a Rank and Wormhole Attack Detection Framework using Machine Learning," *International Conference on Mathematics, Actuarial Science, Computer Science and Statistics (MACS)*. *Consulté le 20/04/2025*.

[65] Fatima-tuz-Zahra, Jhanjhi, N. Z., Brohi, Sarfraz Nawaz, Malik, Nazir A. et Humayun, Mamoona, "Proposing a Hybrid RPL Protocol for Rank and Wormhole Attack Mitigation using Machine Learning," *Consulté le 20/04/2025*.

[66] VMware, « VMware Workstation Player », https://www.blogdumoderateur.com/tools/vmware-workstation-player/. [Consulté le 3 mai 2025].

[67] Rajasekar, V. R. et Rajkumar, S., « Analysis of Blackhole Attack in RPL-based 6LoWPAN Network : A Case Study », dans *Proceedings of the 2021 28th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, Dubaï, Émirats arabes unis, nov.–déc. 2021, p. 1–6. DOI : https://doi.org/10.1109/ICECS53924.2021.9665623.

[68] Evans, D., « Qu'est-ce que l'IoT (Internet des objets) ? Définition, avantages et défis », *Fortinet*, s.d. [Consulté le 7 mars 2025]. Disponible

sur : https://www.fortinet.com/fr/resources/cyberglossary/iot.

ANNEXE A

Environnement de travail

L'utilisation d'outils appropriés a été essentielle pour reproduire un environnement réseau réaliste dans le cadre de notre étude. Ainsi, VMware Workstation Player a été utilisé pour la virtualisation, tandis que Contiki OS et son simulateur Cooja ont permis de simuler le comportement des nœuds et d'analyser les effets de l'attaque de rang sur le protocole RPL.

VMware Workstation Player

VMwareWorkstation Player, anciennement connu sous le nom de VMware Player,était un logiciel de virtualisation destiné aux ordinateurs x64 fonctionnant sous Microsoft Windows ou Linux. Nous allons utiliser la solution VMware Workstation Player pour exécuter une machine virtuelle nommé Instant Contiki OS. VMware Player a la capacité d'exécuter des appareils virtuels existants et de générer ses propres machines virtuelles (qui nécessitent l'installation d'un système d'exploitation pour être opérationnelles) **[66]**.

FIGURE 25 – VMware Workstation Player.

Contiki OS

Contiki OS est un système d'exploitation léger, portable, flexible et open-source conçu pour les dispositifs de mesure dans les réseaux sans fil de capteurs (WSN). Il a été rédigé en langage C afin d'optimiser sa flexibilité, ce qui lui confère une grande portabilité. Une équipe de chercheurs suédois a élaboré ce dernier en 2004. Contiki s'appuie sur un noyau basé sur les événements et propose la fonctionnalité de multitâche préemptive au niveau des processus individuels. Une configuration standard de Contiki requiert approximativement 2 Ko de mémoire vive et 40 Ko de mémoire morte. Pour faire des économies de mémoire.

Contiki fait appel à une notion nommée Protothread, qui représente une méthode intermédiaire entre le multi-threading et la programmation basée sur les événements. Contiki gère deux formes de communication. Pour commencer, il exploite une couche d'échange nommée Rime, qui facilite la communication avec les capteurs avoisinants et le routage. Rime propose une transmission de données à fibre optique.

Par la suite, Contiki se sert d'une seconde couche nommée uIP (micro-IP), qui représente une version allégée de la pile TCP/IP **[68]**.

La figure 26 ci-dessous présente cette architecture, en montrant les différentes couches de Contiki, de la couche matérielle aux applications, ainsi que les modules essentiels comme les Protothreads, uIP et les composants de gestion de noeuds.

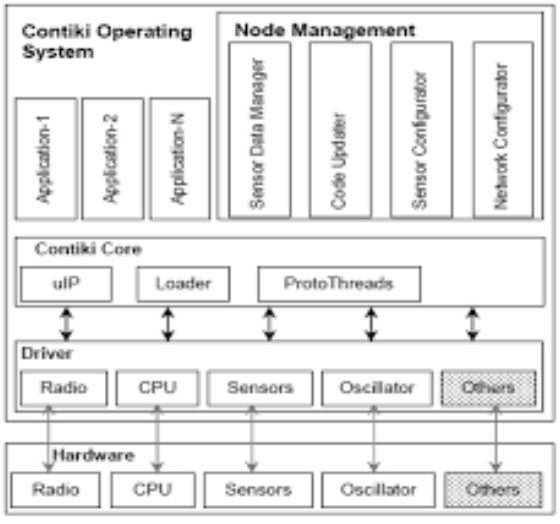

FIGURE 26 – Architecture de Contiki.

Le simulateur Cooja

L'un des outils de Contiki, Cooja, est un simulateur de réseau qui permet la simulation de plaques et de formes physiques réelles. Vous pouvez simuler un réseau de capteurs sans les utiliser réellement en communiquant avec les nœuds du réseau. Cooja offre la possibilité de tester et d'évaluer divers scénarios de réseau en simulant le comportement des nœuds et en permettant la communication inter-nœuds. Cela permet aux développeurs et aux chercheurs de tester et de valider les protocoles, les algorithmes et les applications dans un environnement contrôlé avant de les déployer sur des plateformes matérielles réelles **[38]**.
L'interface de simulateur cooja est composée de plusieurs fenêtres (plugins) :

1. **Network :** affiche graphiquement la topologie du réseau et les nœuds positionnés (motes) avec leur état actuel (address, LED, position, etc.). On peut choisir quels nœuds étudier et observer les connexions et les mouvements. Cette zone est vide au début de la simulation et nécessite l'ajout de nœuds.
2. **Simulation Control :** Cette zone est utilisée principalement pour gérer la simulation, y compris le démarrage, la recharge et l'exécution. Le temps d'exécution et la vitesse de simulation sont répartis de manière égale. Elle est composée des boutons suivants : Démarrer (pour terminer une simulation), Pause (pour arrêter une simulation), Étape (pour contrôler la durée de la simulation) et Recharger (pour redémarrer une simulation).
3. **Notes :** Elle permet d'ajouter des annotations à la simulation en enregistrant les informations supplémentaires sur cette dernière, pour garder une trace de changement ou de commentaire lié à l'expérience.
4. **Mote Output :** Cette zone affiche les sorties des différentes interfaces des nœuds (via printf dans le code Contiki). Elle est utile pour suivre le comportement des nœuds et visualiser les échanges.
5. **Timeline :** c'est une zone qui permet notament d'afficher les echange de donnes entre noeuds dans une ligne de temps (envois,receptions interferences).Très utile pour visualiser les communications et détecter les conflits ou pertes.

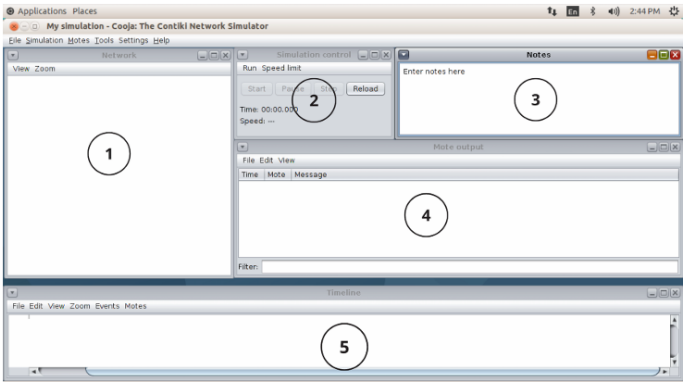

FIGURE 27 – Les fenêtres de cooja.

Google Colab (Collaboratory)

c'est une plateforme de développement cloud fournie par Google, facilitant l'exécution de code Python au sein d'un notebook Jupyter sans nécessité d'installation sur machine locale. C'est un outil particulièrement adapté pour le traitement des données, l'entraînement de modèles d'apprentissage automatique, et il donne un accès libre à des ressources matérielles telles que les GPU. Dans notre étude, nous avons eu recours à Google Colab pour examiner le jeu de données et mettre en œuvre les algorithmes de détection d'attaques.

FIGURE 28 – Google Colab.

Installation Instant Contiki

Instant Contiki est un environnement de dévéloppement simplifié, sous la forme d'une machine virtuelle Vmware, contient tout le code source de Contiki et toutes ses fonctionnalités ainsi le simulateur Cooja. Nous allons utiliser la solution VMware Workstation Player pour exécuter Contiki Instant sur MacBook Pro. Pour cela, suivez les étapes suivantes :

- Télécharger Instant Contiki.zip, qui est une machine virtuelle créée avec toutes les chaînes d'outils et les logiciels nécessaires au développement de ContikiOS qui peut être téléchargé à l'adresse suivante : https://sourceforge.net/projects/contiki/files/ .
- Unefois le package zip est téléchargé, Il faut ensuite décompresser le fichier obtenu.
- Il faut télécharger et installer VMware Workstation Player depuis l'url, https://www.vmware.com/products/workstation-player.html .
- Après avoir ouvert la fenêtre de lecture VMware Workstation Player, clique sur « Open a virtual machine » et importe le fichier instant Contiki virtual machine (vmx), puis configure la machine avec ses caractéristiques, et après, lance la machine virtuelle, entre le mot de passe par défaut : user.
- Ouvrir le terminal de contiki et exécuter les commandes suivantes :
  - « sudo apt-get update ».
  - « sudo apt-get upgrade ».
  - « sudo apt update ».

— « sudo apt upgrade ».
— « git submodule update-–init-–recursive ».
— Ouvrir le simulateur Cooja en tapant sur le « terminal » les commandes « cd contiki/tools/cooja » puis « ant run » (voir Figure 29).

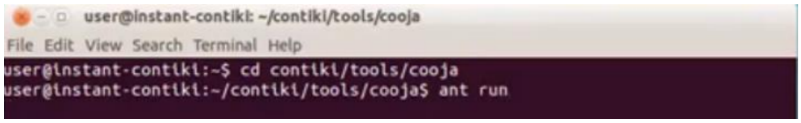

FIGURE 29 – Lancement de simulateur cooja sur terminal.

Paramètres et environnement de développement

Caractéristiques de la machine

La machine utilisée dans la simulation du réseau est caractérisée par les paramètres suivants, tels que présentés dans le tableau 5 .

TABLE 5 – Caractéristiques de la machine utilisée dans la simulation .

Champs	Valeur
Processeur	2.70GHz Intel Core i5-7300U
RAM	8 GO
Disque Dur	SSD

Paramètres de simulation

Dans notre étude, nous avons utilisé l'exemple de rpl-collect et un environnement de simulation comme montre le tableau 6 ci-dessous :

TABLE 6 – Les paramètres de la simulation .

Paramètre	Valeur
Simulateur	Cooja
Contiki	InstantContiki3.0
Couche d'adaptation	6LOWPAN
Protocole de routage	RPL
Nombre de noeuds	15
Type de noeuds	Z1 mote
Nbr de noeuds malveillant	1
Radio Environment	UDGM(Distance Loss)
Temps	10-15min
Topologie	Random

ANNEXE B

Capture de Dataset

Pour extraire un ensemble de données pour une attaque à rang réduit dans Contiki OS avec Cooja, et l'utiliser pour la détection basée sur l'apprentissage automatique, nous avons suivi un ensemble d'étapes. Voici comment procéder :

1. **Démarrer Cooja :** Compilation et exécution du simulateur Cooja.

2. **Modifier le code du nœud :** Nous avons apporté des changements directement dans le fichier rpl-icmp6.c, qui se trouve dans le répertoire contiki/core/net/rpl/ du système d'exploitation Contiki OS. Nous avons introduit des commandes printf() dans les fonctions gérant les messages de contrôle afin enregistrer les fonctionnalités clés comme heure (heure actuelle en secondes), nœud (id du nœud qui envoie le message), rang et le type de message.
3. **Exécuter la simulation :** Deux scénarios ont été simulés pendant 10 min dans l'environnement Cooja (Un scénario normal, sans attaque, représentant le comportement habituel du réseau).

4. **Enregistrer les logs :** Dans menu Cliquez sur tools → "Mote Output". Enregistrez la sortie via File Save Log.

5. **Analyser le file.log avec python pour créer un ensemble de données :** nous avons utilisé un script Python simple pour analyser le journal et enregistrer un CSV.

```
    import re
import csv

with open('cooja_log.txt', 'r') as infile, open('
   dataset.csv', 'w', newline='') as outfile:
    writer = csv.writer(outfile)
    writer.writerow(['node_id', 'time', 'rank', '
       instance_id'])

    for line in infile:
        if "DIO_SENT" in line:
            match = re.search(r'NODE:(\d+), TIME:(\d
               +), RANK:(\d+), INSTANCE_ID:(\d+)',
               line)
            if match:
                writer.writerow(match.groups())
```

6. **Ajouter des étiquettes pour ML :** Dès que les données sont en format CSV, nous ajoutons manuellement une colonne d'étiquettes. 1 → nœud influencé par un comportement malveillant.
   0 → comportement normal.

7. **Former un modèle d'apprentissage automatique :**Former un modèle d'apprentissage automatique à l'aide de Google Colab, puis le combiner et le diviser pour l'entraînement avec 4 modèles d'apprentissage automatique, afin de détecter les anomalies en fonction de la fréquence des messages, des changements de rang, etc.

Résumé

L'Internet des objets (IdO) connaît un essor considérable, avec une multitude d'objets connectés fonctionnant au sein de réseaux contraints appelés LLN (Low-power and Lossy Networks). Pour répondre aux exigences de routage dans ces environnements, le protocole RPL a été conçu. Toutefois, ce protocole présente des vulnérabilités exploitables par des attaques, notamment celles par rang, qui perturbent la topologie du réseau en manipulant les valeurs de rang des nœuds.

Dans ce mémoire, nous avons simulé deux variantes d'attaques par rang (augmentation et diminution) sur un réseau RPL à l'aide du simulateur Cooja sous Contiki OS. L'impact de ces attaques a été évalué en analysant la consommation énergétique moyenne des nœuds et le volume de messages de contrôle échangés.

À partir des données générées durant la simulation de l'attaque de diminution de rang, un ensemble de données (dataset) a été constitué. Ce dernier a servi à entraîner plusieurs modèles d'apprentissage supervisé (KNN, arbre de décision, forêt aléatoire, régression logistique) pour détecter automatiquement l'attaque. Les résultats obtenus montrent que certains modèles offrent une précision de détection élevée tout en conservant un temps d'apprentissage acceptable.

Mots-clés : IoT, RPL, attaque de rang, Cooja, détection, machine learning, apprentissage supervisé.

Abstract

The Internet of Things (IoT) is experiencing considerable growth, with a multitude of connected objects operating within constrained networks called LLN (Low-power and Lossy Networks). To meet the routing requirements in these environments, the RPL protocol was designed. However, this protocol has vulnerabilities that can be exploited by attacks, particularly rank attacks, which disrupt the network topology by manipulating the rank values of the nodes.

In this thesis, we simulated two variants of rank attacks (increase and decrease) on an RPL network using the Cooja simulator under Contiki OS. The impact of these attacks was evaluated by analyzing the average energy consumption of the nodes and the volume of control messages exchanged.

From the data generated during the rank reduction attack simulation, a dataset was created. The latter was used to train several supervised learning models (KNN, decision tree, random forest, logistic regression) to automatically detect the attack. The results obtained show that some models offer high detection accuracy while maintaining an acceptable learning time..

Keywords : IoT, RPL, rank attack, Cooja, detection, machine learning, supervised learning.