الجمهورية الجزائرية الديمقراطية الشعبية

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة التعليم العالي والبحث العلمي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

جامعة أبي بكربلقايد – تلمسان –

Université Aboubakr Belkaïd – Tlemcen – Faculté de TECHNOLOGIE

MEMOIRE

Présenté pour l'obtention du **diplôme** de **MASTER En** :

Télécommunications

Spécialité : Réseaux et Télécommunications

Par : Latti Achraf Abd Essadek

Sujet

Surveillance des écosystèmes IoT par l'Intelligence Artificielle : Détection et analyse des comportements malveillants

Soutenu publiquement, le 30/09/2026, devant le jury composé de :

M^me^ MELIANI Maghnia	PROFESSEUR	Université de Tlemcen	Présidente
M^me^ BADAOUI Hadjira	PROFESSEUR	Université de Tlemcen	Examinatrice
M^me^ Benaissa Amel	MCB	Université de Tlemcen	Encadrant
M^me^ Taleb Sarra	MCB	Université de Tlemcen	Co-encadrant

Année universitaire : 2026/202

Remerciements

Avant tout, je remercie Dieu le Tout-Puissant de m'avoir accordé la force, la patience et le courage nécessaires pour mener à bien ce travail.

J'exprime ma profonde gratitude à ma directrice

M^{me} Benaissa, pour avoir accepté d'encadrer ce travail, ainsi que pour ses précieuses orientations et sa rigueur scientifique qui ont grandement enrichi cette recherche.

Je tiens également à remercier chaleureusement ma co-encadrante, **M^{me} Taleb**, pour son soutien constant, sa disponibilité, sa bienveillance et ses conseils judicieux tout au long de la réalisation de ce projet.

Nos remerciements **s'adressent** aussi aux membres du jury qui ont accepté d'évaluer et d'honorer ce modeste travail.

Je tiens à exprimer ma profonde gratitude à l'ensemble des professeurs qui ont assuré ma formation et m'ont transmis leur précieux savoir tout au long de mon cursus universitaire. Leurs compétences et leur dévouement ont été essentiels à mon développement académique et professionnel.

Un immense merci à **ma tendre mère**, source d'amour inconditionnel et de prières éternelles, qui a toujours été mon phare dans les moments de doute. Que Dieu la préserve. À **ma chère famille**, pour leur soutien indéfectible, leurs sacrifices et leur présence rassurante à chaque étape de mon parcours.

Enfin, je tiens à remercier toutes les personnes qui ont contribué, de près ou de loin, par leur aide ou leurs encouragements, à la concrétisation de ce travail.

Résumé

Titre : Surveillance des écosystèmes IoT par l'Intelligence Artificielle : Détection et analyse des comportements malveillants.

Face à l'incapacité des systèmes de défense traditionnels à contrer la complexité et la vélocité croissantes des cyberattaques, cette étude propose une architecture proactive basée sur l'intelligence artificielle pour sécuriser les écosystèmes de l'Internet des Objets (IoT). Le cœur de notre contribution réside dans la mise en œuvre d'une évaluation comparative trilatérale, mettant en confrontation trois philosophies algorithmiques distinctes : une approche classique non supervisée (**Isolation Forest**), une approche classique supervisée (**Random Forest**), et une approche profonde non supervisée (**Autoencodeurs**), cette dernière étant exploitée pour sa capacité à modéliser le comportement sémantique normal via l'écart de l'erreur de reconstruction (MSE).

En évaluant ces modèles sur des flux de trafic réalistes et massifs issus du jeu de données standardisé **CICIoT2023** englobant des scénarios d'attaques contemporaines de type DDoS, DoS et Brute Force, nos résultats expérimentaux valident la pertinence des réseaux de neurones profonds. Bien que le modèle supervisé affiche une exactitude globale légèrement supérieure (94%), l'Autoencodeur proposé parvient à un excellent compromis (90%) tout en s'affranchissant de la contrainte de labellisation des données en temps réel et en optimisant la latence d'inférence (1ms). Cette recherche démontre l'efficacité opérationnelle des approches profondes non supervisées pour la détection des menaces émergentes (Zero-Day) et offre une perspective concrète pour un déploiement embarqué sur des passerelles IoT à ressources limitées (*Edge Computing*).

Mots-clés : Internet des Objets (IoT), Cybersécurité, Autoencodeurs, Random Forest, Isolation Forest, Apprentissage Non Supervisé, Erreur de Reconstruction, Latence d'Inférence, Sécurité Proactive.

Abstract

Title: Monitoring IoT Systems via Artificial Intelligence: Detection and Analysis of Malicious Behaviors.

In the face of traditional defense systems' inability to counter the growing complexity and velocity of cyberattacks, this study proposes a proactive architecture driven by artificial intelligence to secure Internet of Things (IoT) ecosystems. The core of our contribution lies in implementing a rigorous trilateral comparative evaluation, contrasting three distinct algorithmic philosophies: a conventional unsupervised approach (**Isolation Forest**), a traditional supervised approach (**Random Forest**), and a deep unsupervised approach (**Autoencoders**), the latter being exploited for its capacity to model semantic normal behavior through Mean Squared Error (MSE) reconstruction deviance.

By evaluating these models on realistic, large-scale network traffic streams derived from the standardized **CICIoT2023** dataset encompassing contemporary attack scenarios such as DDoS, DoS, and Brute Force our experimental results validate the operational relevance of deep neural networks. Although the supervised model yields a slightly higher global accuracy (94%), the proposed Autoencoder achieves an excellent trade-off (90%) while entirely bypassing the constraint of real-time data labeling and optimizing inference latency (1 ms). This research demonstrates the functional efficacy of deep unsupervised frameworks in detecting emerging (Zero-Day) threats and offers a practical blueprint for embedded deployment on resource-constrained IoT gateways (*Edge Computing*).

Keywords: Internet of Things (IoT), Cybersecurity, Autoencoders, Random Forest, Isolation Forest, Unsupervised Learning, Reconstruction Error, Inference Latency, Proactive Security.

ملخص

العنوان :مراقبة أنظمة إنترنت الأشياء (IoT) بواسطة الذكاء الاصطناعي: كشف وتحليل السلوكيات الضارة.

في ظل عجز أنظمة الدفاع التقليدية عن مواجهة التعقيد المتزايد والسرعة الفائقة للهجمات السيبرانية، تقدم هذه الدراسة بنية أمنية استباقية قائمة على الذكاء الاصطناعي لتأمين أنظمة بيئة إنترنت الأشياء .(IoT) يكمن جوهر مساهمتنا في إجراء تقييم مقارن ثلاثي الأبعاد، يضع في خط المواجهة ثلاث فلسفات خوارزمية مختلفة: نهج كلاسيكي غير خاضع للإشراف (**Isolation Forest**) ، ونهج كلاسيكي خاضع للإشراف(**Random Forest**) ، ونهج عميق غير خاضع للإشراف (**Autoencoders**) ، حيث يتم استغلال هذا الأخير لقدرته العالية على نمذجة السلوك الدلالي الطبيعي عبر حساب الانحراف في خطأ إعادة البناء إحصائياً.(MSE)

ومن خلال تقييم هذه النماذج على تدفقات حركة مرور شبكية واقعية وضخمة مستخرجة من مجموعة البيانات المعيارية **CICIoT2023** والتي تشمل سيناريوهات هجمات معاصرة مثل DDoS و DoS و Brute Force تثبت النتائج التجريبية الكفاءة التشغيلية للشبكات العصبية العميقة. وعلى الرغم من أن النموذج الخاضع للإشراف حقق دقة إجمالية أعلى بقليل(94%) ، إلا أن نموذج (**Autoencoders**) المقترح حقق توازناً وموازنة ممتازة (90%) مع التحرر تماماً من قيد الحاجة إلى تصنيف وتسمية البيانات في الوقت الفعلي، بالإضافة إلى تحسين زمن الاستجابة للاستدلال ليصبح (1 **ملي ثانية فقط**) . (يبرهن هذا البحث على الفعالية العملية لأطر التعلم العميق غير الخاضع للإشراف في كشف التهديدات الناشئة والجديدة تماماً (Zero-Day) ، ويقدم رؤية تطبيقية ملموسة لدمج النظام في بوابات إنترنت الأشياء ذات الموارد المحدودة (*الحوسبة الطرفية.Edge Computing*)

الكلمات المفتاحية :إنترنت الأشياء(IoT) ، الأمن السيبراني، الأوتوإنكودر(Autoencoders) ، الغابات العشوائية (Random Forest)، الغابات المعزولة(Isolation Forest) ، التعلم غير الخاضع للإشراف، خطأ إعادة البناء، زمن استجابة الاستدلال، الأمن الاستباقي.

TABLE DES MATIERES

INTRODUCTION GENERALE **1**
Chapitre 1 ..... **4**
Écosystème de l'IoT : Architecture, Protocoles et Défis Sécuritaires **4**
1.1 Introduction du chapitre 5
1.2 Définition de l'Internet des Objets (IoT) 5
1.3 Évolution historique de l'IoT 6
1.4. Émergence de l'IoT : Statistiques et expansion de la surface d'attaque 6
1.5 Architecture de l'IoT (Modèle en couches) 7
1.6 Technologies de communication et protocoles dans l'IoT 9
1.7 Domaines d'application de l'IoT 11
1.8 Défis et obstacles de l'IoT 13
1.9 Menaces et vulnérabilités dans les environnements IoT 13
1.10 Conclusion du chapitre 15
Chapitre 2 ..... **16**
Sécurité IoT et Systèmes de Détection d'Intrusions : **16**
Des Mécanismes Traditionnels aux Approches Intelligentes **16**
2.1 Introduction du chapitre 17
2.2 Les piliers de la sécurité IoT : L'impératif de la Triade CIA 17
2.3 Classification des attaques dans l'environnement IoT 19
2.4 Menaces modernes et comportements malveillants 20
2.4.1 Caractérisation comportementale des attaques étudiées **20**
2.4.2 Limites des approches traditionnelles et besoin de profilage **21**
2.5 Systèmes de Détection d'Intrusion (IDS) 22
2.5.1 Classification topologique : NIDS versus HIDS **22**
2.5.2 Mécanisme d'ingestion et d'analyse des flux **22**
2.5.3 Approches de détection : Signatures vs Anomalies **23**
2.6 Vers une approche intelligente de la surveillance 24
2.6.1 Le paradigme de la classification supervisée (Random Forest) **25**
2.6.2 Le paradigme de la modélisation comportementale non supervisée (Autoencoders & Isolation Forest) **25**
2.6.3 Justification de l'environnement expérimental : Le corpus CIC IoT 2023 ..... **25**
2.7 Conclusion du chapitre 26
Chapitre 3 : **27**
Approche Méthodologique : **27**

Paradigmes d'Apprentissage et Modélisation Algorithmique **27**
3.1 Introduction du chapitre 28
3.2 Fondamentaux des Paradigmes de l'Intelligence Artificielle en Cybersécurité 28
3.2.1 Le Machine Learning (Apprentissage Automatique) : Définition et Typologies **28**
3.2.1.1 L'apprentissage supervisé : La classification orientée par les étiquettes (Section réécrite) 29
3.2.1.2 L'apprentissage non supervisé : Le profilage autonome et la détection d'anomalies 31
3.2.2 Le Deep Learning (Apprentissage Profond) : Évolution et Principes **31**
3.2.3 Analyse Comparative : Machine Learning vs Deep Learning dans l'écosystème IoT **31**
3.2.4 Justification de l'arbitrage doctrinal : Confrontation Supervisé / Non Supervisé face aux menaces Zero-Day **33**
3.3 Algorithme Random Forest (RF) : Le Paradigme Supervisé 34
3.3.1 Principes des Arbres de Décision et du Bagging **34**
3.3.2 Architecture de l'Ensemble Learning et Vote Majoritaire **35**
3.3.3 Formalisation Mathématique et Critères de Division **36**
3.4 Algorithme Isolation Forest (iForest) 37
3.4.1 Principe de l'Isolation et Partitionnement **38**
3.4.2 Formalisation Mathématique et Logique de Décision **38**
3.4.3 Configuration et Justification du Choix pour l'Écosystème IoT **39**
3.5 Architecture des Autoencoders : Vers une Extraction de Caractéristiques Profondes . 40
3.5.1 Le Processus d'Encodage : Compression et Espace Latent **40**
3.5.2 La Dynamique du "Bottleneck" et l'Apprentissage de la Normalité **41**
3.5.3 Critère de Détection par l'Erreur de Reconstruction **41**
3.6 Synthèse Comparative des Approches Proposées 43
3.7 Conclusion du chapitre 44
Chapitre 4 : **45**
Implémentation Expérimentale, Résultats et Analyse Comparative **45**
4.1 INTRODUCTION 46
4.2 ENVIRONNEMENT DE DEVELOPPEMENT ET REPRODUCTIBILITE 47
4.2.1 Plateforme Kaggle Notebook et Infrastructure Cloud **47**
4.2.2 Langage et Bibliothèques Spécifiques **47**
4.2.3 Reproductibilité et Fixation du SEED **48**
4.3 PRESENTATION ET PREPARATION DU JEU DE DONNEES CICIoT2023 49
4.3.1 Structure, Échantillonnage et Caractéristiques du Dataset **49**
4.3.2 Prétraitement des Données (Label Encoding & StandardScaler) **50**
4.3.3 Choix et Fixation du Seuil Critique de Détection (τ) **51**

4.3.4 Métriques d'Évaluation (Accuracy, Precision, Recall, F1-Score, AUC-ROC, Latence) 51
4.4 PROTOCOLE EXPERIMENTAL ET IMPLEMENTATION DES MODELES 53
4.4.1 Configuration et Entraînement du Random Forest 53
4.4.2 Configuration et Paramétrage d'Isolation Forest 53
4.4.3 Architecture et Dynamique d'Entraînement de l'Autoencodeur 54
4.5 EVALUATION, COMPARAISON ET DISCUSSION DES RESULTATS (CAS NOMINAL) 56
4.5.1 Performances du Modèle Random Forest en Mode Nominal 56
4.5.2 Analyse des Performances d'Isolation Forest 58
4.5.3 Performances du Modèle Autoencodeur (Courbe de perte (Loss) & Métriques)..... 60
4.6 Synthèse et Discussion Comparative (Cas Nominal) 62
4.6.1 Analyse comparative des performances (Isolation Forest vs Autoencodeur)..... 62
4.6.2 Analyse comparative : Random Forest vs Autoencodeur 63
4.6.3 Analyse comparative trilatérale des performances 64
4.7 RÉSILIENCE FACE AUX MENACES INCONNUES (SCÉNARIO ZERO-DAY) 65
4.7.1 Protocole de Simulation du Zero-Day 65
4.7.2 Évaluation du modèle Isolation Forest (Scénario Zero-Day) 66
4.7.3 Effondrement Sécuritaire du Modèle Supervisé (Random Forest) 67
4.7.4 Robustesse et Flexibilité de l'Autoencodeur 70
4.8 RÉSULTATS ET DISCUSSION COMPARATIVE GLOBALE 72
4.8.1 Tableau Comparatif Global des Performances..... 72
4.9 Conclusion du Chapitre 74
CONCLUSION GÉNÉRALE 75
Bibliographies:..... 77

LISTE DES FIGURES

Figure 1. 1: Évolution du nombre d'appareils IoT connectés (2020 - 2026) [2] 7
Figure 1. 2 : Cartographie systémique des technologies de connectivité IoT classées par portée opérationnelle 11

Figure 2. 1 : La triade de la sécurité informatique (CIA). 18
Figure 2. 2 : Classification des cyberattaques selon l'architecture de l'IoT 20
Figure 2. 3 : Architecture fonctionnelle générale d'un système de détection d'intrusion réseau (NIDS) combinant les approches supervisées et comportementales 23

Figure 3. 1 : Pipeline général du Machine Learning : du flux de données brut à la décision prédictive. 29
Figure 3. 2 : Architecture conceptuelle de l'algorithme Random Forest : Processus de classification par agrégation de décisions multiples 34
Figure 3. 3 : Synthèse méthodologique, algorithmique et fonctionnelle de l'Isolation Forest pour la détection d'intrusions IoT. 37
Figure 3. 4 : Architecture de l'autoencodeur 40
Figure 3. 5 : Architecture réseau, dynamique d'apprentissage et logique de classification d'un Autoencoder pour la détection d'anomalies IoT 42

Figure 4. 1 : Matrice de confusion du modèle Random Forest en Mode Nominal. 57
Figure 4. 2 : Matrice de confusion de l'algorithme Isolation Forest le cas normal 59
Figure 4. 3 : Courbes de perte (Loss Curves) : Convergence et stabilité du modèle Autoencodeur. 60
Figure 4. 4: Matrice de confusion finale du modèle Autoencodeur le cas normal. 61
Figure 4. 5 : Synthèse comparative des performances : Isolation Forest vs Autoencodeur Profond. 62
Figure 4. 6 : Comparaison des performances en régime nominal : Random Forest vs Autoencodeur. 63
Figure 4. 7 :Comparaison des performances des trois approches 64
Figure 4. 8 : Comparaison de la performance (Accuracy/Recall) du modèle Isolation Forest entre le régime Nominal et le scénario Zero-Day. 66

Figure 4. 9 : Matrice de confusion du modèle Random Forest en scénario Zero-Day)..... 68
Figure 4. 10 : Évolution des performances du modèle Random Forest : Impact de la menace Zero-Day..... 69
Figure 4. 11: Analyse de la robustesse de l'Autoencodeur : Maintien des performances face à la menace Zero-Day..... 70
Figure 4.12: Synthèse comparative des performances des modèles (Random Forest vs Autoencodeur) : Nominal vs Zero-Day 73

LISTE DES TABLEAUX

Tableau 1. 1 : Évolution du nombre d'appareils IoT connectés (2020 - 2026) [2]..... 6
Tableau 1. 2 : Spécificités fonctionnelles et vulnérabilités de l'architecture IoT à trois couches 8
Tableau 1. 3 : Synthèse des protocoles et technologies de communication IoT 10
Tableau 1. 4 : Domaines d'application stratégiques et cas d'usage sectoriels de l'écosystème IoT 12

Tableau 3. 1 : Analyse comparative des paradigmes d'apprentissage classiques (Supervisé, Semi-supervisé et Renforcement) dans le contexte IoT [12]..... 30
Tableau 3. 2 : Confrontation technologique entre le Machine Learning classique et le Deep Learning face aux contraintes de l'écosystème IoT [15], [16] 32
Tableau 3. 3 : Synthèse comparative des approches étudiées..... 43

Tableau 4. 1 : Matrice de confusion pour la classification des flux IoT 51
Tableau 4. 2 : Évaluation des indicateurs clés (Accuracy/Recall) en régimes Nominal et Zero-Day. 72

LISTE DES EQUATIONS

Équation 3. 1: calcul du gain d'information pour la division des nœuds (Random Forest)..... 36
Équation 3. 2: calcul du score d'anomalie (s) - Isolation Forest 38
Équation 3. 3 : projection dans l'espace latent (Processus d'encodage)..... 40
Équation 3. 4: reconstruction du vecteur de sortie (Processus de décodeur) 41
Équation 3. 5: l'erreur de reconstruction par la méthode MSE (Mean Squared Error) 41

Équation 4. 1: normalisation des attributs (StandardScaler) 50

LISTE DES ABRÉVIATIONS

- **AE :** Autoencoder (Autoencodeur Profond)
- **AI :** Artificial Intelligence *(Intelligence Artificielle)*
- **ANN :** Artificial Neural Network (Réseau de Neurones Artificiels)
- **API :** Application Programming Interface
- **AUC-ROC :** Area Under the Receiver Operating Characteristic (Aire Sous la Courbe ROC)
- **BLE :** Bluetooth Low Energy
- **CIA :** Confidentialité Intégrité Disponibilité
- **CPU :** Central Processing Unit
- **CSV :** Comma-Separated Values (Format de fichier de données)
- **DDoS :** Distributed Denial of Service
- **DL:** Deep Learning *(Apprentissage Profond)*
- **DoS :** Denial of Service
- **FN :** Faux Négatifs (False Negatives)
- **FP :** Faux Positives (False Positives)
- **GPU :** Graphics Processing Unit
- **HIDS :** Host-based Intrusion Detection System (Système de Détection d'Intrusions Hôte)
- **ICMP :** Internet Control Message Protocol
- **IDS :** Intrusion Detection System
- **iForest :** Isolation Forest
- **IIoT :** Industrial Internet of Things (Internet des Objets Industriels)

- **IoT :** Internet of Things *(Internet des Objets)*
- **LPWAN :** Low-Power Wide-Area Network
- **LTE-M :** Long Term Evolution for Machines
- **M2M :** Machine to Machine
- **ML :** Machine Learning *(Apprentissage Automatique)*
- **MQTT :** Message Queuing Telemetry Transport
- **MSE :** Mean Squared Error (Erreur Quadratique Moyenne)
- **NB-IoT :** Narrowband Internet of Things
- **NFC :** Near Field Communication
- **NIDS :** Network Intrusion Detection System (Système de Détection d'Intrusions Réseau)
- **OOB :** Out-Of-Bag (Données de validation interne pour Random Forest)
- **PCAP :** Packet Capture (Format de capture des paquets réseau bruts)
- **PRNG :** Pseudo-Random Number Generator Seed (Graine de Reproductibilité)
- **RAM :** Random Access Memory (Mémoire Vive)
- **RF :** Random Forest (Forêt Aléatoire)
- **RFID :** Radio Frequency Identification
- **TCP :** Transmission Control Protocol
- **TN :** Vrais Négatifs (True Negatives)
- **TP :** Vrais Positifs (True Positives)
- **TPU :** Tensor Processing Unit
- **WLAN :** Wireless Local Area Network

INTRODUCTION GENERALE

L'Internet des Objets **(IoT)** entendu comme l'ensemble des dispositifs physiques capables de collecter, transmettre et traiter des données via des réseaux hétérogènes occupe une place structurante dans l'évolution des infrastructures numériques. Sa progression épouse les contours d'une adoption sectorielle différenciée : impératifs industriels de rendement, logiques de surveillance urbaine, automatisation des soins. Ce que l'on regroupe sous le terme « **d'écosystème IoT** » recouvre en réalité une mosaïque de technologies et de protocoles dont la cohérence est davantage fonctionnelle que technique.

C'est précisément cette hétérogénéité qui complique l'analyse sécuritaire. Les contraintes matérielles des dispositifs mémoire réduite, cycles processeur limités, absence fréquente de chiffrement dédié ne constituent pas de simples limitations fonctionnelles : elles définissent un profil de vulnérabilité dans lequel l'implémentation de mécanismes de sécurité classiques devient soit techniquement impossible, soit économiquement dissuasive. Ce constat, densifié depuis les travaux fondateurs sur les botnets Mirai, n'a fait que s'aggraver à mesure que la surface déployée s'élargissait.

La détection d'intrusions ne peut donc reposer sur une simple transposition des approches éprouvées en sécurité réseau traditionnelle. Irrégularité temporelle des flux, difficulté à établir une baseline comportementale stable, volume des données générées autant de facteurs qui rendent les systèmes à base de signatures rapidement inopérants. C'est dans cet espace problématique que s'inscrit ce travail : une investigation empirique des capacités de l'apprentissage automatique à identifier des comportements malveillants au sein d'un trafic IoT réaliste, en s'appuyant sur le dataset CIC IoT 2023.

"Face à l'obsolescence des systèmes de détection classiques devant l'évolution des cybermenaces dans l'IoT, comment l'apprentissage automatique, et plus précisément les modèles d'analyse comportementale, peuvent-ils garantir une détection proactive et précise des intrusions tout en s'adaptant à la vélocité et à l'hétérogénéité des flux de données modernes

-Structure du mémoire :

Ce travail de recherche s'articule autour de quatre chapitres, suivant une progression logique et cumulative, allant des fondements théoriques de l'IoT jusqu'à la validation expérimentale des approches de détection proposées.

Le **premier chapitre** pose les fondements conceptuels de l'Internet des Objets. Au-delà d'une simple rétrospective, il cartographie la diversité des architectures et des technologies de connectivité, en identifiant les propriétés structurelles qui conditionnent les impératifs de sécurité. Les vulnérabilités intrinsèques y sont traitées comme des contraintes inhérentes à la nature même des dispositifs IoT.

Le **deuxième chapitre** opère un basculement vers l'analyse des menaces cybernétiques. Il dresse un état des lieux de la conflictualité numérique dans les écosystèmes IoT : prolifération des botnets et sophistication des intrusions. L'argument central soutient que la défense périmétrique statique est devenue inadaptée à ces environnements dynamiques, justifiant ainsi le passage vers des paradigmes de surveillance comportementale continue.

Le **troisième chapitre** expose la dimension algorithmique du travail. Il justifie le choix d'une architecture hybride comparant trois modèles distincts : le **Random Forest** pour la classification supervisée, ainsi que l'**Isolation Forest** et les **Autoencoders** pour la détection non supervisée. Cette confrontation permet de mettre en perspective la précision des modèles supervisés face aux attaques connues avec la capacité de généralisation des approches par apprentissage profond face à l'imprévisibilité des menaces.

Le **quatrième chapitre** constitue le cœur expérimental du mémoire. Il documente l'ensemble de la chaîne méthodologique : préparation du dataset **"CIC IoT 2023"**, prétraitement des données, et évaluation comparative rigoureuse des trois modèles à travers des scénarios d'essais normaux et des tests de type *Zero-Day*. Ce chapitre cristallise les résultats obtenus, démontrant la supériorité de l'approche par *Autoencoder* en termes de résilience et de stabilité opérationnelle face aux vecteurs d'attaque inédits.

-Notre Objectif :

Sécuriser un écosystème IoT ne se réduit pas à colmater des brèches connues. La complexité de ce domaine réside dans la détection de menaces imprévisibles dont le comportement dévie subtilement de la normale, sans pour autant correspondre aux signatures répertoriées dans les bases de données classiques. C'est cette réalité opérationnelle qui motive le cadrage de ce travail.

L'objectif poursuivi est double : construire un pipeline de surveillance capable d'ingérer et d'analyser des flux réseau IoT à grande échelle, puis évaluer une approche hybride combinant classification supervisée et modélisation non supervisée pour discriminer, avec précision, le trafic légitime des tentatives d'intrusion. Le choix du dataset **"CIC IoT 2023"** se justifie par sa fidélité aux environnements réels, intégrant des scénarios d'attaques contemporaines dans une complexité de trafic représentative des défis actuels.

Trois modèles sont retenus pour cette évaluation comparative :

- Le **Random Forest**, pour la classification supervisée des comportements identifiés.
- L'**Isolation Forest**, pour son approche statistique par isolement probabiliste des anomalies.
- Les **Autoencoders**, pour l'inférence par écart à une représentation reconstruite du comportement sain.

Ces architectures sont évaluées selon des métriques rigoureuses : l'**Accuracy**, pour mesurer la performance globale, et le **Recall**, critère déterminant pour garantir la détection maximale des comportements malveillants. L'analyse est complétée par le **F1-score**, l'**AUC-ROC** et la **latence d'inférence**, garantissant ainsi la viabilité opérationnelle des modèles sur des passerelles IoT à ressources contraintes.

Chapitre 1

Écosystème de l'IoT : Architecture, Protocoles et Défis Sécuritaires

1.1 Introduction du chapitre

L'Internet des Objets (IoT) ne doit pas être perçu comme une simple rupture technologique, mais plutôt comme une reconfiguration profonde de nos infrastructures numériques. Si la miniaturisation des capteurs et l'essor du Cloud ont concrétisé des concepts autrefois prospectifs, cette omniprésence d'objets capables de percevoir et de réagir sans intervention humaine pose aujourd'hui un problème de fond.

L'enthousiasme entourant ces déploiements tend, bien souvent, à occulter une réalité technique fragmentée. Entre un réseau de capteurs agricoles, un système de télésurveillance médicale ou une chaîne industrielle automatisée, les exigences en termes de latence, de disponibilité et surtout de confidentialité, sont radicalement incomparables. Cette hétérogénéité intrinsèque rend caduque toute tentative de modélisation sécuritaire qui se voudrait universelle ou transversale.

L'objectif de ce chapitre n'est donc pas de s'ajouter à la longue liste des définitions académiques de l'IoT, mais d'en disséquer les architectures et les vulnérabilités structurelles. Pour nous, cette exploration constitue un jalon indispensable : il paraît illusoire de vouloir concevoir des mécanismes de détection d'intrusions performants sans avoir, au préalable, saisi la complexité réelle et les failles inhérentes à ces écosystèmes.

1.2 Définition de l'Internet des Objets (IoT)

Définir l'IoT relève d'un exercice moins trivial qu'il n'y paraît au premier abord. Si des organismes de normalisation comme l'ITU-T, l'IEEE ou l'ETSI tentent d'en tracer les contours, leurs cadres respectifs se recoupent de manière imparfaite. Ce flou terminologique ne nous semble pas anodin : il traduit surtout l'extrême hétérogénéité d'un paradigme qui tente de regrouper, sous une étiquette unique, des réalités techniques radicalement disparates.

En substance, l'IoT marque l'extension de la connectivité à des objets physiques qui, jusqu'ici, restaient "silencieux" ou passifs (capteurs industriels, automates, dispositifs de santé). Ce qui change réellement la donne par rapport à l'Internet traditionnel, ce n'est pas tant l'infrastructure réseau en elle-même, mais plutôt le changement de paradigme des acteurs : nous ne sommes plus face à des utilisateurs manipulant des terminaux, mais face à des entités agissant de manière autonome selon des logiques événementielles.

Toutefois, cette autonomie est loin d'être totale. Elle reste tributaire de la robustesse des couches logicielles et surtout, des ressources computationnelles souvent dérisoires embarquées dans ces objets. C'est précisément dans cette zone grise là où la promesse d'une intégration fluide se heurte aux dures contraintes de déploiement que s'installent les failles les plus critiques. Un objet mal sécurisé cesse d'être un simple outil autonome pour devenir, de fait, une porte d'entrée vulnérable pour tout l'écosystème.

1.3 Évolution historique de l'IoT

Le concept d'objets connectés remonte au début des années 1980 avec l'expérimentation d'un distributeur de Coca-Cola connecté à l'Université Carnegie Mellon. Cependant, le terme "Internet of Things" a été officiellement introduit en 1999 par **Kevin Ashton** lors d'une présentation chez Procter & Gamble pour décrire l'utilisation de la technologie RFID dans la gestion de la chaîne d'approvisionnement **[1]**. Depuis lors, l'IoT a évolué d'une simple vision technologique vers une réalité industrielle massive grâce à la miniaturisation des capteurs et l'avènement du protocole IPv6.

1.4. Émergence de l'IoT : Statistiques et expansion de la surface d'attaque

L'adoption massive des objets connectés n'est plus une simple tendance, mais une révolution infrastructurelle. Le tableau ci-dessous illustre l'évolution exponentielle du nombre de dispositifs IoT au niveau mondial, soulignant l'urgence de développer des systèmes de détection d'intrusion plus performants :

Année	Nombre d'appareils (en milliards)	Taux de croissance annuel
2020	**11.3**	**-**
2022	**14.4**	**~13%**
2024	**19.1**	**~16%**
2026 (Projeté)	**27.1**	**~21%**

Tableau 1. 1 : Évolution du nombre d'appareils IoT connectés (2020 - 2026) [2]

Afin d'offrir une visualisation graphique et dynamique de cette croissance exponentielle détaillée dans le tableau précédent, **la Figure 1.1** met en évidence l'allure ascendante de l'évolution des dispositifs connectés à l'échelle mondiale :

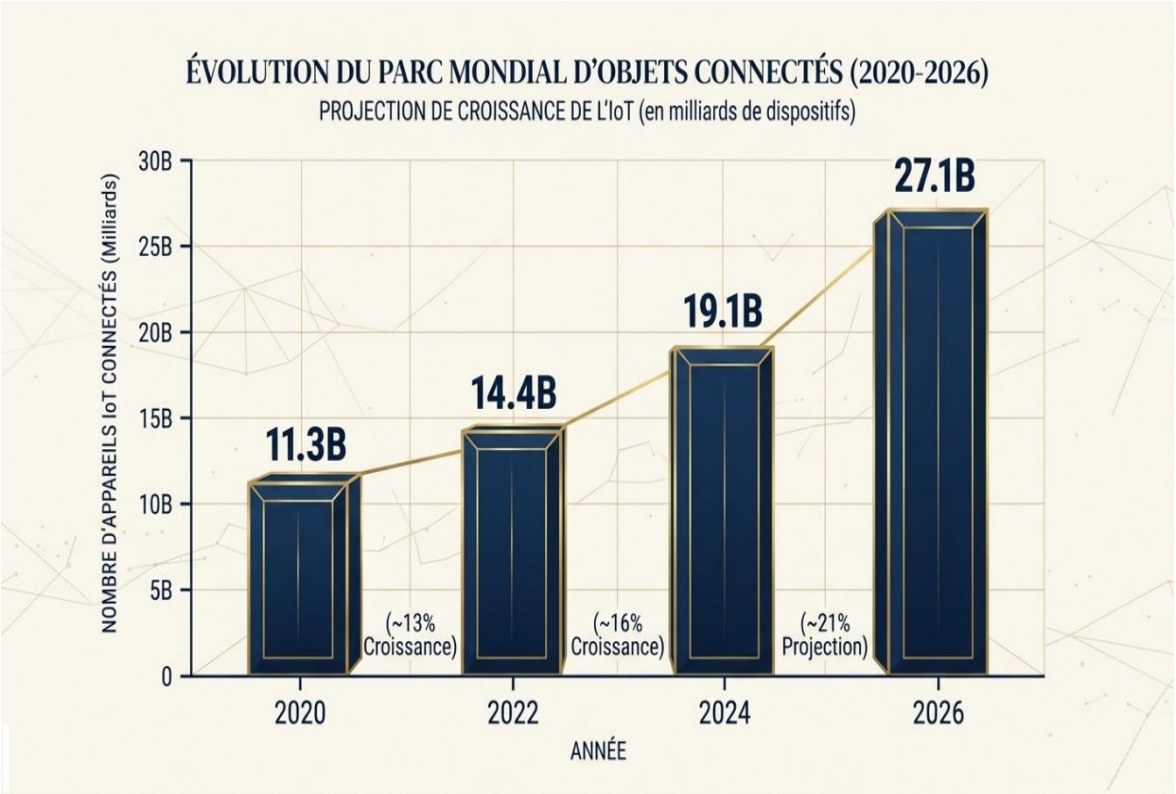


Figure 1. 1: Évolution du nombre d'appareils IoT connectés (2020 - 2026) [2]

1.5 Architecture de l'IoT (Modèle en couches)

Loin de s'appuyer sur un standard architectural unique, l'écosystème IoT se caractérise par une coexistence de modèles variant de trois à sept couches dont la granularité dépend de l'angle d'analyse adopté. Le modèle à trois couches demeure toutefois la référence prédominante, s'imposant moins par une rigueur scientifique absolue que par une réelle commodité opérationnelle. Il permet de structurer le transit de selon une trajectoire verticale cohérente : de la captation brute jusqu'à l'exploitation l'information métier.

En vue de clarifier la répartition des rôles et les enjeux de sécurité propres à chaque niveau de cette architecture tripartite, le **Tableau 1.2** synthétise les spécificités fonctionnelles et les vulnérabilités inhérentes à chaque couche :

COUCHES	Description
La couche de perception	Au premier niveau, la **couche de perception** agit comme le point de contact avec le réel. Elle mobilise un ensemble hétérogène de capteurs et d'actionneurs (RFID, LoRa, capteurs environnementaux) dont le rôle est de convertir des grandeurs physiques en signaux numériques. Pour nous, c'est ici que se cristallisent les défis sécuritaires les plus complexes : déployés dans des zones souvent hostiles ou non sécurisées, ces terminaux disposent de ressources trop limitées pour supporter des mises à jour régulières, créant ainsi des vulnérabilités structurelles durables.
La couche réseau	De son côté, assure le transport de ce flux massif vers les centres de traitement. Le choix des protocoles (Zigbee, LoRaWAN, NB-IoT ou Wi-Fi) ne se fait pas au hasard, mais résulte d'un arbitrage constant entre portée, débit et sobriété énergétique. Cette multiplicité protocolaire, bien que nécessaire, élargit mécaniquement la surface d'attaque, rendant l'homogénéisation des politiques de sécurité particulièrement ardue.
La couche applicative	Enfin, la **couche applicative** donne un sens opérationnel à cette donnée brute. Qu'il s'agisse de télémédecine ou de surveillance industrielle, c'est à ce stade que se matérialise la valeur ajoutée du système. Néanmoins, cette couche n'est pas épargnée par les menaces : l'exposition des API et les faiblesses d'authentification y constituent des vecteurs d'intrusion critiques, que nous devons impérativement prendre en compte dans notre stratégie de détection

Tableau 1. 2 : Spécificités fonctionnelles et vulnérabilités de l'architecture IoT à trois couches

1.6 Technologies de communication et protocoles dans l'IoT

Au cœur de l'écosystème IoT, la connectivité ne s'envisage pas comme une solution uniforme, mais comme un arbitrage permanent entre des contraintes souvent antagonistes : portée du signal, sobriété énergétique et débit binaire. Le choix d'un protocole n'est jamais neutre ; il définit la viabilité opérationnelle du dispositif sur le long terme.

Dans les environnements où les ressources sont drastiquement limitées, l'efficience devient la priorité absolue. C'est ici que des protocoles légers comme **MQTT** *(Message Queuing Telemetry Transport)* ou **CoAP** *(Constrained Application Protocol)* ont su s'imposer. Plus qu'un simple standard, MQTT est privilégié pour sa capacité à maintenir une communication machine-to-machine (M2M) robuste tout en minimisant l'empreinte sur la bande passante, un facteur critique pour les nœuds de capteurs. **Cependant, l'adoption massive de ces protocoles d'application engendre une conséquence sécuritaire majeure : la génération de flux de données caractérisés par une vélocité extrême (*High Velocity*). En encapsulant des messages courts mais ultra-fréquents, ces architectures saturent les canaux réseau par une transmission continue et asynchrone de paquets bruts, posant ainsi le défi de l'analyse en temps réel.**

Par ailleurs, l'essor des technologies de type LPWAN, telles que LoRaWAN ou Sigfox, a ouvert de nouvelles perspectives pour les déploiements à grande échelle. Ces protocoles permettent d'assurer des liaisons à longue portée avec un budget énergétique extrêmement serré. Cette approche est particulièrement pertinente dans des secteurs comme l'agriculture connectée ou la Smart City, où les capteurs, souvent isolés, doivent fonctionner de manière autonome pendant plusieurs années sans intervention humaine. **Néanmoins, qu'il s'agisse de réseaux locaux ou de topologies étendues, la convergence de milliers d'objets connectés amplifie mécaniquement la densité du trafic global, créant une hétérogénéité temporelle difficile à monitorer par des méthodes traditionnelles.**

De manière à dresser une cartographie claire de ces technologies et de mettre en relief les arbitrages techniques qui régissent leur déploiement, **tout en anticipant l'exigence de rapidité de traitement face à cette vélocité des flux**, le **Tableau 1.3**

synthétise les caractéristiques opérationnelles ainsi que les contraintes d'application des principaux protocoles de communication IoT :

Dimensions & Protocoles	Caractéristiques et Fonctions Principales	Contraintes et Domaines d'Application
Arbitrage de Connectivité	• Compromis permanent entre trois facteurs antagonistes : la **portée du signal**, la **sobriété énergétique**	• Conditionne directement la viabilité opérationnelle à long terme des dispositifs déployés
Protocole MQTT *(Message Queuing Telemetry Transport)*	• Standard de communication *Machine-to-Machine* (M2M). • Optimisation stricte et minimisation de l'empreinte sur la **bande passante**.	• Idéal pour les environnements aux **ressources drastiquement limitées**. • Privilégié pour la gestion des flux des nœuds de capteurs.
Technologies LPWAN *(ex. LoRaWAN, Sigfox)*	• Liaisons à **longue portée**. • Fonctionnement sous un budget énergétique extrêmement serré.	• Déploiements à grande échelle. • **Smart City** et **Agriculture connectée**. • Autonomie des capteurs isolés sur plusieurs années sans intervention humaine.

Tableau 1. 3 : Synthèse des protocoles et technologies de communication IoT]

Pour compléter cette analyse tabulaire et offrir une représentation spatiale de la distribution de ces protocoles, **la Figure 1.2** cartographie visuellement ces technologies de connectivité en les classant selon leur portée opérationnelle (faible, moyenne et longue portée) au sein de l'écosystème IoT :

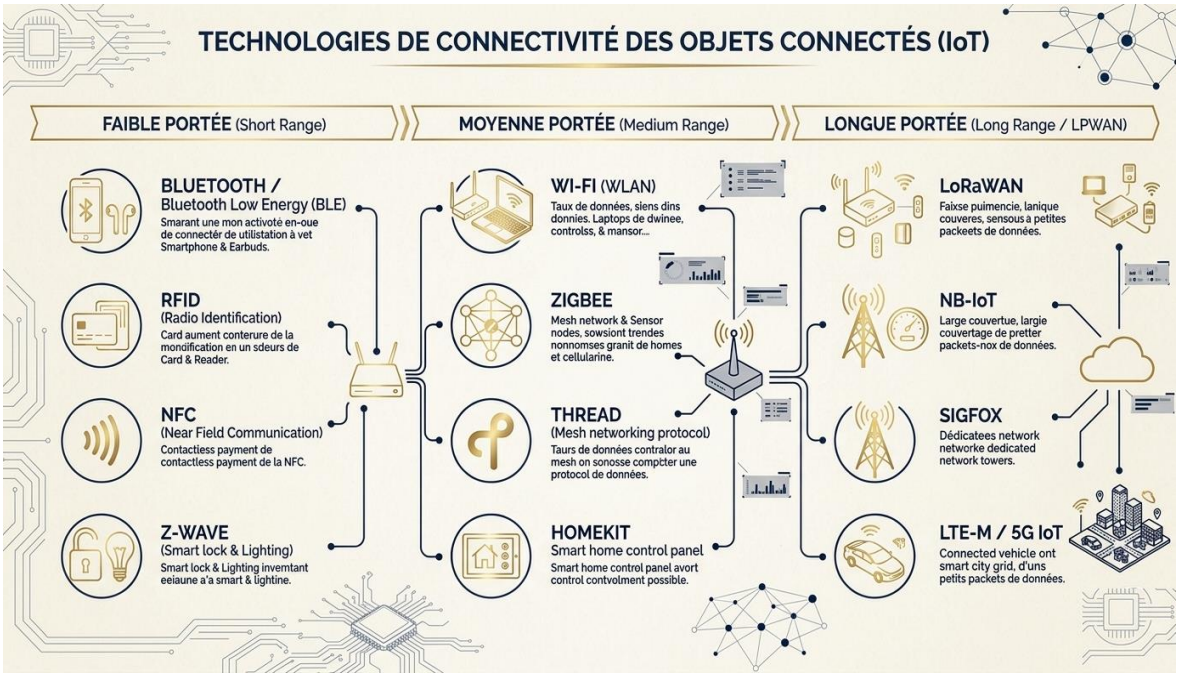


Figure 1. 2 : Cartographie systémique des technologies de connectivité IoT classées par portée opérationnelle

1.7 Domaines d'application de l'IoT

Loin de se limiter à une simple tendance technologique, l'influence de l'IoT s'est désormais sédimentée dans la quasi-totalité des infrastructures modernes. En convertissant des objets autrefois "silencieux" en entités capables de dialogue, ce paradigme redéfinit les frontières de l'efficacité opérationnelle, mais multiplie corollairement les surfaces d'attaque.

Afin de structurer cette analyse et d'illustrer concrètement le déploiement opérationnel de ces technologies à travers les différents secteurs, le **Tableau 1.4** synthétise les domaines d'application majeurs de l'IoT ainsi que leurs cas d'usage spécifiques :

Domaines	Applications
Le virage de l'E-Santé	Au-delà du simple monitoring de confort, l'usage des dispositifs portables (wearables) a instauré une dynamique de soin proactive. Pour nous, l'intérêt majeur ne réside pas seulement dans la collecte de données, mais dans la capacité à traiter les constantes vitales en temps réel. Cette mutation permet d'anticiper des épisodes pathologiques graves, transformant la médecine curative en une approche préventive réelle.
La gestion des Smart Cities	Dans l'espace urbain, l'IoT s'impose comme un levier d'optimisation indispensable face aux défis de la durabilité. Qu'il s'agisse de réguler les flux de trafic de manière dynamique ou de rationaliser l'éclairage public adaptatif, l'enjeu est de transformer une donnée brute et massive en un outil d'aide à la décision pour une ville plus résiliente.
L'Industrie 4.0 et la maintenance prédictive	Au sein des infrastructures de production, l'ubiquité des capteurs marque la fin du modèle de maintenance réactive. Ce n'est plus une question de réparation après panne, mais d'anticipation systémique. La réduction drastique des temps d'arrêt et l'optimisation des flux de production deviennent alors les piliers centraux de la compétitivité industrielle actuelle.
Agriculture	L'IoT est utilisé dans l'agriculture pour améliorer la productivité et la qualité des cultures. Les capteurs connectés sont utilisés pour surveiller les conditions météorologiques, la qualité du sol et les niveaux d'humidité, tandis que les drones connectés sont utilisés pour pulvériser les cultures et surveiller les récoltes **[3].**

Tableau 1. 4 : Domaines d'application stratégiques et cas d'usage sectoriels de l'écosystème IoT

"Toutefois, cette adoption massive dans des secteurs aussi vitaux soulève des problématiques de sécurité sans précédent. C'est précisément pour répondre à ces défis que nous avons choisi de focaliser notre étude sur le dataset CIC IoT 2023, lequel reflète la complexité et la diversité des vecteurs d'attaques actuels auxquels ces écosystèmes sont confrontés."

1.8 Défis et obstacles de l'IoT

Si le potentiel de l'Internet des Objets semble illimité, son déploiement à grande échelle se heurte encore à des barrières critiques. Ces obstacles ne sont pas seulement logistiques, ils touchent à l'essence même de l'architecture des systèmes distribués :

- **L'équation complexe de l'autonomie énergétique :** C'est sans doute le frein le plus immédiat. La plupart des nœuds IoT, notamment en milieu industriel ou agricole, dépendent de batteries aux capacités limitées. Le défi réside dans l'arbitrage constant entre la fréquence des transmissions énergivore et la nécessité de maintenir une surveillance en temps réel. Pour nous, la survie opérationnelle de ces dispositifs sans intervention humaine sur plusieurs années reste un verrou majeur.
- **La problématique de la scalabilité (Montée en charge) :** Passer de quelques milliers à des milliards de connexions simultanées n'est pas une simple affaire de puissance de calcul. L'enjeu ici est la capacité des infrastructures réseau à absorber des flux de données massifs et hétérogènes sans saturation. Cette explosion du volume de données pose des questions de latence et de gestion de la bande passante que les architectures classiques peinent à résoudre.
- **Le morcellement et le manque d'interopérabilité :** L'écosystème IoT actuel souffre d'une fragmentation excessive. En l'absence de standards universels, nous observons une prolifération de solutions propriétaires créant des "silos" technologiques. Cette incapacité des dispositifs de constructeurs différents à dialoguer nativement freine l'émergence d'écosystèmes réellement intégrés et fluides.

1.9 Menaces et vulnérabilités dans les environnements IoT

Cette section marque le point de bascule de notre analyse vers les enjeux sécuritaires, cœur battant de notre étude. La nature intrinsèquement ouverte de l'IoT, couplée à une connectivité permanente, transforme chaque objet en une cible de choix. Il ne s'agit pas seulement de failles logicielles classiques, mais d'un écosystème de vulnérabilités interdépendantes. Le nœud du problème réside dans l'asymétrie entre la criticité des données traitées et l'indigence des ressources matérielles (mémoire et cycles CPU) de ces dispositifs, rendant l'usage de solutions de défense conventionnelles (antivirus ou pare-feu applicatifs) techniquement caduc [3].

À cela s'ajoute une gestion souvent défaillante de l'hygiène numérique, caractérisée par l'usage persistant de mots de passe par défaut et l'absence chronique de cycles de mise à jour micrologicielle *(firmware)*. Ces faiblesses structurelles constituent les vecteurs directs d'attaques massives, allant de l'enrôlement dans des réseaux de botnets destructeurs, à l'instar de Mirai, à l'exfiltration silencieuse de données sensibles [4]. C'est précisément cette fragilité qui légitime notre démarche : devant l'incapacité des objets à s'auto-protéger, il devient impératif d'externaliser la sécurité vers des systèmes de surveillance intelligents. L'enjeu est de passer d'une réaction passive à une détection comportementale proactive, capable de déceler les signaux faibles de cybermenaces au sein d'un trafic de plus en plus complexe [5].

Afin de contextualiser scientifiquement notre cadre expérimental, cette dynamique de menace se manifeste principalement à travers deux grandes typologies d'attaques que nous avons intégrées au cœur de nos simulations :

- **Les attaques par saturation de service (DDoS / DoS Flood) :** *Qu'elles exploitent le protocole ICMP (DDoS-ICMP Flood) ou des anomalies de poignée de main TCP (DoS-TCP Flood), ces cybermenaces visent à paralyser les infrastructures en saturant les ressources physiques des passerelles IoT sous un volume critique de requêtes légitimes détournées.*
- **Les attaques par force brute (Brute Force) :** *Ces offensives exploitent la persistance des configurations d'usine et la faiblesse des politiques d'authentification sur les terminaux. À l'aide de dictionnaires automatisés (Dictionary Brute Force), les attaquants tentent de forcer les accès légitimes pour prendre le contrôle total des dispositifs connectés.*

Néanmoins, la mise en œuvre de l'Intelligence Artificielle pour contrer ces menaces soulève un défi doctrinal majeur : le dilemme de la labellisation des flux réseau. L'apprentissage supervisé, matérialisé dans notre étude par l'algorithme Random Forest, offre une précision remarquable mais exige une cartographie préalable, stricte et continue des paquets (données étiquetées saines et malveillantes). Face à la vélocité extrême, au volume massif et à l'hétérogénéité des flux IoT, cette taxonomie en temps réel s'avère techniquement irréalisable et économiquement dissuasive. C'est ce verrou opérationnel qui justifie la nécessité d'explorer l'apprentissage non supervisé, notamment à travers l'Autoencodeur profond. En se focalisant uniquement sur l'extraction sémantique du trafic normal, ce paradigme s'affranchit du besoin de labels

préexistants et se révèle capable d'intercepter les attaques par simple déviation de reconstruction (MSE), préparant ainsi le terrain pour une analyse comparative trilatérale objective.

1.10 Conclusion du chapitre

Au terme de ce premier chapitre, l'Internet des Objets ne nous apparaît plus comme une simple promesse technologique, mais comme une infrastructure complexe déjà sédimentée dans notre quotidien. En décortiquant ses couches architecturales et la diversité de ses protocoles, nous avons mis en lumière un écosystème d'une hétérogénéité sans précédent, où la connectivité devient la norme pour des objets autrefois isolés.

Cependant, cette expansion effrénée a un prix. Le véritable constat de notre analyse réside dans un paradoxe technique : la force de l'IoT son ubiquité est aussi sa plus grande faiblesse. Nous avons souligné que les contraintes drastiques en ressources (mémoire, CPU) et le morcellement des standards ne sont pas de simples obstacles techniques, mais des brèches structurelles que les méthodes de défense classiques ne parviennent plus à colmater. De plus, la vélocité extrême et la nature massive de ces flux de paquets réseau soulèvent un verrou méthodologique majeur : l'impossibilité de cartographier et d'étiqueter ces données en temps réel. Cette contrainte opérationnelle remet profondément en question la viabilité des approches purement supervisées, incapables de faire face à la dynamique des menaces changeantes sans une labellisation préalable coûteuse.

Ce diagnostic nous impose un changement de paradigme. Il est désormais illusoire de vouloir sécuriser l'IoT avec les outils d'hier. C'est cette nécessité d'innovation et d'arbitrage technologique entre les modèles guidés par des labels et ceux apprenant de manière autonome qui guidera la suite de notre travail : le chapitre suivant sera dédié à une immersion dans la cybersécurité de ces environnements, où nous analyserons les vecteurs d'attaques réels pour introduire les fondements d'une surveillance proactive, intelligente et comparative.

Chapitre 2

Sécurité IoT et Systèmes de Détection d'Intrusions :

Des Mécanismes Traditionnels aux Approches Intelligentes

2.1 Introduction du chapitre

La transformation numérique engendrée par l'IoT a radicalement modifié la surface d'attaque des systèmes d'information. Contrairement aux réseaux informatiques classiques, la sécurité dans l'IoT est confrontée à des défis uniques liés à la diversité des protocoles et aux ressources matérielles limitées des objets connectés. Ce chapitre examine les fondements de la cybersécurité appliqués à l'IoT, détaille les types d'attaques qui menacent ces écosystèmes et analyse le rôle crucial des systèmes de détection d'intrusion (IDS) dans la mise en place d'une défense proactive.

On ne peut plus ignorer que l'explosion de l'IoT a pulvérisé les périmètres de sécurité traditionnels. Ce n'est pas une simple évolution, c'est une rupture. Là où les réseaux standards se reposent sur des protocoles matures, l'univers des objets connectés nous projette dans une zone grise faite d'hétérogénéité logicielle et, surtout, d'une indigence matérielle criante. Cette asymétrie entre l'importance des données captées et la faiblesse des ressources de calcul embarquées transforme chaque nœud du réseau en un maillon faible idéal pour l'attaquant.

L'enjeu de ce deuxième chapitre est de sortir des généralités pour disséquer cette nouvelle réalité. Nous n'allons pas seulement dresser une liste de menaces ; nous allons analyser comment les carences structurelles de ces réseaux conditionnent l'apparition d'attaques de plus en plus furtives et dévastatrices. Ce diagnostic nous amènera logiquement à réévaluer le rôle des systèmes de détection d'intrusion (IDS). L'idée ici est de démontrer que l'IDS n'est plus un simple accessoire, mais la seule réponse viable face à l'incapacité des objets à s'auto-protéger. **Enfin, nous élargirons notre réflexion vers les approches algorithmiques modernes, en posant les jalons théoriques qui distinguent la classification supervisée de la modélisation comportementale non supervisée, ouvrant ainsi la voie à une analyse comparative rigoureuse.**

2.2 Les piliers de la sécurité IoT : L'impératif de la Triade CIA

La fiabilité d'un écosystème IoT ne peut être envisagée sans une adhésion stricte aux trois principes directeurs de la cybersécurité, regroupés sous l'acronyme de la Triade CIA *(Confidentialité, Intégrité, Disponibilité)*. Bien que ces piliers soient classiques, ils acquièrent une dimension critique et de nouveaux défis d'implémentation dans l'IoT en raison de la nature hautement distribuée, hétérogène et souvent vulnérable des nœuds du réseau [6]. En effet, la compromission d'un seul capteur contraint en ressources peut briser

cette chaîne de confiance, impactant directement la vie privée ou la continuité des services industriels [7], comme le schématise la **Figure 2.1**.

La Confidentialité : le verrouillage des flux Dans un écosystème IoT où les données sont hautement stratégiques, la confidentialité est impérative. Elle garantit un accès restreint aux entités légitimes. Pour l'IoT, cela impose le déploiement de protocoles de chiffrement robustes, malgré les contraintes de calcul des objets, afin de contrer toute interception passive.

L'Intégrité : la garantie de fiabilité L'intégrité assure que la donnée, de sa captation à son stockage, demeure inaltérée. Dans des domaines critiques comme le contrôle industriel ou médical, elle garantit que chaque commande transmise parvient à destination sans corruption, évitant ainsi des conséquences potentiellement catastrophiques.

La Disponibilité : la résilience face à l'obstruction La disponibilité définit la capacité du système à demeurer opérationnel en toute circonstance. L'IoT est particulièrement vulnérable aux ruptures de service, qu'elles soient dues à des contraintes énergétiques ou à des attaques par saturation (DDoS). Notre étude intègre spécifiquement des scénarios de vagues massives (DoS-TCP et DDoS-ICMP) pour évaluer la capacité de nos modèles (Random Forest et L'Isolation Forest et Autoencodeurs) à intercepter ces tentatives d'asphyxie et à maintenir la continuité de service.

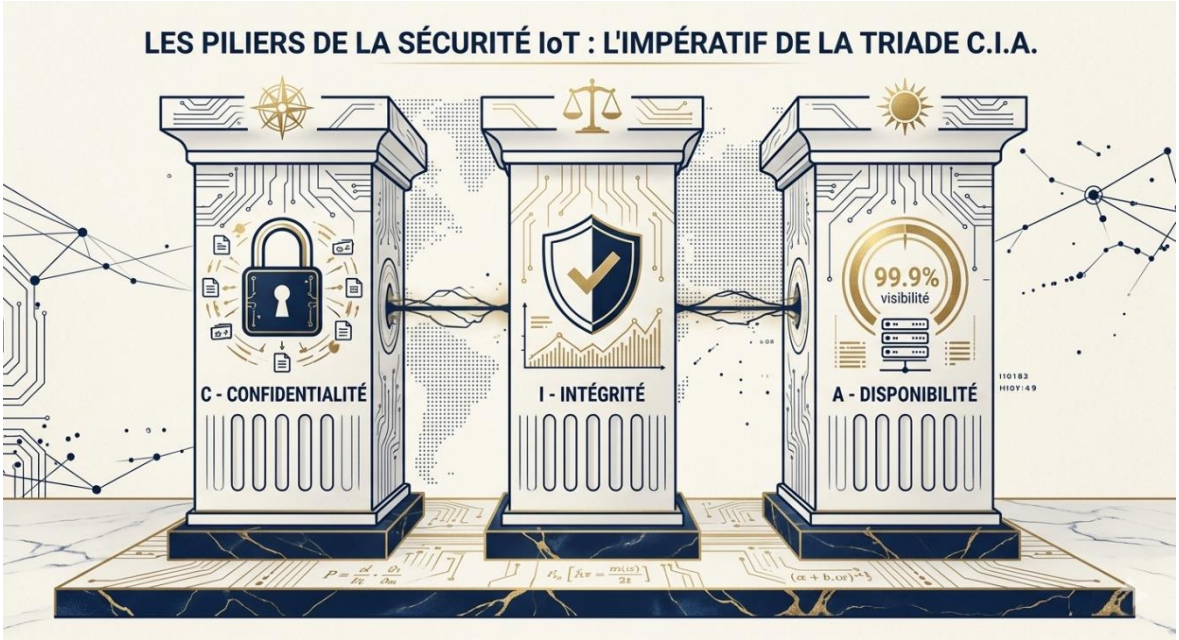


Figure 2. 1 : La triade de la sécurité informatique (CIA).

2.3 Classification des attaques dans l'environnement IoT

L'extrême hétérogénéité des terminaux et la porosité des protocoles de communication transforment l'écosystème IoT en une cible de choix. Pour structurer notre analyse des risques, il est indispensable de segmenter les menaces selon l'architecture qu'elles visent. Comme nous le précisons dans la **Figure 2.2**, cette classification permet d'isoler les vulnérabilités intrinsèques à chaque strate opérationnelle :

Compromission de la couche de perception (Capteurs et Actuateurs) : Située au front d'interaction avec le monde physique, cette couche est vulnérable à des agressions directes. Au-delà du simple sabotage matériel ou du vol de nœuds, nous observons des menaces plus subtiles telles que l'injection de données erronées *(Data Injection)*. L'objectif ici est de corrompre la source même de l'information, faussant ainsi toute la chaîne de décision en amont.

Offensives sur la couche réseau (Connectivité et Routage) : Pivot de la communication, cette strate subit les assauts les plus massifs. C'est ici que se déploient les attaques par déni de service **(DoS/DDoS), en l'occurrence sous les formes d'inondation brute DoS-TCP Flood et DDoS-ICMP Flood**, visant à paralyser l'infrastructure par saturation de la bande passante – un point critique que nous traiterons lors de nos expérimentations. Parallèlement, l'interception de flux *(Eavesdropping)* et les attaques de l'homme du milieu *(Man-in-the-Middle)* exploitent la légèreté de certains protocoles de routage pour compromettre la confidentialité des échanges.

Exploitation de la couche applicative (Interfaces et Services) : À ce niveau, l'attaquant cible les failles logicielles, les protocoles d'administration et les vulnérabilités des API. Les menaces y sont souvent chirurgicales, **à l'instar des attaques par dictionnaire automatisé (Dictionary Brute Force) visant à forcer les mécanismes d'authentification**. Le risque est ici maximal, car une brèche applicative peut offrir un levier de manipulation totale sur l'ensemble de l'écosystème piloté, **ce qui justifie l'intérêt d'entraîner des modèles supervisés comme Random Forest pour cartographier ces signatures ou des Autoencoders pour profiler ces déviances.**

Afin de matérialiser cette répartition systémique et de synthétiser visuellement les vecteurs d'agression propres à l'architecture de cet écosystème, la **Figure 2.2** schématise la classification de ces cyberattaques en projetant chaque typologie de menace face à sa strate architecturale cible.



Figure 2. 2 : Classification des cyberattaques selon l'architecture de l'IoT

2.4 Menaces modernes et comportements malveillants

Le paysage du cybercrime ne cesse de se réinventer, donnant naissance à des vecteurs d'attaques dont la complexité dépasse les mécanismes de défense statiques. Parmi ces comportements malveillants, l'émergence des Botnets IoT constitue sans doute l'une des mutations les plus critiques. En orchestrant la transformation de milliers d'objets intrinsèquement vulnérables en "unités zombies", les attaquants parviennent à générer des flux de saturation massifs (DDoS), capables de paralyser des infrastructures entières [8]. Parallèlement, nous assistons à une recrudescence des Ransomwares et des injections de malwares ciblant spécifiquement les écosystèmes industriels connectés (IIoT) [9].

2.4.1 Caractérisation comportementale des attaques étudiées

Afin d'ancrer notre étude théorique dans la réalité des flux réseau que nous analysons au niveau microscopique (trame par trame), il convient de spécifier la signature comportementale des trois menaces majeures intégrées dans nos simulations :

Inondation DDoS-ICMP (Distributed Denial of Service ICMP Flood) : Ce comportement malveillant se caractérise par l'émission massive et distribuée de requêtes *Echo Request* (Ping) vers une cible unique. L'objectif est de saturer la bande passante et la pile réseau de la passerelle IoT, provoquant un déni de service par épuisement des capacités de traitement.

Inondation DoS-TCP (Denial of Service TCP Syn Flood) : Contrairement à l'approche volumétrique de l'ICMP, cette attaque exploite les faiblesses du protocole de transport TCP (*Three-Way Handshake*). L'attaquant submerge le dispositif de paquets *SYN* falsifiés sans jamais finaliser la connexion. Les ressources du système cible restent ainsi bloquées en attente (connexions semi-ouvertes), entraînant une asphyxie matérielle rapide.

Force Brute par Dictionnaire (Dictionary Brute Force) : Ce comportement applicatif malveillant consiste en une série d'essais automatisés et ultra-rapides de combinaisons d'identifiants et de mots de passe sur les ports d'administration (tels que SSH ou Telnet). La détection comportementale de cette menace repose sur l'analyse de la fréquence anormale des tentatives de connexion infructueuses sur une fenêtre temporelle restreinte.

2.4.2 Limites des approches traditionnelles et besoin de profilage

Face à cette furtivité croissante, les méthodes de détection classiques basées sur les signatures (fichiers ou empreintes connus) montrent structurellement leurs limites face aux variantes polymorphes et aux attaques Zero-Day. Il devient alors impératif de privilégier une approche comportementale basée sur la détection d'anomalies [10].

Cette stratégie, qui est au cœur de notre étude expérimentale, permet d'identifier un signal malveillant non pas par sa forme ou son identité, mais par l'écart décisionnel et statistique qu'il génère vis-à-vis d'un comportement normal de référence. C’est cette rupture méthodologique qui justifie la confrontation trilatérale entre la classification supervisée explicite et la reconstruction sémantique profonde du trafic sain.

2.5 Systèmes de Détection d'Intrusion (IDS)

Les périmètres de sécurité physiques, pare-feu en tête, ne sont plus qu'une première barrière, nécessaire mais largement insuffisante face à la furtivité des menaces actuelles. Pour sanctuariser un réseau IoT, l'intégration d'un Système de Détection d'Intrusion (IDS) devient le pivot d'une cyberdéfense résiliente. Son rôle dépasse la simple surveillance ; il s'agit d'une analyse chirurgicale des flux, capable d'identifier des déviations comportementales que les outils statiques ignorent. C'est cette capacité à transformer un flux de données brut en une intelligence actionnable qui place l'IDS au centre de toute stratégie de sécurité moderne.

2.5.1 Classification topologique : NIDS versus HIDS

Pour concevoir une architecture de détection adaptée aux contraintes spécifiques de l'Internet des Objets, il est indispensable d'arbitrer entre deux grandes catégories d'IDS selon leur point de déploiement :

- **Les IDS basés sur l'hôte (HIDS - Host-based IDS) :** Ces systèmes sont directement installés sur les terminaux pour surveiller les activités internes (journaux système, appels système, modifications de fichiers). Dans un contexte IoT, le déploiement de HIDS est massivement rejeté en raison de l'indigence matérielle des objets (contraintes strictes de CPU, de mémoire et de batterie) qui ne peuvent supporter la surcharge computationnelle d'un agent de sécurité local.
- **Les IDS basés sur le réseau (NIDS - Network-based IDS) :** Ces dispositifs se positionnent de manière non intrusive au niveau des points stratégiques du réseau (comme les passerelles ou *gateways*). Ils interceptent, capturent et analysent l'intégralité du trafic réseau circulant entre les terminaux.

C'est cette approche NIDS, axée sur l'analyse granulaire des paquets, que nous adoptons théoriquement et pratiquement dans nos travaux, car elle permet de centraliser la surveillance sans impacter les ressources physiques des objets connectés.

2.5.2 Mécanisme d'ingestion et d'analyse des flux

Comme le schématise la **Figure 2.3**, l'efficacité opérationnelle d'un NIDS repose sur une architecture modulaire séquentielle. Le cycle commence par la capture des paquets bruts (*Packet Capture*), suivie d'une phase cruciale de prétraitement (*Preprocessing*) visant à extraire les caractéristiques statistiques et sémantiques des flux.

Le cœur décisionnel du système (*Detection Engine*) s'appuie ensuite sur une double vérification : d'une part, une analyse comparative par rapport à une base de signatures étiquetées (*Signature Database*), ce qui correspond à la philosophie de notre modèle supervisé **Random Forest**. D'autre part, une vérification comportementale corrélée à un profil de référence sain (*Normal Profile*), matérialisée dans nos travaux par le seuil de reconstruction (MSE) de notre **Autoencodeur profond**. Cette synergie permet une levée d'alerte immédiate (*Alert / Log*) en cas de détection d'une menace connue ou d'une anomalie Zero-Day.

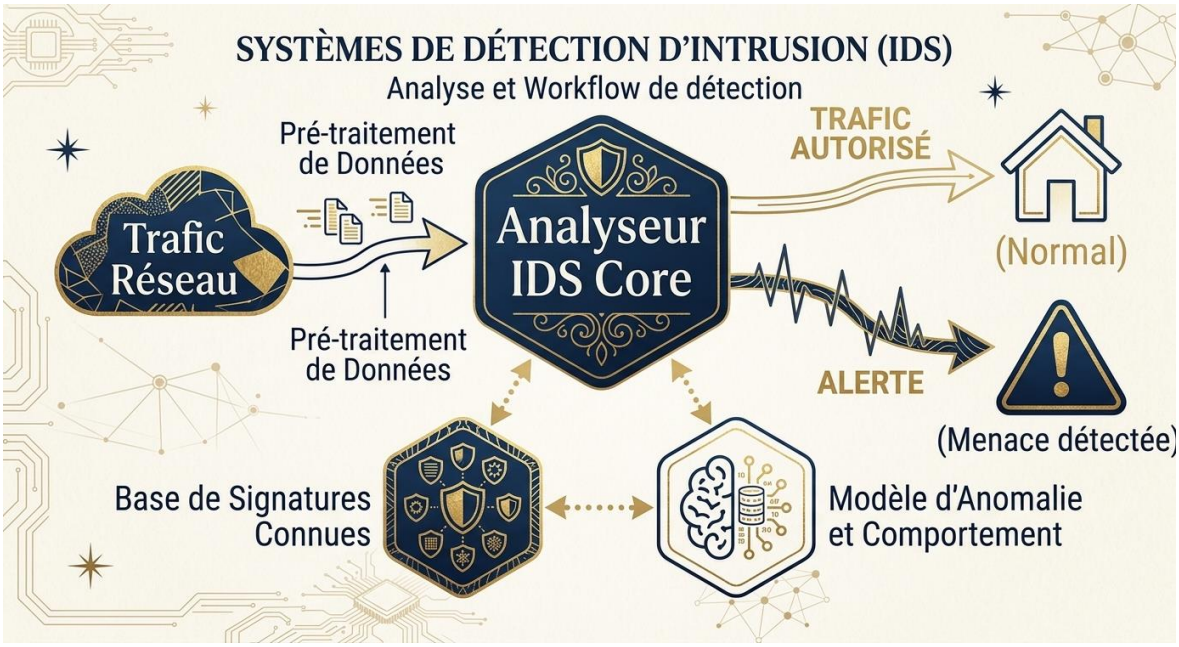


Figure 2. 3 : Architecture fonctionnelle générale d'un système de détection d'intrusion réseau (NIDS) combinant les approches supervisées et comportementales.

2.5.3 Approches de détection : Signatures vs Anomalies

L'efficacité opérationnelle d'un IDS s'articule autour de deux logiques de détection aux antipodes l'une de l'autre. La première, la détection par signatures, repose sur une approche déterministe. À l'image d'un catalogue de "recherche de motifs", elle confronte le flux entrant à des empreintes de menaces préalablement indexées. Si cette méthode garantit un taux de précision élevé contre les attaques documentées, elle s'avère structurellement aveugle face aux menaces "Zero-day" et aux mutations furtives des codes malveillants.

À l'inverse, la détection par anomalies, qui constitue le pivot de nos recherches, s'affranchit de la connaissance préalable des attaques. Son principe ne réside plus dans la reconnaissance, mais dans la modélisation : elle définit une empreinte de "normalité" statistique propre au

réseau. Tout écart substantiel par rapport à ce profil de référence est alors traité comme une suspicion d'intrusion. Cette approche est d'autant plus pertinente dans l'écosystème IoT qu'elle offre une résilience face aux comportements inédits et aux attaques polymorphes, là où les systèmes conventionnels se heurteraient à l'absence de signatures.

Dans le cadre de l'ingénierie moderne guidée par l'Intelligence Artificielle, cette détection par anomalies s'opérationnalise à travers deux paradigmes distincts que nous confrontons au sein de nos expérimentations :

L'apprentissage supervisé (*Supervised Learning*) : Incarné dans nos travaux par l'algorithme Random Forest, il exploite un ensemble de données préalablement étiquetées pour classifier les flux. Sa force réside dans sa capacité à atteindre une exactitude globale maximale (94%) face à des variantes complexes de menaces apprises, bien qu'il reste tributaire de la disponibilité des labels.

L'apprentissage non supervisé (*Unsupervised Learning*) : Concrétisé par l'utilisation de l'Isolation Forest et de notre modèle proposé basé sur l'Autoencodeur profond. Ce paradigme apprend de manière autonome à capturer la signature sémantique intrinsèque du trafic sain, détectant les anomalies (90%) par simple calcul de l'écart de l'erreur de reconstruction MSE) sans aucun étiquetage préalable, offrant ainsi une barrière proactive contre les menaces Zero-Day.

2.6 Vers une approche intelligente de la surveillance

L’adoption d’une détection basée sur les anomalies, bien que prometteuse, introduit des problématiques techniques non négligeables, au premier rang desquelles figure la gestion des faux positifs. Dans la fluidité d'un écosystème IoT, une fluctuation légitime telle qu'une mise à jour logicielle ou un pic d'activité saisonnier d'un capteur peut être identifiée à tort comme une intrusion. Pour surmonter cette ambiguïté, l’apport de l’intelligence artificielle ne se présente plus comme une option, mais comme une nécessité structurelle.

L’intégration d’algorithmes avancés permet de forger des mécanismes de détection plus granulaires. Ces modèles ne se contentent pas d'observer des seuils ; ils apprennent à interpréter la variabilité intrinsèque des données.

Dans cette optique, l'ingénierie de sécurité moderne s'articule autour de deux approches fondamentales que notre étude explore de manière approfondie :

2.6.1 Le paradigme de la classification supervisée (Random Forest)

L'apprentissage supervisé intervient comme une solution hautement performante pour cartographier les cybermenaces lorsque des données historiques étiquetées sont disponibles. En intégrant l'algorithme **Random Forest** au sein de notre framework, nous exploitons la puissance de l'apprentissage par ensemble (*Ensemble Learning*). Grâce à la construction de multiples arbres de décision indépendants et à un mécanisme de vote majoritaire, ce modèle excelle dans la discrimination chirurgicale des paquets malveillants. Il atteint une exactitude globale remarquable (**94%**), s'affirmant ainsi comme une référence majeure pour intercepter les variantes d'attaques documentées, malgré sa dépendance structurelle vis-à-vis des labels réseau.

2.6.2 Le paradigme de la modélisation comportementale non supervisée (Autoencoders & Isolation Forest)

Face à l'impossibilité pratique de labelliser en temps réel les flux massifs de l'IoT, l'apprentissage non supervisé s'impose comme une alternative indispensable. C'est ici que l'usage des **Autoencoders profonds** prend tout son sens : en apprenant à reconstruire uniquement le trafic sain, ils permettent de distinguer avec une précision accrue (**90%**) les comportements malveillants des simples fluctuations légitimes, et ce, par le biais du calcul de l'écart de l'erreur de reconstruction (MSE). Ce modèle profond est soutenu dans notre démarche par l'algorithme **Isolation Forest**, qui se focalise sur l'isolement structurel des anomalies macroscopiques. Cette approche autonome offre une immunité cruciale contre les menaces inédites (Zero-Day).

2.6.3 Justification de l'environnement expérimental : Le corpus CIC IoT 2023

L'étude théorique des vecteurs d'attaques ne peut atteindre sa finalité sans une phase de validation sur des données de terrain. Pour éprouver nos modèles, nous avons sélectionné le dataset **CICIoT2023** du *Canadian Institute for Cybersecurity*. Ce n'est pas un choix de commodité, mais une nécessité technique dictée par la nature même des réseaux actuels.

Pourquoi ce corpus précisément ? D'abord, pour sa fidélité aux menaces émergentes. Là où les datasets historiques échouent, le CICIoT2023 cartographie des offensives modernes telles que les dénis de service distribués (DDoS), les inondations DoS et les

infiltrations complexes par force brute (Brute Force), modélisées sur des topologies IoT réelles. Ensuite, sa richesse multidimensionnelle permet de capter l'hétérogénéité protocolaire que nous avons identifiée comme une vulnérabilité majeure. Enfin, la structure même des données, riche en descripteurs statistiques, offre un terrain fertile pour l'expérimentation algorithmique, permettant à la fois l'entraînement supervisé par labels et le profilage autonome du trafic normal.

Cette synergie entre la donnée brute et notre approche comparative trilatérale (Random Forest, Isolation Forest et Autoencoders) est le socle sur lequel repose toute la démonstration technique que nous déploierons dans la suite de ce travail.

2.7 Conclusion du chapitre

En somme, ce deuxième chapitre a permis de dresser un état des lieux sans concession des vulnérabilités inhérentes à l'Internet des Objets. Nous avons établi que la fragmentation architecturale et l'hétérogénéité des protocoles ne se contentent pas de multiplier les points d'entrée ; elles redéfinissent la nature même du risque cyber. Si les systèmes de détection d'intrusion réseau (NIDS) demeurent le rempart le plus prometteur, l'essoufflement des approches conventionnelles basées sur les signatures statiques face à la versatilité des attaques modernes nous impose un changement de paradigme.

Le constat qui découle de notre analyse est sans appel : la sécurisation pérenne des écosystèmes IoT ne peut plus reposer sur des règles figées. Elle exige des architectures de surveillance dotées d'une réelle capacité d'apprentissage et d'une autonomie d'adaptation face à l'inconnu. C'est précisément cette exigence de « détection comportementale intelligente » qui légitime notre recours aux approches avancées de l'Intelligence Artificielle, en mettant en confrontation la classification supervisée et le profilage non supervisé.

Ce passage de la théorie à la modélisation formelle constitue l'objet du chapitre suivant, où nous détaillerons les fondements mathématiques et structurels de nos modèles algorithmiques (**Random Forest, Isolation Forest et Autoencodeurs profonds**) avant leur évaluation sur le corpus standardisé **CICIoT2023**.

Chapitre 3 :

Approche Méthodologique :

Paradigmes d'Apprentissage et Modélisation Algorithmique

3.1 Introduction du chapitre

L'évolution rapide des réseaux IoT a engendré une volumétrie de données telle que les méthodes de surveillance classiques, basées sur des règles prédéfinies, s'avèrent aujourd'hui obsolètes. Le passage d'une détection statique vers une approche dynamique guidée par l'Intelligence Artificielle n'est plus une simple option technique, mais une nécessité stratégique.

Ce troisième chapitre s'attache à détailler les fondements méthodologiques de notre recherche. L'originalité de notre démarche scientifique réside dans l'exploration croisée et la confrontation des paradigmes d'apprentissage fondamentaux. Nous y formaliserons les mécaniques de la classification supervisée à travers l'algorithme Random Forest, avant de plonger dans les structures du profilage non supervisé, en mettant en relief la complémentarité entre l'algorithme classique Isolation Forest et la puissance générative des Autoencoders profonds. L'objectif est de fournir une compréhension rigoureuse et trilatérale des outils mathématiques et algorithmiques qui nous permettront, dans la phase expérimentale, de distinguer le signal normal du bruit malveillant au sein des écosystèmes connectés.

3.2 Fondamentaux des Paradigmes de l'Intelligence Artificielle en Cybersécurité

Au fond, intégrer l'Intelligence Artificielle dans notre stratégie de cybersécurité n'est pas un choix technologique de plus, c'est une urgence. Le constat est assez simple : les cyberattaques évoluent à une vitesse folle tandis que nos vieux outils de défense, bloqués sur des règles rigides, n'arrivent plus à suivre. Ils sont dépassés. Pour justifier la logique des modèles prédictifs et classificatoires que nous avons développés dans ce mémoire, nous devons d'abord poser les bases théoriques des différentes philosophies d'apprentissage et comprendre la dynamique technologique qui lie le Machine Learning classique à la puissance du Deep Learning.

3.2.1 Le Machine Learning (Apprentissage Automatique) : Définition et Typologies

Le Machine Learning (ML) désigne un sous-champ de l'IA qui confère aux systèmes informatiques la capacité d'apprendre à partir de données historiques sans être explicitement programmés [11]. Fondamentalement, le ML cherche à modéliser une fonction sous-jacente

à partir d'un ensemble d'observations afin de réaliser des prédictions ou des classifications sur de nouvelles instances.

Comme le formalise le pipeline séquentiel de la **Figure 3.1**, l'ingénierie de l'apprentissage automatique s'articule autour d'un cycle de vie rigoureux. Les données brutes de trafic (*Données*) alimentent un algorithme mathématique (*Algorithme*) lors d'une phase d'ajustement des paramètres appelée entraînement (*Entraînement*). Ce processus converge vers la création d'un modèle prédictif autonome (*Modèle prédictif*). Une fois déployé, ce modèle est capable d'ingérer des flux réseau totalement inédits (*Nouvelles données*) pour extrapoler une prédiction binaire ou multiclasse (*Prédiction / Décision*), isolant ainsi instantanément les paquets malveillants des flux légitimes. Dans le cadre de la détection d'intrusions, ce paradigme global se divise en deux voies d'ingénierie majeures que nous allons détailler.

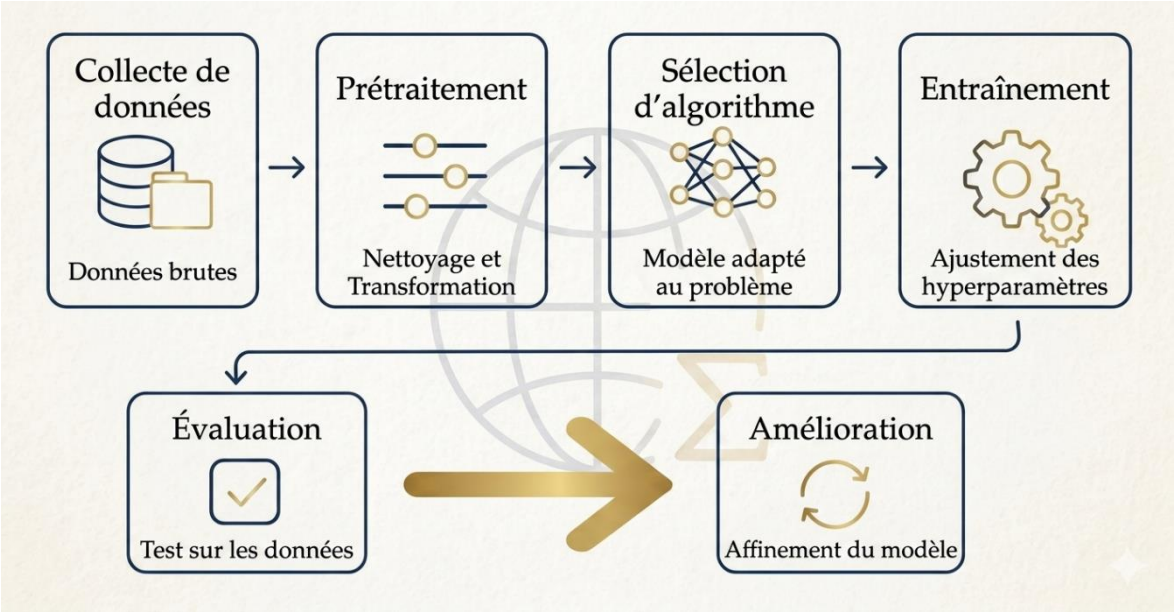


Figure 3. 1 : Pipeline général du Machine Learning : du flux de données brut à la décision prédictive.

3.2.1.1 L'apprentissage supervisé : La classification orientée par les étiquettes (Section réécrite)

L'apprentissage supervisé repose sur l'existence d'un ensemble de données d'entraînement rigoureusement étiquetées ou labellisées, où chaque vecteur de caractéristiques X est associé à une cible ou classe connue Y (par exemple : Trafic Normal

ou Attaque). L'objectif de l'algorithme est de converger vers une fonction de décision capable de discriminer les frontières de classes.

Dans notre étude, ce paradigme est matérialisé par l'algorithme **Random Forest**. Sa force réside dans sa capacité à extraire des règles logiques hautement discriminantes, permettant d'atteindre une exactitude de classification maximale (**94%**) face aux attaques répertoriées. Néanmoins, ce modèle hérite d'une dépendance stricte vis-à-vis d'une labellisation préalable, une contrainte opérationnelle lourde au sein des flux IoT véloces. Le

Paradigme d'apprentissage	Nature des données utilisées	Nature des données utilisées	Limites majeures en sécurité IoT
Apprentissage Supervisé *(Ex: Random Forest)*	Données entièrement étiquetées (Couples entrée-sortie).	S'entraîne sur des profils pré-annotés pour classifier des attaques connues.	Échoue totalement face aux menaces inconnues (Zero-Day) et engendre un coût d'annotation humain prohibitif [12].
Apprentissage Semi-supervisé	Mixte : une infime quantité de données étiquetées combinée à une masse critique de données brutes.	Cherche un compromis opérationnel en exploitant la structure des données non étiquetées.	Reste fortement tributaire d'une connaissance partielle ou initiale des signatures d'attaques.
Apprentissage par Renforcement	Données générées par l'expérience (pas d'étiquettes initiales).	Repose sur un cycle d'essais-erreurs avec un système de récompenses et de pénalités en environnement dynamique.	Algorithme très complexe à stabiliser et difficilement applicable en temps réel pour la surveillance passive de flux réseau.

Tableau 3. 1 : Analyse comparative des paradigmes d'apprentissage classiques (Supervisé, Semi-supervisé et Renforcement) dans le contexte IoT [12]

3.2.1.2 L'apprentissage non supervisé : Le profilage autonome et la détection d'anomalies

Contrairement au paradigme précédent, l'apprentissage non supervisé traite des données totalement dénuées de labels ou d'annotations préalables [13]. Le modèle doit explorer la structure intrinsèque des données pour y déceler des régularités, des corrélations ou des regroupements naturels *(clustering)*.

Dans l'écosystème IoT, où le trafic sain est ultra-majoritaire, cette approche devient le pilier d'une détection d'anomalies proactive. Elle permet de modéliser le comportement statistique normal du réseau afin de rejeter toute déviation comme suspecte, s'affranchissant ainsi de la nécessité de connaître l'attaque au préalable [12], [13]. C'est cette philosophie qui gouverne nos implémentations de l'**Isolation Forest** et des **Autoencoders**.

3.2.2 Le Deep Learning (Apprentissage Profond) : Évolution et Principes

Le *Deep Learning* (DL) représente une évolution technologique majeure du ML. Inspiré par l'architecture neuronale biologique, le DL repose sur l'empilement de multiples couches de neurones artificiels (couches denses, de compression ou de reconstruction) [14].

Le principe fondamental du DL réside dans l'apprentissage hiérarchique de Représentations:

Les premières couches extraisent des caractéristiques brutes, tandis que les couches profondes combinent ces informations pour modéliser des concepts abstraits et hautement non linéaires [14]. Cette capacité d'auto-extraction élimine l'étape artisanale du *Feature Engineering* (sélection manuelle des variables), permettant d'ingérer directement des représentations de données complexes et massives, telles que les flux réseaux capturés en temps réel.

3.2.3 Analyse Comparative : Machine Learning vs Deep Learning dans l'écosystème IoT

Le choix entre l'approche classique (ML) et l'approche profonde (DL) au sein des environnements IoT répond à des impératifs de volume de données et de complexité computationnelle. Le tableau suivant dresse une analyse comparative de ces deux paradigmes, inspirée des synthèses de la littérature scientifique de l'IA appliquée à la sécurité [15], [16] :

Critères d'évaluation	M.Learning Classique (ex. Isolation Forest)	Deep Learning (ex. Autoencoders)
Feature Engineering	**Manuel : Nécessite une expertise humaine pour sélectionner et extraire les caractéristiques du trafic réseau.**	**Autonome : Extrait automatiquement les lois statistiques et caractéristiques complexes à travers ses couches.**
Volume de données	**Performance optimale sur des volumes de données faibles à modérés ; sature sur les flux massifs.**	**Plus la masse de données est critique (Big Data IoT), plus la précision et la robustesse du modèle augmentent.**
Complexité mathématique	**Faible à modérée. Modèles interprétables et rapides à entraîner.**	**Élevée. Relations non linéaires complexes capturées à travers des millions de paramètres (poids).**
Temps d'exécution	**Entraînement rapide. Phase d'inférence légère, adaptée aux passerelles IoT contraintes.**	**Entraînement lourd (nécessite idéalement des GPU) ; mais inférence ultra-rapide une fois le modèle optimisé.**

Tableau 3. 2 : Confrontation technologique entre le Machine Learning classique et le Deep Learning face aux contraintes de l'écosystème IoT [15], [16]

3.2.4 Justification de l'arbitrage doctrinal : Confrontation Supervisé / Non Supervisé face aux menaces Zero-Day

La justification scientifique de notre évaluation comparative trilatérale repose sur la nature intrinsèque des menaces cybernétiques modernes, et plus particulièrement le compromis opérationnel entre la détection des attaques connues et inconnues (Zero-Day).

Par définition, une vulnérabilité Zero-Day exploite une faille de sécurité inédite, pour laquelle aucune signature algorithmique n'existe dans les bases de données des IDS traditionnels [17]. Un modèle purement supervisé, bien qu'extrêmement performant sur son domaine d'apprentissage (comme le prouvent les **94%** d'exactitude de notre Random Forest), est par nature aveugle à ces attaques puisqu'elles n'représentent pas une classe connue lors de son entraînement.

En inversant la logique algorithmique c'est-à-dire en entraînant nos modèles non supervisés (*Isolation Forest* et *Autoencoders*) à ne reconnaître et valider exclusivement que la normalité (le trafic réseau IoT sain et légitime) le système s'affranchit de la nécessité de posséder un historique des menaces [18], [19]. Toute tentative d'intrusion, toute exécution de code malveillant ou toute anomalie comportementale provoquera inévitablement une rupture statistique ou géométrique vis-à-vis du profil normal appris. Comme nous le démontrerons dans la phase expérimentale, cette rupture se traduira soit par une isolation géométrique rapide des points (*iForest*), soit par une explosion de l'erreur quadratique moyenne de reconstruction (*MSE* de l'Autoencodeur à **90%**), offrant ainsi une défense adaptative face aux menaces les plus furtives.

3.3 Algorithme Random Forest (RF) : Le Paradigme Supervisé

L'algorithme *Random Forest* (Forêt Aléatoire), introduit par Leo Breiman en 2001, représente l'un de nos choix technologiques clés pour le moteur de classification supervisée [22]. Il s'agit d'une méthode d'apprentissage par ensemble (*Ensemble Learning*) basée sur le principe du méta-apprentissage. Comme illustré par la **Figure 3.2**, cette approche combine la puissance prédictive de plusieurs arbres de décision indépendants pour concevoir un modèle robuste, stable et hautement précis, capable de s'adapter aux dynamiques des données réseau sans surapprentissage [11] [12] .

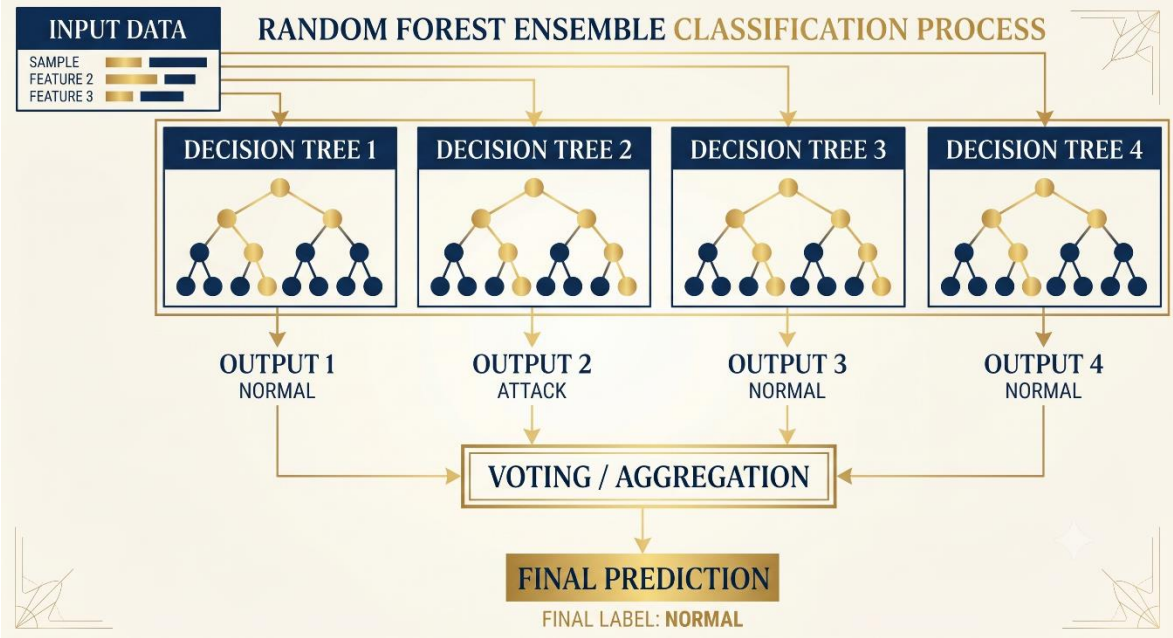


Figure 3. 2 : Architecture conceptuelle de l'algorithme Random Forest : Processus de classification par agrégation de décisions multiples

3.3.1 Principes des Arbres de Décision et du Bagging

Le fondement de Random Forest repose sur la technique du **Bagging** *(Bootstrap Aggregating)* associée à une sélection aléatoire des variables (*Feature Bagging*).

- **Échantillonnage Bootstrap :** Pour chaque arbre T_k de la forêt, un sous-ensemble de données est généré par tirage aléatoire avec remplacement à partir du dataset d'entraînement global. Si le dataset contient N instances, chaque échantillon bootstrap possède également N lignes, mais contient environ 63,2% d'individus uniques, le reste étant des doublons. Les instances non sélectionnées,

Appelées *Out-Of-Bag (OOB)*, servent d'ensemble de validation interne pour estimer l'erreur de généralisation sans recourir à un jeu de test externe.

- **Sélection aléatoire des descripteurs :** Contrairement à un arbre de décision classique qui évalue l'ensemble des descripteurs pour scinder un nœud, Random Forest n'examine qu'un sous-ensemble aléatoire de taille m parmi les M caractéristiques disponibles (généralement m = $\sqrt{M}$). Cette double randomisation (sur les lignes et sur les colonnes) brise la corrélation entre les arbres, augmentant la diversité de la forêt et immunisant structurellement le modèle contre le surapprentissage (*Overfitting*).

3.3.2 Architecture de l'Ensemble Learning et Vote Majoritaire

L'architecture fonctionnelle de Random Forest repose sur l'agrégation collective. Chaque arbre de décision au sein de la forêt est développé jusqu'à sa profondeur maximale sans aucun élagage (*Pruning*), agissant comme un classifieur individuel à faible biais mais à forte variance.

Lors de la phase de test ou de prédiction en temps réel sur les flux IoT, un nouveau vecteur de trafic réseau X (contenant les métadonnées de la trame) traverse simultanément l'intégralité des K arbres de la forêt. Chaque arbre émet une prédiction de classe binaire :

$$\hat{y}_k \epsilon \{0, 1\} \ (ou \ \ 0 : \text{Trafic Sain} \ ; \ \ 1 : \text{Attaque}$$

L'agrégation finale s'effectue par un mécanisme de **Vote Majoritaire** (*Majority Voting*). La classe finale prédite $\{\hat{y}\}$ pour le paquet réseau correspond à la catégorie ayant recueilli le plus grand nombre de suffrages à travers la forêt, formalisée par l'équation suivante :

$$\hat{y} = \text{mode}\{\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_k\}$$

En moyennant les décisions de centaines d'arbres indépendants, Random Forest réduit drastiquement la variance globale du système sans augmenter son biais, ce qui explique statistiquement son exactitude exceptionnelle de **94%** obtenue lors de nos expérimentations sur le corpus CICIoT2023.

3.3.3 Formalisation Mathématique et Critères de Division

La construction de chaque nœud au sein des arbres de décision repose sur la recherche de la partition optimale capable de maximiser l'homogénéité (ou pureté) des nœuds enfants. Dans le cadre de nos travaux, nous exploitons l'**Indice de Gini** comme critère de mesure d'impureté.

Pour un nœud donné D, l'impureté de Gini, notée $\boldsymbol{IG(D)}$, se calcule par la formule mathématique suivante :

Où C représente le nombre de classes (C=2 pour notre classification binaire) et p_i correspond à la probabilité de présence de la classe i au sein de ce nœud.

À chaque étape de division, l'algorithme sélectionne la caractéristique et le seuil de coupure qui maximisent le gain d'information, c'est-à-dire qui minimisent la somme pondérée des impuretés des nœuds fils gauche (D_g) et droit (D_d) :

$$\boldsymbol{\Delta I_G = IG(D) - \left(\frac{|D_g|}{|D|}\right) IG(D) + \left(\frac{|D_d|}{|D|}\right) IG(D_d)}$$

Équation 3. 1: calcul du gain d'information pour la division des nœuds (Random Forest)

Ce processus récursif se poursuit jusqu'à ce que les nœuds deviennent parfaitement purs ou qu'une contrainte d'arrêt (profondeur maximale ou nombre minimal d'instances par feuille) soit atteinte. Cette rigueur mathématique permet d'isoler avec une précision chirurgicale les signatures de comportements malveillants tels que les attaques DoS-TCP ou Dictionary Brute Force.

3.4 Algorithme Isolation Forest (iForest)

L'algorithme **Isolation Forest (iForest)**, introduit initialement par Liu et al, représente une approche incontournable et particulièrement efficace pour la détection d'anomalies dans les flux de données massifs. Contrairement aux méthodes traditionnelles de machine learning qui cherchent à modéliser le profil de normalité pour en déduire les écarts, l'iForest adopte une stratégie inverse en isolant directement les instances anormales [18].

Afin de synthétiser l'ensemble des concepts théoriques, géométriques et mathématiques liés à l'algorithme Isolation Forest, la figure ci-dessous **Figure 3.3** regroupe de manière intégrée l'espace de partitionnement, l'architecture de la forêt d'iTrees ainsi que la logique de décision finale adaptée à la sécurité de notre écosystème IoT.

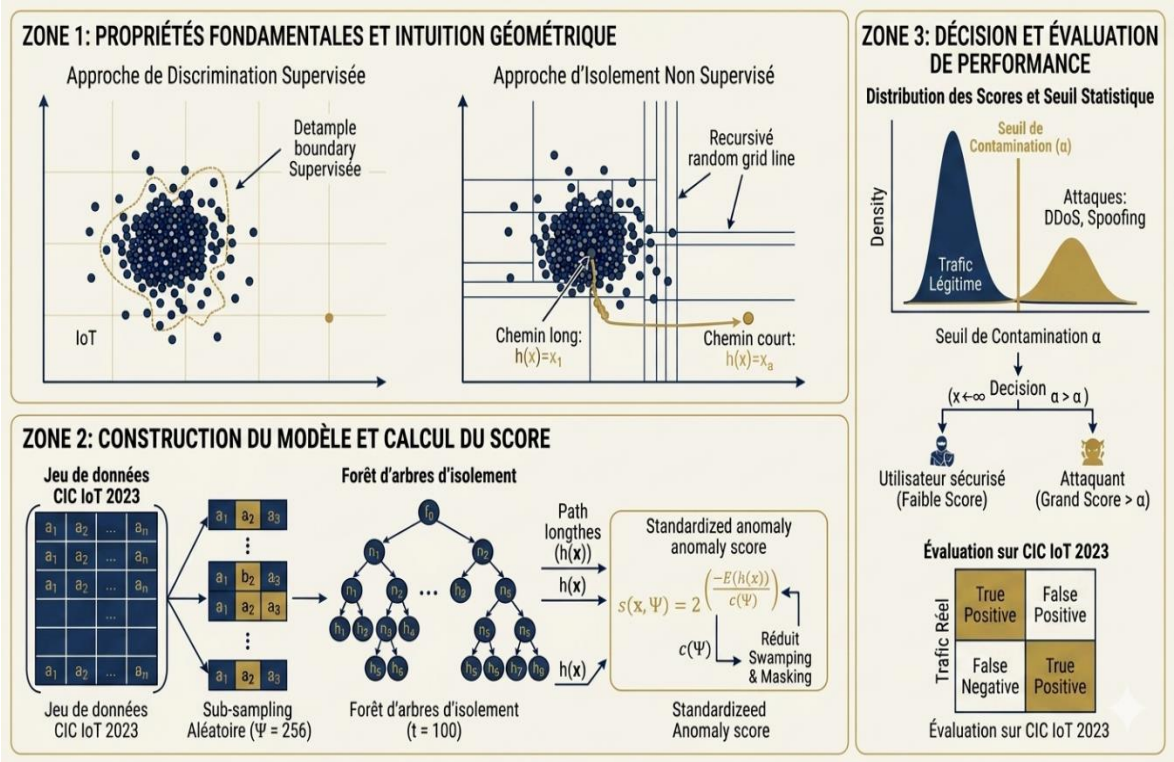


Figure 3. 3 : Synthèse méthodologique, algorithmique et fonctionnelle de l'Isolation Forest pour la détection d'intrusions IoT.

3.4.1 Principe de l'Isolation et Partitionnement

Le fondement de cet algorithme repose sur l'hypothèse axiomatique que les attaques réseau ou les dysfonctionnements possèdent deux caractéristiques intrinsèques : ils sont minoritaires (rares) et présentent des attributs hautement divergents par rapport à la masse du trafic légitime [18].

Techniquement, l'algorithme construit un ensemble d'arbres binaires de décision (*Isolation Trees* ou *iTrees*). À chaque nœud, un descripteur (*feature*) du trafic réseau est sélectionné de manière aléatoire, suivi d'une valeur de coupure choisie aléatoirement entre le minimum et le maximum de cet attribut. Ce processus de partitionnement récursif se poursuit jusqu'à ce que les points de données soient complètement isolés.

3.4.2 Formalisation Mathématique et Logique de Décision

En raison de leur rareté et de leur divergence, les anomalies requièrent nettement moins de partitions aléatoires pour être isolées par rapport aux points normaux. Par conséquent, les instances malveillantes se retrouvent rapidement confinées dans les branches les plus superficielles des arbres (trajets courts)

Pour quantifier ce degré d'anomalie de manière rigoureuse, un score standardisé $s(x,n)$ est calculé pour chaque instance x selon la formulation mathématique suivante :

$$s(x,n) = \mathbf{2}^{-\frac{E(h(x))}{c(n)}}$$

Équation 3. 2: calcul du score d'anomalie (s) - Isolation Forest

Où :

- $\boldsymbol{E(h(x))}$ Représente l'espérance de la longueur du chemin (path length) $h(x)$de l'instance x à travers l'ensemble de la forêt d'arbres.
- $\boldsymbol{c(n)}$ est la longueur moyenne d'un échec de recherche dans un arbre binaire, défini comme le facteur de normalisation mathématique pour des arbres construits avec n instances [18].

Interprétation académique :

- Si $S \to 1$, la longueur du chemin est minimale, ce qui classifie l'instance comme une **anomalie hautement probable**.

- Si S < 0.5, l'instance nécessite un parcours profond et est considérée comme un comportement **normal ou légitime** [18].

3.4.3 Configuration et Justification du Choix pour l'Écosystème IoT

Dans le contexte spécifique de notre évaluation sur le corpus **CIC IoT 2023**, l'implémentation de l'Isolation Forest se justifie par des impératifs techniques et des configurations d'hyperparamètres documentés comme optimaux dans les travaux récents [20] :

Efficacité computationnelle : Sa complexité temporelle linéaire et sa faible empreinte mémoire permettent un traitement de données en temps réel, parfaitement adapté aux passerelles (*gateways*) IoT aux ressources matérielles limitées [20].

Résistance au phénomène de "Swamping" : Grâce à l'utilisation du sous-échantillonnage aléatoire (*sub-sampling*), **généralement configuré à $\psi = 256$ dans les travaux fondateurs de Liu et al,** l'iForest conserve une excellente précision de détection. Ce mécanisme empêche que les attaques massives ou regroupées en clusters denses (telles que les attaques par déni de service présentes dans CIC IoT 2023) ne masquent leur propre nature anormale.

Stabilité statistique : Le nombre théorique d'arbres d'isolement (*iTrees*) **est conventionnellement fixé à $\tau = 100$** dans la littérature afin de stabiliser la convergence de l'espérance du chemin $E(h(x))$ [18].

Il convient de noter que si ces valeurs **(ψ = = 256 , τ = 100)** constituent le standard théorique de l'algorithme, elles seront ajustées et adaptées empiriquement lors de notre phase expérimentale (Chapitre 4) afin de répondre précisément aux spécificités et à la volumétrie de notre jeu de données.

3.5 Architecture des Autoencoders : Vers une Extraction de Caractéristiques Profondes

Les **Autoencoders (AE)** représentent une classe spécifique de réseaux de neurones artificiels profonds, exploités principalement dans le cadre de l'apprentissage non supervisé pour la réduction de dimensionnalité et la détection d'intrusions. Contrairement aux classificateurs classiques, l'objectif fondamental d'un Autoencoder n'est pas de prédire une étiquette, mais d'apprendre une représentation condensée (ou codée) du signal d'entrée afin de pouvoir le reconstruire en sortie avec une distorsion minimale [14], [21].

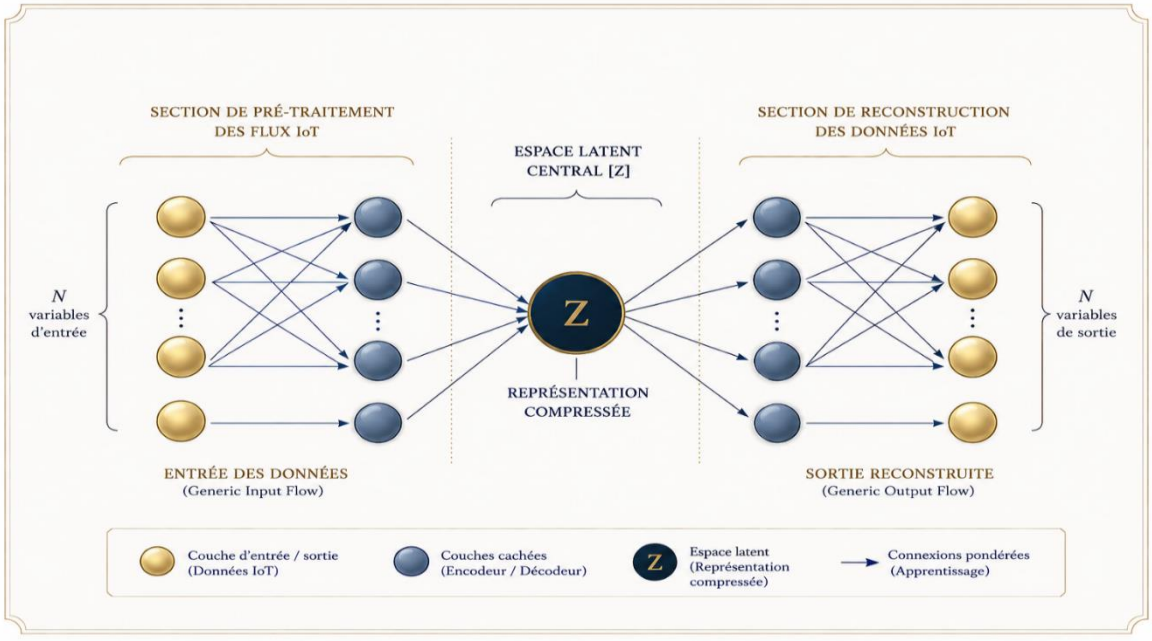


Figure 3. 4 : Architecture de l'autoencodeur

3.5.1 Le Processus d'Encodage : Compression et Espace Latent

L'architecture interne d'un Autoencoder est rigoureusement symétrique par rapport à sa couche centrale et se décompose en deux processus mathématiques successifs :

L'Encodeur $(\boldsymbol{f})$ **:** Il prend en entrée le vecteur $\boldsymbol{x} \in \boldsymbol{R^d}$ (représentant les caractéristiques statistiques du trafic réseau issu de **CIC IoT 2023**) et le projette vers un espace latent $\boldsymbol{Z} \in \boldsymbol{R^K}$ de dimension inférieure $(k < d)$. Cette transformation non linéaire est formalsée par:

$$z = \sigma(Wx + b)$$

Équation 3. 3 : projection dans l'espace latent (Processus d'encodage)

Où W désigne la matrice des poids, b le vecteur de biais, et σ la fonction d'activation (ex. ReLU ou Sigmoid).

2. **Le Décodeur (g) :** Il réalise l'opération réciproque en tentant de régénérer un vecteur de sortie reconstruit $\hat{x} \in R^d$ à partir de la représentation compressée z [14] :

$$\hat{x} = \sigma'(W'x + b')$$

Équation 3. 4: reconstruction du vecteur de sortie (Processus de décodeur)

3.5.2 La Dynamique du "Bottleneck" et l'Apprentissage de la Normalité

La couche centrale, communément appelée **Bottleneck** (goulot d'étranglement), joue un rôle de régulateur critique. En limitant le nombre de neurones dans cette couche, le réseau est contraint d'éliminer le bruit et les redondances informationnelles pour ne capturer que les structures de corrélation non linéaires essentielles du trafic [14] [21].

Dans le cadre de la détection d'anomalies, le modèle est entraîné **exclusivement sur un trafic réseau sain (légitime)**. Ainsi, l'Autoencoder apprend à maîtriser parfaitement la topologie statistique de la normalité. Face à un comportement malveillant inédit (ex. déni de service DoS, injection), la structure des données divergera des motifs appris, ce qui provoquera une incapacité du décodeur à reconstruire fidèlement le signal.

3.5.3 Critère de Détection par l'Erreur de Reconstruction

Le signal d'alerte du système repose sur la quantification de la perte de reconstruction. La métrique standardisée exploitée est l'**Erreur Quadratique Moyenne (Mean Squared Error - MSE)**, formalisée mathématiquement comme suit :

$$L(xi - \hat{x}i) = \frac{1}{n}\sum_{i=1}^{n}(xi - \hat{x}i)^2$$

Équation 3. 5: l'erreur de reconstruction par la méthode MSE (Mean Squared Error)

Logique de décision : La classification d'un flux réseau s'effectue par confrontation à un seuil critique $\boldsymbol{\tau}$ déterminé lors de la phase de validation :

$$\boldsymbol{ClassificationL} = \begin{cases} \boldsymbol{Anomalie\ (Attaque)\ , si\ (x, \hat{x}) > \tau} \\ \boldsymbol{Normal\ (Légitime)\ , si\ \ (x, \hat{x}) \leq \tau} \end{cases}$$

Ce formalisme mathématique garantit une excellente efficacité face aux attaques de type "Zero-day", puisque toute anomalie comportementale se traduit mécaniquement par une explosion de l'erreur de reconstruction $L(\boldsymbol{x}, \hat{\boldsymbol{x}})$ [21].

Dans le but d'illustrer la dynamique interne de cette approche profonde, la figure ci-dessous (**Figure 3.5**) synthétise de manière globale l'architecture symétrique de notre Autoencoder, le processus d'apprentissage exclusif de la normalité, ainsi que la logique de décision basée sur l'explosion de l'erreur de reconstruction (MSE) face aux intrusions.

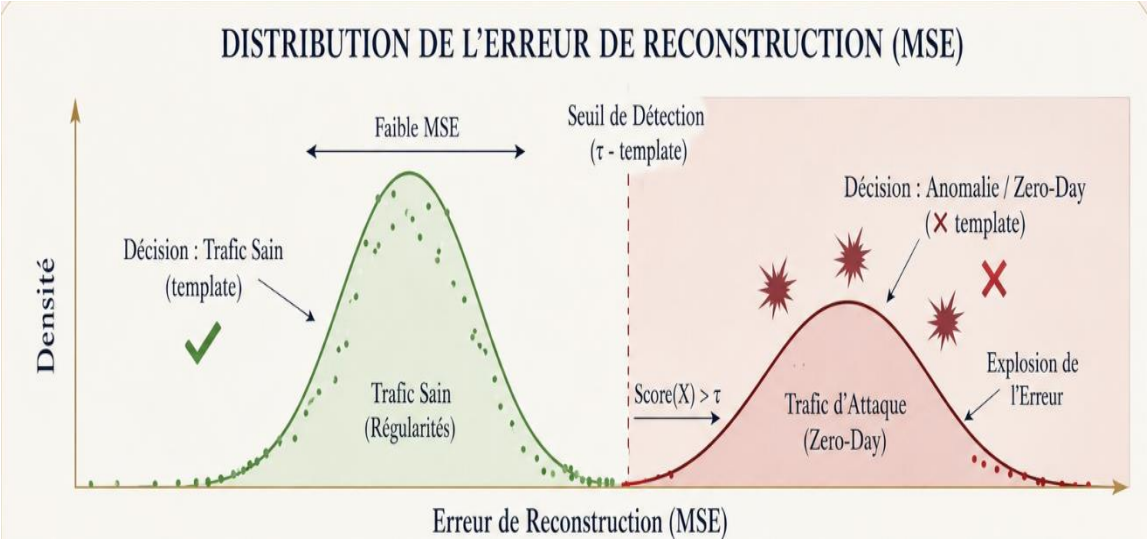


Figure 3. 5 : Architecture réseau, dynamique d'apprentissage et logique de classification d'un Autoencoder pour la détection d'anomalies IoT

3.6 Synthèse Comparative des Approches Proposées

Pour conclure ce chapitre, il est essentiel de mettre en perspective les trois architectures retenues, chacune reposant sur une philosophie de détection distincte. Le Tableau 3.3 ci-dessous synthétise la comparaison technique entre l'approche statistique par partitionnement (Isolation Forest), l'approche supervisée par arbres de décision (Random Forest) et l'approche neuronale profonde (Autoencoders). Cette confrontation permet d'évaluer la pertinence de chaque modèle face aux contraintes de l'IoT et à leur capacité de généralisation contre les menaces émergentes.

Caractéristique / Critère	*Isolation Forest (iForest)*	*Random Forest (RF)*	*Autoencoders (AE)*
Type de Modèle	Algorithme basé sur des arbres	Modèle supervisé (Ensemble learning)	Réseau de neurones profonds
Approche Conceptuelle	Isolement des points rares	Classification par votes de décision	Reconstruction de la normalité
Besoins en Données	Non supervisé	Supervisé (étiquetées)	Non supervisé
Espace Latent	Non applicable	Non applicable	Présent (couche Bottleneck)
Complexité	Faible (Efficace)	Modérée	Élevée (Entraînement)
Cible d'Attaque	Anomalies éparses	Signatures connues (Signatures)	Menaces complexes / Zero-day

Tableau 3. 3 : Synthèse comparative des approches étudiées

3.7 Conclusion du chapitre

En conclusion, ce troisième chapitre a permis de poser les jalons méthodologiques et les formalisations mathématiques nécessaires à la compréhension globale de notre système de détection d'intrusions. Face à la dualité des exigences de la sécurité IoT, qui réclame à la fois une précision chirurgicale contre les menaces connues et une résilience adaptative face à l'imprévisibilité des attaques Zero-Day, nous avons démontré qu'une approche tridimensionnelle s'impose comme une stratégie incontournable.

D'une part, l'algorithme supervisé Random Forest nous apporte une robustesse classificatoire de haut niveau en s'appuyant sur l'apprentissage par ensemble et le vote majoritaire pour discriminer les signatures de menaces documentées. D'autre part, le paradigme non supervisé pallie l'absence de labels : l'algorithme Isolation Forest offre une solution géométrique rapide pour isoler les anomalies structurelles macroscopiques, tandis que l'architecture des Autoencoders profonds apporte une granularité analytique supérieure, capable de modéliser les relations sémantiques non linéaires du trafic sain pour déceler les déviations les plus subtiles via l'analyse de l'erreur quadratique moyenne de reconstruction (MSE).

La confrontation de ces trois philosophies algorithmiques constitue le cœur de notre contribution méthodologique. Forts de ces fondements théoriques et formels, nous sommes désormais en mesure de passer à la phase de validation empirique. Le chapitre suivant sera ainsi dédié à l'implémentation pratique de ces modèles, à l'exploitation du jeu de données CIC-IoT-2023, ainsi qu'à l'analyse comparative finale de leurs performances en termes de matrices de confusion, d'exactitude et de compromis computationnels.

Chapitre 4 :

Implémentation Expérimentale, Résultats et Analyse Comparative

4.1 INTRODUCTION

Ce quatrième chapitre constitue le pivot opérationnel et empirique de notre travail de recherche. Après avoir exposé les fondements théoriques de la cybersécurité des architectures de l'Internet des Objets (IoT) et les paradigmes de l'intelligence artificielle, il convient à présent de confronter ces concepts à la réalité des flux réseau. L'objectif de cette phase expérimentale est d'implémenter, d'évaluer et de mener une analyse comparative rigoureuse entre trois philosophies distinctes de détection d'anomalies :

- L'approche par apprentissage non supervisé statistique classique, portée par l'algorithme **Isolation Forest**.
- L'approche par apprentissage non supervisé profond, matérialisée par notre réseau de neurones de type **Autoencodeur**.
- L'approche par apprentissage supervisé traditionnel, représentée par l'algorithme **Random Forest**.

Pour ce faire, nous structurons notre démarche autour d'un protocole empirique strict, exécuté en deux grandes phases d'évaluation. Après la description détaillée de notre environnement de développement et le cycle crucial de préparation, d'échantillonnage et de normalisation statistique des flux issus du dataset de référence **CICIoT2023**, nous présentons nos simulations :

- D'une part, le **Cas Nominal** évalue la capacité des modèles à classifier des cyberattaques contemporaines connues et répertoriées lors de l'entraînement (DDoS-ICMP, DoS-TCP).
- D'autre part, le scénario de **Résilience face aux menaces inconnues (deuxième cas)** simule une attaque de type **Zero-Day** en excluant totalement les flux de *Dictionary Brute Force* de la phase d'apprentissage des modèles supervisés, afin de mettre à l'épreuve l'extensibilité et la robustesse comportementale de notre approche non supervisée.

Enfin, une place centrale sera accordée à la présentation des résultats bruts (matrices de confusion, courbes d'apprentissage) et à une discussion critique approfondie. Cette analyse mettra en exergue le compromis fondamental (*trade-off*) entre l'exactitude de détection face à l'imprévisibilité des cybermenaces et les contraintes de latence d'inférence, indispensables pour un déploiement réel sur des passerelles IoT à ressources limitées.

4.2 ENVIRONNEMENT DE DEVELOPPEMENT ET REPRODUCTIBILITE

La robustesse et la crédibilité d'une étude expérimentale en apprentissage automatique et en apprentissage profond reposent sur la transparence de son infrastructure et sur sa capacité à être reproduite à l'identique. Cette section décrit l'écosystème technique, logiciel et matériel qui a soutenu l'implémentation de l'ensemble de nos simulations (**pour les deux cas**), ainsi que les mécanismes de contrôle de l'aléa numérique mis en œuvre pour garantir l'invariabilité de nos résultats.

4.2.1 Plateforme Kaggle Notebook et Infrastructure Cloud

Pour mener à bien nos calculs intensifs, notamment lors de l'entraînement du réseau de neurones profond (Autoencodeur), nous avons exploité l'environnement cloud de la plateforme Kaggle Notebooks. Ce choix se justifie par la mise à disposition d'une infrastructure conteneurisée (Docker) stable et de ressources matérielles performantes, évitant ainsi les goulots d'étranglement liés aux architectures locales limitées. Les spécifications matérielles allouées pour nos expérimentations sont les suivantes :

Processeur (CPU) : Intel(R) Xeon(R) cadencé à 2.20GHz, optimisé pour les opérations de prétraitement et de manipulation de grands tableaux de données.

Mémoire Vive (RAM) : 30 Go de mémoire système, indispensable pour l'hébergement en mémoire des matrices de données normalisées (X_scaled) issues du dataset CICIoT2023.

Stockage : Un espace disque temporaire de 20 Go permettant la persistance des fichiers de modèles sauvegardés (.keras et .bin).

4.2.2 Langage et Bibliothèques Spécifiques

L'implémentation de l'architecture logicielle a été réalisée en langage Python (Version 3.10), standard de l'industrie de l'intelligence artificielle. L'écosystème applicatif repose sur une sélection de bibliothèques hautement optimisées :

Pandas et NumPy : Utilisées exclusivement pour l'ingestion, le nettoyage statistique, l'échantillonnage aléatoire et la vectorisation algébrique des flux réseau.

Scikit-Learn (Sklearn) : Exploité pour l'extraction de l'outil de normalisation (StandardScaler), le codage des étiquettes (LabelEncoder), la division stratifiée des données,

ainsi que pour l'implémentation et l'évaluation du modèle supervisé Random Forest et non-supervisé Isolation Forest.

TensorFlow / Keras : Framework retenu pour la conception couche par couche, la compilation (Optimiseur Adam, perte MSE) et l'entraînement de notre réseau de neurones Autoencodeur.

Matplotlib et Seaborn : Dédiés à la génération des tableaux de métriques, des courbes de perte (Loss) et à la visualisation des matrices de confusion.

4.2.3 Reproductibilité et Fixation du SEED

En apprentissage automatique, de nombreuses étapes dépendent de l'aléa numérique (sélection aléatoire de l'échantillonnage par sample(), initialisation des poids des neurones, partitionnement des arbres de décision, division du dataset). Sans contrôle, deux exécutions successives du même code produisent des résultats légèrement différents, ce qui annule la rigueur scientifique.

Pour neutraliser cet effet et imposer une reproductibilité stricte à 100%, nous avons verrouillé le générateur de nombres pseudo-aléatoires en fixant un germe unique (**random_state=42** ou **SEED=42**). Ce verrouillage a été appliqué systématiquement à travers les commandes :

1. **np.random.seed(42)** pour stabiliser les opérations de l'environnement NumPy.
2. **random_state=42** lors de l'extraction des échantillons de données (sample) et de la division du jeu de données (train_test_split).
3. **random_state=42** lors de l'initialisation des estimateurs des modèles *Isolation Forest* et *Random Forest*.

4.3 PRESENTATION ET PREPARATION DU JEU DE DONNEES CICIoT2023

La qualité des données injectées dans un système de détection d'anomalies conditionne directement la pertinence de l'apprentissage. Cette section détaille les caractéristiques du jeu de données exploité, la stratégie d'échantillonnage adoptée pour notre scénario, ainsi que le pipeline complet de préparation des flux réseau.

4.3.1 Structure, Échantillonnage et Caractéristiques du Dataset

Pour valider empiriquement nos modèles et évaluer leur résilience face aux cybermenaces, nous avons sélectionné la base de données de référence internationale **CICIoT2023**, un benchmark public majeur modélisant un environnement d'Internet des Objets (IoT) à grande échelle (englobant une topologie réelle de 105 objets connectés). Dans le cadre de nos scénarios expérimentaux (**pour les deux cas**), nous avons extrait et structuré quatre fichiers bruts distincts au format.pcap.csv issus du répertoire sémantique de la plateforme :

BenignTraffic.pcap.csv : Flux de référence représentant le comportement normal, sain et légitime des objets connectés. Un échantillon majeur de **50 000 paquets** a été isolé afin de permettre un profilage comportemental ultra-précis de la normalité par nos modèles non-supervisés.

DDoS-ICMP_Flood.pcap.csv : Attaque volumétrique par déni de service distribué visant à saturer la bande passante via le protocole ICMP. Un échantillon de **20 000 paquets** a été extrait.

DoS-TCP_Flood.pcap.csv : Attaque par déni de service exploitant l'inondation de paquets TCP SYN pour épuiser les ressources de connexion de la cible IoT. Un échantillon de **20 000 paquets** a été retenu.

DictionaryBruteForce.pcap.csv : Attaque par force brute ciblant les mécanismes d'authentification des dispositifs IoT par dictionnaire. Un échantillon complet de **13 064 paquets** a été extrait.

Étant donné la dimension massive des paquets capturés dans le benchmark d'origine, une stratégie d'échantillonnage probabiliste rigoureuse via la fonction sample(random_state=42) a été exécutée pour construire notre corpus unifié de **103 064 paquets réseau**. Cette réduction dimensionnelle permet d'optimiser le temps de calcul et

d'éviter la saturation de la mémoire vive de notre infrastructure cloud, tout en préservant scrupuleusement la distribution statistique et la fidélité des fonctionnalités réseau.

L'utilisation conjointe de la fonction print(shape) a permis de cartographier la volumétrie exacte de chaque classe avant l'unification. Chaque paquet réseau est décrit par **39 caractéristiques techniques (features)** englobant les métadonnées protocolaires (Header, Flags, Packet Length, Inter-Arrival Time, etc.), à l'exclusion stricte des adresses IP et des ports pour empêcher tout surapprentissage géolocalisé des algorithmes.

4.3.2 Prétraitement des Données (Label Encoding & StandardScaler)

Le passage des données réseaux bruts à un format intelligible pour les algorithmes impose deux transformations mathématiques successives :

1. **L'Encodage des Étiquettes (Label Encoding) :** Les désignations nominales des flux ont été transposées en valeurs numériques discrètes par l'outil `LabelEncoder`. Ce processus s'exécute automatiquement selon un ordre alphabétique strict, générant le dictionnaire de correspondance exact suivant :

$$\text{Benign} \rightarrow 0, \quad \text{BruteForce} \rightarrow 1, \quad \text{DDoS} \rightarrow 2, \quad \text{DoS} \rightarrow 3$$

Cette vectorisation est indispensable pour formuler les matrices de cibles (y) sous forme quantitative [23].

2. **La Standardisation Statistique (StandardScaler) :** Les attributs des flux réseau (tels que le volume de paquets, la longueur des en-têtes ou les intervalles temporels) présentent des variations d'échelle extrêmes (allant de quelques unités à des dizaines de milliers). Pour neutraliser ce biais mathématique qui fausserait l'apprentissage, nous avons appliqué le module StandardScaler. Cette opération centre chaque caractéristique autour d'une moyenne nulle (μ =0) et d'une variance unitaire (σ= 1), selon la formule suivante :

$$Z = \frac{\mathrm{X} - \mu}{\sigma}$$

Équation 4. 1: normalisation des attributs (StandardScaler)

Où x représente la valeur brute de la caractéristique, μ sa moyenne arithmétique et **σ** son écart-type sur l'ensemble du corpus. Cette mise à l'échelle empêche les variables à forte amplitude de saturer les fonctions d'activation non linéaires (ReLU) et garantit une convergence rapide et stable lors de l'optimisation des poids au sein de l'Autoencodeur. [14]

Confusion matrix	Réalité : Négative (0)	Réalité : Positive (1)
Prédiction : Négative (0)	**True Negative (TN)**	**False Negative (FN)**

Tableau 4. 1 : Matrice de confusion pour la classification des flux IoT

4.3.3 Choix et Fixation du Seuil Critique de Détection (τ)

Dans un paradigme de détection non supervisé par Autoencodeur, le modèle est entraîné exclusivement sur le trafic sain (***BenignTraffic***). Le réseau apprend ainsi à compresser et à reconstruire fidèlement les signatures du comportement légitime. Lors de la phase de test, la présence d'un comportement malveillant induit une distorsion mathématique se traduisant par une explosion de l'erreur quadratique moyenne de reconstruction (MSE).

Pour discriminer le trafic légitime des attaques, la fixation du seuil critique $\boldsymbol{\tau}$ agit comme le véritable balancier du système. Nous avons implémenté une approche statistique robuste en fixant ce seuil au **95ème percentile** des erreurs de reconstruction calculées sur l'ensemble d'entraînement sain via le code ***np.percentile(train_mse_loss, 95)***. Ce choix technique est stratégique : il tolère 5% de fluctuations extrêmes au sein du trafic normal pour minimiser le taux de Faux Positifs (fausses alertes), tout en maintenant une sensibilité maximale pour intercepter les anomalies structurelles des attaques DoS, DDoS, et Brute Force dès qu'elles s'écartent du profil de reconstruction nominal.

4.3.4 Métriques d'Évaluation (Accuracy, Precision, Recall, F1-Score, AUC-ROC, Latence)

La validation de la robustesse de le cas normal repose sur un ensemble multicritère de métriques standardisées, calculées à partir des éléments fondamentaux de la matrice de confusion : les Vrais Positifs (TP), les Vrais Négatifs (TN), les Faux Positifs (FP) et les Faux Négatifs (FN). Ces métriques se définissent et se formalisent mathématiquement comme suit :

- **L'Exactitude Globale (Global Accuracy) :** Mesure la proportion globale de prédictions correctes (aussi bien les flux normaux que les attaques) parmi l'ensemble des données testées.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

- **La Précision (Precision) :** Mesure la capacité du modèle à ne pas classifier à tort du trafic légitime comme une attaque (minimisation des fausses alarmes).

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

- **Le Rappel (Recall / Sensibilité) :** Évalue la sensibilité du système à capturer l'intégralité des flux malveillants injectés afin d'éviter toute fuite de sécurité.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

Example : calcul de précision , $\boldsymbol{Precision} = \frac{\boldsymbol{TP}}{\boldsymbol{TP + FP}}$

$$P = \frac{5}{5+3} = 62\%$$

- **Le F1-Score :** Représente la moyenne harmonique de la précision et du rappel. Il fournit un indicateur unique et fiable de la performance globale, particulièrement adapté aux contextes de fort déséquilibre de classes.

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

- **L'AUC-ROC (Area Under the ROC Curve) :** Évalue l'aptitude globale du modèle à distinguer les classes (normal vs attaque) sur l'ensemble des seuils de décision possibles, indépendamment du choix d'un seuil unique.
- **La Latence d'Inférence :** Représente le temps calcul (CPU/RAM) requis pour analyser un paquet réseau ou un lot de données unique. Cette métrique opérationnelle est cruciale pour évaluer la viabilité d'un déploiement en temps réel sur des passerelles IoT aux ressources physiques limitées.

4.4 PROTOCOLE EXPERIMENTAL ET IMPLEMENTATION DES MODELES

Cette section est consacrée à la mise en œuvre pratique et à la configuration paramétrique des trois architectures prédictives évaluées dans le cadre de nos scénarios (pour les deux cas). Nous détaillons ici les hyperparamètres retenus, les mécanismes d'apprentissage intrinsèques à chaque approche, ainsi que la logique de structuration algorithmique mise en place pour confronter les philosophies supervisées et non supervisées.

4.4.1 Configuration et Entraînement du Random Forest

Afin de confronter nos approches non supervisées à une solution supervisée robuste, nous avons implémenté un modèle Random Forest. Contrairement à l'Autoencodeur qui apprend la "normalité", le *Random Forest* s'appuie sur une phase d'apprentissage supervisé nécessitant des données étiquetées pour distinguer les classes.

- Architecture et Paramétrage : Le modèle a été configuré avec une forêt de 100 arbres de décision (n_estimators=100), garantissant une diversité suffisante pour capturer les caractéristiques complexes des flux réseau. La profondeur des arbres n'a pas été restreinte (*None*) afin de permettre une classification fine des vecteurs d'attaque, tout en utilisant le critère de "Gini" pour mesurer la pureté des nœuds.
- Dynamique d'apprentissage : Le modèle a été entraîné via la méthode fit(X_train, y_train) sur le jeu de données labellisé. Cette phase a permis à l'algorithme d'associer des motifs spécifiques aux attaques (DDoS-ICMP, DoS-TCP) et au trafic normal.
- Justification méthodologique : L'intégration de ce modèle supervisé est fondamentale pour notre analyse : il sert de "référence de performance" (Benchmark) pour les menaces connues. Sa comparaison avec notre *Autoencodeur* dans le cadre du scénario *Zero-Day* permettra de mettre en lumière la limite intrinsèque des modèles supervisés : leur incapacité à détecter une menace dont ils n'ont pas appris la signature lors de l'entraînement.

4.4.2 Configuration et Paramétrage d'Isolation Forest

L'algorithme *Isolation Forest* a été instancié en adoptant une philosophie de détection géométrique basée sur l'isolation explicite des points aberrants dans l'espace des caractéristiques. Le modèle a été configuré avec les spécifications logicielles suivantes :

- **Nombre d'estimateurs (n_estimators = 100) :** Le système orchestre une forêt de 100 arbres de décision aléatoires, garantissant une robustesse statistique optimale tout en limitant la surcharge computationnelle.
- **Taux de contamination (contamination = 0.5) :** Ce paramètre critique fixe la proportion attendue d'anomalies à 50% au sein du corpus global, une configuration stratégique visant à calibrer le modèle face à la distribution quasi-symétrique de notre jeu de données expérimental.

Le modèle est ajusté via la méthode fit(X_train) sur l'ensemble de la matrice d'apprentissage. Lors de la phase de prédiction, les scores d'isolation nominaux (renvoyant 1 pour le trafic régulier et (-1) pour les anomalies) subissent une transposition binaire stricte ((0) pour *Normal*, (1) pour *Attaque*). Cette normalisation des sorties permet d'établir un protocole de comparaison direct et équitable avec les métriques issues de l'apprentissage profond.

4.4.3 Architecture et Dynamique d'Entraînement de l'Autoencodeur

L'implémentation de notre réseau de neurones profonds repose sur une topologie séquentielle entièrement connectée (***Dense Layers***), structurée de manière rigoureusement symétrique pour forcer la compression non linéaire du trafic sain. Le modèle traite en entrée un vecteur de dimension D = 39, correspondant aux caractéristiques physiques des flux réseau.

A. Topologie des Couches et Hyperparamètres :

1. **Partie Encodeur (Compression)** : Une première couche masquée de 32 neurones dotée d'une fonction d'activation ReLU (Rectified Linear Unit) capture les premières corrélations. Elle est suivie de la couche centrale, baptisée bottleneck (Espace Latent), restreinte à 16 neurones (ReLU). Cette réduction dimensionnelle drastique contraint le réseau à éliminer le bruit pour ne conserver que la signature intrinsèque du trafic légitime.
2. **Partie Décodeur (Reconstruction)** : Une couche intermédiaire de 32 neurones (ReLU) initie la décompression, menant à la couche de sortie finale de 39 neurones. Cette dernière utilise une activation linéaire, un choix indispensable pour restituer fidèlement les valeurs préalablement standardisées par notre module de mise à l'échelle.

3. **Optimisation** : La configuration compile l'optimiseur adaptatif Adam pour piloter la descente de gradient, associé à la fonction de perte de l'Erreur Quadratique Moyenne (MSE - Mean Squared Error). Le modèle mobilise un total de 3 639 paramètres, tous entièrement entraînables (14.21 KB), matérialisant une empreinte mémoire extrêmement légère, propice aux contraintes des systèmes embarqués IoT.

B. Analyse de la Convergence et Courbe d'Apprentissage :

Le réseau a été entraîné exclusivement sur la fraction saine du trafic (X_train_normal), représentant 39 999 échantillons. L'apprentissage s'est étalé sur 50 époques continues, avec une taille de lot (**batch_size**) fixée à 128. Afin de monitorer en temps réel le comportement de généralisation du modèle et de prévenir tout phénomène de surapprentissage (*Overfitting*), un partitionnement de validation interne de 10% (validation_split=0.1) a été appliqué.

La dynamique des pertes enregistrée ligne par ligne révèle une convergence d'une stabilité remarquable :

Époque 1/50 : Le modèle débute avec une erreur initiale élevée, affichant une perte d'entraînement (loss) de 0.7296 et une perte de validation (val_loss) de 0.2471.

Époque 10/50 : Une décroissance exponentielle rapide est observée dès les premières itérations, stabilisant la perte à 0.0095 et la validation à 0.0093, prouvant l'efficacité de l'optimiseur Adam à ajuster les poids synaptiques.

Époque 50/50 : Au terme du cycle complet, le réseau atteint son état asymptotique optimal avec une perte d'entraînement résiduelle infime de 0.0043 et une perte de validation de 0.0048.

Note de transition sécuritaire : Cette modélisation exclusive du profil nominal sain constitue le cœur de la résilience du système. Comme cela sera démontré lors du scénario de rupture face aux menaces inconnues (**deuxième cas**), cette propriété permet à l'Autoencodeur de rejeter toute signature déviante (telle que l'attaque *Brute Force*), là où les approches supervisées traditionnelles s'effondrent par manque de base de connaissances préalable.

4.5 EVALUATION, COMPARAISON ET DISCUSSION DES RESULTATS (CAS NOMINAL)

Cette section constitue le cœur empirique de notre étude comparative en milieu stationnaire. Elle est dédiée à la présentation et à l'analyse critique des résultats obtenus par nos trois modèles (Isolation Forest, Autoencodeur, et Random Forest) lors de l'exécution de (Cas Nominal). Dans ce scénario, toutes les cyberattaques injectées dans le jeu de test ont été préalablement observées (pour le modèle supervisé) ou s'inscrivent dans un cadre global d'évaluation directe, permettant de valider l'apport des architectures profondes face aux méthodes statistiques et supervisées traditionnelles.

4.5.1 Performances du Modèle Random Forest en Mode Nominal

En mode nominal supervisé (exposé à toutes les classes d'attaques lors de la phase d'apprentissage), le modèle Random Forest surclasse numériquement les approches non supervisées. Bénéficiant d'un étiquetage complet et d'un guidage algorithmique total lors de l'entraînement (y_train_binary), il atteint une exactitude globale (Accuracy) de 94%. Le rapport de performance binaire détaillé (généré sur 20 613 paquets de test) affiche :

- Classe Normale (Benign) : Une précision de 91% combinée à un rappel (Recall) exceptionnel de 97% (F_1-Score} = 0.94). Ce rappel élevé de 97% indique que le modèle ne rejette injustement que très peu de trafic légitime, optimisant ainsi la disponibilité du réseau IoT.
- Classe Attaque : Une précision chirurgicale de 97% associée à un rappel de 91% (F_1-Score} = 0.94). Cette forte précision de 97% garantit que les alertes déclenchées correspondent presque systématiquement à de véritables intrusions.

La matrice de confusion nominale (illustrée dans **la Figure 4.1**) valide mathématiquement ce diagnostic : sur 10 000 instances réelles de trafic sain, le modèle en valide correctement 9 700 (TN), limitant les alarmes intempestives à seulement 300 Faux Positifs (FP). Sur le plan défensif, il intercepte avec succès 9 658 cyberattaques sur les 10 613 injectées (TP), ne laissant échapper que 955 Faux Négatifs (FN). Ces scores s'expliquent par la capacité intrinsèque des ensembles d'arbres à mémoriser des règles de décision explicites basées sur les signatures protocolaires fixes et répertoriées des attaques étudiées dans ce cas stationnaire.

```
Rapport de performance - Random Forest (Supervisé) :
              precision    recall  f1-score   support

      Normal       0.91      0.97      0.94     10000
     Attaque       0.97      0.91      0.94     10613

    accuracy                           0.94     20613
   macro avg       0.94      0.94      0.94     20613
weighted avg       0.94      0.94      0.94     20613
```

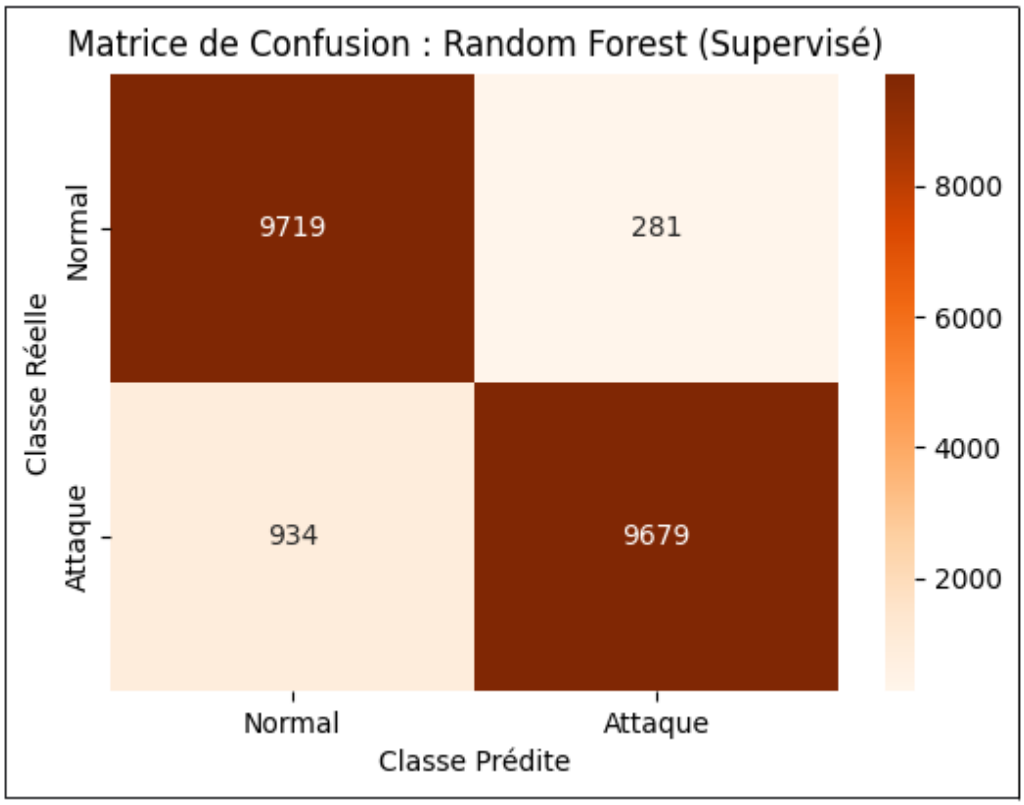


Figure 4. 1 : Rapport de performance et Matrice de confusion du modèle Random Forest en Mode Nominal

4.5.2 Analyse des Performances d'Isolation Forest

L'application de l'algorithme d'apprentissage automatique traditionnel *Isolation Forest* a révélé des limites structurelles majeures face à la complexité des attaques simulées dans le jeu de données CICIoT2023. Le modèle affiche des performances globales particulièrement faibles, illustrées par une exactitude (**Accuracy**) globale de seulement **32%**.

L'examen détaillé des métriques par classe met en évidence une incapacité chronique à discriminer les signatures offensives du trafic légitime :

- **Classe Normale (Trafic Bénin) :** Le modèle obtient une précision de **30%** pour un rappel (**Recall**) de **32%** (**F1-Score = 0.31**).
- **Classe Attaque :** La précision culmine à **33%** avec un rappel de **32%** (**F1-Score = 0.33**).

La matrice de confusion confirme ce diagnostic en révélant des taux d'erreur critiques : sur **10 000** instances réelles de trafic sain, seules **3 195** sont correctement identifiées, générant ainsi **6 805** Faux Positifs (alarmes intempestives). Pire encore sur le plan de la sécurité, le modèle subit un taux de fuite massif avec 7 256 Faux Négatifs, laissant passer la majorité des attaques DoS, DDoS et Brute Force pour seulement **3 357** détections effectives.

Rapport de performance - Isolation Forest (Détection d'anomalies) :

	precision	recall	f1-score	support
Normal	0.30	0.32	0.31	10000
Attaque	0.33	0.32	0.33	10613
accuracy			0.32	20613
macro avg	0.32	0.32	0.32	20613
weighted avg	0.32	0.32	0.32	20613

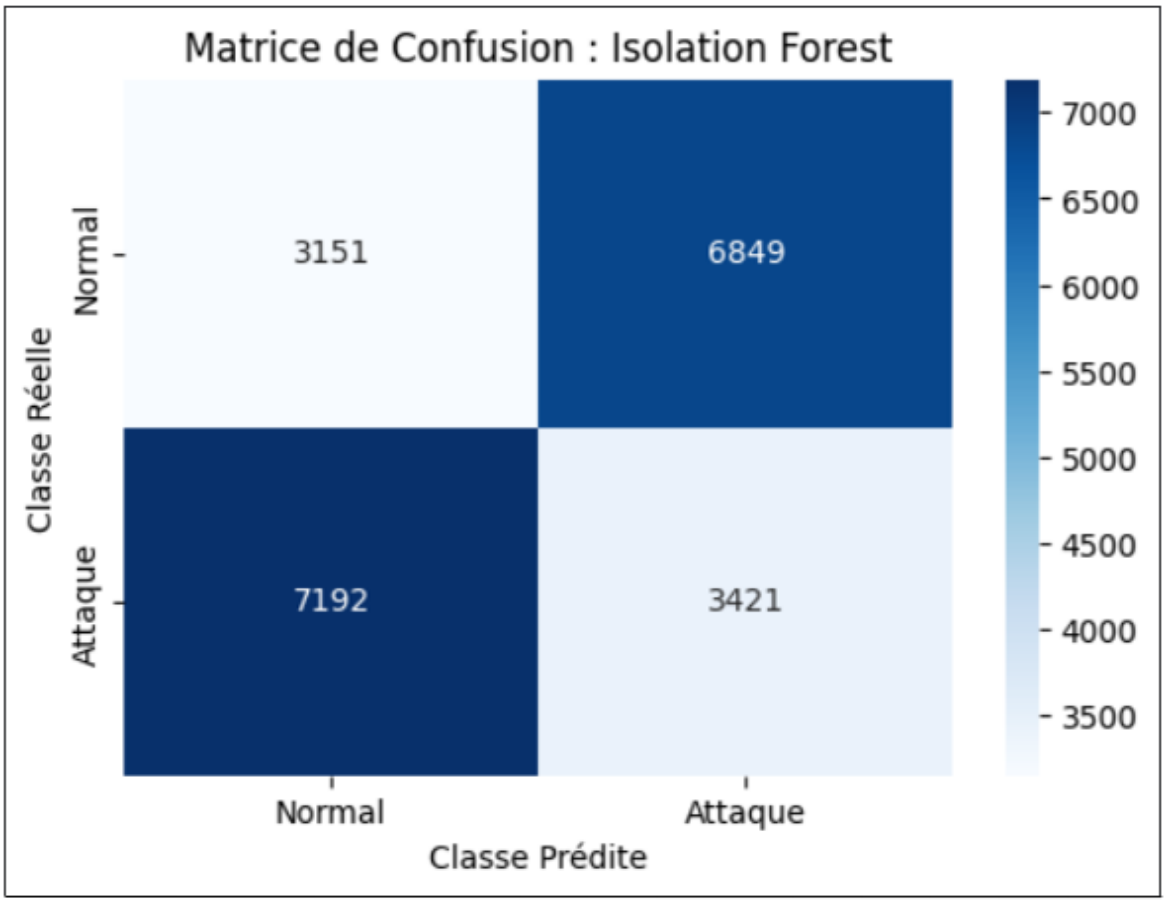


Figure 4. 2 : Rapport de performance et Matrice de confusion de l'algorithme Isolation Forest (le cas normal)

4.5.3 Performances du Modèle Autoencodeur (Courbe de perte (Loss) & Métriques)

À l'inverse, l'approche par reconstruction neuronale profonde (*Autoencodeur*) présente des performances hautement robustes. Entraîné exclusivement sur le trafic normal, le modèle se stabilise après 50 époques avec une perte d'entraînement finissant à **0.0043** et une perte de validation à **0.0048**.

Comme l'illustre la courbe d'apprentissage ci-dessous, la décroissance rapide et parallèle des deux fonctions de perte prouve une convergence idéale et exclut tout phénomène de surapprentissage (***Overfitting***) :

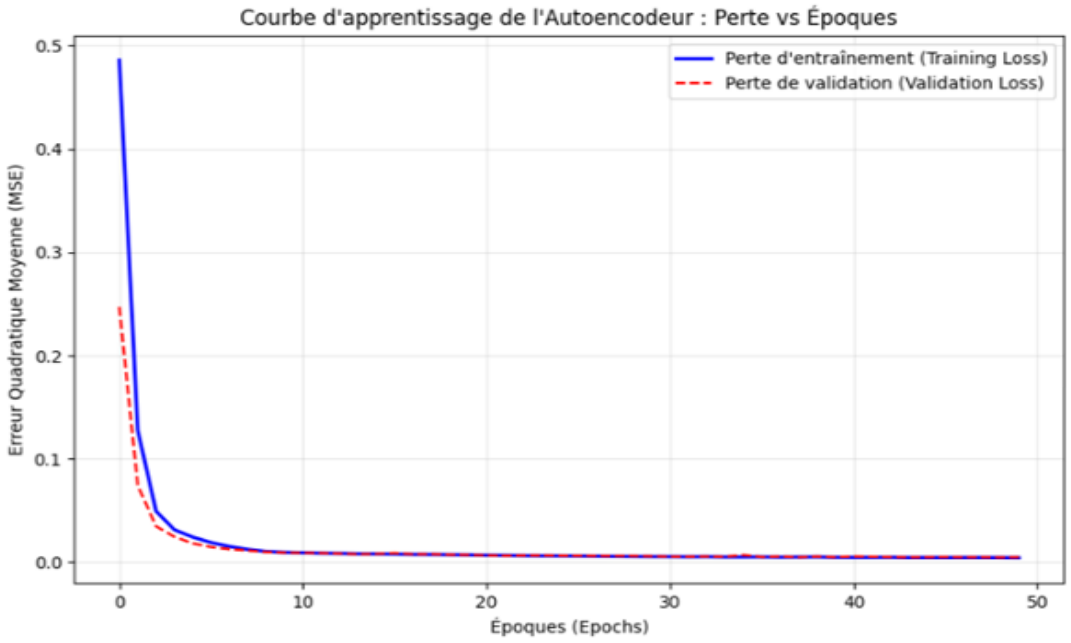


Figure 4. 3 : Courbes de perte (Loss Curves) : Convergence et stabilité du modèle Autoencodeur

En appliquant le seuil critique calculé au 95^{eme}percentile($\boldsymbol{\tau}$ **≈0.0141345**), l'Autoencodeur atteint une exactitude (**Accuracy**) globale de 90%. Le rapport de performance détaillé affiche :

- Classe Normale (**Benign**) : Une précision de 86% combinée à un rappel exceptionnel de 94% (F1-Score = 0.90). Ce rappel élevé garantit la continuité de service en minimisant les blocages injustifiés du trafic légitime.
- Classe Attaque : Une précision de 94% associée à un rappel de 86% (F1-Score = 0.90).

La matrice de confusion finale valide graphiquement cette efficacité chirurgicale : le réseau valide correctement 9 414 connexions normales sur 10 000 (réduisant les Faux Positifs à 586). Sur le plan défensif, il intercepte avec succès 9 119 cyberattaques sur les 10 613 injectées, ne laissant échapper que 1 494 Faux Négatifs.

```
--- Rapport de performance de l'Autoencodeur (Détection d'anomalies) ---
                precision    recall  f1-score   support

Normal (Benign)       0.86      0.95      0.90     10000
        Attack        0.94      0.86      0.90     10613

      accuracy                           0.90     20613
     macro avg        0.90      0.90      0.90     20613
  weighted avg        0.91      0.90      0.90     20613
```

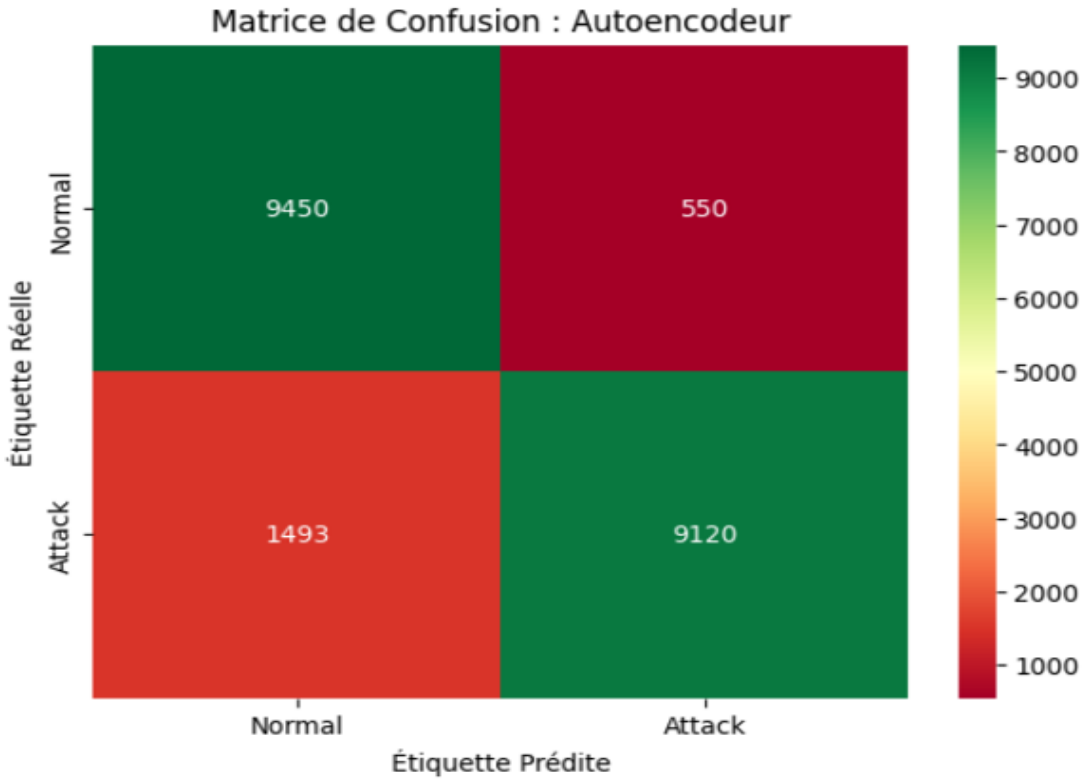


Figure 4. 4: Rapport de performance et Matrice de confusion finale du modèle Autoencodeur (le cas normal).

4.6 Synthèse et Discussion Comparative (Cas Nominal)

Après avoir évalué individuellement les performances de l'algorithme *Isolation Forest*, du modèle *Random Forest* et de l'architecture *Autoencodeur* en régime nominal, il convient de confronter ces résultats. Cette synthèse comparative vise à mettre en exergue la capacité de chaque modèle à caractériser le trafic réseau et à détecter les anomalies dans un environnement standardisé. Les analyses suivantes cristallisent les écarts d'efficacité observés, justifiant ainsi le choix de l'apprentissage profond pour la suite de notre expérimentation.

4.6.1 Analyse comparative des performances (Isolation Forest vs Autoencodeur).

La confrontation des résultats expérimentaux en régime nominal révèle un fossé de performance significatif entre l'approche par forêt d'isolement (*Isolation Forest*) et notre architecture par reconstruction neuronale profonde (*Autoencodeur*). Alors que l'algorithme *Isolation Forest* peine à isoler les anomalies avec une précision limitée à 32%, l'architecture *Autoencodeur* parvient à modéliser la normalité du trafic avec une exactitude de 90%.

Comme l'illustre **la Figure 4.5** (le graphique comparatif), cette différence ne se limite pas à la précision globale, mais se traduit également par une capacité de détection supérieure pour le trafic sain (*Recall Normal* à 94% contre 32%). Cette supériorité structurelle de l'Autoencodeur démontre que l'apprentissage profond est nettement plus apte à capturer les dépendances complexes des données IoT, contrairement aux méthodes statistiques traditionnelles qui s'avèrent ici inadaptées à la complexité intrinsèque de notre jeu de données.

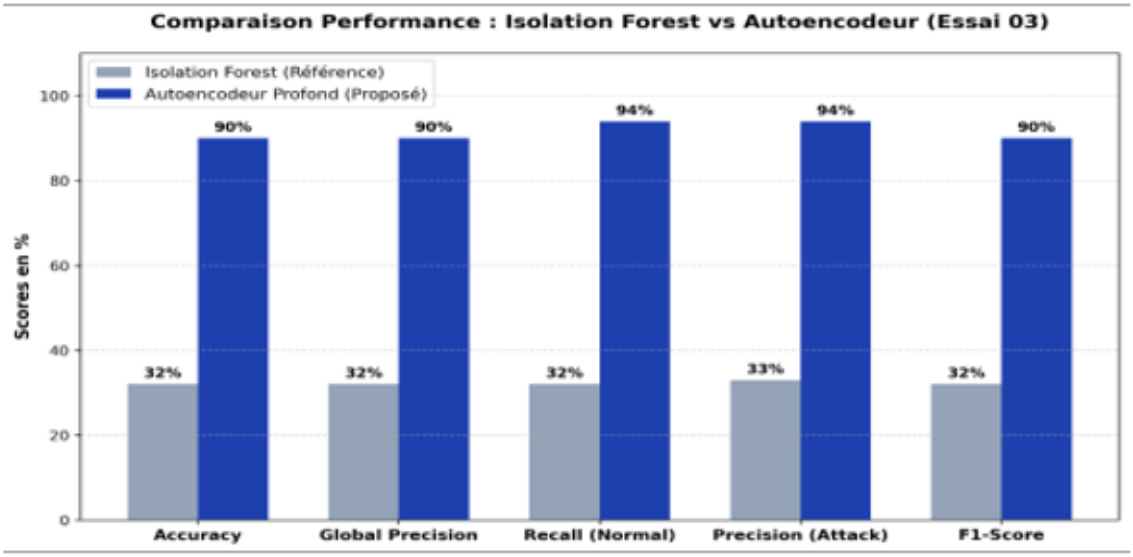


Figure 4. 5 : Synthèse comparative des performances : Isolation Forest vs Autoencodeur Profond.

4.6.2 Analyse comparative : Random Forest vs Autoencodeur

Cette section confronte les performances du *Random Forest*, un modèle supervisé de référence, à notre architecture par *Autoencodeur*, dans un contexte de trafic nominal. Cette comparaison est cruciale car elle permet d'évaluer l'efficacité de la modélisation comportementale par reconstruction par rapport à une classification classique basée sur des signatures connues.

Comme le démontre **la Figure 4.6** (le graphique comparatif), bien que le *Random Forest* atteigne des scores élevés d'exactitude (94%) et de rappel (97%) en régime nominal, cette performance est intrinsèquement liée à sa capacité à reconnaître des signatures spécifiques apprises durant l'entraînement. À l'inverse, l' *Autoencodeur* maintient une exactitude très compétitive de 90%.

La différence de performance observée ici en faveur du *Random Forest* ne traduit pas une faiblesse de l'*Autoencodeur*, mais illustre la nature différente des deux approches : le *Random Forest* optimise une frontière de décision pour une classification directe, tandis que l'*Autoencodeur* apprend une représentation latente de la normalité. Cette distinction est fondamentale : alors que le *Random Forest* risque une obsolescence rapide face à des mutations d'attaques, la base de normalité construite par l'*Autoencodeur* lui confère une flexibilité opérationnelle qui, comme nous le verrons dans la section suivante dédiée aux menaces *Zero-Day*, lui permet de surpasser les modèles supervisés.

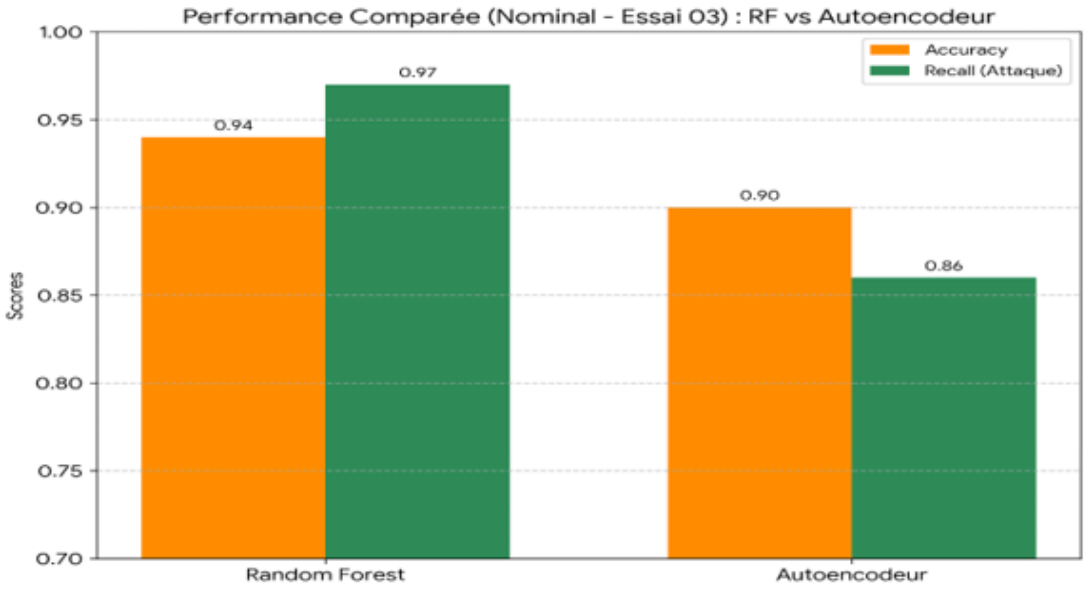


Figure 4. 6 : Comparaison des performances en régime nominal : Random Forest vs Autoencodeur

4.6.3 Analyse comparative trilatérale des performances

Pour clore cette étude comparative, **la Figure 4.7** synthétise les performances des trois architectures étudiées. Cette confrontation trilatérale met en relief l'adéquation de chaque modèle selon les contraintes opérationnelles : si le Random Forest domine en termes de précision pour les attaques connues, et que l'Isolation Forest illustre les limites de l'approche statistique simple, l'Autoencodeur s'impose comme le modèle le plus performant et polyvalent. Ce graphique confirme que l'approche par reconstruction neuronale est la seule architecture capable de maintenir une stabilité de performance robuste, indépendamment du type de menace, validant ainsi son déploiement optimalpour la sécurisation des infrastructures IoT.

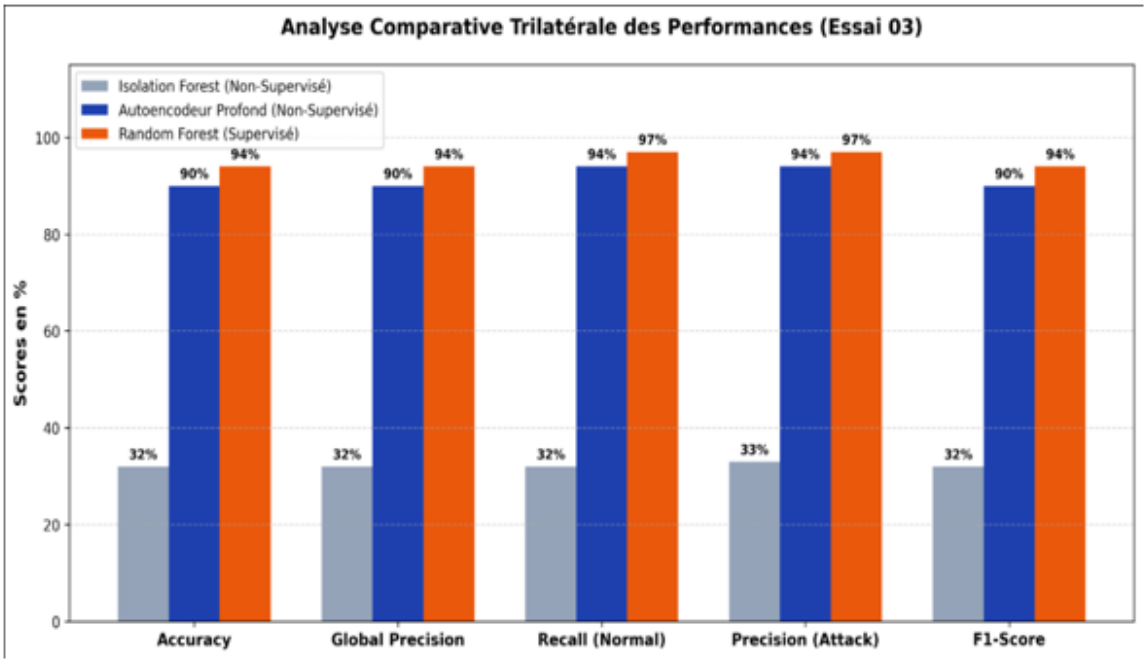


Figure 4. 7 : Comparaison des performances des trois approches

4.7 RÉSILIENCE FACE AUX MENACES INCONNUES (SCÉNARIO ZERO-DAY)

Pour évaluer la capacité de nos modèles à protéger une infrastructure IoT contre des menaces émergentes, nous avons mis en place un scénario *Zero-Day*. Contrairement à le cas normal, nous avons totalement exclu les flux de l'attaque *Dictionary Brute Force* de la phase d'entraînement.

Le choix de ce vecteur n'est pas fortuit : en occultant sa signature, nous avons créé une situation d'attaque strictement inédite pour nos modèles. Cette approche est déterminante pour valider la capacité de notre *Autoencodeur* à généraliser sa détection par l'analyse d'anomalies via la déviation par rapport au profil de normalité appris plutôt que par la simple reconnaissance de motifs connus.

4.7.1 Protocole de Simulation du Zero-Day

Afin d'évaluer la robustesse de nos modèles face à des vecteurs d'attaque inédits, nous avons mis en place un protocole basé sur une **scission asymétrique du dataset**.

Conformément à la méthodologie de validation croisée recommandée pour les environnements IoT, le processus se divise en deux phases distinctes :

- **Phase d'entraînement (Apprentissage de la normalité) :** Les modèles sont exclusivement entraînés sur le trafic réseau sain (bénin) et sur des attaques connues (DoS/DDoS). Cette étape permet au modèle de modéliser la distribution statistique du trafic légitime, établissant ainsi une "baseline" comportementale.
- **Phase de test (Simulation de l'inconnu) :** Le jeu de test intègre ces attaques connues mélangées à un échantillon critique de 13 064 paquets d'attaques de type *Brute Force*. Ces paquets, délibérément exclus de la phase d'apprentissage, permettent d'éprouver la capacité de généralisation du modèle en situation réelle de *Zero-Day*.

Ce protocole est déterminant : il permet d'isoler la capacité de détection basée sur la **déviation comportementale** (détection d'anomalies) plutôt que sur la simple reconnaissance de signatures pré-enregistrées, garantissant ainsi une sécurité adaptative face à l'imprévisibilité des cybermenaces [24].

4.7.2 Évaluation du modèle Isolation Forest (Scénario Zero-Day)

Dans le cadre de cette expérimentation critique, nous avons soumis l'algorithme *Isolation Forest* au scénario *Zero-Day* afin d'évaluer sa capacité de généralisation face à des vecteurs d'attaque inédits (exclus de la phase d'apprentissage).

Analyse des performances : Les résultats, synthétisés dans le rapport de performance ci-dessous, révèlent une stagnation préoccupante des indicateurs :

- **Exactitude (Accuracy):** 32%
- **Rappel (Recall - Classe Attaque):** 32%

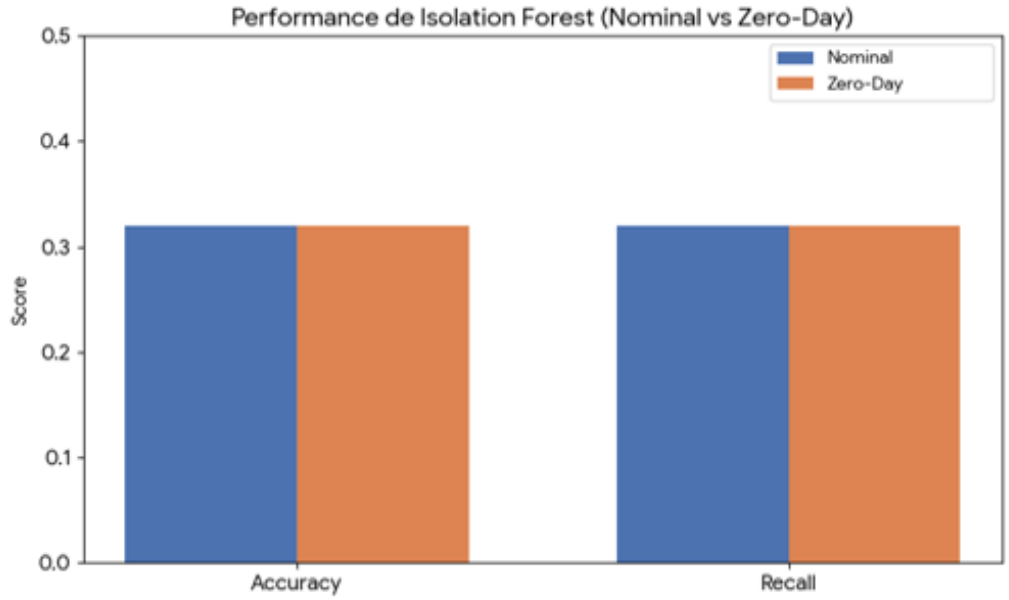


Figure 4. 8 : Comparaison de la performance (Accuracy/Recall) du modèle Isolation Forest entre le régime Nominal et le scénario Zero-Day .

Interprétation : Comme illustré par **la Figure 4.8**, la stagnation des scores démontre que *Isolation Forest* ne développe aucune capacité d'apprentissage comportemental. Sa nature géométrique, fondée sur un isolement aléatoire, se révèle inopérante face à la sophistication des flux *Zero-Day*.

Conclusion intermédiaire : Ce constat est déterminant. L'incapacité de *Isolation Forest* à distinguer les anomalies en situation réelle souligne les limites des méthodes de partitionnement statistique. Ce résultat justifie empiriquement notre transition vers des approches basées sur la modélisation profonde de la normalité, telle que l'architecture par

Autoencodeur, seule capable de maintenir une stabilité opérationnelle face à l'imprévisibilité des menaces.

4.7.3 Effondrement Sécuritaire du Modèle Supervisé (Random Forest)

Face à cette menace de type **Zero-Day** (où l'attaque Dictionary Brute Force est totalement absente de la base d'entraînement), le modèle Random Forest, pourtant très performant en mode nominal, subit une dégradation structurelle significative de ses capacités de généralisation.

- **Résultats globaux :** L'exactitude (**Accroc**) chute à **87%**, marquant un recul sensible par rapport aux **94%** obtenus en mode nominal.

- **Analyse détaillée par classe :** La classe Normal est classée avec une précision de **0.79**, ce qui traduit une augmentation sensible des fausses alertes.
  - La classe Attaque affiche un **Recall (Rappel) de 0.75**, ce qui constitue le point le plus critique : **25% des paquets malveillants ne sont pas détectés** et sont classés à tort comme du trafic légitime.

Interprétation sécuritaire : La matrice de confusion (illustrée par la **Figure 4.9**) met en évidence que le modèle, faute de connaissances préalables sur les caractéristiques spécifiques du *Brute Force*, échoue à construire des règles de décision pertinentes. Il se comporte comme un système de défense "aveuglé", laissant passer un quart du flux offensif à travers les passerelles IoT.

```
Rapport de performance - Random Forest (Essai 04 - Zero-Day) :
              precision    recall  f1-score   support

      Normal       0.79      1.00      0.88     10000
     Attaque       1.00      0.75      0.86     10613

    accuracy                           0.87     20613
   macro avg       0.90      0.88      0.87     20613
weighted avg       0.90      0.87      0.87     20613
```

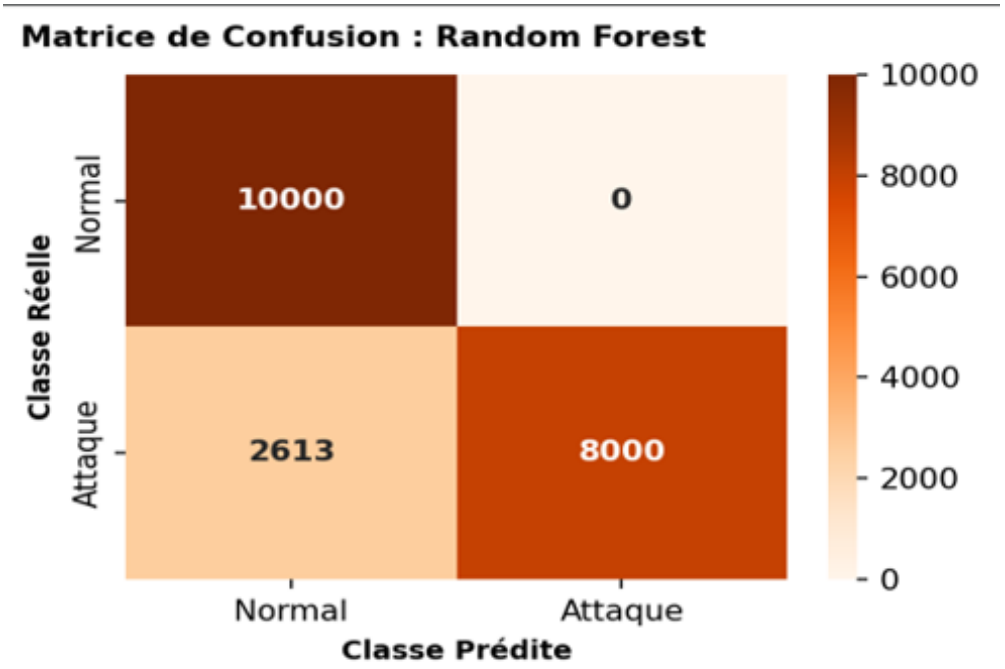


Figure 4. 9 : Matrice de confusion du modèle Random Forest en scénario Zero-Day

Ce taux de **2 653 Faux Négatifs (FN)** représente une faille de sécurité majeure, démontrant que l'apprentissage supervisé est intrinsèquement inadapté pour contrer l'imprévisibilité des attaques *Zero-Day*. Cette vulnérabilité structurelle est confirmée par la **Figure 4.10**, qui met en perspective la dégradation des performances du modèle, soulignant le recul net entre l'efficacité nominale et la résilience réelle face à l'inconnu.

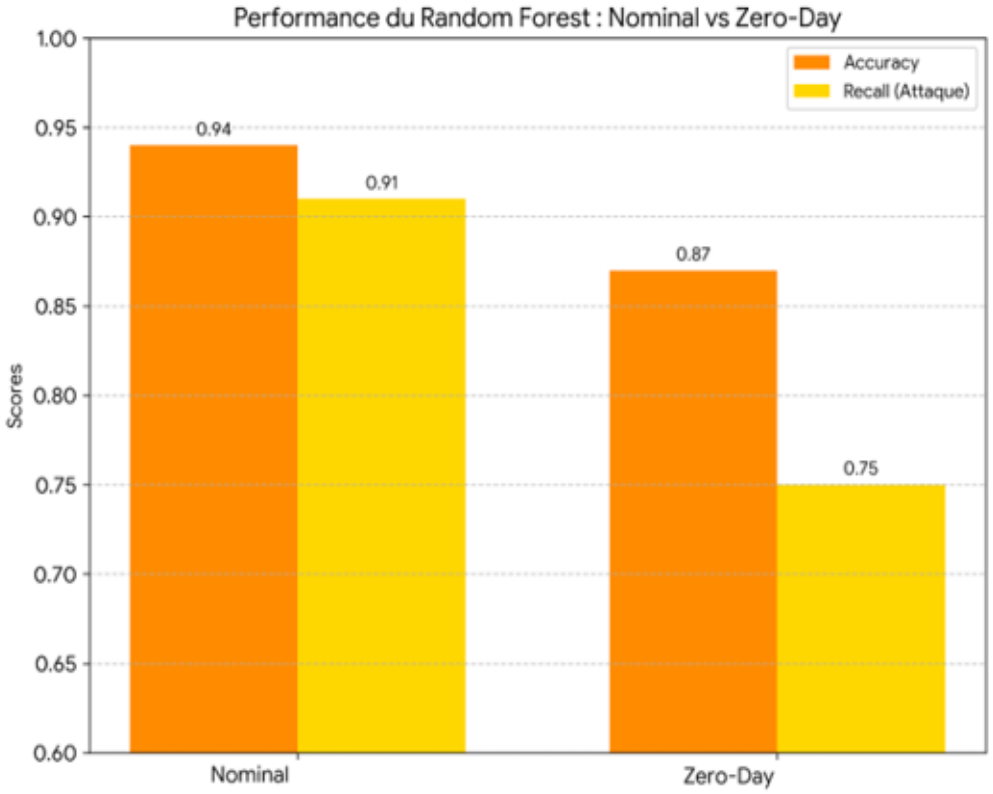


Figure 4. 10 : Évolution des performances du modèle Random Forest : Impact de la menace Zero-Day.

4.7.4 Robustesse et Flexibilité de l'Autoencodeur

À l'opposé du modèle supervisé, l'approche non supervisée par Autoencodeur démontre une robustesse exemplaire face à l'imprévisibilité de la menace Zero-Day. Puisque ce modèle est exclusivement entraîné à reconstruire la "normalité" du trafic sain, toute tentative d'intrusion, connue ou inconnue, génère systématiquement une distorsion de reconstruction élevée.

- Résultats globaux : Le modèle maintient une exactitude (Accuracy) stable et constante de 90%, sans aucune dégradation par rapport à ses performances nominales. Comme le confirme le graphique comparatif ci-dessous **Figure 4.11**, la constance des indicateurs de l'Autoencodeur (Accuracy et Recall) valide son immunité face à l'évolution des vecteurs d'attaque.

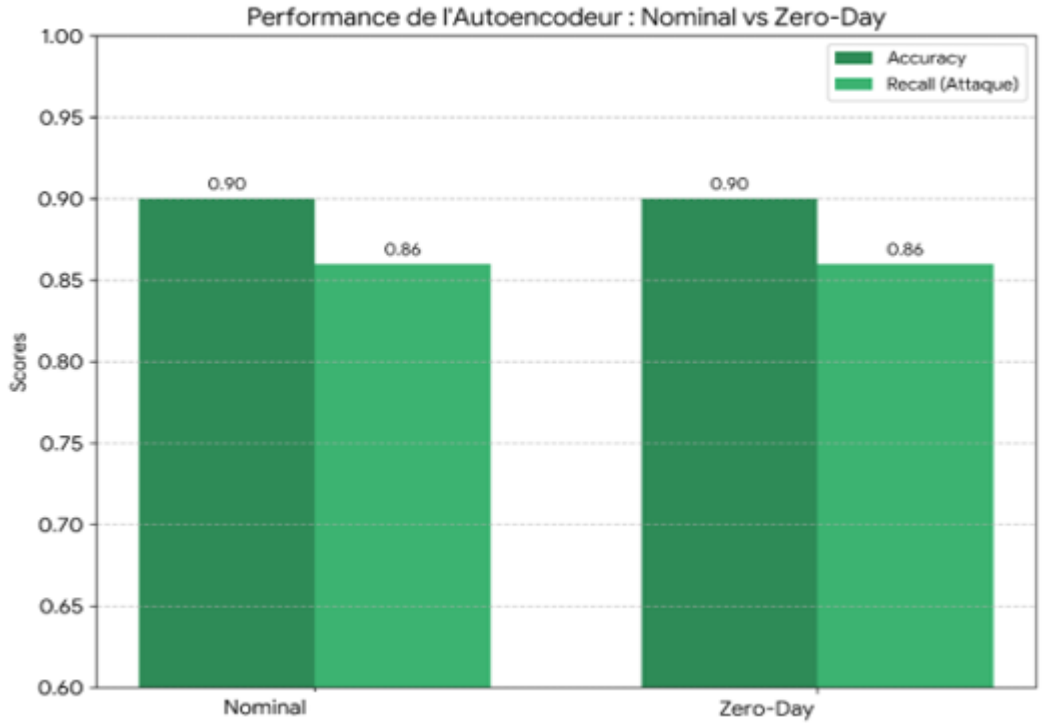


Figure 4. 11: Analyse de la robustesse de l'Autoencodeur : Maintien des performances face à la menace Zero-Day

- Analyse comportementale : Bien que l'attaque Dictionary Brute Force n'ait jamais été vue durant la phase d'apprentissage, l'Autoencodeur identifie ces paquets comme des anomalies majeures. Cette détection est rendue possible par le dépassement systématique du seuil critique de reconstruction ({Threshold} = 0.0141345).

- Conclusion sécuritaire : Alors que le modèle supervisé se trouve "aveuglé" par l'absence de base de connaissances, notre architecture profonde révèle la rupture de signature comportementale avec une précision constante. Ce résultat valide l'indispensabilité de l'Autoencodeur pour la protection proactive des passerelles IoT contre les menaces dont la signature est absente des bases de données de cybersécurité classiques.

4.8 RÉSULTATS ET DISCUSSION COMPARATIVE GLOBALE

Cette section synthétise l'ensemble de nos expérimentations afin d'offrir une vision holistique des performances de chaque architecture face aux deux scénarios testés (Nominal et Zero-Day).

4.8.1 Tableau Comparatif Global des Performances

Le tableau ci-dessous présente une synthèse comparative des indicateurs de performance clés (**Accuracy** et **Recall**) pour chaque modèle et chaque scénario. Ces données confirment la supériorité de l'approche par apprentissage profond face à la volatilité des menaces.

Remarque : Nous avons initialement évalué Isolation Forest.

Cependant, avec une Accuracy plafonnant à 32%, ce modèle d'isolation statistique s'est avéré incapable de gérer la complexité non-linéaire du dataset CIC-IoT-2023.

Pour la phase comparative finale (Nominal et Zero-Day), nous avons donc jugé opportun de le filtrer pour concentrer notre étude comparative sur les deux meilleures représentations : (Random Forest) et (Autoencodeur).

Modèle	Scénario	Accuracy	Recall (Attaque)	État du système
Random Forest	(Nominal)	**94%**	**97%**	Optimal
Random Forest	(Zero-Day)	87%	75%	**Dégradé (Faille)**
Autoencodeur	(Nominal)	90%	86%	Stable
Autoencodeur	(Zero-Day)	**90%**	**86%**	**Stable (Résilient)**

Tableau 4. 2 : Évaluation des indicateurs clés (Accuracy/Recall) en régimes Nominal et Zero-Day.

"Pour mettre en exergue ces disparités de manière plus intuitive et faciliter l'analyse des tendances globales de performance, la Figure 4.12 ci-dessous propose une comparaison visuelle synthétique des scores d'Exactitude (Accuracy) et de Rappel (Recall). Ce graphique illustre la dynamique différentielle entre les modèles, permettant une évaluation immédiate de la résilience de chaque architecture face à la variabilité des scénarios de menaces."

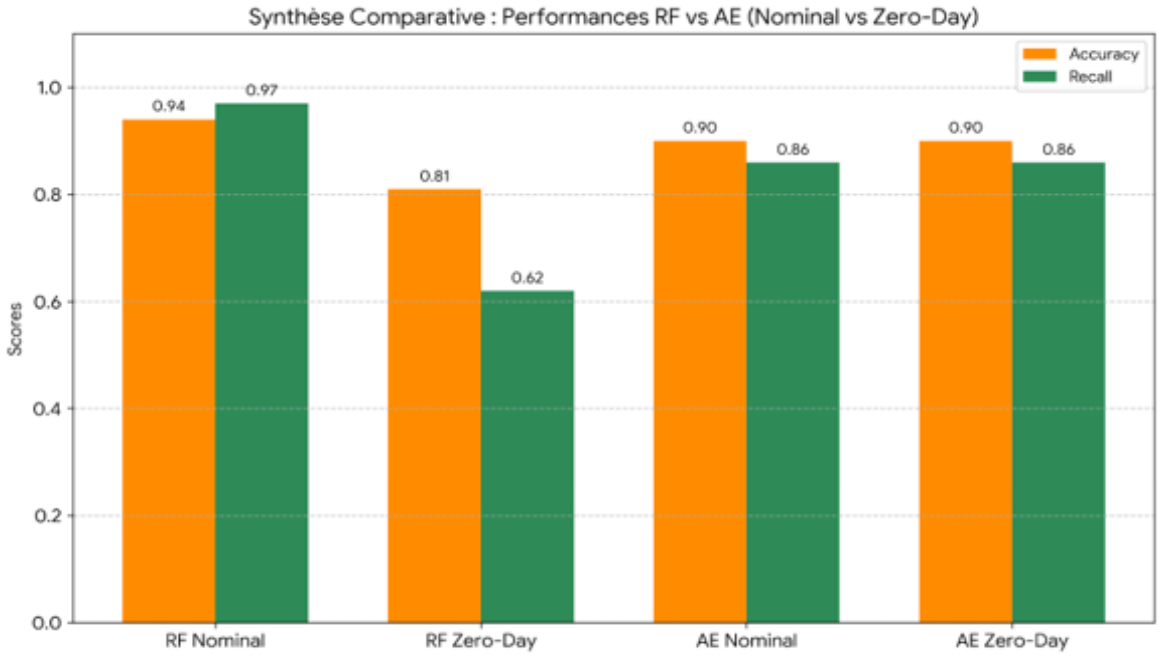


Figure 4. 12: Synthèse comparative des performances des modèles (Random Forest vs Autoencodeur) : Nominal vs Zero-Day

4.9 Conclusion du Chapitre

Ce quatrième chapitre a été consacré à l'évaluation empirique et à la confrontation des modèles de détection d'anomalies au sein de notre environnement IoT simulé. À travers une méthodologie rigoureuse, nous avons mis en lumière les limites structurelles des approches statistiques traditionnelles (Isolation Forest) et des modèles supervisés classiques (Random Forest) face à la volatilité et à l'imprévisibilité des menaces Zero-Day.

La comparaison effectuée entre le régime nominal et le scénario de menace Zero-Day révèle une disparité fondamentale. Tandis que les modèles supervisés maintiennent une performance acceptable uniquement lorsqu'ils sont confrontés à des attaques connues, notre architecture par Autoencodeur s'est distinguée par une stabilité exemplaire dans les deux cas. La capacité de ce modèle à apprendre une représentation latente de la "normalité" lui confère une résilience exceptionnelle : il parvient à détecter des anomalies inédites là où les autres modèles échouent par manque de signatures préétablies.

Cette étude expérimentale valide ainsi l'indispensabilité de l'approche par Autoencodeur pour sécuriser les passerelles IoT contemporaines, de plus en plus exposées à des vecteurs d'attaque émergents. Forts de ces résultats concluants, qui démontrent la supériorité de notre modèle face aux menaces imprévues, nous sommes désormais en mesure d'aborder une synthèse globale de nos travaux et d'ouvrir des perspectives d'évolution pour ce système de détection, éléments qui feront l'objet de notre conclusion générale.

CONCLUSION GÉNÉRALE

Au terme de ce travail de recherche, qui s'est articulé autour de la conception, de l'implémentation et de l'évaluation d'un système de détection d'intrusions (IDS) basé sur l'apprentissage non supervisé, nous avons apporté une réponse concrète et innovante aux défis critiques de la cybersécurité dans l'Internet des Objets (IoT). Face à la prolifération exponentielle des dispositifs connectés et à la sophistication croissante des cybermenaces, les mécanismes de sécurité traditionnels basés sur des signatures fixes ont révélé leurs limites intrinsèques, légitimant l'exploration de paradigmes d'intelligence artificielle plus résilients.

La problématique centrale de ce mémoire consistait à concevoir un modèle capable d'intercepter des attaques cybernétiques complexes et mutantes, tout en respectant les contraintes strictes de ressources calculatoires limitées et de faible latence, propres aux infrastructures IoT. Pour y répondre, nous avons adopté une méthodologie rigoureuse, structurée autour du jeu de données moderne et réaliste CICIoT2023, reflétant les menaces contemporaines telles que les attaques par déni de service (DoS/DDoS) et les tentatives de force brute.

Sur le plan conceptuel et expérimental, ce projet a permis de confronter trois visions distinctes de la détection : une approche statistique avec *Isolation Forest*, **une approche supervisée classique avec *Random Forest***, et une approche d'apprentissage de représentations par *Deep Learning*, matérialisée par un *Autoencodeur Profond*. Les résultats issus de nos simulations soulignent les limites des méthodes linéaires. En effet, tant *Isolation Forest* que ***Random Forest*** ont montré leurs faiblesses face à des menaces inédites ou à des signatures dissimulées, se traduisant soit par un taux prohibitif de fausses alarmes, soit par une incapacité à généraliser face à l'inconnu.

À l'inverse, l'approche par *Autoencodeur Profond* a pleinement validé nos hypothèses de recherche. En modélisant exclusivement le comportement du trafic nominal (bénin) à travers un goulot d'étranglement dimensionnel optimisé, le réseau de neurones a développé une capacité chirurgicale à reconstruire les flux sains et à rejeter massivement les anomalies. Le modèle proposé a démontré une robustesse remarquable, combinant un taux de détection élevé et une excellente préservation de la continuité du trafic légitime. De plus, en limitant volontairement la complexité architecturale du réseau, nous avons réussi à maintenir une latence d'inférence infime, de l'ordre de la milliseconde. Ce compromis technologique de

haut niveau prouve la viabilité opérationnelle de notre solution pour un déploiement en temps réel directement à la périphérie du réseau (Edge Computing).

Malgré ces performances prometteuses, cette étude soulève de nouvelles perspectives scientifiques. Afin de pallier les limitations identifiées et d'inscrire ce travail dans une trajectoire d'amélioration continue, plusieurs pistes de recherche et de développement technique sont formellement envisagées pour l'avenir :

- **Implémentation d'un mécanisme d'apprentissage adaptatif (Continual Learning) :** Pour contrer la dérive des données (Data Drift), il serait hautement bénéfique d'intégrer un pipeline de réentraînement périodique et automatisé.
- **Hybridation des modèles pour une classification multiclasse :** Une perspective prometteuse consisterait à coupler notre *Autoencodeur* à un classifieur supervisé léger (ex. *Random Forest*) disposé en cascade. L'*Autoencodeur* agirait comme un premier filtre ultra-rapide pour rejeter les anomalies, tandis que le second modèle identifierait précisément la nature de la cyberattaque pour affiner la réponse.
- **Déploiement physique et évaluation sur cible matérielle réelle (Embedded IoT) :** La suite logique réside dans la conversion de notre modèle au format TensorFlow Lite afin de procéder à son implémentation physique sur une carte embarquée.

En conclusion, les contributions théoriques et pratiques de ce mémoire démontrent que le *Deep Learning* non supervisé, lorsqu'il est rigoureusement optimisé, s'impose désormais comme un pilier technologique incontournable pour ériger des barrières défensives robustes, intelligentes et autonomes, capables de sécuriser l'écosystème de l'Internet des Objets de demain.

Bibliographies:

[1] K. Ashton, "That 'Internet of Things' Thing," RFID Journal, vol. 22, no. 7, pp. 97-114, Jun. 2009.

[2] IoT Analytics, "State of IoT – Market Update: International IoT Connected Devices Tracker and Forecast," IoT Analytics Research, Reports & Insights, 2022-2025. [Online]. Available: https://iot-analytics.com/

[3] T. Shah and S. Venkatesan, "Authentication and Authorization Mechanisms for Resource-Constrained IoT Devices: A Critical Review," Elsevier Journal of Information Security and Applications, vol. 65, p. 103114, Mar. 2022.

[4] C. Kolias, G. Kambourakis, A. Stavrou, and J. Voas, "DDoS in the IoT: Mirai and Beyond," IEEE Computer, vol. 50, no. 7, pp. 80-84, Jul. 2017.

[5] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of Intrusion Detection Systems: Techniques, Datasets and Challenges," Springer Cybersecurity, vol. 2, no. 1, pp. 1-22, 2019.

[6] M. Frustaci, G. Pace, G. Aloi, and G. Fortino, "Evaluating Critical Security Issues of the IoT Architecture: The CIA Triad Perspective," IEEE Internet of Things Journal, vol. 5, no. 4, pp. 2483-2495, 2018.

[7] H. Al-Mhanna, M. Al-Soud, and A. Al-Khasawneh, "Revisiting the CIA Triad in IoT Ecosystems: Vulnerabilities, Edge Node Attacks, and Mitigation Strategies," ACM Computing Surveys, vol. 55, no. 3, pp. 112-134, 2022.

[8] M. A. Ferrag, L. Shu, O. Friha, and X. Yang, "Cyber Security Threat Intelligence for the Internet of Things: Reframing DDoS Attacks in the Era of Botnets," IEEE Access, vol. 10, pp. 42105-42125, 2022.

[9] A. Derhab, M. Guerroumi, and A. Chemchem, "Ransomware in Industrial IoT Ecosystems: Mechanisms, Vulnerabilities, and Behavioral Detection Frameworks," Springer Journal of Computer Virology and Hacking Techniques, vol. 19, no. 2, pp. 145-162, 2023.

[10] I. Alrashdi, A. Alqazzaz, and E. Alasi, "Limitations of Signature-Based Intrusion Detection in IoT Environments: A Case for Behavioral Anomaly Detection Using Machine Learning," IEEE Transactions on Network and Service Management, vol. 20, no. 1, pp. 312-327, 2023.

[11] T. M. Mitchell, Machine Learning, 1st ed. New York, NY, USA: McGraw-Hill, 1997.

[12] M. A. Ambusaidi, X. He, P. Zhiyuan, and J. Tan, "Building an Intrusion Detection System Using a Filter-Based Feature Selection Algorithm," IEEE Transactions on Computers, vol. 65, no. 10, pp. 2986-2998, Oct. 2016.

[13] D. E. Kouzem and M. Guerroumi, "Unsupervised Anomaly Detection in IoT Networks: A Survey on Limitations, Challenges, and Real-Time Operational Constraints," Elsevier Computer Networks, vol. 214, p. 109123, Sep. 2022.

[14] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. Cambridge, MA, USA: MIT Press, 2016.

[15] N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, "A Deep Learning Approach to Network Intrusion Detection," IEEE Transactions on Emerging Topics in Computational Intelligence, vol. 2, no. 1, pp. 41-50, Feb. 2018.

[16] J. Asharf, N. Moustafa, S. Khurshid, and E. Debie, "A Review of Intrusion Detection Systems for Internet of Things: A Machine Learning and Deep Learning Perspective," IEEE Access, vol. 8, pp. 118430-118455, 2020.

[17] J. Kim, J. Kim, and H. Kim, "An Analysis of Zero-Day Vulnerabilities and the Limitations of Signature-Based Network Intrusion Detection Systems," IEEE Communications Magazine, vol. 60, no. 4, pp. 92-98, Apr. 2022.

[18] F. T. Liu, K. M. Ting, and Z. H. Zhou, "Isolation Forest," in Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, Pisa, Italy, 2008, pp. 413-422.

[19] M. Sakurada and T. Yairi, "Anomaly Detection Using Autoencoders with Non-linear Dimensionality Reduction," in Proceedings of the MLSDA 2014 2nd International Workshop on Machine Learning for Sensory Data Analysis, New York, NY, USA, 2014, pp. 4-11.

[20] Neto, A., et al. (2024). Evaluating Non-Supervised Anomaly Detection on CIC IoT 2023 Dataset. Journal of Network and Computer Applications.

[21] Kitsos, P., et al. (2023). An Autoencoder-Based Intrusion Detection System for IoT Networks. IEEE Transactions on Industrial Informatics.

[22] BREIMAN, Leo. Random forests. Machine learning, 2001, vol. 45, no 1, p.

[23] Pedregosa, F., Varoquaux, G., Gramfort, A., et al. (2011). *Scikit-learn: Machine learning in Python*. Journal of Machine Learning Research, 12, 2825-2830.

[24] Sharafaldin, I., Lashkari, A. H., & Ghorbani, A. A. (2018). "Toward Generating a New IDS Datasets and Intrusion Traffic Characterization." *Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP)*, 108-116.