



République Algérienne Démocratique et Populaire



Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

**Université des Sciences et de la Technologie Houari Boumediene**

Faculté d’Informatique

Département des systèmes informatiques.

## Mémoire de Master

Filière : Informatique

**Spécialité : Sécurité des Systèmes Informatiques (SSI)**

Thème

**Smart power grid protection based on blockchain  
and machine learning**

**Sujet Proposé par :**

Prof. Guerroumi Mohamed

Dr. Derhab Abdelouahid

Dr. Aoufi Souhila

**Soutenu le : 28/06/2022**

**Présenté par :**

Saoudi Yanis

Mellah Mouloud

**Devant le jury composé de :**

**Pr A.BELKHIR  
Mme M.CHENAIT**

**Président  
Membre**

Binôme N° : *SSI\_I\_001/2022*

## ***Remerciements***

*Nous tenons à exprimer notre profonde gratitude à Monsieur GUERROUMI MOHAMED, Monsieur DERHAB ABDELOUAHID et Madame AOUIFISOUHILA pour nous avoir fait confiance et pour nous avoir encouragés tout au long de ce projet. On les remercie également pour leurs disponibilités, leurs conseils judicieux, leurs critiques constructives, et leurs suggestions pertinentes.*

*Nos vifs remerciements vont ensuite à nos très chers parents pour leur soutien moral et leur encouragement.*

*Nos remerciements vont aussi aux membres de jury pour avoir accepté de juger ce travail.*

*Finalement, on remercie du plus profond de nos cœurs toute personne ayant contribué de près ou de loin à l'aboutissement de ce travail.*

## Résumé

Notre projet se consacre à l'étude, la conception et la réalisation d'un système intégrant le machine learning dans la technologie blockchain afin de renforcer la protection d'un réseau intelligent et détecter les anomalies. D'une part, ce système permettra la protection des entités contre l'attaque de l'injection de fausses données (FDIA), et de l'autre, il fournira un mécanisme de détection de vol d'énergie ou de manipulation anormale de l'énergie consommée.

En venir à bout de ces deux attaques est l'objectif majeur de notre solution.

Pour atteindre ces objectifs, on a eu recours à différentes technologies modernes afin de créer ce système qui garantit la détection du vol d'énergie et la dissipation de l'FDIA comme l'utilisation d'un réseau de neurones convolutifs pour le développement d'un modèle deep learning, l'utilisation de la blockchain, et l'utilisation d'un mécanisme de calcul de réputation pour un nœud.

**Mots clés :** Réseau intelligent, Vol d'énergie, Attaque d'injection de fausses données, Intelligence artificielle, Deep learning, Blockchain, Ethereum, Python, Solidity

## Abstract

Our project is dedicated to the study, design and realization of a system integrating machine learning with blockchain technology in order to enhance the protection of a smart grid and detect anomalies. On the one hand, this system will allow the protection of entities against False Data Injection attack (FDIA), and on the other hand, it will provide a detection mechanism of energy theft or abnormal manipulation of the energy consumed.

Overcoming these two attacks is the major objective of our solution.

To achieve these objectives, we have used different modern technologies to create this system that guarantees the detection of energy theft and the dissipation of FDIA such as the use of a convolutional neural network for the development of a deep learning model, the use of the blockchain, and the use of a reputation calculation mechanism for a node.

**Key words :** Smart grid, Energy theft, False data injection attack, Artificial intelligence, Apprentissage profond, Blockchain, Ethereum, Python, Solidity

## ملخص

مشروعنا مخصص لدراسة وتصميم وتنفيذ نظام يدمج التعلم الآلي في تقنية blockchain من أجل تعزيز حماية الشبكة الذكية واكتشاف الحالات الشاذة. هذا النظام سيسمح بحماية البيانات ضد هجوم حقن البيانات الخاطئة (FDIA) من جهة ، ومن جهة أخرى ، سيوفر آلية للكشف عن سرقة الطاقة أو التلاعيب الغير الطبيعي بالطاقة المستهلكة. النغلب على هذين الهجومين هو الهدف الرئيسي لحلنا.

لتحقيق هذه الأهداف ، تم استخدام العديد من التقنيات الحديثة لإنشاء هذا النظام الذي يضمن الكشف عن سرقة الطاقة وتبييد هجوم حقن البيانات الخاطئة (FDIA) مثل استخدام الشبكة العصبية التلاميفية ونموذج التعلم العميق ، استخدام تقنية blockchain ، و استخدام آلية حساب سمعة العقدة.

**الكلمات الدليلية :** الشبكة الذكية، سرقة الطاقة، هجوم حقن البيانات المزيف، ذكاء اصطناعي، التعلم العميق، Solidity، Python، Ethereum، Blockchain

# Table de Matières

Introduction générale .....	1
<b>Chapitre 1 : Etat de l'art.....</b>	<b>3</b>
Introduction.....	4
1.    Réseaux électriques intelligents (Smart Grid) .....	4
1.1.    Définition des Smart Grids.....	4
1.2.    Architecture des Smart grids .....	4
1.3.    Les principales composantes d'un Smart Grid.....	6
1.3.1.    Les compteurs intelligents .....	6
1.3.2.    Les systèmes de communication : The Smart Metering Networks.....	7
1.3.3.    Les systèmes de réseau électrique.....	9
1.4.    La Sécurité dans les Smart Grids .....	11
1.4.1.    Les attaques sur les smart metering networks.....	12
1.4.2.    Les exigences de sécurité dans les smart metering network .....	13
2.    False data injection attack (FDIA) et vol d'énergie .....	15
2.1.    False data injection .....	15
2.2.    Le vol d'énergie .....	16
2.3.    La sécurité du système électrique.....	17
2.4.    Classification FDIA basée sur le système cible .....	17
2.5.    Impact de False data injection attack et le vol d'énergie .....	18
Conclusion .....	19
<b>Chapitre 2 : Blockchain et Intelligence Artificielle.....</b>	<b>20</b>
Introduction.....	21
1.    Les contre-mesures basées sur la blockchain.....	21
1.1.    La technologie de la blockchain .....	21
1.1.1.    L'arbre de Merkle .....	22
1.1.2.    Concept de fonctionnement de la blockchain .....	23
1.1.3.    Les mécanismes de consensus .....	23
1.1.4.    Les catégories de la blockchain .....	26
1.2.    Sécurité des smart grids basée sur la blockchain distribuée .....	26
1.2.1    Passage au système décentralisé de réseau intelligent.....	27
1.2.2    Motivations de l'application de la blockchain dans le paradigme SG .....	28

1.3 Protection des données distribué basé sur la blockchain contre FDIA .....	30
1.3.1 La Reconfiguration de réseau .....	30
1.3.2 Mécanisme de fonctionnement .....	31
1.3.3 Mécanisme de Consensus .....	32
1.4. La gestion décentralisée de données multi-chaînes pour les systèmes d'alimentation .	33
1.4.1 La reconfiguration de réseau.....	33
1.4.2 Mécanisme de Fonctionnement .....	34
1.4.3 Mécanisme de Consensus .....	34
1.5. A Novel Multitier Blockchain Architecture to Protect Data in Smart Metering Systems .....	35
1.5.1 La reconfiguration de réseau.....	35
1.5.2 Mécanisme de fonctionnement .....	37
1.5.3 Mécanisme de Consensus .....	39
1.6. Tableau comparatif des solutions trouvées .....	39
2. Détection de menaces avec l'intelligence artificielle.....	40
2.1. Définition .....	40
2.2. Les Modèles Deep Learning .....	41
2.2.1. Convolutional Neural Network (Réseau de neurones à convolution) .....	41
2.2.2. Recurrent Neural Network (Réseau de neurones récurrent) .....	41
2.2.3. Generative Adversarial Networks (Réseaux antagonistes génératifs) .....	41
2.2.4. Deep Reinforcement Learning (Apprentissage par renforcement profond) .....	42
2.2.5. Geometric Deep Learning (Apprentissage profond géométrique).....	42
2.3. Protection contre le vol d'énergie basé sur l'IA .....	42
2.3.1. Analyse basée sur la modélisation ARIMA et validation des relevés de consommations. ....	42
2.3.2. Algorithme CPBETD (Consumption pattern-based energy theft detector).....	43
2.3.3. Algorithme GBTD (Gradient boosting theft detector).....	44
2.3.4. Analyse basée sur des caractéristiques comportementales .....	45
2.3.5. Analyse basée sur le modèle hybride CNN-RF ( Convolutional Neural Network and Random Forest ) .....	46
2.3.6. Analyse basée sur le modèle hybride CNN-SVM (Convolutional Neural Network and Support vector machine) .....	47
3. Revue et comparaison de quelques recherches sur la combinaison de la blockchain et le deep learning .....	47
3. 1. A Privacy-Preserving Framework based Blockchain and Deep Learning for Protecting Smart Power Networks .....	48

3. 2. DeepCoin: A Novel Deep learning and Blockchain-based Energy Exchange Framework for Smart Grids .....	48
3.3 Blockchain-enabled Data Collection and Sharing for Industrial IoT with Deep Reinforcement Learning .....	49
3.4 Decentralized firewall for malware detection.....	49
3.5. When Machine Learning Meets Blockchain: A Decentralized, Privacy-preserving and Secure Design .....	49
3.6. BlockDeepNet: A Blockchain-Based Secure Deep Learning for IoT Network .....	49
3.7. Comparaison entre les solutions .....	50
3.8. Synthèse .....	51
Conclusion .....	52
<b>Chapitre 3 : Conception d'une contre-mesure de détection FDIA basée Deep learning et Blockchain.</b> .....	<b>53</b>
Introduction.....	54
1. Architecture générale AMI .....	54
2. Vecteurs d'attaque .....	55
2.1 Exemple d'une Attaque FDIA interne .....	56
2.2 Exemple d'une Attaque FDIA externe .....	57
3. Architecture de la solution proposée.....	57
Étape 1 : Récoltes des mesures par la blockchain au niveau NAN .....	59
Etape 2 : Validation d'un bloc et le calcul de la réputation au niveau NAN .....	63
1. Réputation .....	63
2. Structure de bloc .....	64
3. Processus de consensus PoT .....	65
Etape 3 : Détection de vol d'énergie au niveau DC.....	76
1. Choix de Dataset .....	76
2. Choix du modèle IA utilisé .....	77
2.1. Analyse des données et prétraitement.....	78
2.2. Génération d'ensembles de données d'entraînement et de test.....	79
2.3. Classification à l'aide du modèle CNN .....	79
Etape 4 : Calcule de la nouvelle réputation .....	80
Etape 5 : Blockchain niveau WAN.....	81
1. Construction de donnée D.....	82
2. Structure de bloc .....	83
4. Discussion .....	83

4.1. Buts atteints.....	85
Conclusion .....	85
<b>Chapitre 4 : Réalisation et simulation.....</b>	<b>86</b>
Introduction.....	87
1.    Architecture de Simulation .....	87
2.    Environnement de travail .....	88
Python .....	88
Tensorflow .....	89
cuDNN (NVIDIA CUDA Deep Neural Network).....	89
Pandas .....	90
Scikit-learn.....	90
Ganache gui .....	91
Brownie.....	91
3.    Evaluation .....	92
3.1.    Modèle IA :.....	92
3.1.1.    Les métriques :.....	95
3.2.    Blockchain :.....	99
3.2.1.    Simulations, résultats, et analyses :.....	100
3.3.    Combinaison Blockchain et IA : .....	107
Conclusion générale et perspectives .....	112
Webographie .....	i
Bibliographie.....	ii
Annexe 1: Sous-systèmes de la sécurité du système électrique en ligne.....	i
Annexe 2 : CA Attack.....	i
Annexe 3 : Classification FDIA basée sur le système cible (Suite) .....	ii
3.3.1.    State estimation attack .....	ii
3.3.2.    Contingency Analysis Attack (CA attack).....	iv
3.3.3.    SCOPF Attack.....	iv
3.3.4.    SCED attack.....	v
3.3.5.    DSSE attack .....	v
Annexe 4 :L'algorithme Proof-of-Efficiency .....	vi

## Table de Figures

<b>Figure 1 :</b> Architecture des Smart Grids .....	6
<b>Figure 2 :</b> Exemple d'un réseau domestique (HAN) .....	7
<b>Figure 3 :</b> Exemple d'un building area network (BAN) .....	8
<b>Figure 4 :</b> Exemple d'un réseau de voisinage (NAN).....	8
<b>Figure 5 :</b> Relation entre WAN, HAN et NAN.....	9
<b>Figure 6 :</b> Architecture de réseau électrique .....	10
<b>Figure 7 :</b> Prérequis en sécurité des SMNs. ....	14
<b>Figure 8 :</b> Exemple d'un scénario d'attaque par injection de fausses données (FDIA).....	16
<b>Figure 9 :</b> Structure générale de la blockchain .....	22
<b>Figure 10 :</b> L'arbre de Merkle.....	22
<b>Figure 11 :</b> Principe de fonctionnement de la blockchain .....	23
<b>Figure 12 :</b> Réseau compteur-noeud .....	31
<b>Figure 13 :</b> Contenu des blocs et connexions des chaînes. ....	32
<b>Figure 14 :</b> Vue d'ensemble du réseau. ....	33
<b>Figure 15 :</b> Blockchain data structure. ....	34
<b>Figure 16 :</b> Architecture proposée basée sur quatre zones élémentaires de l'AMI. ....	36
<b>Figure 17 :</b> Le stockage des données, DB dans SA, SM et DC. ....	36
<b>Figure 18 :</b> Structure de données de la blockchain mise en œuvre au niveau du réseau HAN. ....	37
<b>Figure 19 :</b> Structure de données de la blockchain mise en œuvre dans le niveau NAN. ....	38
<b>Figure 20 :</b> Structure de données de la blockchain mise en œuvre au niveau du FAN/WAN	38
<b>Figure 21 :</b> L'intelligence artificielle. ....	40
<b>Figure 22 :</b> Flux de détection de vol d'énergie avec CNN-RF.....	46
<b>Figure 23 :</b> Architecture du modèle CNN-SVM.....	47
<b>Figure 24 :</b> Architecture général AMI .....	55
<b>Figure 25 :</b> Types d'attaques FDIA. ....	56
<b>Figure 26 :</b> Architecture Globale de la solution proposée. ....	58
<b>Figure 27 :</b> Chiffrement des mesures collectées dans smart meter.....	59
<b>Figure 28 :</b> Chiffrement du chiffré avec la clé publique du smart meter dans smart meter ...	60
<b>Figure 29 :</b> Hash du message M avec une fonction de hashage dans smart meter .....	60
<b>Figure 30 :</b> Signature de M avec clé privée dans smart meter.....	61
<b>Figure 31 :</b> Envoie de la signature et du signé du smart meter .....	61
<b>Figure 32 :</b> Déchiffrement du signé avec la clé publique du smart meter émetteur .....	62
<b>Figure 33 :</b> Hash du message M avec une fonction de hashage dans un nœud .....	62
<b>Figure 34 :</b> Vérification de l'égalité des deux hashs dans le nœud.....	62
<b>Figure 35 :</b> Structure d'un bloc .....	65
<b>Figure 36 :</b> Envoi d'un vote. ....	68
<b>Figure 37 :</b> Comparaison de performance entre plusieurs modèles de détection de vol d'énergie. ....	77
<b>Figure 38 :</b> Flux de détection de vol d'énergie avec CNN .....	80
<b>Figure 39 :</b> Structure d'un bloc niveau WAN.....	83
<b>Figure 40 :</b> Architecture de simulation .....	88

<b>Figure 41:</b> Architecture modèle CNN utilisé .....	93
<b>Figure 42 :</b> Comparaison d'Adam avec d'autres algorithmes d'optimisation .....	93
<b>Figure 43 :</b> Taux de prédiction en fonction de l'époque d'entraînement.....	94
<b>Figure 44 :</b> Taux de prédiction en fonction du nombre de filtres .....	95
<b>Figure 45 :</b> Prédiction/perte en fonction des époques d'entraînement.....	96
<b>Figure 46 :</b> Evaluation du modèle entraîné .....	96
<b>Figure 47 :</b> Matrice de confusion du modèle généré .....	97
<b>Figure 48 :</b> Temps d'exécution + résultat de la Prédiction Numéro 1 .....	98
<b>Figure 49 :</b> Temps d'exécution + résultat de la Prédiction Numéro 2 .....	98
<b>Figure 50 :</b> Temps d'exécution + résultat de la Prédiction Numéro 3 .....	98
<b>Figure 51 :</b> Temps d'exécution + résultat de la Prédiction Numéro 4 .....	98
<b>Figure 52 :</b> Temps d'exécution + résultat de la Prédiction Numéro 5 .....	99
<b>Figure 53 :</b> Temps de détections des nœuds malicieux par rapport à alpha .....	101
<b>Figure 54 :</b> Coût de communication .....	102
<b>Figure 55 :</b> Coût de Transactions au déploiement selon la taille de réseau .....	103
<b>Figure 56 :</b> Cout des transactions selon la taille du NAN.....	104
<b>Figure 57 :</b> Latence par cycle de consensus selon taille de réseau du NAN.....	105
<b>Figure 58 :</b> Latence par cycle d'exécution dans le réseau WAN .....	105
<b>Figure 59 :</b> La détection et l'élimination des nœuds validateurs malicieux .....	106
<b>Figure 60 :</b> La diminution de réputation dans l'attaque externe.....	107
<b>Figure 61 :</b> La diminution de réputation dans l'attaque interne .....	108
<b>Figure 62 :</b> La diminution de réputation dans l'attaque interne et externe.....	109
<b>Figure 63 :</b> Taux de détection de chaque solution .....	109
<b>Figure 64 :</b> Taux de détection selon la durée moyenne .....	110
<b>Figure 65 :</b> Fonction du State Estimation. ....	iii

# Liste des Tableaux

<b>Tableau 1 : Probabilité de l'attaque et la gravité associée.....</b>	<b>13</b>
<b>Tableau 2 : Systèmes et sous-systèmes d'un système.....</b>	<b>17</b>
<b>Tableau 3 : Classification de FDIA selon système cible.....</b>	<b>18</b>
<b>Tableau 4 : Une brève comparaison entre le Smart Grid et le Smart Grid décentralisé.....</b>	<b>28</b>
<b>Tableau 5 : Objectifs communs de sécurité, et la manière dont Blockchain peut aborder.....</b>	<b>29</b>
<b>Tableau 6 : Item et signification.....</b>	<b>35</b>
<b>Tableau 7 : Comparaison des solutions blockchain trouvées.....</b>	<b>39</b>
<b>Tableau 8 : Performance moyenne de détection et comparaisons de 5000 clients entre le classificateurs CPBETD et GBTD .....</b>	<b>45</b>
<b>Tableau 9 : Résumé des scores de classification de CNN-RF.....</b>	<b>47</b>
<b>Tableau 10 : Comparaison des solutions.....</b>	<b>51</b>
<b>Tableau 11 : Classification des attaques FDIA interne.....</b>	<b>56</b>
<b>Tableau 12 : Diminution de la réputation niveau NAN.....</b>	<b>63</b>
<b>Tableau 13 : Signification des attributs d'un bloc .....</b>	<b>65</b>
<b>Tableau 14 : Explication du détournement des attaques.....</b>	<b>84</b>
<b>Tableau 15 : Score de classification pour le modèle CNN généré .....</b>	<b>95</b>
<b>Tableau 16 : Résultat de la solution de l'apprentissage automatique contre les attaques internes.</b>	<b>99</b>
<b>Tableau 17 : Les paramètres utilisés dans l'évaluation de la blockchain.....</b>	<b>100</b>

## Liste des Acronymes

**ACE** : Area Control Error

**AGC** : Automatic Gain Control

**AMI** : Advanced Metering Infrastructure

**ARIMA** : Autoregressive Integrated Moving Average

**BAN** : Building Area Network

**BAS** : Blockchain Autonomy System

**BDD** : Bad Data Detection

**BS** : Base Station

**BSN** : BS Node

**CA** : Contingency Analysis

**CIA** : Confidentiality, Integrity and Availability

**CMP** : Node compromise Attack

**CNN** : Convolutional Neural Network

**CPBETD** : Consumption pattern-based energy theft detector

**DB** : DataBase

**DC** : Data Concentrator

**DER** : Distributed Energy Resources

**DL** : Deep learning

**DMS** : Distribution Management Systems

**DOS** : Denial Of Service

**DR** : Detection rate

**DRL** : Deep Reinforcement Learning

**DS** : Distribution System

**DSSE** : Distribution System State Estimation

**ED** : Economic Dispatch

**EMS** : Energy Management Systems

**ESB** : Electricity Supply Board

**EVD** : Eavesdropping Attack

**EW** : Event Window

**EWMA** : Exponentially Weighted Moving Average

**FDIA** : False Data Injection Attack

**FN** : False Negative

**FNR** : False Negative Rate

**FP** : False Positive

**FPR** : False Positive Rate

**GAN** : Generative Adversarial Networks

**GBC** : Gradients Boosting Classifiers

**GBTD** : Gradient boosting theft detector

**GPU** : Graphics Processing Unit

**GRD** : Gestionnaire de Réseau de Distribution

**GRU** : Gated Recurrent Unit

**GSM** : Global System for Mobile Communications

**HAN** : Home Area Network

**IA** : Intelligence artificielle

**IDS** : Intrusion Detection System

**IED** : Intelligent Electronic Device

**IMP** : Impersonation Attack

**IOT** : Internet Of Things

**LDOS** : Low-rate Denial Of Service

**LMP** : Locational Marginal Price

**LSTM** : Long Short-Term Memory

**LTC** : Load Tap Changer

**LTE** : Long Term Evolution

**MDMS** : Meter Data Management System

**MITM** : Men In The Middle

**ML** : Machine learning

**MMS** : Market Management Systems

**MP** : Memory Pool

**MR-D** : Merkle Root Data

**MR-R** : Merkle Root Reputation

**NAN** : Neighborhood Area Network

**NLP** : Natural Language Processing

**NTIC** : National Institute of Standards and Technology

**NTL** : Non-Technical Loss

**OPF** : Optimal Power Flow

**P2P** : Peer-to-Peer

**PoA** : Proof of Authority

**PBFT** : Practical Byzantine Fault Tolerance  
**PoEf** : proof-of-efficiency  
**PoR** : Proof of Reputation  
**PoS** : Proof of Stack  
**PoW** : Proof of Work  
**QoS** : Quality Of Service  
**REP** : Replay Attack  
**RF** : Random Forest  
**RNN** : Recurrent Neural Network  
**RTU** : Remote Terminal Unit  
**SA** : smart appliances  
**SCADA** : Supervisory Control and Data Acquisition  
**SCED** : Security Constrained Economic Dispatch  
**SCOPF** : Security Constrained Optimal Power Flow  
**SE** : State Estimation  
**SG** : Smart Grid  
**SGD** : Stochastic Gradient Decentralized  
**SMN** : Smart Metering Networks  
**SMOTE** : Synthetic Minority Over-Sampling Technique  
**SN** : Sensor Node  
**SPD** : Signal Price Determination  
**SVM** : Support vector machine  
**TIC** : Information and Communication Technology  
**TN** : True Negative  
**TP** : Topology Attack  
**TP** : True Positive  
**TS** : Transmission System  
**UI** : User Interface  
**VAE** : Variational Automatic Encoder  
**VVC** : Volt/Var Control  
**WAN** : Wide Area Network  
**WiMAX** : Worldwide Interoperability for Microwave Access  
**WLAN** : Wireless Lan Area Network

## Introduction générale

Au cours des dernières décennies, les systèmes énergétiques centralisés traditionnels basés sur les combustibles fossiles ont été confrontés à des défis majeurs tels que la transmission à longue distance, les émissions de carbone, la pollution de l'environnement et la crise énergétique. Afin de construire une société durable en relevant ces défis, l'utilisation d'énergies renouvelables provenant de diverses sources ainsi que l'amélioration de l'efficacité de l'utilisation de l'énergie sont les deux principales solutions potentielles. Ces dernières années, le concept de réseau intelligent qui implique la technologie de communication, le système d'alimentation interconnecté, la technologie de contrôle avancée et les compteurs intelligents a été appliqué pour améliorer l'utilisation des sources d'énergie renouvelables et soulager la crise énergétique d'une manière ou d'une autre.

Le concept de réseau intelligent a été introduit comme une nouvelle vision du réseau électrique conventionnel pour trouver un moyen efficace d'intégrer les technologies d'énergie verte et renouvelable. De cette manière, le réseau intelligent connecté à Internet, également appelé Internet de l'énergie, apparaît également comme une approche innovante et bidirectionnelle d'énergie et d'informations afin de trouver un moyen efficace de fournir, de gérer et d'intégrer les technologies d'énergie et de garantir une sécurité des plus optimal.

Cependant, l'intégration et la coordination d'un grand nombre de connexions croissantes peuvent être un défi pour le système de réseau centralisé traditionnel. De plus, installer des équipements autonomes comme le smart meter (SM) au niveau des clients peut aussi s'avérer critique en vue d'une possible manipulation de données ou d'une pré-méditation de disfonctionnement de l'appareil. Par conséquent, le réseau intelligent subit une transformation vers la topologie décentralisée à partir de sa forme centralisée, ce qui permettra d'introduire la technologie de la blockchain et pallier au problème de l'injection de fausses données. D'une autre part, l'intelligence artificielle jouera un rôle primordial dans la détection d'un quelconque vol ou manipulation de l'énergie au niveau de mesures envoyées par le smart meter.

En conséquence, afin d'aboutir à notre objectif qui consistent à combiner l'IA et la blockchain pour sécuriser les réseaux intelligents, notre travail est présenté en 4 chapitres :

- Le Premier chapitre « Etat de l'art » sera consacrée à la définition du réseau intelligent, ses principaux composantes et à la sécurité dans les smart grid incluant les deux attaques de vol d'énergie et la FDIA.

- Le deuxième chapitre « Blockchain et Intelligence Artificielle » sera consacrée aux différentes méthodes et techniques utilisées pour pallier aux principaux problèmes rencontrés dans les smart grids à savoir, l'attaque FDIA (False Data Injection Attack) et le vol d'énergie. Et l'utilité du deep learning et de la blockchain dans la protection de chaque une de ces attaques.
- Le troisième chapitre sera consacré à la Conception d'une contre-mesure de détection FDIA combinant des techniques d'apprentissage automatique et la technologie blockchain.
- Dans le quatrième chapitre, nous présenterons l'environnement de travail, puis nous réaliserons des expériences sur la solution proposée combinant le deep learning et la blockchain et nous évaluerons ses performances à l'aide de métriques bien définies.

# **Chapitre 1**

## Etat de l'art

## Introduction

Depuis l'avènement des technologies Internet, le développement des réseaux informatiques n'a cessé d'accroître. Avec l'arrivée des technologies de l'IOT (Internet Of Things), celui-ci s'est amplifié et a pris une place prépondérante dans la vie quotidienne grâce à la richesse des services offerts.

L'idée d'un « réseau intelligent » a vivement occupé le devant de la scène, cette évolution des technologies de pointe permet de rendre possible la disponibilité d'un réseau électrique plus intelligent, plus efficace, sûr et durable.

Ces technologies visent à relever les défis complexes auxquels sont confrontés les systèmes de réseau aujourd'hui, qui découlent en grande partie de leurs infrastructures vieillissantes et de leurs cycles de vie très limités, ce qui affecte négativement leurs fiabilités et leurs efficacités.

Dans ce premier chapitre, nous décrivons le smart grid, son architecture ainsi que ses composantes. Puis, nous nous intéresserons à la sécurité dans ce type de réseau où on évaluera la réalité des risques en remettant en perspective l'impact, les acteurs possibles et le potentiel perturbateur dans les smart grids.

## 1. Réseaux électriques intelligents (Smart Grid)

### 1.1. Définition des Smart Grids

Le Smart Grid ou « *réseau électrique intelligent* » est une nouvelle technologie qui modernise le système électrique à travers l'intégration des nouvelles technologies de l'information et de la communication (*NTIC*) dans le réseau, comme par exemple les objets connectés.

Ce réseau intelligent est capable de transmettre des informations en temps réel sur les usages et les consommations d'électricité entre la source d'approvisionnement (centrale électrique, éolienne...etc), et le consommateur, et cela afin d'ajuster les flux d'électricité et garantir une meilleure efficacité énergétique dans le réseau [Net 1].

### 1.2. Architecture des Smart grids

L'intégration des systèmes de réseaux électriques avec les systèmes de communication a amélioré la gestion et le contrôle des ressources des systèmes de réseau électrique.

En effet, le réseau intelligent repose sur l'implémentation de deux interfaces informatiques, l'une placée chez la source d'approvisionnement (central électrique par exemple) et l'autre généralement chez le client. Grâce à une communication *bidirectionnelle* entre le fournisseur

et le client, l'échange de données et l'exécution de commande de contrôle peuvent se faire d'une manière synchronisée. En outre, les données recueillies par l'interface du client sont directement envoyées au GRD (Gestionnaire de réseau de distribution), qui lui se chargera d'ajuster l'approvisionnement en électricité de manière intelligente et en temps réel pour le client [Net 2].

Les systèmes de réseau intelligent comme mentionné ci-dessus peuvent être divisés en deux systèmes distincts qui sont : systèmes de communication et systèmes des réseaux électriques. Comme le montre la Fig 1, les systèmes de communication du réseau intelligent consistent en plusieurs types de topologies de réseau mises en œuvre et structurées de manière hiérarchique. Comme on peut le voir, la station de base agit comme un collecteur de données acquises à partir du réseau de distribution englobant le *réseau étendu (WAN)*, la *zone de voisinage réseau local (NAN)*, *réseau de bâtiment (BAN)* et *réseau domestique (HAN)*. Ces quatre topologies de réseau sont le réseau de communication de base dans le réseau intelligent et seront encore discutées dans la section suivante.

Du point de vue des réseaux électriques, l'énergie peut être générée par une variété de centrales de production d'énergie telles que le nucléaire, l'hydroélectricité, l'éolien et les systèmes solaires. L'énergie générée est ensuite distribuée via deux sous-stations électriques différentes. La première sous-station connue sous le nom de sous-station de transmission est normalement située dans une position de la centrale électrique. Cette sous-station est chargée de fournir des quantités de tension de la centrale de production d'électricité à la prochaine sous-station connue sous le nom de sous-station de distribution. Fondamentalement, le poste de distribution est situé à proximité de sites industriels ou zones résidentielles où il est chargé de convertir l'alimentation haute tension en alimentation électrique de moyenne tension avant de la distribuer aux colonnes d'alimentation de basse tension. Le bas de pilier d'alimentation de tension distribue ensuite l'alimentation électrique basse tension à l'industriel ou les zones résidentielles telles que les bâtiments commerciaux et les maisons.

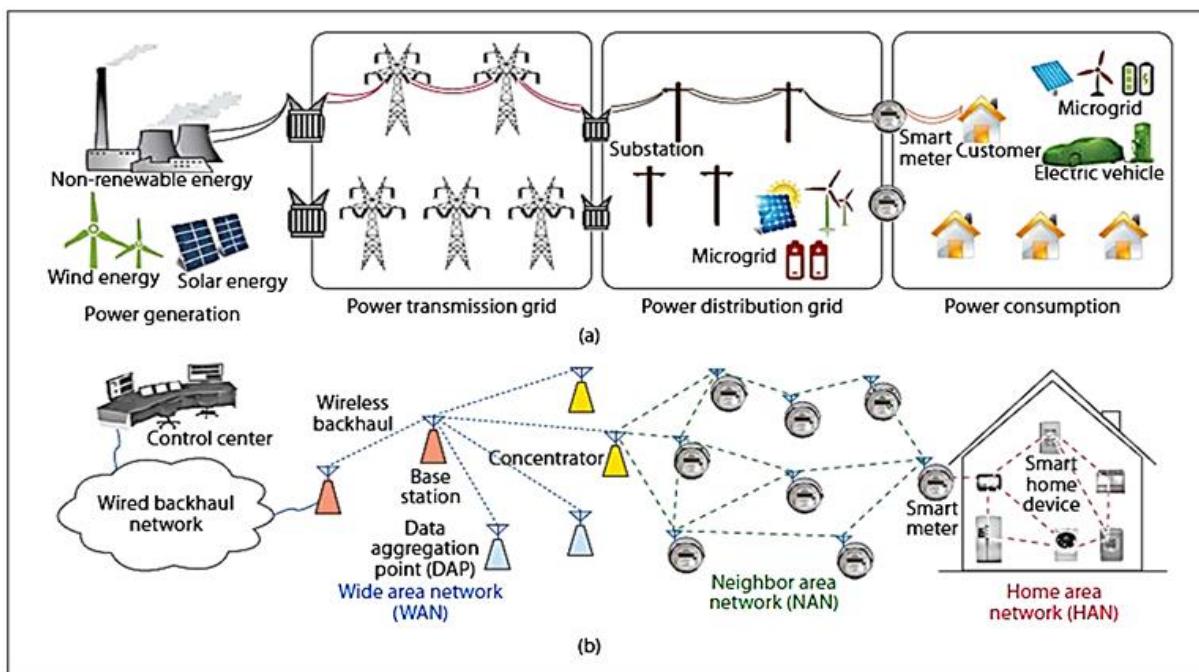


Figure 1 : Architecture des Smart Grids [Net 3]

### 1.3. Les principales composantes d'un Smart Grid

Le compteur intelligent a certainement été l'élément déclencheur des premières technologies de réseau intelligent. Toutefois, un smart grid représente bien plus qu'un simple compteur intelligent, parmi les autres éléments figurent les lignes et les sous-stations de distribution ainsi que les différentes technologies et mécanismes de communication.

#### 1.3.1. Les compteurs intelligents

Un compteur intelligent est un compteur d'énergie avancé qui identifie consommation d'énergie plus détaillée par rapport à un compteur d'énergie conventionnel.

Un compteur intelligent peut accéder aux données concernant la consommation d'électricité en temps réel et de l'électricité produite localement. Il a aussi la capacité de lire d'autres compteurs de marchandises dans les mêmes locaux ou à sa portée [3].

Le compteur intelligent est l'un des composants essentiels de l'infrastructure AMI (Advanced Metering Infrastructure) qui est chargé d'assurer un contrôle et un suivi efficaces des consommations d'énergie électrique. Les tâches qu'un compteur intelligent exécute sont: la communication bidirectionnelle entre fournisseurs de services publics et les consommateurs, gestion des données de comptage, fourniture de rapports d'anomalies, analyse des défauts et la qualité de l'énergie, ce qui montre simplement qu'il existe une énorme quantité de données échangés dans le Smart Metering Networks (SMN) [1].

### 1.3.2. Les systèmes de communication : The Smart Metering Networks

La communication entre les smart compteurs, forme un réseau appelé réseau de comptage intelligent (SMN). Le réseau est également connu sous le nom “Advanced Metering Infrastructure (AMI)” [1].

Pour faciliter la communication des données dans le réseau intelligent, plusieurs types de réseaux ont été introduits et sont :

- **Home Area Network (HAN)**

HAN est un réseau dédié reliant tous les appareils intelligents qui fonctionnent dans un réseau domestique. Pour permettre la communication dans ce réseau, un compteur intelligent sans fil est placé dans la maison du consommateur qui sert ensuite de passerelle HAN chargée de gérer toutes les communications de données.

Grâce à ce réseau, les consommateurs peuvent contrôler et surveiller l'énergie consommée et le flux de données entre les appareils électroménagers tels que les thermostats, les climatiseurs, les réfrigérateurs, les rondelles, les séchoirs et les cuisinières [1].



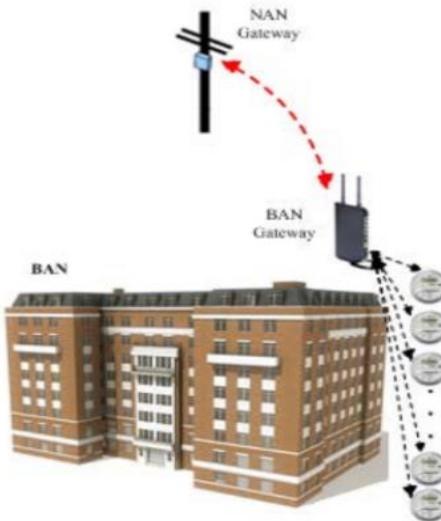
*Figure 2 : Exemple d'un réseau domestique (HAN)*

- **Building Area Network (BAN)**

Aussi appelé *Basement Area Network*, BAN est un réseau qui couvre l'ensemble d'un bâtiment, il peut s'agir d'un ensemble de réseaux locaux plus petits. Par exemple, si chaque étage est considéré comme un seul HAN, alors la combinaison de chaque HAN par étage est considérée comme un BAN [1].

La Fig. 3 montre la topologie du réseau BAN qui consiste en plusieurs HANs. Afin de maintenir les liens de communication entre BAN et HANs, un dispositif appelé

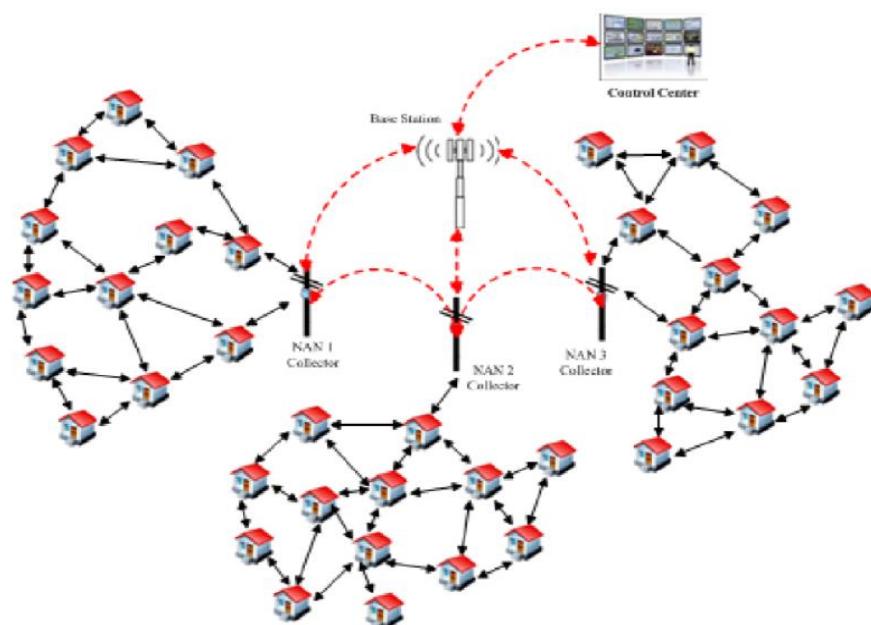
passerelle BAN est installé dans le bâtiment. Les fonctions de cette passerelle sont d'agrégier, de surveiller et de fournir des informations telles que les besoins de la consommation en énergie de ses HANs, où ces informations seront envoyées au centre de collecte de données le plus proche ou à la passerelle NAN.



*Figure 3 : Exemple d'un building area network (BAN)*

#### ➤ Neighborhood Area Network (NAN)

Un réseau de voisinage (NAN) est une branche des points d'accès Wi-Fi et des réseaux locaux sans fil (WLAN), qui permettent aux utilisateurs de se connecter à Internet rapidement et à très peu de frais. Ne couvrant qu'un petit nombre de blocs à proximité d'un point d'accès 802.11, ce réseau est généralement installé pour desservir une ou plusieurs voisins (l'installation se fait généralement par un particulier).



*Figure 4 : Exemple d'un réseau de voisinage (NAN)*

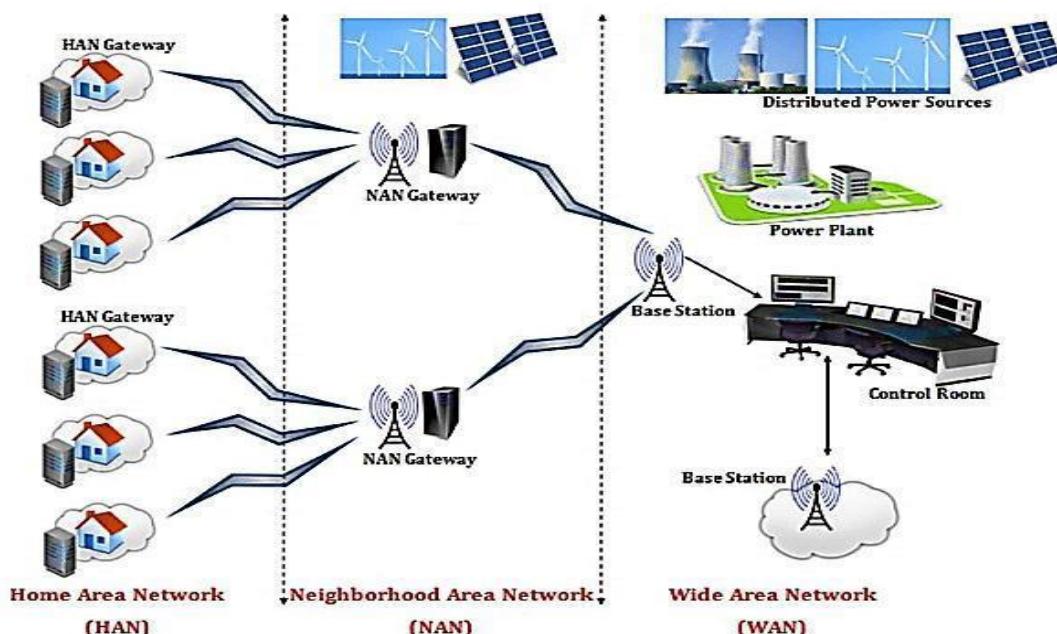
### ➤ Wide Area Network (WAN)

Le WAN est un réseau informatique capable de couvrir une zone géographique très vaste, il se situe au-dessus du HAN, BAN et NAN.

Celui-ci couvre la transmission de données depuis NAN passerelle vers le centre de contrôle du service public. En effet, l'interface entre le WAN et les NANs se compose de stations de base, tandis que l'interface entre le NAN et les BANs n'est qu'une passerelle BAN qui est ensuite connectée à des compteurs intelligents qui agissent comme une interface. Ceci est bien illustré dans la figure 3.

Pour permettre et faciliter les communications au sein du WAN, les technologies de communication telles que WiMAX, cellulaire GSM 3G, LTE et fibre optique sont parmi les technologies appropriées qui peuvent être utilisées pour répondre aux exigences d'une communication WAN.

La figure suivante reflète la relation existante entre le réseau WAN et les réseaux HAN et NAN [Net 4].



*Figure 5 : Relation entre WAN, HAN et NAN.*

#### 1.3.3. Les systèmes de réseau électrique [Net 5].

Cette partie explique la structure de base d'un système de transmission et de distribution d'énergie électrique comme montré dans la figure 6 en mettant l'accent sur la mission, les fonctions et les composants d'un système de contrôle d'électricité des services publics [Net 5].

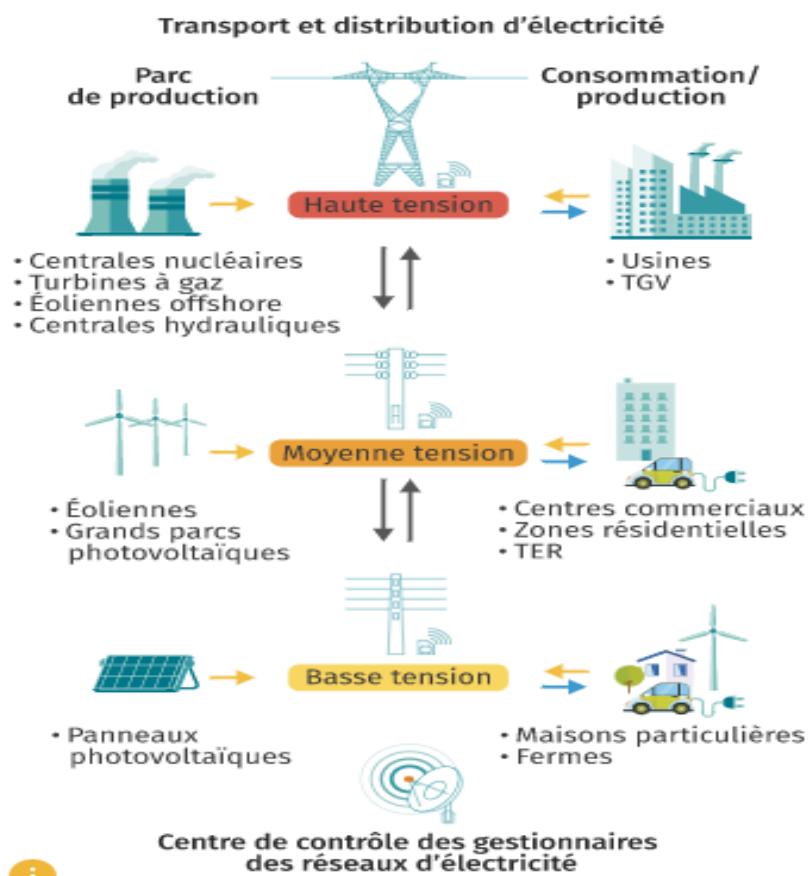


Figure 6 : Architecture de réseau électrique

### ➤ Système de transmission et de distribution d'énergie électrique

Il se compose d'un système de génération, un système de transmission, un système de sous-transmission, un système de distribution et d'un centre de contrôle [Net 5].

### ➤ Le centre de contrôle

Le centre de contrôle surveille les centrales de production, les systèmes de transmission et de sous-transmission, les systèmes de distribution et les charges des clients d'un service public. Les principales fonctions d'un centre de contrôle des services publics d'électricité sont de :

- Fournir une surveillance centralisée des opérations du système électrique.
- Conserver les données historiques.
- Permettre le contrôle manuel et automatique de l'équipement du terrain.

### ➤ Système de gestion de l'énergie

Un système de gestion de l'énergie (EMS) du centre de contrôle abrite généralement les systèmes du service public, bases de données, les applications et affichages opérationnels, et la fonction de génération de rapports sur le système électrique. La nécessité de diffuser des données précieuses sur le système électrique au sein d'un

service public a conduit de nombreux services publics à connecter leurs systèmes EMS à leur réseau local (LAN) ou réseau étendu (WAN) d'entreprise pour faciliter le partage de données avec d'autres services. Un système de gestion de l'énergie (EMS) d'un centre de contrôle se compose généralement de quatre éléments principaux [Net 5].

- **Le système de contrôle de surveillance et d'acquisition de données (SCADA)** : collecte les données du système électrique sur le terrain via une série de processeurs frontaux, déclenche des alarmes pour le personnel d'exploitation et émet des commandes de contrôle sur le terrain selon les instructions des applications du système de centre de contrôle.
- **Le système de contrôle automatique de la production (AGC)** : contrôle les unités de production du service public pour s'assurer que la charge optimale du système est satisfaite, avec la production la plus économique disponible.
- **Les applications et la base de données de gestion de l'énergie** : sont les programmes et les ensembles de données associés que le personnel d'exploitation des services publics utilise pour gérer l'estimation de l'état, le flux de puissance, l'analyse des contingences, le flux de puissance optimal, la prévision de charge et l'allocation des unités de production.
- **Le système d'interface utilisateur (UI)** : une interface interactive pour surveiller les performances du système électrique, gérer les conditions d'alarme du système et étudier les conditions potentielles du système pour s'assurer que les critères de sécurité du réseau sont respectés.

#### 1.4. La Sécurité dans les Smart Grids

Bien que les avancées technologiques augmentent l'efficience et l'innovation dans le domaine des réseaux, celles-ci peuvent être un double tranchant pour ce qui en est de la sécurité des technologies de l'information.

En effet, l'utilisation croissante des systèmes d'alimentation électrique basés sur les technologies des smart grids augmente le nombre de vulnérabilités dans ce système [Net 6].

À l'exemple des réseaux de capteurs sans fil et les réseaux Ad-hoc mobile, les SMNs (réseau de comptage intelligent) sont également vulnérables aux différents types de menaces de sécurité telles que l'usurpation d'identité, false data injection, et les attaques par compromission de nœud [1].

Dans cette partie nous nous chargerons d'énumérer les différentes attaques informatiques sur les SMNs, comme nous explorerons les exigences de sécurité requises sur ces réseaux afin d'assurer la sécurité des données communiquées.

#### 1.4.1. Les attaques sur les smart metering networks

Bien qu'il se base sur l'informatique et les technologies avancées de l'IOT, le smart grid s'appuie sur un plus large éventail de technologies. La transition des réseaux électriques traditionnels vers le système intelligent repose sur divers autres facteurs.

Les technologies de pointe sur lesquelles le réseau intelligent se base font de lui une innovation technologique remarquable et lui permettent de répondre aux besoins attendus. Cependant, Niveau sécurité, celui-ci peut s'avérer vulnérable aux menaces courantes liées aux réseaux informatiques, et comme exemples de violations de sécurité, on dénombre les attaques suivantes :

##### ➤ **Phishing**

Les pirates pourraient utiliser les informations des clients, les factures ou le reçu de paiement obtenu par des techniques d'ingénierie sociale afin d'obtenir des informations cruciales sur le fournisseur d'électricité [4].

##### ➤ **False data injection (FDI)**

Constitue une sorte d'attaque qui tente de manipuler l'intégrité des données en injectant de fausses données dans le réseau pour but d'induire le centre de contrôle en erreur pour qu'il prenne une décision erronée en cas d'urgence sur les processus d'analyse, distribution de puissance et de facturation [1].

##### ➤ **Déni de service (DOS)**

Le DoS dans les SMNs est une attaque de brouillage qui se produit essentiellement dans les réseaux sans fil à l'aide d'un brouilleur qui tente de perturber les fréquences radio utilisées par les compteurs intelligents en transmettant un autre signal radio afin d'interrompre le processus de transmission de données [1].

##### ➤ **Eavesdropping Attack (EVD) et Analyse du trafic**

Une autre attaque courante dans les SMNs qui peut être définie comme un acte d'écoute secret et d'enregistrement des communications de données sur les compteurs intelligents voisins [1].

➤ **Attaque d'usurpation d'identité “*Impersonation Attack*” (IMP)**

L'attaque IMP ou man-in-the-middle est une attaque où un compteur intelligent malveillant s'empare de l'identité d'autres compteurs intelligents légitimes, ce qui permet à un attaquant de recevoir et de modifier le contenu des messages reçus [1].

➤ **Attaque par compromission de nœud “*Node compromise Attack (CMP)*”**

Cette attaque se base sur l'accès physique au nœud du compteur intelligent dans le but de prendre le contrôle de la communication et ainsi obtenir un accès non autorisé aux données sensibles telles que les informations cryptographiques, ce qui va permettre à l'attaquant d'injecter de fausses données dans le réseau [1].

		Gravité de l'attaque		
		Faible	Moyen	Élevé
Probabilité de l'attaque pour être effectué	Élevé	- Eavesdropping attack et Analyse de trafic [5]		- DOS [5,1] - Virus, vers, cheval de Troie [5]
	Moyen	- Phishing [5]	- MITM [5,1]	- False data injection [1] - Attaque par compromission de nœud [1]
	Faible			

*Tableau 1 : Probabilité de l'attaque et la gravité associée.*

#### 1.4.2. Les exigences de sécurité dans les smart metering network

Un réseau intelligent intègre le réseau électrique traditionnel avec les technologies de l'information et de la communication (TIC). Une telle intégration permet aux fournisseurs de services publics d'électricité et aux consommateurs d'améliorer l'efficacité et la disponibilité du système électrique tout en surveillant, contrôlant et gérant en permanence les demandes des clients [4], la figure suivante reflète des objectifs en termes de sécurité qui doivent être pris en compte dès la conception initiale du réseau SMN.

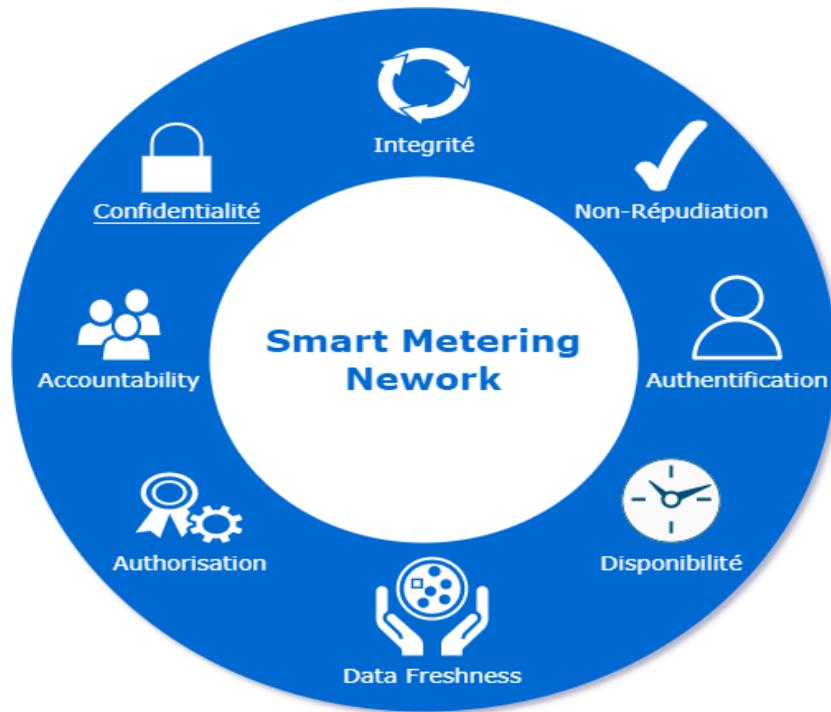


Figure 7 : Prérequis en sécurité des SMNs.

#### ➤ Confidentialité

La confidentialité des données doit être préservée pour garantir que le contenu des données transmises ne sera jamais exposé à des parties non autorisées. Elle est assurée en utilisant les techniques de cryptage [1].

#### ➤ Intégrité

L'intégrité fournit l'assurance que les données et les commandes de contrôle n'ont pas été altérées ou modifiées sans autorisation [6].

#### ➤ Disponibilité

Dans les SMNs, la disponibilité des données garantit que le réseau est actif et que les données sont accessibles même en présence d'attaque DoS [1].

#### ➤ Non réputation

La propriété de non-réputation peut détecter qu'un individu ait accompli de fausses actions tels que vol d'énergie et false data injection et maintenant nie ou n'assume la responsabilité de l'action [6].

#### ➤ Data Freshness

Une autre exigence de sécurité importante pour garantir que les données transmises sont récentes, Le but d'avoir une telle exigence est de protéger les données d'être manipulées par rejouer l'attaquant [1].

### ➤ Authentification

L'authentification ou l'identification est une méthode logique pour prouver la légitimité et l'identification d'une entité, telle qu'un utilisateur final, un compteur, etc...[6].

### ➤ Autorisation

De nombreux acteurs vont partager des données d'usage et de consommation, qui dépendent de l'intérêt de leurs applications, telles que la demande d'offre. Cependant, ces acteurs sont nécessaires pour appliquer des politiques d'autorisation adéquates, afin que les données des clients ne soient pas accessibles sans autorisation [6].

### ➤ Accountability et audit

L'audit périodique et l'accountability sont des exigences primordiales pour permettre une analyse approfondie et la traçabilité de réseau [6].

Il existe une grande variété de cyberattaques sur les smart grids, parmi ces attaques ayant une gravité élevée et une probabilité d'occurrence moyenne, l'attaque *False Data Injection (FDI)*.

Les détails concernant cette attaque seront traités dans la section suivante.

## 2. False data injection attack (FDIA) et vol d'énergie

### 2.1 False data injection

L'attaque *False Data Injection (FDI)* est l'une des cyber-attaques les plus dangereuses dans les smart grids. Elle vise à injecter de fausses données pour lancer une attaque judicieuse pouvant causer de graves dommages [7]. Le support sans fil et la communication basée sur la diffusion sont utilisés pour transmettre des données dans le SMN, les données transmises par les compteurs intelligents utilisant des supports de communication sans fil peuvent être facilement capturées ou interceptées par l'attaquant. Cette attaque fait partie des attaques qui sont difficilement détectables dans les SMNs en raison de l'absence ou de l'insuffisance du mécanisme de détection des comportements inappropriés utilisé dans les schémas de sécurité. Par conséquent, lorsque ces nœuds sont compromis avec succès ou piratés, l'injection de fausses données peut être facilement effectuée par l'attaquant [1]. Pour bien comprendre le fonctionnement de cette attaque, on présente ci-dessous un exemple de scénario de celle-ci.

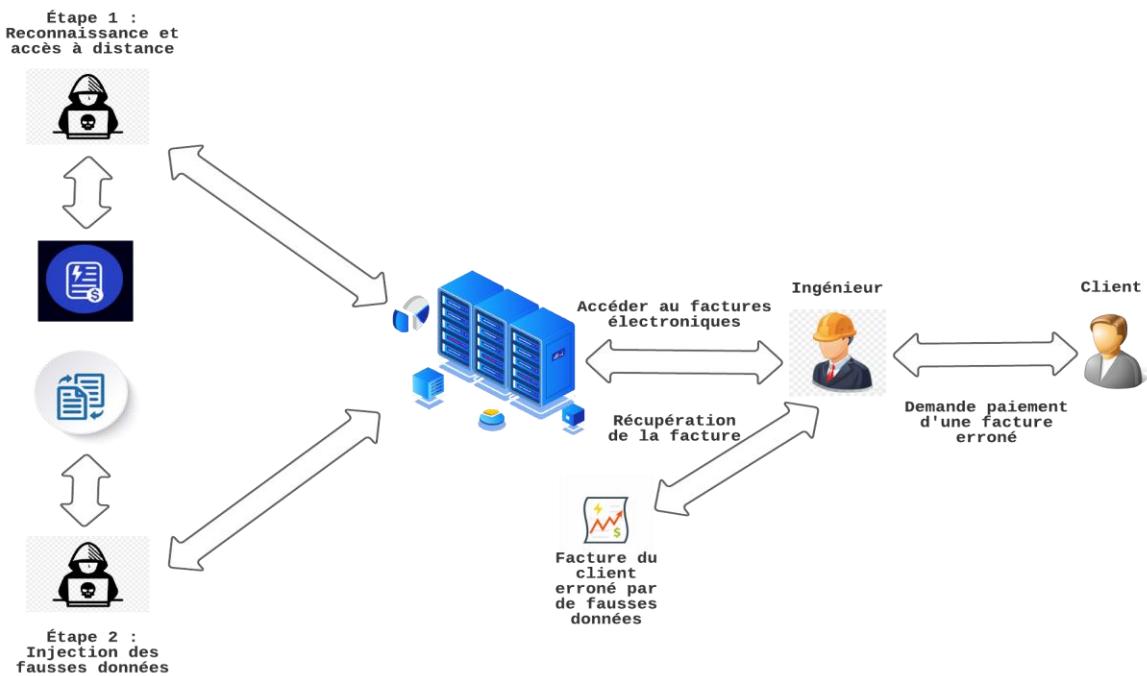


Figure 8 : Exemple d'un scénario d'attaque par injection de fausses données (FDIA).

Les étapes d'exploitation de ce scénario d'attaque sont comme suit :

- **Étape 1 :** Grâce à une phrase de reconnaissance et de récupération de données sensibles, l'attaquant obtient un accès à distance au serveur contenant les données des clients qui sont mises à jour en temps réel.
- **Étape 2 :** Une fois l'accès établi, l'attaquant injecte des fausses données concernant un ou plusieurs clients. Par exemple : gonflement du taux d'électricité consommée par un client.
- **Étape 3 :** à un moment donné, les factures d'électricité de chaque client seront validées que ça soit de manière automatique par le système lui-même ou bien par des entités responsables de cette tâche, et donc les factures erronées seront elles aussi validés. Comme conséquence, les clients auront donc à payer des factures qui ne leurs correspondent pas causant ainsi des pertes financières catastrophiques.

## 2.2 Le vol d'énergie

Le vol d'électricité constitue une part importante du NTL (pertes non techniques), Le vol d'électricité comprend le contournement, la falsification du compteur d'énergie et d'autres méthodes physiques pour échapper au paiement. Appui illégal de l'électricité de la ligne d'alimentation et la falsification du compteur sont les moyens de vol les plus identifiés et les plus comptabilisés [48].

La forme la plus courante de vol d'électricité consiste à prélever de l'électricité directement sur le réseau de distribution et à falsifier le compteur d'énergie.

Afin de comprendre l'étendu et les variétés des attaques FDIA et le vol d'énergie, on se charge d'étudier la sécurité d'un système électrique.

### 2.3. La sécurité du système électrique

Le centre de contrôle surveille la sécurité du système électrique à la fois dans les systèmes de transmission (TS) et de distribution (DS) et estime un état appelé "*State Estimation*" (SE).

Sur la base de l'état estimé, différentes décisions sont prises par diverses applications de nombreux systèmes dont les systèmes de gestion d'énergie (EMS), systèmes de gestion du marché (MMS), et systèmes de gestion de la distribution (DMS).

Les systèmes précédents eux-mêmes se composent de plusieurs sous-systèmes qui sont montrés dans le tableau ci-dessous

Systèmes	Sous-systèmes
Système de gestion d'énergie (EMS)	<ul style="list-style-type: none"> <li>- Analyse de contingence (CA)</li> <li>- Flux de puissance optimal sous la contrainte de sécurité (SCOPF)</li> </ul>
Systèmes de gestion du marché (MMS)	<ul style="list-style-type: none"> <li>- Répartition économique contrainte par la sécurité (SCED)</li> </ul>
Systèmes de gestion de la distribution (DMS)	<ul style="list-style-type: none"> <li>- Contrôle automatique de la génération</li> <li>- Le contrôle volt/var (VVC)</li> <li>- Détermination du prix du signal (SPD)</li> </ul>

Tableau 2 : Systèmes et sous-systèmes d'un système électrique.

### 2.4. Classification FDIA basée sur le système cible

L'estimation d'état est essentielle à la surveillance et au contrôle des réseaux intelligents. Récemment, l'attaque par injection de fausses données (FDIA) est apparue comme une menace sérieuse pour l'estimation d'État. Les approches conventionnelles de détection des FDIA sont limitées par leurs fortes hypothèses de connaissances statistiques, leur complexité et leur coût matériel. De plus, la plupart des approches actuelles de détection des FDIA se concentrent sur la détection de la présence de FDIA, alors que les informations importantes sur les emplacements et les systèmes cibles exacts de l'injection ne sont pas accessibles.

Dans cette section, en s'inspirant de l'étude faite par [7] on va donner une classification qui est récapitulée dans le tableau 4 de cette attaque selon le système cible.

Le tableau suivant classe les attaques FDIA selon le système cible.

Système	Sous-Système	Attaque
EMS	State estimation	Bypass BDD
		Observability
		Topology
	Contingency analysis	CA
		OPF
	SCOPF	SCOPF
		AGC
MMS	SCED	SCED
DS	VOLT/VAR control	vol/var control

*Tableau 3 : Classification de FDIA selon système cible [7]*

## 2.5. Impact de False data injection attack et le vol d'énergie

L'impact de FDIA et le vol d'énergie sur le système d'alimentation peut être classé en deux classes économique et physique.

Pour l'aspect économique:

- Financement illégale et cela à travers :
  - Falsification du LMP qui est calculée par le marché de l'électricité.
  - Maximisation de la charge générée et du coût lors de la résolution OPF, AGC, SCOPF.
- Vol d'énergie par la modification de la charge profil des clients finaux dans le DS.

Pour l'impact physique, les attaques FDIA peuvent causer des dommages à la fois aux clients et aux services publics et peuvent également entraîner des pertes financières et avoir un impact social car cela affecte le confort du client.

## Conclusion

Dans ce chapitre, nous avons décrit le smart grid, son architecture ainsi que ses composantes. Puis nous nous sommes intéressés à sa sécurité où on a décrit les différentes attaques possibles sur ce type de réseau et leurs impacts, cependant notre intérêt s'est porté sur une attaque particulière qui est l'injection de fausse données et le vol d'énergie. Cette attaque de grande envergure peut cibler n'importe quel composant transmetteur ou distributeur dans un smart grid. De ce fait, des contre-mesures et systèmes de préventions sont primordiaux à mettre en place en utilisant les dernières technologies disponibles et pouvant être efficaces à cela, à savoir la blockchain et l'intelligence artificielle.

Le chapitre suivant sera donc consacré à la technologie de la blockchain et les modèles de prédiction et détection basé sur le deep learning.

# **Chapitre 2 :**

## Blockchain et Intelligence Artificielle.

## Introduction

Mettre en place un smart grid demande une cohésion et une collaboration de la part de tous les acteurs du réseau. Or, plus il y a d'acteurs, plus le projet se complexifie, aussi bien en matière d'organisation que d'exploitation des données. D'autre part, cette complexité de la technologie des réseaux intelligents nécessite une implémentation d'une stratégie de défense complète afin de couvrir toutes sortes de menaces et de vulnérabilités que ça soit en termes de détection ou de prise de mesures proactives.

La pérennité d'un réseau intelligent réside donc avant tout dans la synergie entre ses différents acteurs. C'est là qu'apparaît des solutions technologiques : blockchain et deep learning.

Ce chapitre sera consacré à étudier les technologies de la blockchain et du deep learning, où on va énoncer les différentes solutions proposées dans la littérature pour leurs intégration et implémentation dans les smart grids.

### 1. Les contre-mesures basées sur la blockchain

La technologie des smart grids permet l'intégration et la coordination d'un grand nombre de connexions croissantes. Cependant elle constitue un défi pour les autorités compétentes par rapport au système de réseau centralisé traditionnel. Par conséquent, le réseau intelligent est en train de passer d'une forme centralisée à une topologie décentralisée. D'autre part, la blockchain présente d'excellentes caractéristiques qui en font une application prometteuse pour le paradigme du réseau intelligent.

Dans cette section, nous visons à définir cette technologie et fournir une étude sur l'application de la blockchain dans le réseau intelligent.

#### 1.1. La technologie de la blockchain

La définition la plus simple de la blockchain est une chaîne construite à partir de nombreux blocs contenant des informations [13], ou un registre distribué comprenant une collection de blocs qui enregistre différents types de données ou des informations sur les transactions.

Les blocs sont attachés ensemble par une chaîne où chaque bloc fait référence au hachage cryptographique des données des blocs précédents comme le montre la figure 9. Dans le réseau blockchain, les blocs nouvellement générés sont ajoutés en continu à la chaîne, à des intervalles réguliers, et cette chaîne est répliquée parmi les membres du réseau. Chaque bloc peut également inclure un horodatage, un nonce, un arbre de hachage appelé arbre de Merkle, des scripts de contrats intelligents, etc. Le hachage et l'arbre de Merkle permettent de vérifier que le contenu du bloc n'a pas été modifié, c'est-à-dire qu'il garantit l'intégrité [14], cela fait

de la technologie blockchain une méthode très sûre pour le transfert de propriétés, d'argent et de données sans avoir besoin d'un agent intermédiaire tiers comme les gouvernements ou les banques [13].

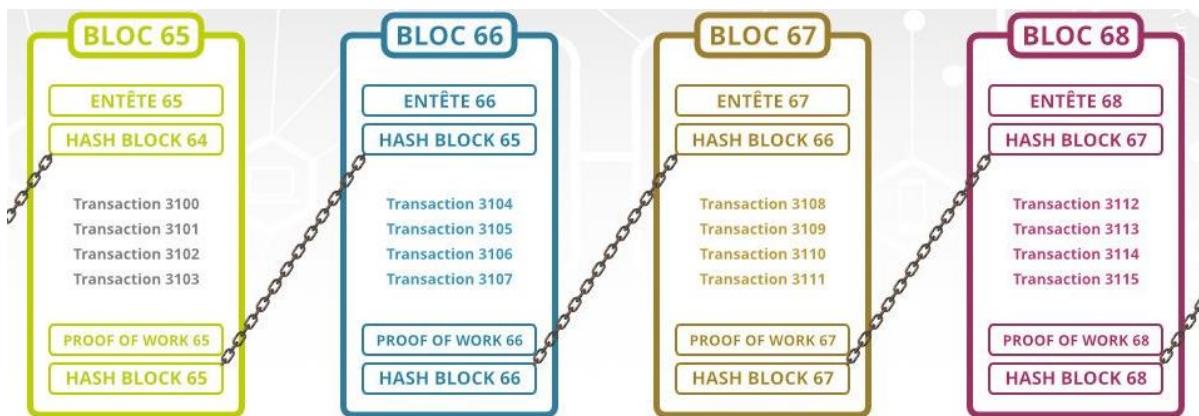


Figure 9 : Structure générale de la blockchain

### 1.1.1. L'arbre de Merkle

Une méthode qui divise les données en blocs, exécute des valeurs de hachage pour chaque bloc, les introduit dans les nœuds feuilles et répète le hachage sous la forme d'un arbre de hachage binaire jusqu'à ce qu'une seule valeur de hachage soit créée. Dans ce cas, la valeur de hachage racine peut être utilisée comme signature. Cette méthode est adaptée à la vérification de l'intégrité des données. La figure suivante montre une configuration typique de Merkle-Tree [25] :

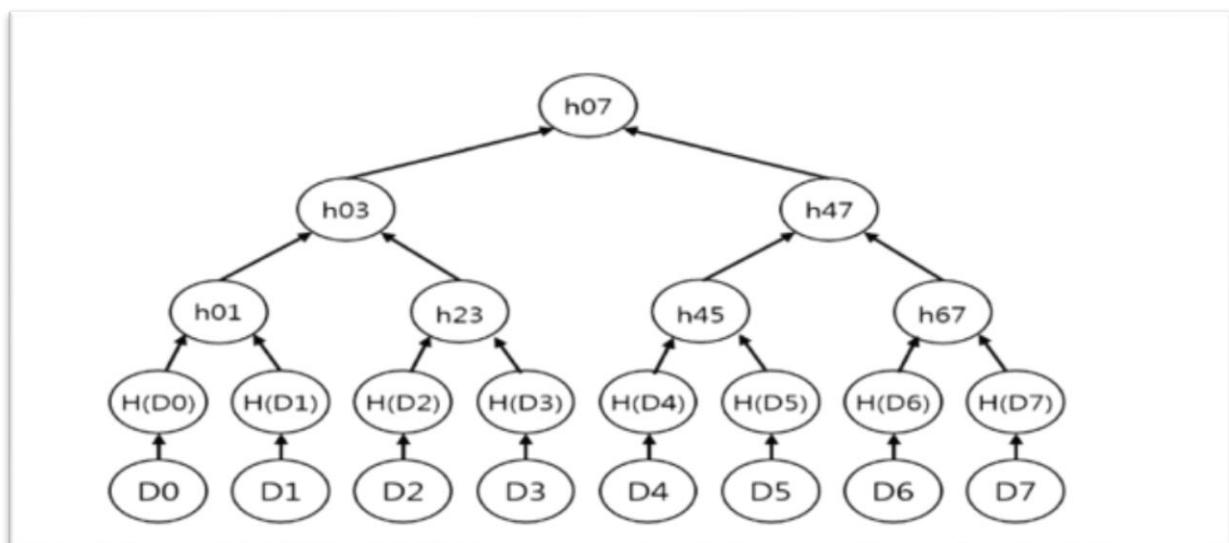


Figure 10 : L'arbre de Merkle

### 1.1.2 Concept de fonctionnement de la blockchain

La blockchain fonctionne selon un processus généralisé qui est illustré à la figure 11. Le processus commence par une demande de transaction qui peut être initiée par n'importe quel utilisateur. Ensuite, la transaction est diffusée à tous les utilisateurs du réseau.

Après cela, le processus de vérification a lieu où tous les nœuds vérifient les transactions via les hachages. Une fois la vérification terminée, la transaction est contenue dans un nouveau bloc qui est connecté à la blockchain précédente, ce qui la rend perméable et inaltérable [13]. L'utilisation de hachages fournit une méthode efficace pour sécuriser la blockchain. Cependant, avec l'aide des ordinateurs super rapides, les pirates informatiques pourraient modifier les informations d'un seul bloc et ensuite recalculer tous les hashs des blocs suivants de la chaîne en quelques minutes. Pour surmonter ce problème, plusieurs algorithmes ont été créés pour obtenir ce que l'on appelle le consensus qu'on va énoncer juste après.

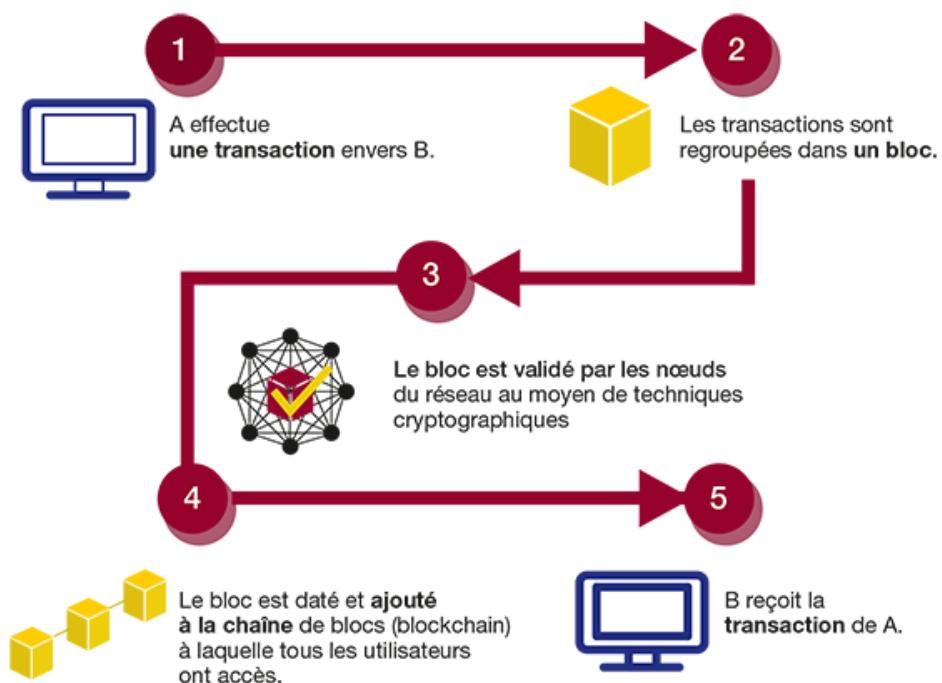


Figure 11 : Principe de fonctionnement de la blockchain

### 1.1.3 Les mécanismes de consensus

Les mécanismes de consensus (également appelés protocoles de consensus ou algorithmes de consensus) permettent aux systèmes distribués (réseaux d'ordinateurs) de fonctionner ensemble et de rester sécurisés. Un mécanisme de consensus dans un système crypto économique permet également de prévenir certains types d'attaques économiques. En théorie,

un attaquant peut compromettre le consensus en contrôlant 51 % du réseau. Les mécanismes de consensus sont conçus pour rendre cette "attaque des 51 %" irréalisable [Net 7].

### ➤ Proof-of-work (PoW)

Le PoW est le premier consensus public de la blockchain qui a été introduit dans le Bitcoin [14]. L'idée principale derrière ce mécanisme est que les nœuds de consensus sont en compétition pour créer de nouveaux blocs remplis de transactions traitées. Le gagnant partage le nouveau bloc avec le reste du réseau. La course est remportée par l'ordinateur le plus rapide à résoudre une énigme mathématique, qui établit le lien cryptographique entre le bloc actuel et le bloc précédent. La résolution de cette énigme est le travail de la "proof-of-work" [Net 7]. La sécurité du réseau est assurée par le fait qu'il faudrait 51 % de la puissance de calcul du réseau pour frauder la chaîne [Net 7].

Cependant le processus de PoW présente certains inconvénients tels qu'un débit inefficace, une latence élevée, et une consommation d'énergie élevée qui rendent le PoW inadapté à de nombreuses applications blockchain [14].

### ➤ Proof-of-stake (PoS)

Dans le cadre d'une approche Proof of Stake (PoS), il n'y a pas de processus de mining. Au lieu de cela, le travail nécessaire pour effectuer le processus de vérification est réparti entre les " validateurs " [15], l'algorithme détermine de manière aléatoire les validateurs pour créer les nouveaux blocs, et la probabilité qu'un nœud valide le prochain nouveau bloc est proportionnelle aux enjeux/actifs (p. ex., pièces) qu'il possède. En d'autres termes, au lieu d'effectuer des calculs élevés de résolution d'énigmes, dans PoS, les validateurs doivent prouver leur part dans le réseau en fonction de la chaîne actuelle [14].

Un système proof-of-stake est sécurisé par le fait qu'il faut 51 % du total des validateurs pour frauder la chaîne. De plus, en cas de comportement malveillant, une réduction de votre mise est appliquée [Net 7].

Cette approche réduit la complexité du processus de vérification décentralisé et peut donc permettre de réaliser d'importantes économies d'énergie et de coûts de fonctionnement et d'exploitation. Elle peut également contribuer à réduire les risques de centralisation par rapport au PoW en raison des économies d'échelle importantes des investissements miniers, et rendre les attaques sur le réseau plus coûteuses [15].

### ➤ Proof-of-authority (PoA)

Selon ce mécanisme, avant de devenir une autorité pour publier un bloc, le participant doit confirmer son identité dans le réseau. Contrairement à PoS, au lieu d'avoir des pièces de monnaie ou d'autres actifs, le PoA considère l'identité d'un participant comme un enjeu. En outre, il est supposé que les autorités sont présélectionnées et fiables pour publier un bloc.

Il est également pratique de détecter les autorités malveillantes et d'en informer les autres [14].

### ➤ PBFT (Practical Byzantine Fault Tolerance)

Principalement utilisé dans des environnements de confiance ou semi confiance et basé sur un algorithme qui requiert que 2/3 des nœuds du réseau pour considérer la transaction comme valide. PBFT est le modèle le plus amélioré des algorithmes de consensus, dont la complexité est presque réduite à un niveau polynomial. Il contient cinq états [26] :

- Demande (Le client envoie une demande au nœud maître et un horodatage est attribué)
- Pré-préparation (Le nœud maître enregistre la demande et diffuse un message de pré-préparation aux autres nœuds serveurs)
- Préparation (Les nœuds serveurs acceptent la demande et diffusent un message de préparation aux autres nœuds)
- Commit (Chaque nœud envoie un message commit et exécute les instructions du message de demande.)
- Reply (Les nœuds du serveur répondent aux clients)

### ➤ Proof-of-Reputation (PoR)

L'idée principale du PoR est de s'assurer que chaque nœud dispose d'un registre de réputation convenu, enregistrant la valeur de réputation de chaque nœud. Les nœuds à forte réputation publient des blocs sur le réseau peer-to-peer grâce à la preuve de réputation. La réputation de chaque nœud est modélisée et nécessite également d'être enregistrée et acceptée par tous les autres nœuds, ce qui nécessite une chaîne de données pour stocker la réputation des nœuds. Les nœuds à forte réputation vérifient le bloc de transaction reçu et les signatures. Dans le PoR, il y a quatre facteurs affectant la valeur de réputation d'un nœud : les transactions historiques, l'âge de la monnaie, la participation au consensus et le comportement illégal des nœuds [27].

### 1.1.4. Les catégories de la blockchain

#### ➤ Permissioned vs Permissionless

La distinction fondamentale entre ces deux types de blockchain est claire à partir des termes eux-mêmes. Une blockchain avec autorisation nécessite une approbation préalable avant d'être utilisée, tandis qu'une blockchain sans autorisation permet à quiconque de participer au système [Net 8]. Dans une chaîne sans autorisation, n'importe qui peut rejoindre le réseau de la blockchain et participer à la création d'un nouveau bloc. En revanche, dans une chaîne avec autorisation, seuls des nœuds prédéfinis et autorisés peuvent le faire [14].

#### ➤ Private vs. Public

Les blockchains peuvent également être classées en deux catégories : publiques et privées.

Les blockchains publiques sont véritablement décentralisées et sans permission, elles permettent une participation ouverte et le maintien d'une copie de la chaîne par n'importe qui. Habituellement, ce type de blockchain a un grand nombre d'utilisateurs anonymes. Contrairement aux chaînes publiques, dans les blockchains privées, certains utilisateurs sélectionnés/prédéfinis et de confiance sont autorisés à valider et à participer à la publication des nouveaux blocs, les autres utilisateurs publics ou autorisés du réseau ne peuvent lire que les données contenues dans les blocs donc la chaîne privée peut être partiellement décentralisée, ce type de blockchain est généralement développé pour être contrôlé par une organisation [14].

## 1.2 Sécurité des smart grids basée sur la blockchain distribuée

Actuellement, la plupart des solutions de sécurité des smart grids sont construites sur des modèles centralisés où les composants du réseau intelligent dépendent d'une autorité centrale pour obtenir des services tels que la facturation, la surveillance, les enchères, le commerce de l'énergie, etc. Bien que ces solutions fonctionnent correctement, plusieurs problèmes difficiles sont associés au système actuel de réseau intelligent : la topologie du réseau s'adapte et passe d'une topologie centralisée à un réseau décentralisé et entièrement automatisé pour permettre une plus grande interaction entre les composants. De même, le marché des réseaux intelligents est en train de passer d'un réseau de producteurs centralisé à un réseau interactif décentralisé. Dans cette évolution vers des systèmes décentralisés, l'application de la blockchain offre une opportunité de faciliter cette évolution.

### 1.2.1 Passage au système décentralisé de réseau intelligent

Dans une topologie centralisée, les générations d'énergie, les réseaux de transport et de livraison, et les marchés en quelque sorte dépendent d'entités centralisées ou intermédiaires. Dans ce système centralisé, les éléments du réseau intelligent interagissent et communiquent avec des entités centralisées qui peuvent surveiller, collecter et traiter les données et soutenir tous les éléments avec des signaux de contrôle appropriés. De plus, la transmission de l'énergie se fait généralement par le biais d'un réseau long distance afin de fournir de l'énergie aux utilisateurs finaux par le biais du réseau de distribution.

Malheureusement, en raison de la fragilité des énergies renouvelables, la conception actuelle du réseau intelligent soulève des inquiétudes, parmi elles : l'extensibilité, les lourdes charges de calcul et de communication, les attaques de disponibilité et l'incapacité de contrôler les futurs systèmes électriques.

En tant que tel, la transformation en système décentralisé est une tendance dans les réseaux intelligents pour apporter plus de dynamisme, d'intelligence et d'efficacité. L'infrastructure du réseau lui-même subit également une adaptation et évolue vers un réseau entièrement automatisé ayant des topologies décentralisées [14].

Le tableau 5 représente une brève comparaison entre le réseau intelligent normal et le futur réseau intelligent décentralisé.

Smart Grid	Smart Grid décentralisé
Transformé en utilisant plus d'énergies renouvelables et en s'intégrant avec les réseaux centralisés.	S'oriente vers la construction d'un système décentralisé en intégrant diverses ressources énergétiques distribuées.
Généralement, l'accent est mis sur l'intégration de technologies avancées de détection et de contrôle dans le réseau traditionnel.	Il s'agit essentiellement de la surveillance en temps réel, de l'ajustement automatique du contrôle et d'optimisation.
S'appuie sur des intermédiaires et des marchés centralisés.	Permet à un certain nombre d'utilisateurs de générer leur propre énergie et de partager les excédents par le biais du peer-to-peer.
Utilise des technologies de communication	Dominé par l'internet de l'énergie pour

avancées.	réaliser un partage de l'énergie et de l'information en continu, comme sur Internet.
Mise en place de communications bidirectionnelles.	Prend en charge les fonctionnalités plug-and-play avancées.
Coûts de calcul et de communication élevés.	Les coûts sont répartis entre les entités sur le réseau.
Moins d'options pour étendre le réseau.	Ont la possibilité d'étendre rapidement et en grand nombre de connectivité.
Serait affecté par un point unique de défaillance.	Résilience contre un point de défaillance unique.
Intégré uniquement aux réseaux d'énergie électrique.	Intégré avec d'autres réseaux d'énergie également.
Dépend toujours du système régional de contrôle du système.	Permet l'accès en douceur à des ressources énergétiques distribuées massives.

*Tableau 4 : Une brève comparaison entre le Smart Grid et le Smart Grid décentralisé [14]*

### 1.2.2 Motivations de l'application de la blockchain dans le paradigme SG

L'utilisation de la technologie blockchain pour sécuriser les SG contre les cyberattaques pourrait être une approche prometteuse. En raison de ses caractéristiques et de ses fonctionnalités, la blockchain pourrait rendre la SG plus robuste contre les FDIA [16].

La blockchain permet d'améliorer la sécurité, la confidentialité et la confiance tout en aidant à éliminer les obstacles pour devenir un système plus décentralisé et résilient. Nous présentons dans le tableau 6 les objectifs de sécurité, de confidentialité et de confiance nécessaires pour le réseau intelligent, et aussi, comment la blockchain peut atteindre ces objectifs [14].

Objectif	Comment la blockchain peut le réaliser
<b>Confidentialité</b>	Techniques cryptographiques
<b>Intégrité</b>	Structure de données protégée crypto-graphiquement par une fonction de hachage, arbre de Merkle, nonce (chiffres utilisés une fois) et horodatage. Les enregistrements manipulés peuvent être détectés et empêchés
<b>Authentification</b>	Enregistrements signés à l'intérieur des blocs par les clés privées des utilisateurs, de sorte qu'il est possible de vérifier que seul l'utilisateur valide l'a envoyé.
<b>Auditabilité</b>	Enregistrements/transactions publiquement disponibles dans la blockchain publique
<b>Autorisation et contrôle d'accès</b>	Autorisation et contrôle d'accès sont définis par l'utilisateur et reposant sur un contrat intelligent et les certificats d'attributs
<b>Confiance</b>	Algorithmes de consensus, la confiance n'est pas placée sur des centrales/intermédiaires, elle est plutôt distribuée parmi les entités du réseau
<b>Transparence</b>	Transparence totale grâce au maintien d'un registre distribué immuable comprenant tous les enregistrements, les transactions et les événements et journaux
<b>Disponibilité</b>	Architecture distribuée permettant à plusieurs entités d'établir des connexions avec d'autres entités et de répliquer une copie complète de la blockchain
<b>Automaticité</b>	La blockchain et les contrats intelligents offrent l'automaticité où les entités peuvent communiquer et échanger des valeurs de pair à pair par la blockchain et exécuter des actions automatiquement par un contrat intelligent.

*Tableau 5 : Objectifs communs de sécurité, et la manière dont Blockchain peut aborder.*

La vulnérabilité des données est devenue un problème incontournable, dans les systèmes électriques conventionnels, une attaque est considérée comme réussie si les cybers attaquants altèrent les données de mesure des compteurs au niveau local, remplacent les paquets de données transmis au centre de contrôle via un canal de communication ou bien piratent le centre de contrôle.

Dans ce qui suit, on va étudier les différentes solutions basées sur la blockchain contre l'attaque false data injection dans les smart grid, et on va fournir une comparaison selon les spécificités de chaque solution.

### 1.3 Protection des données distribué basé sur la blockchain contre FDIA

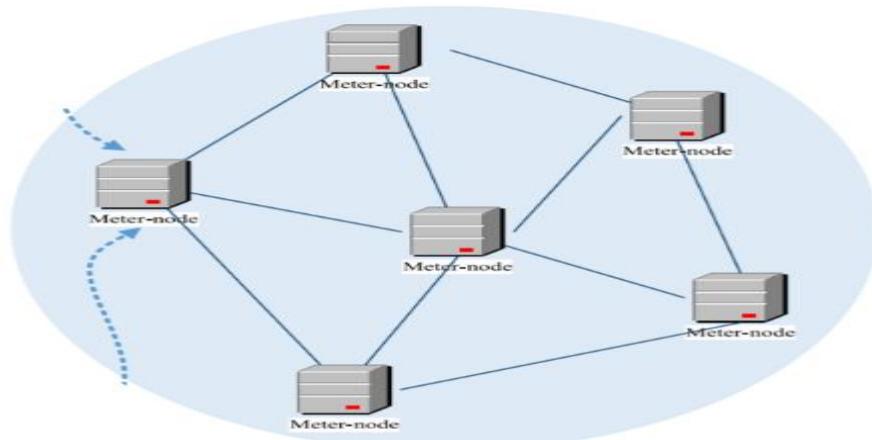
Dans cette section on va voir le framework proposé par [17] qui est basé sur le consensus et exploite les caractéristiques particulières de l'environnement du réseau électrique, où une attaque n'aboutit à une manipulation réussie que si l'attaquant manipule (ou remplace) des paquets de données sur une majorité de canaux, ou s'introduit dans un nombre suffisant de compteurs pour manipuler les données.

#### 1.3.1 La Reconfiguration de réseau

Certaines infrastructures du système doivent être mises à jour ou remplacées pour faciliter le fonctionnement de cette solution :

- Les données et les mesures sont collectées en temps réel du réseau.
- La couche de communication est isolée de l'internet.
- Les smart meters sont distribués géographiquement et ils sont composés d'un dispositif de collecte des données, d'un émetteur de signaux, d'un récepteur de signaux et d'un dispositif de traitement des données.

Le compteur/capteur agit comme un nœud, le graphe correspondant au réseau des compteurs-nœud est illustré dans la figure 14.



*Figure 12 : Réseau compteur-noeud*

- Il existe un chemin de communication reliant chaque paire distincte de nœuds.
- Seuls les compteurs/capteurs qui sont autorisés par le réseau peuvent effectuer la fonction d'acquisition de données, donc le réseau de compteurs-noeuds peut être considéré comme un réseau de blockchain privé.
- Les interactions entre les nœuds dans le réseau sont automatiquement réalisées sur la base d'un certain mécanisme de consensus (sans interaction humaine).

Chaque compteur doit posséder des caractéristiques fonctionnelles qui ne sont pas courantes dans les compteurs largement déployés aujourd'hui tels que :

- Une adresse unique.
- Un logiciel spécifique permettant de générer une clé publique et une clé privée.
- Une mémoire vive, du matériel de calcul, un dispositif de collecte de données, un émetteur et récepteur de signaux et un dispositif de traitement des données.
- La communication entre les compteurs par des canaux de communication avec ou sans fil.

### 1.3.2 Mécanisme de fonctionnement

Les données collectées sont stockées dans un grand livre sous forme de blocs connectés qui existent sous forme distribuée dans la mémoire de chaque compteur. Avant le stockage, certaines procédures sont nécessaires pour garantir l'exactitude des données.

- Diffusion des données.
- Vérification des données via un mécanisme de vote.
- Accumulation du contenu des données dans le bloc.
- Processus d'extraction.
- Vérification du résultat de l'extraction via un mécanisme de vote.

### 1.3.3 Mécanisme de Consensus

Chaque bloc est constitué de:

- **Numéro de bloc** : utilisé comme titre de bloc.
- **Données** : représente toute la data encapsulée dans le bloc.
- **Timestamp** : le temps où la data est encapsulé dans le bloc.
- **Previous hash results** : le résultat de hach de block précédent.
- **Hash result** : le résultat de hash de bloc courant.
- **Nonce solution** : solution du problème de puzzle du bloc.

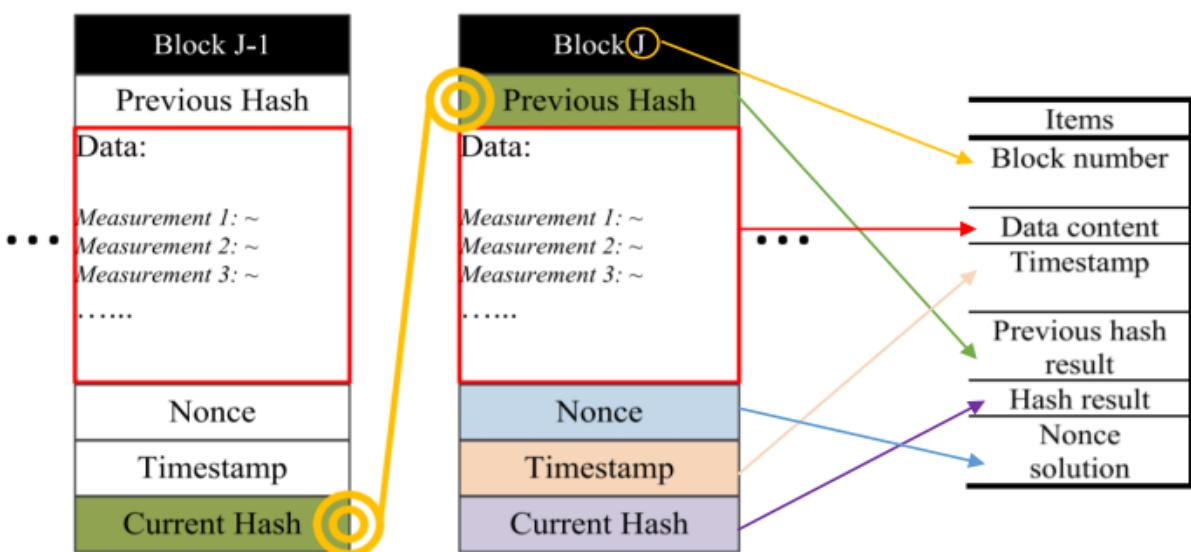


Figure 13 : Contenu des blocs et connexions des chaînes.

1- Pré-traitement :  $S = \text{block number} + \text{data content} + \text{time point} + \text{previous hash result} + \text{nonce}$

2-  $\text{FinalHash} = \text{hash}(\text{SHA256}, \text{hash}(\text{SHA256}, S))$

Le problème du puzzle est de trouver la valeur de nonce appropriée pour que la valeur de Final Hash soit inférieure à une cible T donnée  $\text{FinalHash} \leq T$

3- Certains nœuds peuvent fonctionner comme des mineurs en essayant de résoudre le problème du puzzle de manière indépendante.

4- Une fois que le premier mineur a trouvé le nonce, il diffuse la valeur aux autres nœuds pour leur permettre de vérifier si la solution est correcte en validant le nonce où elle satisfait  $\text{FinalHash} \leq T$

5- Le mécanisme de vote distribué basé sur l'adresse est à nouveau utilisé pour voter sur le résultat de la vérification.

## 1.4. La gestion décentralisée de données multi-chaînes pour les systèmes d'alimentation

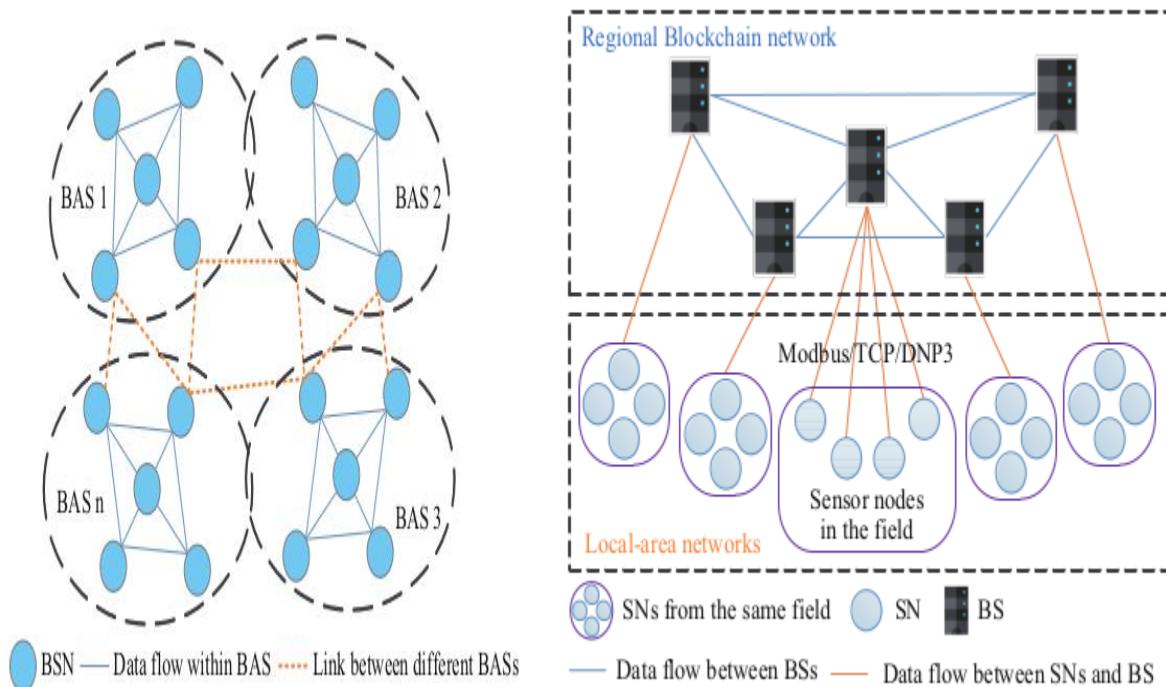
La solution présentée dans [24] propose un framework multi-chaînes basé sur la blockchain, où les mesures des capteurs sont extraites en blocs par des stations de base en utilisant le protocole de consensus Practical Byzantine Fault Tolerance (PBFT).

### 1.4.1 La reconfiguration de réseau

Le framework proposé permet de conserver des enregistrements des données dispersées dans chaque station de base BS du système en appliquant la technologie blockchain.

Les participants à un réseau de blockchain doivent avoir une bonne puissance de calcul et de bonnes capacités de communication. Les unités de contrôle telles que les disjoncteurs, les relais, les actionneurs, etc. sont exclues de ce cadre, alors que seuls les capteurs/compteurs sont inclus et agissent en tant que fournisseurs de données.

Le réseau est composé de nœuds de capteurs (SN), BS et du réseau de communication. Les SN sont chargés d'acquérir des données de mesure à partir du réseau et de les télécharger vers la station de base. Tandis que la station de base collecte les données, les vérifie et les partage avec d'autres BS, puis génère finalement une blockchain des données de mesure.



*Figure 14 : Vue d'ensemble du réseau.*

### 1.4.2 Mécanisme de Fonctionnement

Le processus pour que les données de mesure soient capturées par les SNs sur le terrain jusqu'à leur encapsulation dans un bloc de données du grand livre unifié passe principalement par trois étapes, les BSs collectent les données auprès des SNs, les BSs partagent les données entre elles dans le même BAS, et les données sont extraites en un bloc et enfin enchaînées au grand livre.

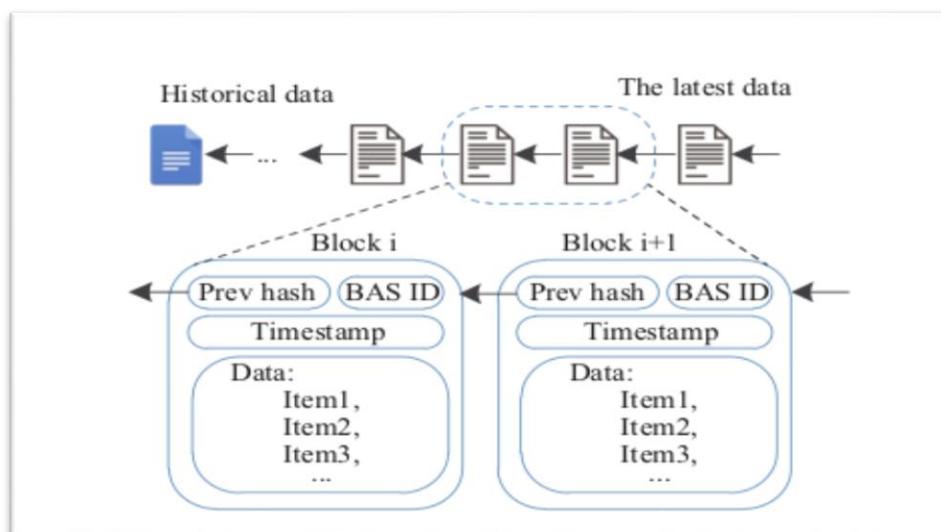
Après avoir collecté suffisamment de données de mesure, ou lorsque le temps est écoulé, le BSN diffuse les données aux autres BSN appartenant au même BAS. Par conséquent, le nœud doit savoir où se trouvent les autres nœuds et comment les localiser car les nœuds fonctionnent dans un réseau privé et fonctionnent de manière complètement automatique.

### 1.4.3 Mécanisme de Consensus

Les données de mesure dans le MP proviennent de SNs connectés ou d'autres BSNs, constituent des transactions non concurrentes du réseau. Ces transactions sont triées par ordre chronologique et seront finalement extraites dans un bloc pour être diffusées à l'ensemble du BAS. Lorsqu'un bloc est accepté comme étant valide, les données de mesure de ce bloc sont effacées du MP pour libérer la mémoire. Les blocs sont liés dans une chaîne appelée "grand livre".

#### ➤ Structure de bloc

Il y a quatre composants dans un bloc, ce sont le numéro de bloc, le hachage précédent, le corps des données et l'horodatage, comme le montre la figure 15.



*Figure 15 : Blockchain data structure.*

La signification de ces composants est illustrée dans le Tableau 7.

Items	Signification
hash précédent	Le résultat du hachage du bloc précédent
ID BAS	L'id de BAS
Horodatage	L'horodatage lorsque le bloc est généré
Données	Les données de mesure qui sont encapsulées dans le bloc

*Tableau 6 : Item et signification.*

#### ➤ **Algorithme de mining**

Dans ce système, seuls les nœuds autorisés sont acceptés pour rejoindre le réseau, en d'autres termes, il s'agit d'un réseau blockchain privé avec une quantité limitée de nœuds. Dans cette situation, le protocole Practical Byzantine Fault Tolerance (PBFT) serait adopté qui est un algorithme pratique pour la réPLICATION de machines à état qui tolère les fautes byzantines.

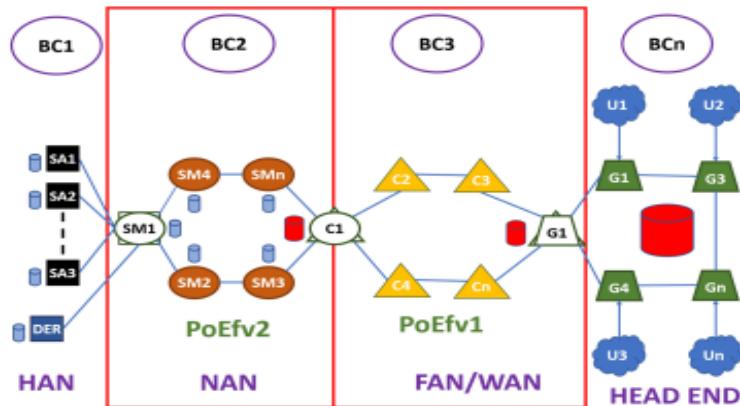
### **1.5. A Novel Multitier Blockchain Architecture to Protect Data in Smart Metering Systems**

La solution [25] propose une architecture de cyber sécurité basée sur la blockchain visant à garantir l'intégrité et la sécurité des données. Les principales contributions de ce travail sont la blockchain multi-niveaux pour et son implémentation dans le SM, ainsi qu'un algorithme de consensus de base et plus adaptable appelé proof-of-efficiency (PoEf) pour le réseau électrique.

#### **1.5.1 La reconfiguration de réseau**

L'architecture proposée est basée sur l'architecture AMI, où l'on distingue quatre domaines élémentaires : Le réseau domestique, le réseau de voisinage (Neighborhood Area Network) (NAN), Réseau de zone de terrain ou réseau étendu et le Réseau du service public (tête de réseau).

L'architecture proposée pour protéger les données est présentée dans la figure 16

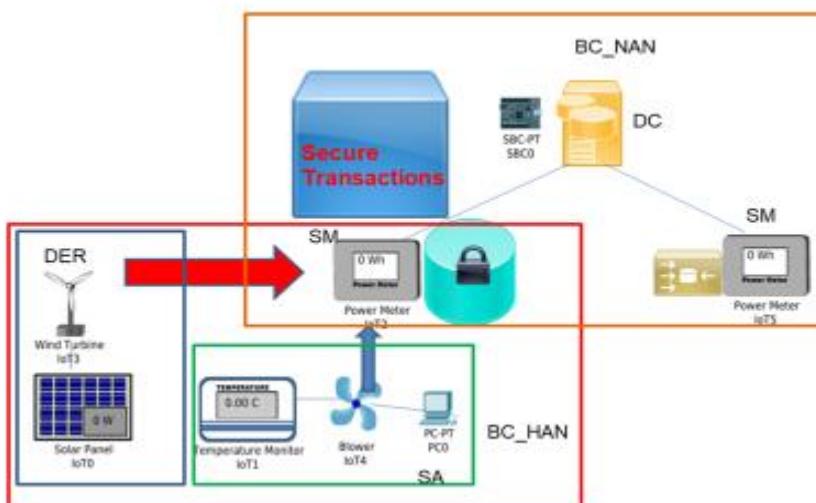


*Figure 16 : Architecture proposée basée sur quatre zones élémentaires de l'AMI.*

Dans le réseau HAN, il y a des SAs qui permettent de mesurer une consommation d'énergie et de la signaler au SM pour qu'il puisse l'évaluer pour générer une blockchain par maison, bureau ou industrie. La consommation de tous les appareils ménagers conventionnels est stockée dans la SM.

Un autre composant important est le DER qui permet de produire de l'énergie. L'excédent de production d'énergie est injecté dans le réseau et la transaction est enregistrée dans la blockchain.

Le SM est à l'intérieur du HAN. Veuillez noter que chaque SM est associé à un HAN particulier, et tous les SM sont associés à un DC. Dans le HAN, une autre blockchain est formée avec les données de tous les SMS du réseau. Les données stockées ici ne sont qu'un résumé des données de chaque SM, car la confidentialité des données doit être protégée à tout prix. Le SM et le DC ont besoin d'une base de données (DB) pour stocker les données des transactions effectuées comme montré dans la figure 17.



*Figure 17 : Le stockage des données, DB dans SA, SM et DC.*

Le niveau suivant de l'architecture se situe au niveau du FAN/WAN où les DC résument les données de chaque WAN et requièrent donc une plus grande capacité de stockage. Enfin, dans les serveurs de l'entreprise, les données de tous les clients sont stockées dans une blockchain qui peut être interopérable avec d'autres services publics.

### 1.5.2 Mécanisme de fonctionnement

#### ➤ Blockchain dans le HAN (Niveau 1)

Tous les appareils électriques consomment de l'énergie électrique et le SM mesure la consommation d'énergie de ces appareils connectés dans le réseau HAN. En outre, SM mesure la production d'énergie des DERs. Dans le cas des SAs et des DERs, une base de données embarquée pour stocker leur consommation/production d'énergie. La structure de données dans ce niveau de la blockchain est présentée par figure 18.

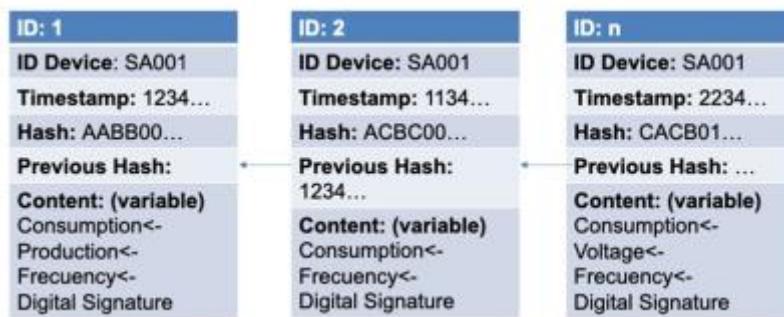


Figure 18 : Structure de données de la blockchain mise en œuvre au niveau du réseau HAN.

#### ➤ Blockchain dans NAN (Niveau 2)

Le SM rapporte des informations concentrées sur la consommation/production de tous les dispositifs de HAN et résume un bloc avec ces informations sous forme d'une transaction énergétique. Cette transaction énergétique est signée entre le SM et le service public d'électricité afin de passer à la blockchain et procéder à sa validation.

Les transactions dans les SMS peuvent être, en plus des relevés de mesure et de production d'énergie électrique, des événements de connexion/reconnexion, des alarmes, et toute autre situation qui est signalée dans les journaux du SM ou qui est notifiée par la compagnie service public d'électricité.

La structure des données de la blockchain à ce niveau est illustrée à la figure 19.



Figure 19 : Structure de données de la blockchain mise en œuvre dans le niveau NAN.

#### ➤ Blockchain dans FAN/WAN (Niveau 3)

À ce niveau, la blockchain est constituée de tous les DCs d'une région. Cela dépend de la densité des nœuds ou des distances entre les nœuds, le nombre de niveaux de combien de niveaux de blockchains supplémentaires doivent être mis en œuvre. La figure 20 montre le condensé des transactions de ce niveau de blockchain.

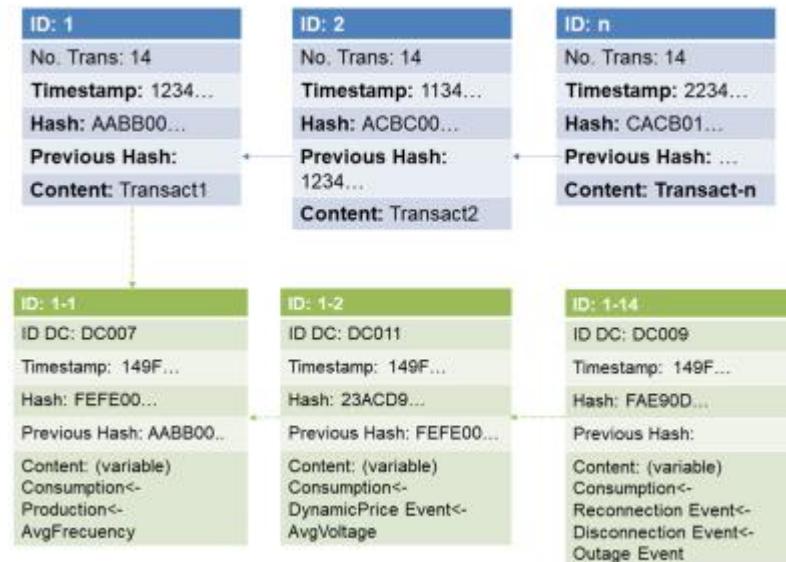


Figure 20 : Structure de données de la blockchain mise en œuvre au niveau du FAN/WAN

#### ➤ Blockchain dans datacenter (Niveau N)

Les serveurs de tête de l'AMI concentrent toutes les transactions dans les différents niveaux de blockchains. Cette blockchain pourrait être énorme et les données ne sont jamais effacées comme dans une blockchain traditionnelle.

### 1.5.3 Mécanisme de Consensus

#### ➤ Proof-of-Efficiency

Cet algorithme est mis en œuvre aux différents niveaux de la blockchain en tenant compte des caractéristiques de chaque niveau. En particulier, son implémentation dans le HAN sera décrite, dans laquelle les nœuds qui ont eu la meilleure consommation d'énergie en fonction de leurs performances antérieures seront récompensés en effectuant de l'analyse des données dans la blockchain au cours d'une période donnée, en recherchant les modèles de consommation (vérifiant l'efficacité énergétique) et la production d'énergie (en tenant compte de facteurs tels que la qualité de l'énergie produite). Il est considéré que toutes les transactions de production d'énergie qui ne respectent pas les paramètres de qualité de l'énergie sont rejetées. La description de cet algorithme est présentée dans l'annexe 3.

## 1.6. Tableau comparatif des solutions trouvées

Dans ce qui suit, on va fournir un tableau comparatif des différentes solutions blockchain [17][24][23] qu'on a déjà présenté :

Framework	[17]	[24]	[23]
Type de blockchain	privée	privée	privée
Niveau où la blockchain est utilisée	Dans un seul niveau : Réseau de distribution	Dans un seul niveau: réseau WAN/FAN	Dans 4 niveaux : HAN, NAN, WAN et le Centre de contrôle
Algorithme de consensus	Proof-of-work	PBFT	PoA (Proof of Authority) avec un mélange de PoW.
Noeuds de la blockchain	Les smarts meter	DC au niveau WAN/FAN	HAN: équipement électriques NAN: LES Smarts meter WAN: DC Centre de contrôle: serveurs
Noeuds Validateurs	Pré-spécifiés/au hasard	Élu par l'algorithme PBFT	Choisie en fonction de ses capacités matérielles la première fois que la blockchain commence à fonctionner

Tableau 7 : Comparaison des solutions blockchain trouvées.

## 2. Détection de menaces avec l'intelligence artificielle

Pour assurer un service fiable et prévisible dans le réseau électrique, il est important de mesurer le niveau de confiance des composants critiques et les sous-stations. Un autre problème majeur dans le smart grid est le vol d'énergie car il génère d'énormes pertes financières pour les entreprises de services publics. Il n'est pas possible d'inspecter manuellement un tel vol dans une grande quantité de données. De ce fait, pour détecter cette attaque, des mesures peuvent être faites avec précision en utilisant de l'apprentissage automatique, des méthodes statistiques, ainsi que l'analyse comportementale afin de détecter un quelconque déroutement d'énergie dans le smart grid.

### 2.1. Définition

Le Deep Learning est un sous-ensemble du Machine Learning, qui est lui-même un sous-ensemble de l'Intelligence Artificielle. L'intelligence artificielle est un terme général qui fait référence aux techniques qui permettent aux ordinateurs d'imiter le comportement humain alors que le deep learning représente un ensemble d'algorithmes entraînés sur des données qui rendent tout cela possible. Inspiré de la structure du cerveau humain, les algorithmes du deep learning tentent de tirer des conclusions similaires à celles des humains en analysant continuellement les données avec une structure logique donnée. Pour y parvenir, ils utilisent une structure multicouche d'algorithmes appelés réseaux de neurones [Net 10].

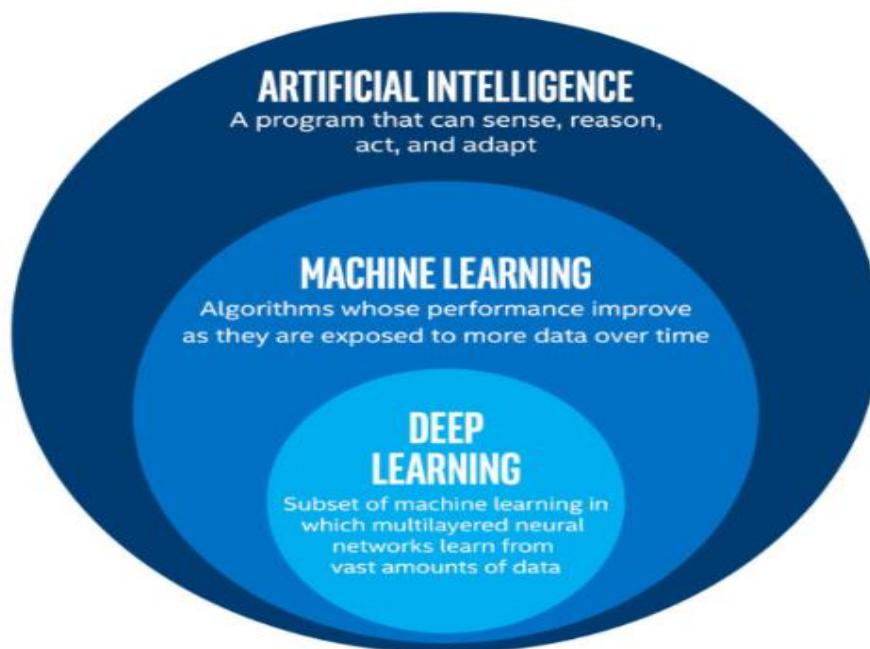


Figure 21 : L'intelligence artificielle.

## 2.2. Les Modèles Deep Learning

Un modèle d'apprentissage en profondeur traite les données collectées et crée des modèles utiles à la prise de décision dans divers cas d'utilisation. Sur la base de la configuration des couches de réseaux de neurones, les modèles d'apprentissage en profondeur utilisés pour la prise de décision dans plusieurs domaines d'application sont classés en cinq grandes catégories [29]. Les catégories des modèles d'apprentissage en profondeur sont citées comme suit :

### 2.2.1. Convolutional Neural Network (Réseau de neurones à convolution)

Les ConvNets sont conçus pour traiter des données qui se présentent sous la forme de multiples, par exemple une image couleur composée de trois matrices 2D contenant les intensités des pixels dans les trois canaux de couleur. De nombreuses modalités de données se présentent sous la forme de tableaux multiples : 1D pour les signaux et les séquences, y compris le langage, 2D pour les images ou les spectrogrammes audio et 3D pour les images vidéo ou volumétriques [30].

### 2.2.2. Recurrent Neural Network (Réseau de neurones récurrent)

Un modèle CNN surpasse les données d'images utilisées en entrée. Cependant, le réseau de neurones récurrent (RNN) utilise des données séquentielles ou chronologiques pour générer des modèles. Les applications célèbres de RNN pour les solutions basées sur la blockchain incluent la reconnaissance vocale, la conversion parole-texte, la recherche vocale et le traitement du langage naturel (NLP). De plus, les données d'entrée sont indépendantes les unes des autres dans CNN ; tandis que les entrées précédentes sont liées et influencent la sortie dans les modèles RNN [29].

### 2.2.3. Generative Adversarial Networks (Réseaux antagonistes génératifs)

Le modèle génératif apprend les modèles de manière non supervisée et est capable de générer des données uniques. Plus précisément, il s'agit d'une forme de modélisation générative qui utilise des techniques d'apprentissage en profondeur telles que les réseaux de neurones convolutionnels. De par sa conception, le modèle GAN se compose d'un générateur et d'un réseau discriminateur. Le générateur est chargé de produire de nouveaux exemples, tandis que le discriminateur apprend à classer les données comme vraies ou fausses [29].

### **2.2.4. Deep Reinforcement Learning (Apprentissage par renforcement profond)**

DRL s'inspire des théories du comportement humain basées sur l'écologie comportementale et permet aux systèmes experts de comprendre les données plus précisément. Les agents intelligents prennent des mesures dans un environnement composé de modèles DRL pour apprendre. De plus, les agents sont implicitement validés ou pénalisés en fonction de leur comportement. Les comportements qui conduisent au résultat souhaité sont récompensés, ce que l'on appelle un modèle basé sur l'apprentissage renforcé [29].

### **2.2.5. Geometric Deep Learning (Apprentissage profond géométrique)**

Il s'agit d'une variante d'apprentissage profond qui se concentre sur le développement de réseaux de neurones basés sur des données non euclidiennes. Un graphe est un exemple spécifique de données non euclidiennes. La modélisation des données peut être effectuée avec moins d'efforts et de ressources tout en utilisant des données basées sur des graphiques. Les graphiques sont entrés dans les modèles géométriques d'apprentissage en profondeur plutôt que les données sous la forme traditionnelle des réseaux de neurones génériques [29].

## **2.3. Protection contre le vol d'énergie basé sur l'IA**

### **2.3.1. Analyse basée sur la modélisation ARIMA et validation des relevés de consommations [19].**

L'intervalle de confiance ARIMA fournit une limite aux mesures et sert de bon détecteur de mesures invalides pour les compteurs défectueux. Cependant, ces limites ne sont pas suffisantes pour détecter les attaques dans lesquelles l'attaquant a une connaissance complète du système. Nous considérons un modèle d'attaque spécifique dans lequel l'attaquant vole de l'électricité à un voisin pour un gain monétaire. Soit la consommation de l'attaquant à l'instant  $t$  est  $A_t$  et celle de son voisin est  $X_t$ . L'attaquant compromet le compteur intelligent de son voisin et lui fait signaler une consommation  $X't > X_t$ . En même temps, l'attaquant déclare sa propre sous-consommation en compromettant la lecture de son compteur à  $A't = A_t - (X't - X_t)$ . Par conséquent, il vole un montant positif au voisin qui est de  $(X't - X_t)$ . Il est facturé pour  $A't < A_t$  alors que le voisin est facturé pour  $X't > X_t$ .

Notons qu'avec ceci, l'attaquant a évité la vérification de l'équilibre faite dans le réseau de distribution électrique car on trouverait que la somme attendue telle que rapportée par les compteurs intelligents ( $X't + A't$ ) correspond à la somme mesurée de ( $X_t + A_t$ ).

Ainsi, l'attaquant a évité la vérification de l'équilibre. Pour ce faire, il a augmenté les relevés de consommation des compteurs intelligents de son voisin.

Sans le mécanisme de détection ARIMA en place, l'attaquant peut voler une quantité arbitraire d'électricité. Il n'est limité que par les limites physiques du système de distribution électrique. Plus précisément, les lignes de distribution électrique sont classées en fonction du courant maximal qu'elles peuvent transporter. Si la demande de l'attaquant augmente (alors que la tension de distribution est maintenue approximativement constante par la compensation de la puissance réactive), le courant dans les lignes de distribution augmente. Cela génère de la chaleur sous la forme de pertes  $I^2R$ , où  $I$  est le courant et  $R$  est la résistance. Si le courant augmente au-delà du seuil normal, les lignes dépassent leurs limites thermiques et les dommages qui s'ensuivent peuvent entraîner des coupures de courant ou d'autres pannes d'équipement, qui sont une indication évidente de consommation anormale. Par conséquent, nous supposons que l'attaquant essaiera d'éviter la détection en opérant de manière à ce que sa propre consommation se situe dans ces limites physiques.

### 2.3.2. Algorithme CPBETD (Consumption pattern-based energy theft detector) [20]

Le CPBETD est conçu pour détecter des anomalies dans le modèle de consommation, il comporte deux phases, à savoir la phase d'entraînement et la phase d'application.

**La phase d'entraînement** se produit comme suit :

- L'algorithme est formé pour estimer le TL dans les lignes de transmission au sein du réseau de proximité (NAN).
- L'étape suivante est le prétraitement des données.
- Une fois que les données sont converties dans le format approprié, l'algorithme k-means est exécuté sur l'ensemble de données.
- L'étape suivante consiste à préparer un dataset pour entraîner le classifieur. Bien qu'un dataset d'échantillons bénins pour chaque client soit facilement obtenu à l'aide de données historiques, des échantillons malveillants peuvent ne pas être disponibles, car le vol d'énergie peut ne jamais ou rarement se produire pour un client donné.
- La prochaine étape est la formation du classificateur. Plusieurs techniques de régression et de classification existent, par exemple SVM qui a de bonnes performances.
- Enfin, Les paramètres SVM peuvent être ajustés pour atteindre différentes performances en termes de DR/FPR.

Concernant **la phrase d'application**, son déroulement est comme suit :

- Pour chaque quartier, un ou plusieurs compteurs à transformateur, mesurent l'électricité totale fournie aux clients de la zone, ETM(t). Cette valeur est comparée à la quantité totale de consommation rapportée par les compteurs intelligents du transformateur de distribution correspondant
- Chaque nouvel échantillon est prétraité et converti en format approprié compatible avec l'ensemble de formation.
- SVM est appliqué à un nouvel échantillon pour déterminer s'il appartient à la classe bénigne ou attaque.
- Si l'étape 1 n'a pas détecté d'anomalie et que le nouvel échantillon a été classé bénin par le SVM, le nouvel échantillon est ajouté dataset bénignes et l'attaque correspondante des modèles sera générée et ajoutée dataset d'attaque.
- Si NTL a été détecté à l'étape 1 et que le classificateur a reconnu une attaque, un comportement suspect du compteur intelligent est signalé. Si un vol est confirmé, des échantillons de la base de données temporaire sont ajoutés au dataset d'attaque.

Des résultats de comparaison entre cet algorithme et d'autres techniques de détection ont été faits et seront illustrés un peu plus en bas.

### 2.3.3. Algorithme GBTD (Gradient boosting theft detector) [51]

Alors que la plupart des algorithmes ML existants se concentrent sur le réglage des hyper paramètres des classificateurs, GBTD, se concentre sur le prétraitement basé sur l'ingénierie des caractéristiques pour améliorer les performances de détection ainsi que la complexité temporelle. GBTD améliore à la fois le taux de détection (DR) et le taux de faux positifs (FPR) de ces GBC en générant des caractéristiques stochastiques comme l'écart type, la moyenne, la valeur minimale et maximale de la consommation électrique quotidienne.

L'algorithme GBTD se base sur les trois derniers gradients boosting classifiers (GBCs) : boosting de gradient extrême (XGBoost), amplification catégorielle (Cat Boost) et lumière méthode d'amplification de gradient (LightGBM). Sur les trois GBCs, en termes de DR, les deux LightGBM et CATBoost sont plus performants que XGBoost. Cependant, LightGBM s'est avéré être le classificateur le plus rapide, avec le meilleur FPR. Il a été prouvé numériquement que le GBTD avec l'ingénierie des caractéristiques minimise non seulement le FPR mais réduit également l'espace de stockage des données du client ainsi que le temps de traitement.

La procédure est la suivante :

### 1. Génération de nouveaux cas de vol

On génère les six cas de vol mis à jour de la référence 10. L'idée fondamentale ici est de générer des modèles de comportements malveillants plus pratiques en temps réel et de les étiqueter en vue de l'utilisation d'algorithmes ML supervisés. Si l'utilisation réelle du client est  $xt$ , alors voici des modèles de vol révisés ( $t \in [1,48]$ ) :

1.  $t1(xt) = xt * random(0.1,0.9)$
2.  $t2(xt) = xt * rt$  ( $rt = random(0.1,1.0)$ )
3.  $t3(xt) = xt * random[0,1]$
4.  $t4(xt) = mean(x) * random(0.1,1.0)$
5.  $t5(xt) = mean(x)$
6.  $t6(xt) = xT - t$ , (où  $T$  est la taille de l'échantillon par jour, c'est à dire 48)

### 2. Evaluation de l'algorithme avec les cas de vol

Les nouveaux cas de vol sont utilisés pour générer des échantillons malveillants. L'évaluation de l'algorithme GBTD et sa comparaison avec l'algorithme CPBETD existent. Pour 3 les nouveaux cas de vol, le tableau 9 confirme qu'en termes de DR moyen (94 à 97% pour GBTD vs 88% pour CPBETD) et FPR (5 à 7% pour GBTD vs 15% pour CPBETD), le premier algorithme est le plus efficace, le premier est plus dominant que le second. Cela implique également que ces nouveaux cas de vol sont plus difficiles à détecter pour l'algorithme CPBETD basé sur un SVM.

Old/New	CBETD classifier	GBTD classifiers			
		SVM	XGBoost	CATBoost	LightGBM
<b>DR (%)</b>	94/88	95/94	96/97	96/97	96/97
<b>FPR (%)</b>	11/15	7/6	8/5	7/7	7/7

Tableau 8 : Performance moyenne de détection et comparaisons de 5000 clients entre les classificateurs CPBETD et GBTD.

#### 2.3.4. Analyse basée sur des caractéristiques comportementales

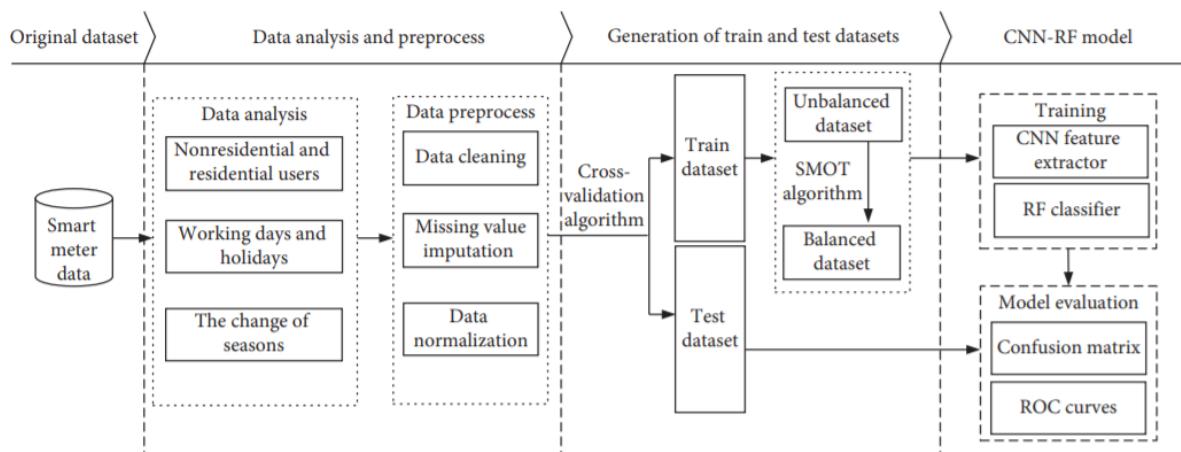
Les communications du réseau sont effectuées utilisant de nombreux protocoles applicatifs, données cellulaires, maillage radiofréquence (RF) et filaire ou sans fil TCP/IP.

De nombreuses détections d'anomalies comportementales traditionnelles des procédés inspectent une trame de transport ou des éléments de paquet pour détecter des trafics suspects. Cependant, dans de nombreux cas, les messages sont cryptés, ce qui rend difficile ou impossible l'inspection d'éléments essentiels de trame ou de paquet.

Dans de tels cas, l'analyse des données de couche d'application non chiffrées sont utilisées. Des méthodes de détection d'anomalies comportementales qui s'appuient sur des données capturées à partir des journaux système non cryptés dans le Smart grid (syslogs) sont utilisées. Le journal système SG contient l'authentification client enregistrée, actions de contrôle d'accès, actions d'administration du réseau et du système. Chaque enregistrement de syslog contient un horodatage, la gravité, installation, étiquette et champ de message. De plus, chaque enregistrement syslog contient un champ lisible où des informations complémentaires concernant les enregistrements y sont insérées. Dans la recherche, les paquets de sous-station SG syslog et TCP/IP les données de capture sont échantillonnées à des vitesses d'échantillonnage spécifiées et intervalles de temps avec chaque intervalle appelé fenêtre d'événement (EW). Les données collectées sont organisées en jeux de données horodatés [18].

### **2.3.5. Analyse basée sur le modèle hybride CNN-RF ( Convolutional Neural Network and Random Forest )**

CNN-RF est un nouveau modèle hybride de réseau neuronal à convolution et de forêt aléatoire (CNN-RF) pour la détection automatique du vol d'électricité. Ce modèle permet de fournir aux services publics une liste classée de leurs clients, en fonction de leur probabilité d'avoir une anomalie dans leur compteur électrique. Comme le montre la figure 22, la détection de vol d'électricité est divisée en trois étapes principales : analyse des données et prétraitement, génération d'ensembles de données d'entraînement et de test, classification à l'aide du modèle CNN-RF [21].



**Figure 22 :** Flux de détection de vol d'énergie avec CNN-RF

Le tableau suivant montre des résultats de classification de CNN-RF où la classe 0 est un modèle d'anomalie et la classe 1 est un modèle normal.

	Precision	Recall	F1 score
<b>Class 0</b>	0.97	0.96	0.96
<b>Class 1</b>	0.98	0.98	0.98
<b>Average/total</b>	0.97	0.97	0.97

Tableau 9 : Résumé des scores de classification de CNN-RF.

### 2.3.6. Analyse basée sur le modèle hybride CNN-SVM (Convolutional Neural Network and Support vector machine)

Le schéma hybride basé sur un réseau neuronal convolutif et une machine à vecteurs de support est utilisé pour l'identification du vol électrique dans le réseau électrique. Le réseau neuronal convolutif est développé pour l'extraction de caractéristiques significatives et la machine à vecteurs de support classe les caractéristiques extraites en vol et non-vol [28].

Le cadre de la méthode proposée est illustré dans la figure suivante :

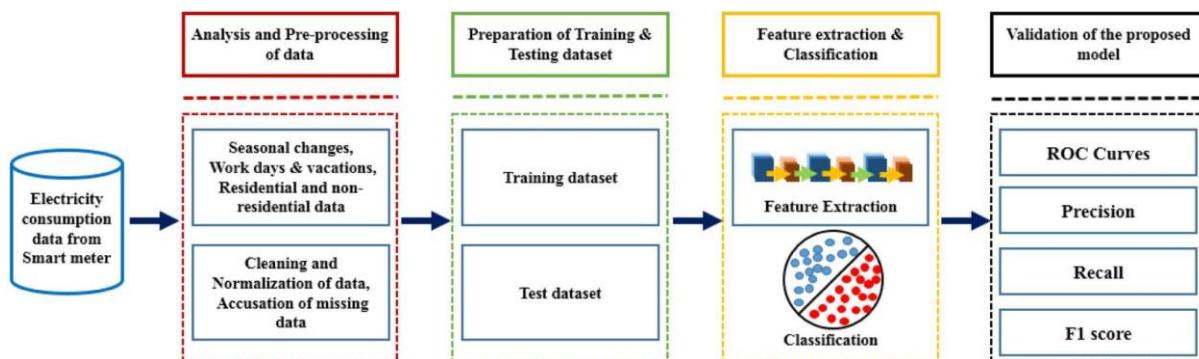


Figure 23 : Architecture du modèle CNN-SVM.

Le système de détection de vol d'électricité proposé comprend principalement quatre parties : analyse et prétraitement des données, préparation des données de formation et de test, extraction des caractéristiques utiles et classification, validation du modèle proposé.

## 3. Revue et comparaison de quelques recherches sur la combinaison de la blockchain et le deep learning

La collaboration entre l'intelligence artificielle et la blockchain pourrait révolutionner la technologie des smart grids dans plusieurs aspects et en particulier celui de la sécurité. En effet, être en mesure de détecter les attaques de vol d'énergie et FDIA dans un smart grid constitue une tâche coriace, cependant avec l'implémentation de l'IA et la Blockchain, de nouveaux mécanismes de détection peuvent être développés et appliquer afin de garantir une sécurité des plus optimisées aux clients dans un réseau intelligent. Dans cette partie nous nous

intéressons aux différentes solutions possibles permettant la conception d'une contre-mesure de détection des attaques précédentes dans les smart grids en utilisant le machine learning et la technologie de la blockchain.

### **3. 1. A Privacy-Preserving Framework based Blockchain and Deep Learning for Protecting Smart Power Networks [31]**

Une solution conçue pour faciliter la détection des attaques dans les réseaux électriques intelligents. Premièrement, ils ont proposé des techniques de confidentialité à deux niveaux. Le premier niveau comprend le développement d'une technique améliorée de technique proof-of-work pour authentifier les enregistrements de données et empêcher les attaques par empoisonnement. Le deuxième niveau contient un codeur automatique variationnel (VAE) utilisé pour convertir les données originales en un format codé afin d'atténuer les attaques par inférence.

Ensuite, la détection d'anomalies basée sur le deep learning est appliquée pour évaluer les données avant et après l'application des techniques de protection de confidentialité à deux niveaux. La technique utilisée est une technique d'apprentissage profond LSTM (Long Short-Term Memory) en raison de son efficacité à détecter les anomalies à partir de séries temporelles de données telles que le temps, l'heure et la date.

### **3. 2. DeepCoin: A Novel Deep learning and Blockchain-based Energy Exchange Framework for Smart Grids**

La solution proposée par [32] qui est un framework basé deep learning et blockchain pour les réseaux intelligents, il est intitulé *DeepCoin* et utilise deux schémas.

Le schéma basé sur la blockchain se compose de cinq phases : la phase d'installation, phase d'accord, phase de création d'un bloc, phase de consensus et la phase de changement de vue.

Il intègre un nouveau système d'énergie pair-à-pair fiable basé sur l'algorithme pratique de tolérance aux pannes et atteint un débit élevé.

Afin de prévenir les attaques contre les réseaux intelligents, la solution proposée permet de générer des blocs en utilisant des signatures courtes et des fonctions de hachage.

L'autre schéma proposé, basé sur le deep learning, est un système de détection d'intrusion (IDS), qui utilise des réseaux neuronaux récurrents (RNN) pour détecter les attaques de réseaux et les fraudes dans le réseau énergétique basé sur la blockchain.

### **3.3 Blockchain-enabled Data Collection and Sharing for Industrial IoT with Deep Reinforcement Learning**

Le framework présenté par [33] propose un schéma de collecte de données et de partage sécurisé des données combinant la blockchain Ethereum et l'apprentissage par renforcement (deep reinforcement learning DRL) pour créer un environnement fiable et sûr. Le DRL est utilisé pour atteindre la quantité maximale de données collectées, et la technologie blockchain est utilisée pour assurer la sécurité et la fiabilité du partage des données. Les résultats de simulations approfondies montrent que le schéma proposé peut atteindre un niveau de sécurité plus élevé et une plus grande résistance aux attaques qu'un schéma traditionnel de partage de données basé sur une base de données pour différents niveaux de sécurité.

### **3.4 Decentralized firewall for malware detection**

Ce framework décrit la conception et le développement d'un système de pare-feu décentralisé alimenté par un nouveau moteur de détection de logiciels malveillants. Les auteurs dans [34] utilisent une fusion unique de la technologie blockchain et de l'apprentissage en profondeur afin de concevoir une solution heuristique à toute épreuve.

### **3.5. When Machine Learning Meets Blockchain: A Decentralized, Privacy-preserving and Secure Design [35]**

Pour répondre aux préoccupations de confidentialité et de sécurité dans les algorithmes distribués d'apprentissage automatique et le paradigme classique d'apprentissage automatique distribué, les auteurs dans article de référence 35 explorent la blockchain pour concevoir un système d'apprentissage automatique décentralisé, préservant la confidentialité et la sécurité, appelé LearningChain, en considérant un modèle d'apprentissage général et sans serveur central de confiance. Il existe deux principaux défis pour un tel apprentissage automatique distribué : l'un est de savoir comment protéger la confidentialité des détenteurs de données, et l'autre est de savoir comment garantir la résilience du système aux attaques d'utilisateurs malveillants.

### **3.6. BlockDeepNet: A Blockchain-Based Secure Deep Learning for IoT Network**

L'infrastructure du système proposée dans [36] décrit une nouvelle approche décentralisée d'analyse de données volumineuses dans laquelle la tâche d'apprentissage est effectuée au niveau de l'appareil et distribuée en utilisant la technologie blockchain. Un réseau IoT moderne utilise trois étapes de base pour l'analyse du Big Data :

- a) les données sont collectées à partir de l'appareil IoT et prétraitées sur le serveur connecté.

- b) Les paradigmes d'apprentissage intelligent sont utilisés pour analyser les données traitées.
- c) A l'aide des données analysées, le dispositif IoT est contrôlé à distance. Une gestion et un contrôle centralisés sont soutenus par les mécanismes d'analyse de données volumineuses existants.

Le système BlockDeepNet proposé réduit remarquablement la possibilité que les données soient manipulées de manière négative en facilitant un paradigme DL sécurisé et collaboratif. Par conséquent, certains composants du réseau IoT doivent être reconfigurés pour prendre en charge la procédure de travail de BlockDeepNet.

### 3.7. Comparaison entre les solutions

Dans cette section nous allons fournir une comparaison entre les 6 travaux trouvés [31],[32],[33],[34],[35],[36] :

La solution	[31]	[32]	[33]	[34]	[35]	[36]
Type Blockchain :	privée	privée	privée	privée	privée/publique	privée
Blockchain : Algorithme de consensus	un algorithme de mining de PoW amélioré (ePoW) qui n'exige pas une grande puissance de calcul	l'algorithme pratique de Byzantin tolérance aux pannes (PBFT)	Les informations ne sont pas fournies	proof-of-work où les noeuds parviennent à un consensus sur la nature d'un fichier	proof-of-work en résolvant un puzzle mathématique	l'algorithme pratique de Byzantin tolérance aux pannes (PBFT)
méthode de Deep Learning	Apprentissage profond LSTM (Long Short-Term Memory) appliqué sur des données codées avec VAE	réseaux neuronaux récurrents RNN (recurrent neural networks)	DRL (deep reinforcement learning)	Deep Belief Neural Network (DBN)	Les informations ne sont pas fournies	Convolution Neural Network (CNN)
Deep Learning : Datasets	Power System dataset et UNSW-NB15 qui comprend une combinaison d'enregistrements normaux et d'attaques actuels	CICID 2017 Power System dataset et Bot-IoT	Les informations ne sont pas fournies	MALIMG dataset pour obtenir L'ensemble de données malveillant qui sont des images grayscale	Synthetic dataset, Wisconsin breast cancer dataset , et MNIST dataset	PASCAL VOC 2012 dataset
Méthode de collaboration entre Deep learning et la blockchain	Pendant l'exécution de l'algorithme de consensus, Utilisation d'un	Un IDS utilise des réseaux neuronaux récurrents (RNN) pour vérifier que	Après chaque cycle de collecte de données, les résultats de l'approche de	Le résultat numérique produit par le moteur deep learning représente la	auteurs explorent la blockchain pour concevoir un système	décrit une nouvelle approche décentralisée d'analyse de

	auto encodeur variationnel (VAE) sur les données provenant de power système ensuite classer ces données par un Module de détection d'anomalie basé sur LSTM	les trames s'exécutant sur le réseau de transaction énergétique sont conformes à un ensemble de règles	collecte de données basée sur le DRL (deep reinforcement learning) peuvent être des demandes de stockage et des demandes d'interrogation, qui seraient les entrées du système de stockage basé sur la blockchain.	probabilité que le fichier soit malveillant. Ceci est haché par la propre clé des nœuds et ajouté à la blockchain en tant que transaction, et il représente une mesure directe de la confiance de ce nœud dans le réseau.	d'apprentissage automatique décentralisé, préservant la confidentialité et la sécurité, appelé LearningChain, en considérant un modèle d'apprentissage général et sans serveur central de confiance	données volumineuses dans laquelle la tâche d'apprentissage est effectuée au niveau de l'appareil et distribuée en utilisant la technologie blockchain
<b>Simulation</b>	Pour ce travail, le framework a été développé dans le langage de programmation R.	L'expérience est réalisée sur Google Colaboratory sous python 3 en utilisant la bibliothèque TensorFlow et trois types d'accélérateurs matériels, quatre packages ont été utilisés : NumPy, Pandas, Scikit-learn et Keras.	implémentation des DRL par les MTs et l'envoi de demandes de stockage de données par Python, Cet environnement a été configuré sur la machine où se trouvent la blockchain et la base de données comparée. implémentation de réseau blockchain privé sur Go Ethereum	Utilisation de réseau Ethereum pour implémenter la blockchain, De plus utilisation le langage de solidité pour mettre en œuvre les contrats intelligents, moteur de détection, il est effectué à l'aide de scripts python qui utilisent la bibliothèque pillow pour le traitement d'image	LearningChain a été implémenté avec Python et en utilisant Ethereum comme blockchain principale	Go-ethereum a été utilisé pour mettre en place la plateforme blockchain et le langage solidity a été employé pour écrire des contrats intelligents, l'interaction entre les applications IoT et la blockchain a été prise en charge en utilisant Node.js comme interface. Ensuite, la version 3.6.4 de Python et la version 1.7.0 de Tensorflow ont été configurées pour une opération DL

*Tableau 10 : Comparaison des solutions.*

### 3.8. Synthèse

L'intégration de l'apprentissage en profondeur avec la blockchain peut faciliter en termes de sécurité et de confidentialité des données les systèmes existants dans plusieurs applications principalement liées à la sécurité de la blockchain, la gestion du trafic de données et faciliter le processus de vérification des données et d'identification des attaques malveillantes et les transactions malhonnêtes dans la blockchain, surtout que les systèmes existants basés sur la blockchain sont incapables de gérer efficacement les problèmes de qualité des données.

Les articles examinés montrent des techniques d'apprentissage automatique et de blockchain qui sont utilisés soit pour étudier le système et la structure de la blockchain elle-même, soit la mise en œuvre de techniques pour améliorer l'apprentissage automatique, par exemple, l'apprentissage collaboratif/distribué et aussi des techniques de combinaison séparément à divers domaines.

Enfin, Dans cette petite étude comparative de quelques recherches sur la combinaison de la blockchain et les technologies d'apprentissage automatique, nous avons pu démontrer qu'ils peuvent collaborer efficacement, et donc il existe une croissance rapide dans l'intérêt de les intégrer pour un partage de données plus sûr et efficace. Cependant la majorité des recherches peuvent être classées comme application d'une technique à une autre, donc il est juste de dire que le courant de la recherche actuelle est encore très préliminaire et nous attendons des nouvelles recherches surtout dans les domaines de l'IOT pour le partage sécurisé des données et les recherches de stratégies optimale pour le Mining des blocks en utilisant l'apprentissage automatique.

## Conclusion

Dans ce chapitre, nous nous sommes intéressés aux différentes méthodes et techniques utilisées pour pallier aux principaux problèmes rencontrés dans les smart grids à savoir, l'attaque FDIA (False Data Injection Attack) et le vol d'énergie.

Nous avons démontré aussi que d'un côté, la technologie de la blockchain peut être utilisée pour sécuriser les communications et donc venir à bout de l'attaque FDIA et de l'autre côté, l'intelligence artificielle et plus précisément le deep learning de par ses performances peut être envisagé pour l'identification et la détection d'un quelconque vol d'énergie grâce aux modèles de prédiction et enfin nous avons passé en revue de la littérature existante axée sur l'intégration de la blockchain avec l'apprentissage en profondeur et une comparaison de ces différentes solutions.

Le chapitre suivant sera consacré à la conception de notre propre solution de détection des attaques précédentes.

# **Chapitre 3**

Conception d'une contre-mesure de détection FDIA basée  
Deep Learning et Blockchain.

## Introduction

La structure en chaîne de Blockchain peut garantir la non manipulation et la traçabilité des données et ainsi garantir la réduction de toute sorte d'attaque FDIA [37]. En ce sens, au lieu de centraliser toutes les données de mesure dans un centre de contrôle, notre solution proposée dans ce chapitre a pour objectif de réduire considérablement le risque de manipulation des données réussie en prévoyant un mécanisme permettant de stocker les données de manière distribuée et sécurisée.

L'intelligence artificielle de son côté peut garantir la détection de fraude et de vol d'énergie dans le smart grid grâce à des méthodes d'apprentissage profond, tentant de modéliser les données à partir de grands ensembles de données apprises. Permettant ainsi une prise de décision et de supervision optimale à l'égard d'un comportement suspect d'un client.

Dans ce chapitre, nous commençons par définir l'architecture générale AMI, puis nous décrivons les vecteurs d'attaque et leurs types tout en expliquant les cinq niveaux qui peuvent être touchés par des attaques, ensuite nous présentons notre conception de la solution proposée pour sécuriser le réseau et les différentes étapes d'opération nécessaires, et enfin nous conclurons le chapitre par les avantages ajoutés par notre solution pour la sécurité dans smart grid ainsi qu'une conclusion.

### 1. Architecture générale AMI

L'infrastructure de mesure avancée (AMI) et le système de gestion des données de compteur (MDMS) sont des composants de base du réseau intelligent. AMI collecte et transmet les données des compteurs intelligents entre les appareils et MDMS facilite la collecte, le stockage et la gestion des données. Le système de réseau intelligent applique des dispositifs de détection, de mesure et de contrôle AMI avec des communications bidirectionnelles aux segments de production, de transport, de distribution et de consommation du réseau électrique pour permettre la tarification, la surveillance et la conservation en temps réel.

Ces technologies communiquent des informations sur les conditions du réseau aux utilisateurs du système, aux opérateurs et aux dispositifs automatisés, ce qui permet de réagir dynamiquement aux changements de condition du réseau [Net 11].

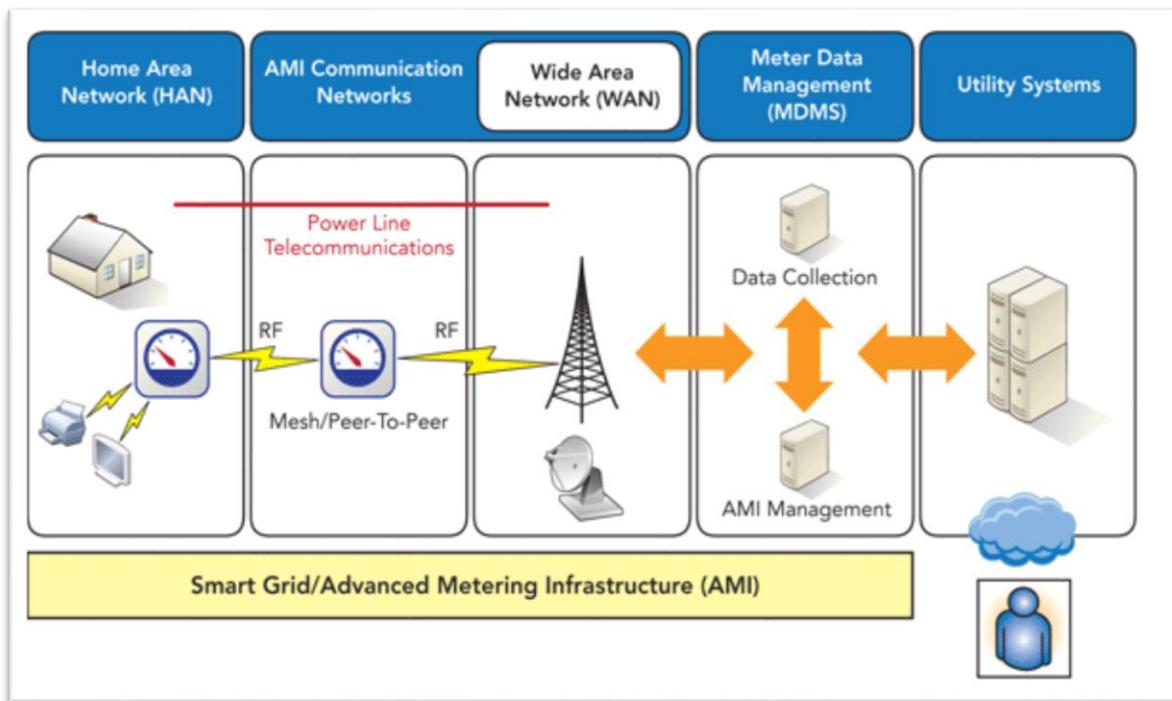


Figure 24 : Architecture général AMI [Net 11]

## 2. Vecteurs d'attaque

Les exigences en matière de sécurité de l'information dans le réseau AMI (infrastructure de mesure avancée) comprennent comme tout autre système trois principales propriétés de sécurité : la confidentialité, l'intégrité et la disponibilité (CIA).

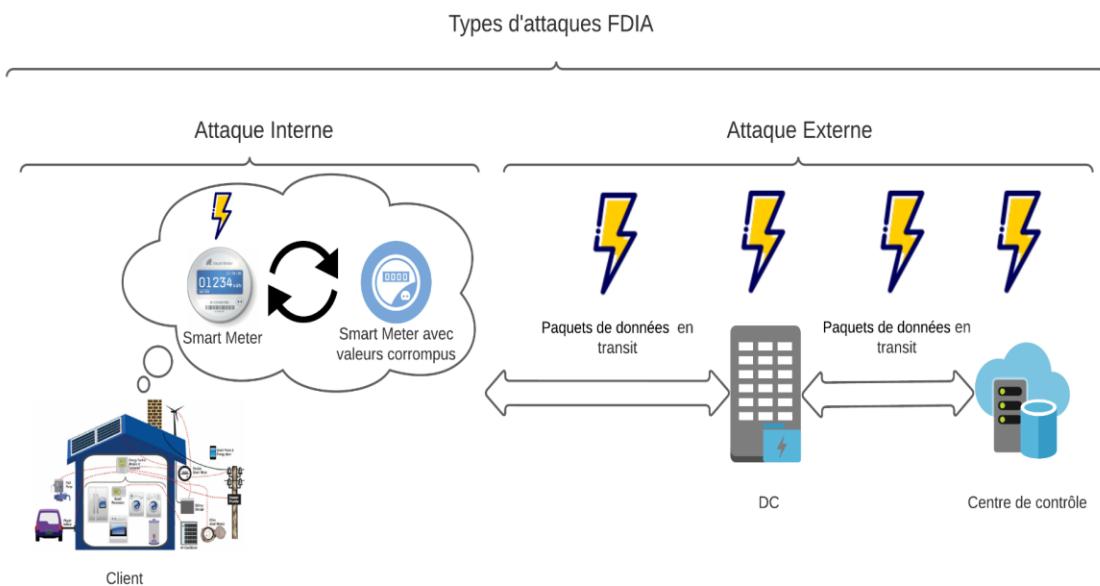
Le réseau intelligent est une infrastructure critique qui peut être la cible de diverses cyber-attaques, la sécurité a été largement reconnue comme un problème majeur aux implications potentiellement catastrophiques [38]. Le réseau cyber-physique est exposé à de nouveaux risques provenant des vulnérabilités des réseaux informatiques, ainsi qu'aux risques hérités des vulnérabilités du réseau électrique.

L'une des attaques les plus dangereuses est la False data injection (FDIA) [38] qui peut être lancé en trois manières: (1) manipuler les données sur le terrain, (2) intercepter et modifier les paquets de données pendant la transmission, et (3) envahir la base de données du centre de contrôle [39].

Selon le niveau d'attaque par rapport au compteur intelligent, et ce qui est applicable dans le cadre de notre étude, on distingue deux types d'attaques FDIA : interne et externe.

1. **Interne** : Permet de compromettre localement les mesures des compteurs intelligents.
2. **Externe** : Permet de falsifier les paquets de données échangés entre le compteur intelligent et le centre de contrôle.

La figure 25 reflète ces deux types d'attaques :



*Figure 25 : Types d'attaques FDIA.*

## 2.1 Exemple d'une Attaque FDIA interne

FDIA intérieur vise à modifier les mesures au niveau du compteur intelligent, les attaquants ont accès aux données de mesures pour modifier les profils de charge et viser à maximiser le bénéfice du vol d'énergie tout en minimisant la probabilité d'être détecté [38].

Il existe Trois grandes classes de vol d'énergie présentées dans le tableau 12 [40] :

Classe d'attaque	Technique
<b>Physique</b>	Déconnecter le compteur
	Contourner le compteur pour supprimer les charges de la mesure
<b>Cyber</b>	Intercepter/altérer les communications
<b>Données</b>	Arrêtez de rapporter toute la consommation
	Modifier le profil de charge de l'appareil pour masquer les charges importantes
	Déclarer zéro consommation

*Tableau 11 : Classification des attaques FDIA interne.*

## 2.2 Exemple d'une Attaque FDIA externe

Dans cette partie nous considérons le scénario où un attaquant tente de falsifier les données dans les systèmes d'alimentation en interceptant et falsifiant les paquets de données pendant la transmission et envahir la base de données du centre de contrôle par les manières suivantes:

- Intercepter et falsifier des paquets de données lors de la transmission de SM vers DC ou entre les SMs.
- Manipuler les DCs sur le terrain.
- Intercepter et falsifier des paquets de données lors de la transmission des DCs au centre de contrôle dans les systèmes électriques existants ou entre les DCs.
- Envahir la base de données dans le centre de contrôle des systèmes électriques existants.

Pour limiter la portée de notre projet on va se focaliser essentiellement sur :

1. La sécurité les paquets de données pendant la transmission (attaque externe).
2. La détection de vol d'énergie à cause d'une manipulation des smart meters sur le terrain (attaque interne).

Et on suppose que :

1. Les données traitées et envoyées par un DC ne sont pas falsifiées par le fait que les DCs dans le réseau WAN sont résistant aux manipulations et donc fiables.
2. Les données sont stockées dans le centre de contrôle d'une manière sécurisé.

## 3. Architecture de la solution proposée

Notre Architecture est basé sur l'infrastructure de mesure avancée (AMI) inspirée de ces solutions [43][41][42][39] et a une approche cloud-fog-edge :

- Chaque compteur de réseau NAN doit posséder des caractéristiques fonctionnelles : Une adresse unique, Un logiciel spécifique permettant de générer une clé publique et une clé privée, Une mémoire vive, du matériel de calcul, d'un dispositif de collecte de données, d'un émetteur de signaux, d'un récepteur de signaux et d'un dispositif de traitement des données et doivent rapporter leurs données au DC afin qu'elles puissent arriver au centre de contrôle.
- Chaque réseau NAN contient un nombre N de smart meters qui sont distribués géographiquement et chaque paire distincte peut communiquer par un canal de communication avec ou sans fil.
- Le centre de contrôle contient les serveurs de l'entreprise qui stockent les données de tous les clients.

- Il existe un chemin de communication reliant chaque paire distincte de nœuds dans chaque réseau (entre DCs dans le WAN et les serveurs dans le centre de contrôles).
- Un compteur observateur est installé sur chaque DC pour enregistrer l'électricité totale fournie à chaque zone. Donc la quantité totale d'électricité fournie au groupe de compteurs de réseau NAN.
- Un système de détection de vol d'énergie basé deep learning est installé sur le DC de chaque région.

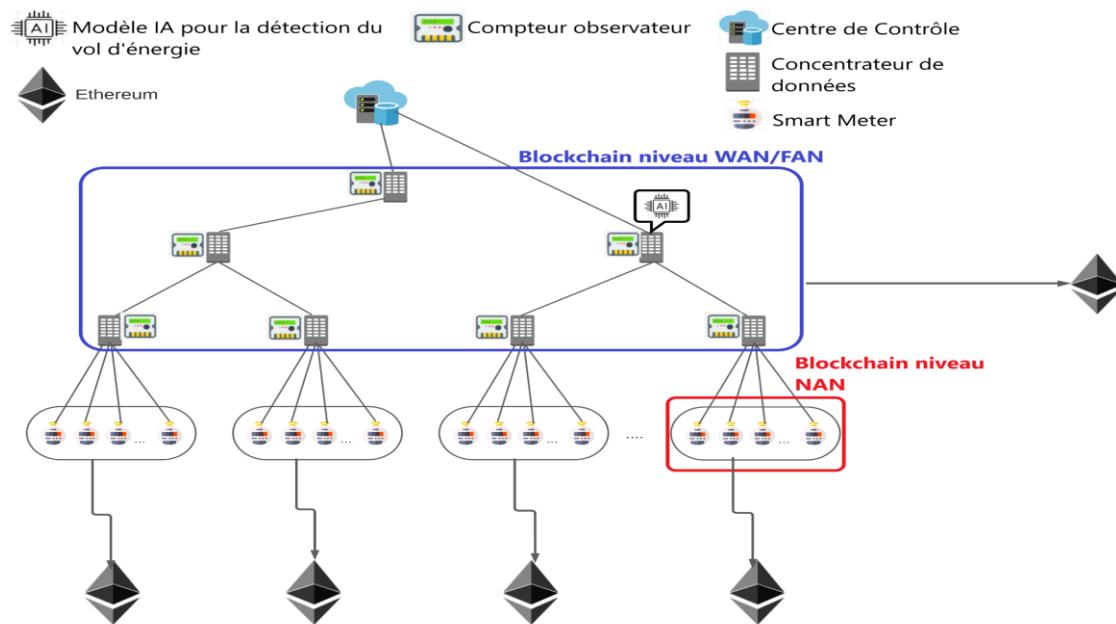


Figure 26 : Architecture Globale de la solution proposée.

Chaque compteur intelligent dans le réseau NAN collecte les mesures de consommation d'un client et les envoie aux autres compteurs et son concentrateur de données DC, afin de les stocker dans un grand livre sous forme de blocs connectés (blockchain régional (NANs)) qui existent sous forme distribuée dans la mémoire de chaque compteur et DC du même NAN. Ce dernier partage les données entre les autres DCs dans le même WAN pour former une blockchain en zone étendu. Enfin, le DC va les enchaîner au grand livre situé au centre de contrôle.

La détection de vol d'énergie est mise en œuvre en deux étapes :

1. Dans la première étape, les données transmises par les compteurs intelligents sont prétraitées par le DC de chaque région. Si les pertes d'un réseau NAN sont supérieures à 4%, on suppose alors qu'une consommation illégale peut exister. Les pertes de distribution peuvent être facilement calculées à partir de l'énergie enregistrée par les

compteurs des clients et les compteurs observateurs. Ensuite, les données sont collectées pour que le consommateur illégal soit identifié à l'étape suivante [43].

2. Dans la deuxième étape, une anomalie dans les données de consommation d'énergie des clients est extraite au niveau des DCs pour détecter le vol d'électricité par une technique deep learning se basant sur le modèle CNN-RF (Convolutional Neural Network and Random Forest).

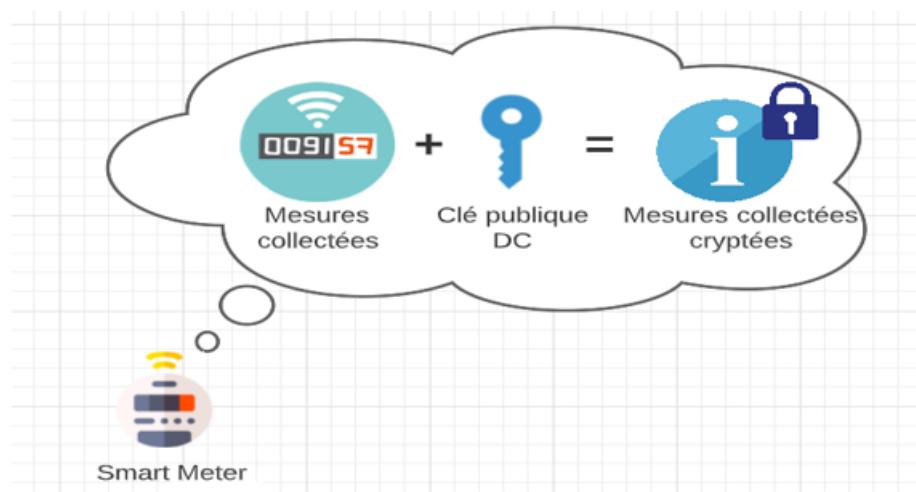
Les étapes de traitement des entités seront traitées en détails ci-dessous.

### **Étape 1 : Récoltes des mesures par la blockchain au niveau NAN**

Chaque SM rapporte des informations concentrées sur la consommation des appareils dans son réseau HAN sous la forme d'une donnée. Cette donnée est cryptée entre le SM et le DC situé au même réseau NAN, et elle est diffusée à tous les autres nœuds dans le même NAN afin de vérifier sa validité, ce processus est détaillé comme suit :

1. Le compteur calcule le chiffré ( $C$ ) des mesures collectées dans une période en utilisant la clé publique du DC :

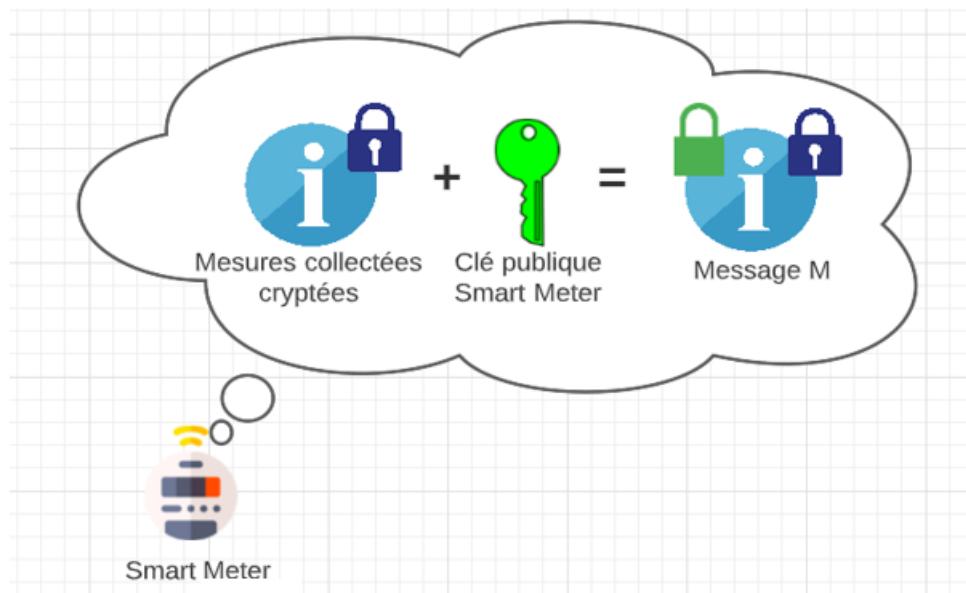
$$C = CH_{CléPubliqueDC}(mesures)$$



*Figure 27 : Chiffrement des mesures collectées dans smart meter*

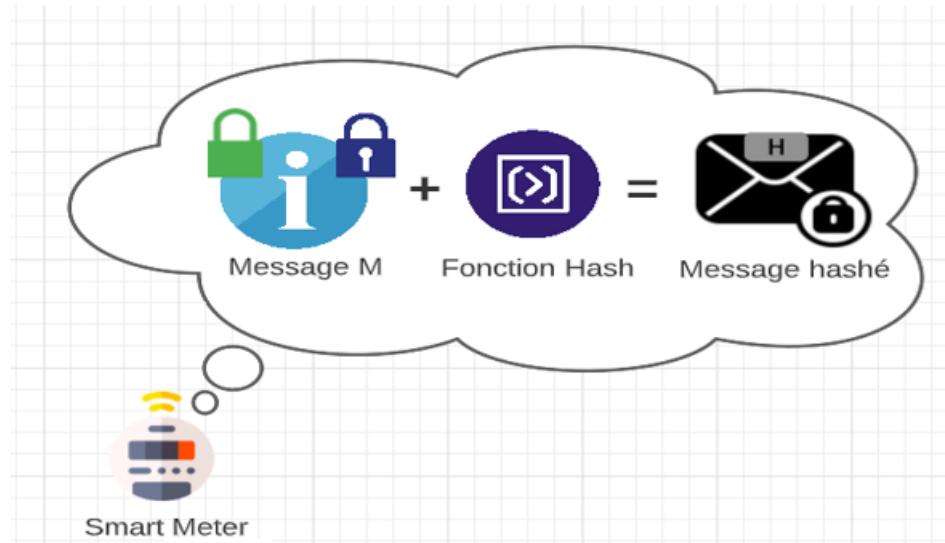
2. Le compteur combine le chiffré  $C$  avec sa clé publique pour construire  $m$  :

$$m = Clépublique || C$$



*Figure 28 : Chiffrement du chiffré avec la clé publique du smart meter dans smart meter*

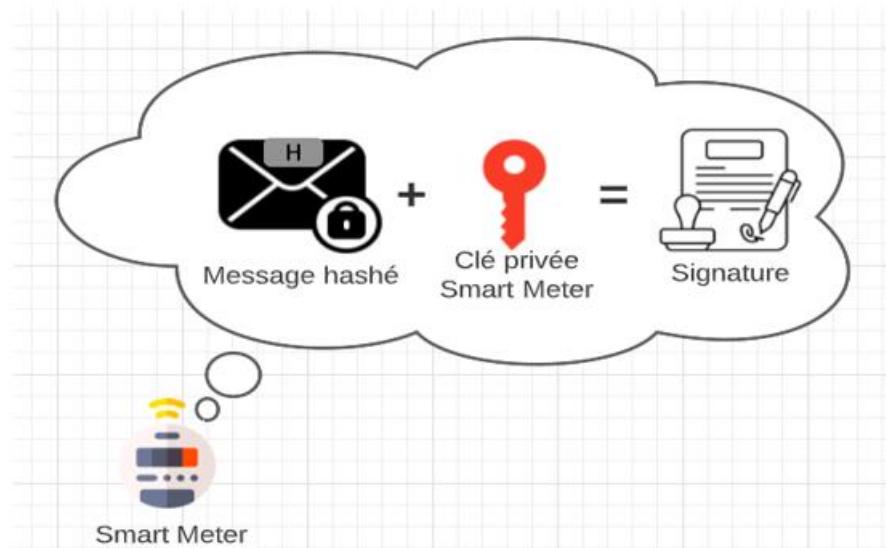
3. Le compteur Hach le message  $m$  avec une fonction de hachage  $H(m)$



*Figure 29 : Hash du message  $M$  avec une fonction de hashage dans smart meter*

4. Ensuite, il construit la signature  $S$  de ce dernier avec sa clé privée:

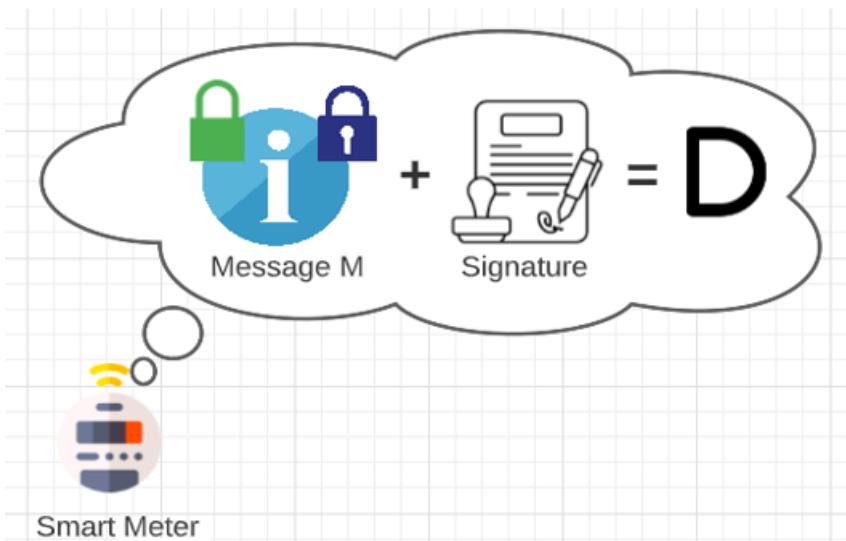
$$S = CH_{Clé Privé}(H(m))$$



*Figure 30 : Signature de M avec clé privée dans smart meter*

5. Enfin il construit les messages à envoyer  $D$ :

$$D = m \parallel S$$

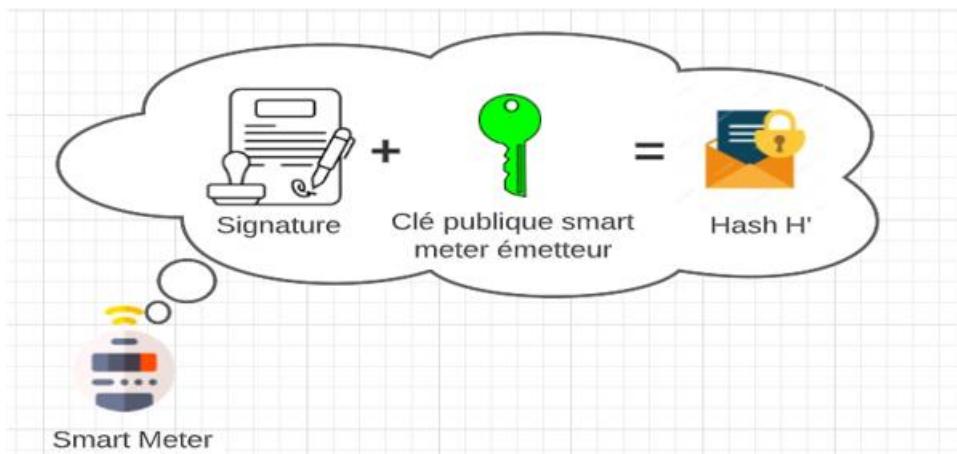


*Figure 31 : Envoie de la signature et du signé du smart meter*

À la réception, chaque nœud vérifie l'intégrité des données comme ceci:

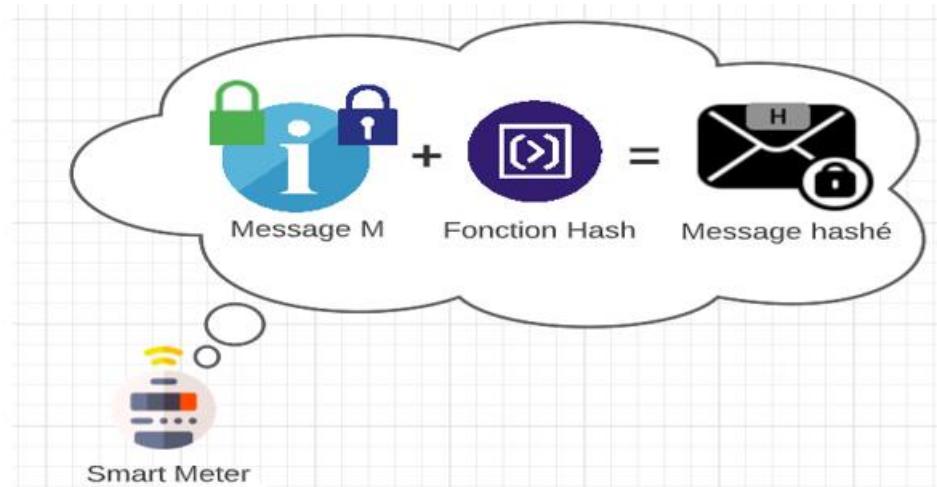
1. Il déchiffre  $S$  avec la clé publique du compteur émetteur. Il obtient un hach  $H'$ .

$$H' = CH_{CléPublique}(S)$$



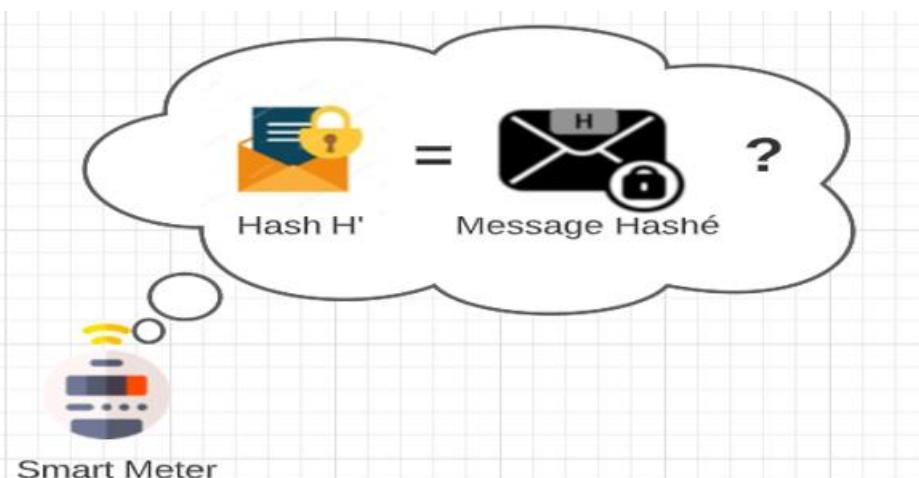
*Figure 32 : Déchiffrement du signé avec la clé publique du smart meter émetteur*

2. Il hach le message  $m$  pour obtenir  $H(m)$ .



*Figure 33 : Hash du message  $M$  avec une fonction de hashage dans un nœud*

3. Si  $H(m) = H'$  alors la signature est vérifiée, c'est-à-dire que l'intégrité du message  $m$  est vérifiée.



*Figure 34 : Vérification de l'égalité des deux hashs dans le nœud*

## Etape 2 : Validation d'un bloc et le calcul de la réputation au niveau NAN

Les données qui passent la vérification seront stockées dans une mémoire locale, sinon elles sont rejetées et le nœud émetteur sera pénalisé. La mémoire locale de chaque nœud contient des données qui n'ont pas été enregistrées dans un bloc. Les données sont validées et exploitées en blocs par un nœud leader en appliquant un algorithme de consensus. Dans ce cadre l'algorithme PoT est adopté afin de valider les blocs en sélectionnant le nœud le mieux réputé.

Dans ce qui suit on détaillera:

- Comment la réputation d'un nœud est mesurée.
- La structure de bloc blockchain dans PoT.
- Le mécanisme de consensus.

### 1. Réputation

La valeur de réputation d'un nœud peut diminuer, elle se résume à la fiabilité de celui-ci.

Les cas de diminution de la fiabilité d'un nœud sont présentés dans le tableau suivant :

Les cas de diminution de réputation
<ul style="list-style-type: none"> <li>- Non envoi d'une transaction dans la période valide préétablie.</li> <li>- Non validité d'une transaction.</li> <li>- Le nœud est leader et ne respecte pas le délai de création et la diffusion de bloc.</li> <li>- Ne diffuse pas son vote ou fait un vote incorrect.</li> <li>- S'il vol de l'énergie.</li> </ul>

*Tableau 12 : Diminution de la réputation niveau NAN*

En utilisant la méthode EWMA (Exponentially Weighted Moving Average), la nouvelle réputation d'un nœud peut être calculée à partir de sa réputation précédente ainsi que sa réputation courante, c'est-à-dire qu'il n'est pas nécessaire de stocker les anciennes réputations, ce qui est souhaitable en raison de problèmes d'extensibilité (scalability) [47]. EWMA est défini comme suit :

$$Rn = \alpha * Rp + (1 - \alpha) * Rc$$

- $Rn$  : nouvelle valeur de réputation où **0** est la plus mauvaise réputation et **100** la meilleure.
- $Rp$  : réputation précédente.
- $Rc$  : réputation courante.

- $\alpha \in [0, 1]$  est un facteur qui détermine le poids approprié appliqué à la valeur de réputation précédente du nœud

Si la réputation **Rn** est supérieure à un threshold, alors on calcule **Rtheft** de la même formule pour mesurer le comportement du nœud par rapport au vol d'énergie :

$$Rtheft = \alpha * Rtheftp + (1 - \alpha) * Rtheftc$$

- **Rtheft** : nouvelle valeur de réputation où **0** est la plus mauvaise réputation et **100** la meilleure.
- **Rtheftp** : réputation précédente.
- **Rtheftc** : réputation courante.

Et enfin on calcul la réputation **Ragrégé** qui est l'agrégation des deux réputations **Rtheft** et **Rn** via la formule suivante:

$$Ragrégé = \alpha * Ragrégép + \frac{1 - \alpha}{2} * Rthefp + \frac{1 - \alpha}{2} * Rn$$

Où **Ragrégép** et **Rthefp** sont récupérées dans le bloc précédemment validé par ancien tour de consensus. **Rn** est la réputation du nœud par rapport au comportement actuel dans la blockchain NAN.

## 2. Structure de bloc

Comme illustré dans la figure 35, chaque bloc de la blockchain va contenir les cinq éléments de base : le numéro du bloc (id), le hachage du bloc précédent, le timestamp, le corps de données ainsi que le hachage du bloc. Afin d'appliquer le cadre PoT, d'autres éléments doivent être ajoutés: le merkleroot des données, la liste des réputations, le merkleroot de la liste des réputations. La signification de chaque élément de bloc est comme suit :

Champ de bloc	Signification
<b>ID</b>	Le numéro successif du bloc courant, qui est utilisé comme titre du bloc.
<b>Timestamp</b>	La date à laquelle les données vérifiées sont encapsulées dans le bloc actuel.
<b>Data</b>	Les données de mesure reçus des différents compteurs intelligents.
<b>hachage précédent</b>	Le résultat du hachage du bloc précédent.
<b>Résultat hachage</b>	Le résultat du hachage du bloc actuel.

<b>Merkleroot</b>	Peut être compris comme la signature de toutes les données incluses dans un seul bloc, c'est le top hash de l'arbre de hachage (MerkleTree) qui est une structure utilisée pour valider efficacement de grandes quantités de données.
<b>Liste des réputations</b>	Une liste qui va contenir la réputation de chaque nœud.
<b>Le merkleroot de la liste des réputations</b>	Arbre Merkle dans l'organisation de la liste de réputations.

Tableau 13 : Signification des attributs d'un bloc.

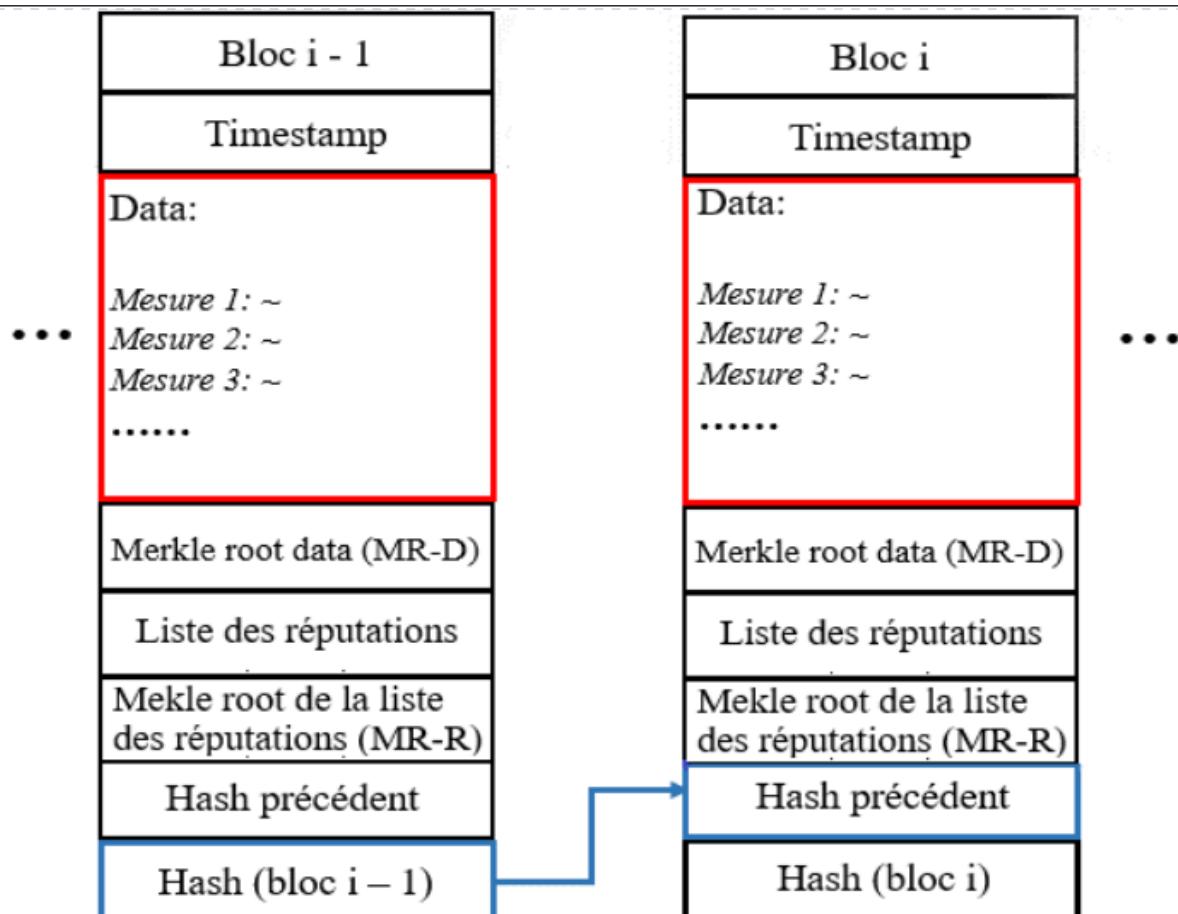


Figure 35 : Structure d'un bloc

### 3. Processus de consensus PoT

#### ➤ Initialisation de l'algorithme

**Threshold** : Indique le nombre de données reçues et validées nécessaires pour entamer la création de bloc.

**Rank** : Signifie la valeur de réputation (de confiance) de chaque nœud.

**P%** : Indique un groupe de nœuds de haute réputation dans la liste des réputations. Seuls ces nœuds fiables contribueront à la vérification des blocs, ce qui donnera plus de crédibilité au résultat et réduira la charge sur le réseau.

Un bloc est validé en deux phases afin d'ajouter la liste de réputation à la blockchain après chaque tour de consensus.

### ➤ Phase 1: Validation des données

Lorsque le nombre de données reçues par chaque nœud du réseau NAN/WAN atteint le Threshold dans la mémoire locale, ils seront regroupés pour créer un bloc. Ce processus s'effectuera comme suit :

1. **Sélection du leader et construction du bloc de données  $B_1$** : Pour construire le bloc de données  $B_1$ , chaque nœud compare les valeurs Rank qui sont dans la liste des réputations stockées dans le dernier bloc de la blockchain. Si le nœud lui-même est sur l'entête de liste (leader), il construit un bloc en effectuant les étapes suivantes :

- Récupérer le hash  $H(i-1)$  du dernier bloc à partir de sa copie dans la blockchain.
- Calculer l'id du bloc ( $i$ ) à créer.
- Chercher le temps actuel (timestamp).
- Calculer le merkle root des données MR-D.
- Concaténer ces informations avec les données reçues.

$$B1 = H(i-1) // id(i) // TimeStamp // MRD // Data$$

- Calculer le hash du bloc  $i$  :  $H(i) = \text{hash}(B1)$
- Construire le bloc  $B_1$  avec les informations  $H(i-1)$ ,  $id(i)$ ,  $TimeStamp$ ,  $MRD$ ,  $H(i)$  et les données.
- Une fois le bloc est construit par le leader, il le publie après l'avoir signé avec sa clé privée.

### Algorithme 1 : Création de bloc $B_1$

#### Les entrées :

*List\_rep\_bloc* : la liste des réputations stockée au niveau du dernier bloc de la blockchain.

#### Paramètre :

Nœud  $i \in (1, \dots, n)$ .

*List\_trans* : Un tableau qui contient les données reçues par les nœuds.

*List\_rep* : Copie de *List\_rep\_bloc*, qui est utilisée pour mettre à jour la liste des réputations telle que la case  $j$  qui contient la réputation du nœud  $i$  ( $i = j$ ).

*List\_rep\_copy* : Copie de *List\_rep\_bloc*, utilisée pour la sélection du leader et des P% meilleurs nœuds.

*N*: Le nombre de fois où le leader n'a pas envoyé le bloc dans le délai prévu.

*TimeOut* : La durée d'attente des mesures.

**SelectIdLeader** (*List\_rep\_bloc*, *m*) : Fonction qui retourne l'id du nœud qui a la *m* ème meilleur réputation.

### Les sorties :

L'id du leader (*id\_leader*)

Création et diffusion du bloc  $B_1$

**1:** *List\_rep* = *List\_rep\_bloc*

**2: if** ((**Taille** (*List\_trans*) == Threshold) **OR** *TimeOut*) **then**

**3:**     *List\_rep\_copy* = *List\_rep\_bloc*

**4:**     *N* = 0

**5:**     *id\_leader* = **selectIdLeader** (*List\_rep\_copy*, *N*+1)

**6:**     **if** (nœud *i* = *id\_leader*) **then**

**7:**         *B1* = **Createblock()**

**8:**             **Diffusion** (*i*, message ( $B_1$  , signature))

**9:**     **end if**

**10: end if**

---

**2. Vérification de bloc  $B_1$**  : Chaque nœud qui reçoit le bloc de données et qui est parmi les P% meilleurs nœuds en termes de réputation, va vérifier l'authenticité du celui qui le crée (leader) pour s'assurer que ce bloc a été effectivement envoyé par le nœud le plus confiant. Ensuite chaque nœud va constituer et comparer le merkleroot des données reçues précédemment stockées dans la mémoire locale avec le merkleroot des données qui est dans le bloc reçu (MR-D).

Si le bloc est vérifié par un nœud (vérifie: la signature de l'émetteur, les données reçues et le hash du bloc précédent), ce dernier vote par un message « Accepté » en l'envoyant à tous les nœuds du réseau,

Sinon il diffuse le message « Refusé ».

Le vote est envoyé avec le hash signé en utilisant la clé privée du noeud émetteur.

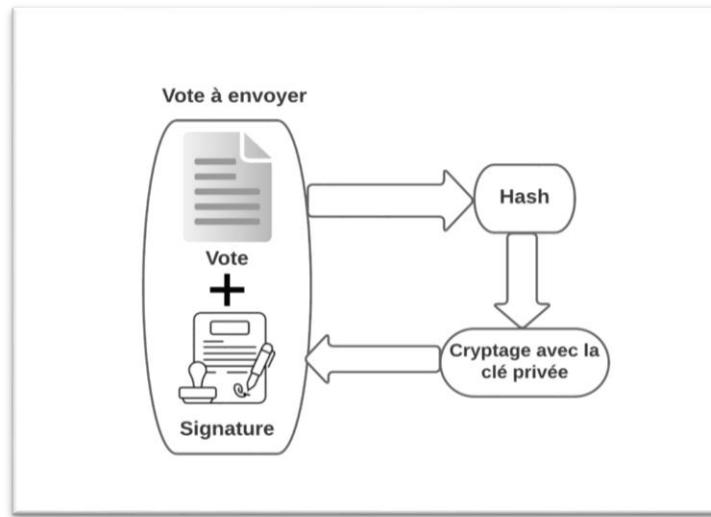


Figure 36 : Envoi d'un vote.

**Algorithme 2: Vérification de bloc  $B_1$** **Les entrées :**

$B_1$  : Le bloc à vérifier.

MR-D : Merkleroot des données localement enregistrées.

N: Le nombre de fois où le leader n'a pas envoyé le bloc dans le délai prévu.

List\_rep : Copie de List\_rep\_bloc (stockée au niveau du dernier bloc de la blockchain), qui est utilisée pour mettre à jour la liste des réputations.

List\_agréé: Copie de List\_agréé\_bloc (stockée au niveau du dernier bloc de la blockchain)

List\_theft: Copie de List\_theft\_bloc (stockée au niveau du dernier bloc de la blockchain)

List\_rep\_copy : Copie de List\_rep\_bloc, utilisée pour la sélection du leader et des P% meilleurs nœuds.

Repmin : La valeur minimale de la réputation pour laquelle les nœuds malveillants commencent à être signalés

**Paramètre :**

Nœud  $i \in (1, \dots, n)$ .

$M = P\%$  des meilleurs nœuds dans List\_rep\_bloc.

TimeOut : La durée d'attente du bloc  $B_1$  est terminée.

Rc : Réputation courante de nœud.

**Les sorties :**

Des votes sur la vérification du bloc  $B_1$ .

```

1: if (Recevoir( $B_1$ ) OR TimeOut) then
2:   if (TimeOut) then
        //Réduire la réputation du leader qui n'a pas envoyé le bloc B 1 dans le délai spécifié.
3:     List_rep[id_leader] = ( $\alpha * List\_rep[id\_leader]$ ) + ((1 -  $\alpha$ ) * Rc) ; //Rc = Rp - 10
4:     N = N + 1
5:     if (List_rep[id_leader] <= Rep minimale ) then
6:       SendAlertToAdmin(id_Leader)
  
```

```

7:           else:
8:               List_agrégé[id_leader]=(α * List_agrégé[id_leader]) +
               (1-α)/2List_theft[id_leader] + (1-α)/2 List_rep[id_leader]
9:               if (List_agrégé[id_leader] <= Rep minimale ) then
10:                  SendAlertToAdmin(id_Leader)
11:              End if
12:              //Re-sélection d'un nouveau leader.
13:              id_Leader = selectIdLeader(List_rep_copy, N + 1);
14:              Aller à ligne 6 de l'algorithme 1;
14:          else if (nœud i∈ M) then
15:              //Vérification de la validité du bloc  $B_1$  et la diffusion du vote en fonction de la vérification.
16:              if (Vérifie (B1.signature) && B 1 .MR-D == MR-D && B 1 .PrevHash == HashDerBloc)
17:                  then
18:                      Diffusion (Noeud i, message (accepté, signature))
19:                  else
20:                      Diffusion (Noeud i, message (refusé, signature))
21:                  end if
22:              end if
22:          end if

```

---

**3. Validation de bloc  $B_1$**  : Chaque nœud vérifiera l'authenticité du vote qui lui a été transféré, en décryptant la signature et en comparant le hash avec hash du message reçu. Ensuite il calcul le nombre de nœuds qui lui ont envoyés le message « Accepté» ainsi que « Refusé ». Après, il fait la somme des réputations des nœuds qui lui ont envoyés le message « Accepté » (Ra) et celle des nœuds qui lui ont envoyés le message « Refusé » (Rr)

Le bloc est valide lorsque:

$$F2 = Ra > 2/3 Rt \quad \&\& \quad VotePos > 2/3 NV$$

**Ra** : Somme des réputations des nœuds qui lui ont envoyés le message « Accepté ».

**Rt** : Somme des réputations des nœuds validateurs.

**vote<sub>pos</sub>**: Le nombre de votes Positifs.

**NV** : Nombre de votes.

Le bloc est non valide lorsque :

$$F1 = Rr > 2/3 Rt \quad \&\& \quad VoteNeg > 2/3 NV$$

**Rr** : Somme des réputations des nœuds qui lui ont envoyés le message « Refusé »

**Rt** : Somme des réputations des nœuds validateurs

***vote<sub>neg</sub>***: Le nombre de votes Négatifs.

Si ni F1 et ni F2 ne sont vérifiés, Nous le considérerons le bloc comme invalide sans faire mettre à jour la liste des réputations.

### Algorithme 3 : validation de bloc

#### Les entrées :

Tab\_vote : Tableau qui contient les votes reçus de chaque noeud initialisé par -1, tel que la case i contient le vote du noeud i.

#### Paramètre :

TimeOut : La durée d'attente des votes.

#### Les sorties :

Validation du bloc (Vote\_Globale).

//Ajout de B 2 au blockchain (dans le cas où il s'agit de la deuxième phase)

Vote\_pos = 0 ; // nombre de vote accepté reçu.

Vote\_neg = 0 ; // nombre de vote refusé reçu.

**1: While Recevoir (i && message) && !TimeOut do**

//Réception et vérification de la validité des votes reçus.

**2: If (vérifie (message.signature) && message.vote == accepté) then**

**3:     Tab\_vote[i] == 1**

**4:     Vote\_pos ++**

**5: else if (vérifie (message.signature) && message.vote == refusé) then**

**6:     Tab\_vote[i] == 0**

**7:     Vote\_neg ++**

**8: end if**

**9: end while**

**10: If (F1) then**

**11:     Vote\_Globale = True**

// Le bloc est validé (dans la 2 ème phase B2 est ajouté dans la blockchain).

**12: else if (F2) then**

**13:     Vote\_Globale = False // le bloc n'est pas validé.**

**14: else**

**15:     aller à l'algorithme 1.**

**16: end if**

#### 4. Mise à jour de la liste de réputations

Chaque noeud calcule localement la mise à jour de la liste des réputations en se basant sur le résultat de la validation du B1.

Si le bloc est valide, la réputation des noeuds qui ont diffusés « Refusé » va diminuer.

Si le bloc est rejeté, chaque noeud diminue la valeur de réputation du leader, ainsi que celle des noeuds qui lui ont envoyé le message « Accepté ».

L'algorithme 4 présente le processus de mise à jour de la liste des réputations.

---

**Algorithme 4 : Mise à jour de la liste de réputation**

---

**Les entrées :**

*Vote\_Globale* : Résultat du vote.

*Tab\_vote* : Tableau qui contient les votes reçus de chaque nœud, tel que la case *i* qui contient le vote du nœud *i*.

*List\_rep* : Copie de *List\_rep\_bloc*, qui est utilisée pour mettre à jour la liste des réputations.

*List\_agrégré*: Copie de *List\_agrégré\_bloc* (stockée au niveau du dernier bloc de la blockchain)

*List\_theft*: Copie de *List\_theft\_bloc* (stockée au niveau du dernier bloc de la blockchain)

*List\_trans* : Un tableau qui contient les données reçues par les nœuds.

*Tour\_actu* : Représente le numéro du tour de consensus courant.

**Paramètre:**

Nœud *i* ∈ (1, ..., *n*).

*M* = P% des meilleurs nœuds dans *List\_rep\_bloc*.

**Les sorties :**

La liste des réputations mise à jour (*List\_rep*).

**1: For** *j* allant de 0 à Taille (*List\_trans*) **do**

//Réduire la réputation des nœuds qui n'ont pas envoyé ces mesures.

**2: If** (*List\_trans[j]* == vide) **then**

**3: List\_rep[j] =** ( $\alpha * \text{List\_rep}[j]$ ) + ((1 -  $\alpha$ ) \* *Rc*) //*Rc* = *Rp* - 10

**4 if** (*List\_rep[id\_leader]* <= Rep minimale ) **then**

SendAlertToAdmin(*id\_Leader*)

**5: else:**

*List\_agrégré[id\_leader]*= ( $\alpha * \text{List\_agrégé}[id\_leader]$ ) +

**8: (1- $\alpha$ )/2***List\_theft[id\_leader]* + (1- $\alpha$ )/2 *List\_rep[id\_leader]*

**9: if** (*List\_agrégré[id\_leader]* <= Rep minimale ) **then**

SendAlertToAdmin(*id\_Leader*)

**11: end if**

**12: End if**

**13: done**

**14: if** (*Vote\_Globale* == false) **then**

//Réduction de la réputation du leader en fonction du résultat final du vote.

**15: List\_rep[id\_leader] =** ( $\alpha * \text{List\_rep}[id\_leader]$ ) + ((1 -  $\alpha$ ) \* *Rc*) //*Rc* = *Rp* - 10

**16: if** (*List\_rep[id\_leader]* <= Rep minimale ) **then**

SendAlertToAdmin (*id\_Leader*)

**18: else:**

*List\_agrégré[id\_leader]*= ( $\alpha * \text{List\_agrégé}[id\_leader]$ ) +

**20: (1- $\alpha$ )/2***List\_theft[id\_leader]* + (1- $\alpha$ )/2 *List\_rep[id\_leader]*

**21: if** (*List\_agrégré[id\_leader]* <= Rep minimale ) **then**

SendAlertToAdmin(*id\_Leader*)

**23: end if**

```

24: end if
25: for nœud i ∈ M do
26: if (Tab_vote[i] != Vote_Globale) then
    //Réduction de la réputation des nœuds en fonction du résultat final du vote.
27:     List_rep[i] = ( $\alpha$  * List_rep[i]) + ((1 -  $\alpha$ ) * Rc) //Rc = Rp - 10
28:     if (List_rep[i] <= Rep minimale ) then
29:         SendAlertToAdmin(i)
30:     else:
31:         List_agrégé[i]= ( $\alpha$  * List_agrégé[i]) +
32:             (1- $\alpha$ )/2List_theft[i] + (1- $\alpha$ )/2 List_rep[i]
33:     end if
34:     if (List_agrégé[i] <= Rep minimale ) then
35:         SendAlertToAdmin (i)
36:     end if
37: end if
38: done
39: Fait

```

---

## ➤ Phase 2 : Validation de la liste des réputations

Dans cette phase, les nœuds se mettront d'accord sur la liste des réputations qui sera utilisée lors du prochain tour de consensus.

### 1. Composition et diffusion du bloc B2

Si le bloc n'a pas été validé lors de la première phase, les mesures des nœuds se mettent à 0 dans B1 et il y aura une sélection d'un nouveau leader, qui a une meilleure réputation après le leader corrompu dans la liste des réputations stockée dans le dernier bloc de la blockchain, à condition qu'il n'a pas envoyé un vote incorrect lors de ce tour de consensus.

Sinon, les données et le leader ne changent pas. Leader actuel ajoute la liste des réputations locale mise à jour et son merkleroot (MR-R) au bloc de données B1, formant ainsi un nouveau hachage du bloc, comme indiqué ci-dessous :

$$\mathbf{B2} = \mathbf{B1} // List\text{-}rep // MR\_R$$

$$H(i) = \text{hash } (B_2)$$

Le bloc B2 est diffusé par le leader après l'avoir signé avec sa clé privée.

---

**Algorithme 5 : Composition et diffusion de bloc  $B_2$** 

---

**Les entrées :**

*List\_rep* : Copie de *List\_rep\_bloc*, qui est utilisée pour mettre à jour la liste des réputations.

*List\_rep\_bloc* : La liste des réputations stockée au niveau du dernier bloc de la blockchain.

*B<sub>1</sub>.valide* : la validation du bloc.

**Paramètre :**

Nœud i ∈ (1, ..., n).

*List\_rep\_copy* : Copie de *List\_rep\_bloc*, utilisée pour la sélection du leader et des P% meilleurs nœuds.

N: Le nombre de fois où le leader n'a pas envoyé le bloc dans le délai prévu.

*SelectIdLeader* (*List\_rep\_bloc*, m, *Vote\_Globale*) : Fonction qui retourne l'id du nœud qui a la m ème meilleure réputation dans *List\_rep\_bloc*, à condition qu'il n'ait pas envoyé un vote incorrect lors de ce tour de consensus.

**Les sorties :**

Composition et diffusion du bloc  $B_2$

**1:** N2 = N + 1

**2:If** !(*B<sub>1</sub>.valide*) **then**

**3:**     N2 = N2 + 1

**4:**     *B<sub>1</sub>.Data* = 0

**5:**     *List\_rep\_copy* = *List\_rep\_bloc*

**6:**     *Id\_leader* = *SelectIdLeader*(*List\_rep\_copy*, N2, *Vote\_Globale*)

**7: End if**

**8: if** (nœudi = *id\_leader*) **then**

**9:**     *B<sub>2</sub>* = *UpdateBlock*(*B<sub>1</sub>*, *List\_rep*)

    // ajouter la liste des réputations mise à jour et son merkle root.

**10:**    Diffusion (Noeud i, message (*B<sub>2</sub>*, signature))

**11: end if**

---

**2. Vérification de B2**

Le processus de vérification dans cette phase est le même que celui du bloc dans la première phase, en ajoutant simplement la vérification du merkleroot de la liste des réputations de telle sorte que chaque nœud des P% meilleurs nœuds fait une comparaison entre merkel root de la liste des réputations stocké localement et le merkleroot des réputations contenu dans le bloc B2 reçu (MR-R).

---

**Algorithme 6 : vérification de bloc  $B_2$** 

---

**Les entrées :**

$B_2$  : Le bloc à vérifié.

$MR-D$ : Merkle root des données.

$MR-R$ : Merkle root de la liste des réputations (List\_rep) ajouté dans le bloc  $B_2$ .

$N2$ : Le nombre de fois où le leader n'a pas envoyé le bloc dans le délai prévu.

$List_{rep\_bloc}$  : La liste des réputations stockée au niveau du dernier bloc de la blockchain.

$List_{rep}$  : Copie de  $List_{rep\_bloc}$ , qui est utilisée pour mettre à jour la liste des réputations.

$List_{agrégé}$ : Copie de  $List_{agrégé\_bloc}$  (stockée au niveau du dernier bloc de la blockchain)

$List_{theft}$ : Copie de  $List_{theft\_bloc}$  (stockée au niveau du dernier bloc de la blockchain)

**Paramètre:**

Nœud  $i \in (1, \dots, n)$ .

$M = P\%$  des meilleurs nœuds dans  $List_{rep\_bloc}$ .

$List_{rep\_copy}$  : Copie de  $List_{rep\_bloc}$ , utilisée pour la sélection du leader et des  $P\%$  meilleurs nœuds.

TimeOut : La durée d'attente de bloc  $B_2$  est terminée.

**Les sorties :**

Des votes sur la vérification d'un bloc  $B_2$

```

1: if (Recevoir( $B_2$ ) OR TimeOut) then
2:   if (TimeOut) then
        //Réduire la réputation du leader qui n'a pas envoyé le bloc dans le délai spécifié.
3:     List_rep[id_leader] = ( $\alpha * List_{rep}[id\_leader]$ ) + ((1 -  $\alpha$ ) * Rc) //Rc = Rp - 10
4:     List_rep_copy = List_rep_bloc
5:     N2 = N2 + 1
6:     if(List_rep[id_leader] <= Rep minimale ) then
        SendAlertToAdmin(id_Leader)
7:   else:
8:     List_agrégé[id_leader]= ( $\alpha * List_{agrégé}[id\_leader]$ ) +
9:       (1- $\alpha$ )/2List_theft[id_leader] + (1- $\alpha$ )/2 List_rep[id_leader]
10:    if(List_agrégé[id_leader] <= Rep minimale ) then
11:      SendAlertToAdmin(id_Leader)
12:    end if
13:  end if

14:  id_Leader = selectIdLeader(List_rep_copy, N2, Vote_Globale)
    //Resélection d'un nouveau leader.
15:  aller à ligne 8 de l'algorithme 5.
16:  elseif (nœudi  $\in M$ ) then
        //Vérification de la validité du bloc  $B_2$  et la diffusion du vote en fonction de la vérification.
17:    if (Vérifie ( $B_2$  .signature) && $B_2$  .MR-D == MR-D && $B_2$  .MR-R == MR-R)
18:      then

```

---

```

19:           Diffusion (i, message (accepté, signature))
20:     else
21:           Diffusion (i, message (refusé, signature))
22:     end if
23:   end if
24: end if

```

---

### 3. Validation de bloc $B_2$

Le processus de validation du bloc ne change pas, en comptant les votes reçus et en appliquant la formule F1 qui a été présentée précédemment.

### 4. Mise à jour de la liste des réputations

Si le bloc  $B_2$  est validé, il sera ajouté à la blockchain, sinon une mise à jour de la réputation du leader est déclenchée, où chaque nœud calcule localement la liste des réputations mise à jour en diminuant la valeur de réputation du leader, ainsi, un nouveau leader sera sélectionné, et la deuxième phase est relancée jusqu'à ce que le bloc  $B_2$  devienne valide afin de l'ajouter au blockchain.

Après la validation de  $B_2$ , si les données contenues dans le bloc lors de la première phase ( $B_1$ ) n'étaient pas valides, le prochain tour de consensus validera les mêmes données.

---

### **Algorithme 7 : Mise à jour de la liste de réputation**

---

#### Les entrées :

*Vote\_Globale* : Résultat du vote (bloc  $B_2$  non valide).

*List\_rep* : Copie de *List\_rep\_bloc*, qui est utilisée pour mettre à jour la liste des réputations.

*N2*: Le nombre de fois où le leader n'a pas envoyé le bloc dans le délai prévu.

*List\_rep\_bloc* : La liste des réputations stockée au niveau du dernier bloc de la blockchain.

*List\_agrégré*: Copie de *List\_agrégré\_bloc* (stockée au niveau du dernier bloc de la blockchain)

*List\_theft*: Copie de *List\_theft\_bloc* (stockée au niveau du dernier bloc de la blockchain)

#### Paramètre:

M = P% des meilleurs nœuds dans *List\_rep\_bloc*.

*List\_rep\_copy*: Copie de *List\_rep\_bloc*, utilisée pour la sélection du leader et des P% meilleurs nœuds.

#### Les sorties :

La liste des réputations mise à jour (*List\_rep*).

**1: if** (*Vote\_Globale* == false) **then**

//Réduction de la réputation du leader en fonction du résultat final du vote.

**2:** *List\_rep[id\_leader]* = ( $\alpha * \text{List\_rep[id\_leader]}$ ) + ((1 -  $\alpha$ ) \* *Rc*) //*Rc* = **Rp - 10**

```

3:   if (List_rep[id_leader] <= Rep minimale ) then
4:       SendAlertToAdmin(id_Leader)
5:   else:
6:       List_agrégré[id_leader]=( $\alpha$  * List_agrégré[id_leader]) +
7:           (1- $\alpha$ )/2List_theft[id_leader] + (1- $\alpha$ )/2 List_rep[id_leader]
8:       if (List_agrégré[id_leader] <= Rep minimale ) then
9:           SendAlertToAdmin(id_Leader)
10:      end if
11:      N2 = N2 + 1

12:     List_rep_copy = List_rep_bloc
13:     Id_leader = SelectIdLeader(List_rep_copy, N2, Vote_Globale) //Resélection d'un
nouveau leader.
14:     Aller à ligne 8 dans l'algorithme 5.
15: End if

```

---

### **Etape 3 : Détection de vol d'énergie au niveau DC**

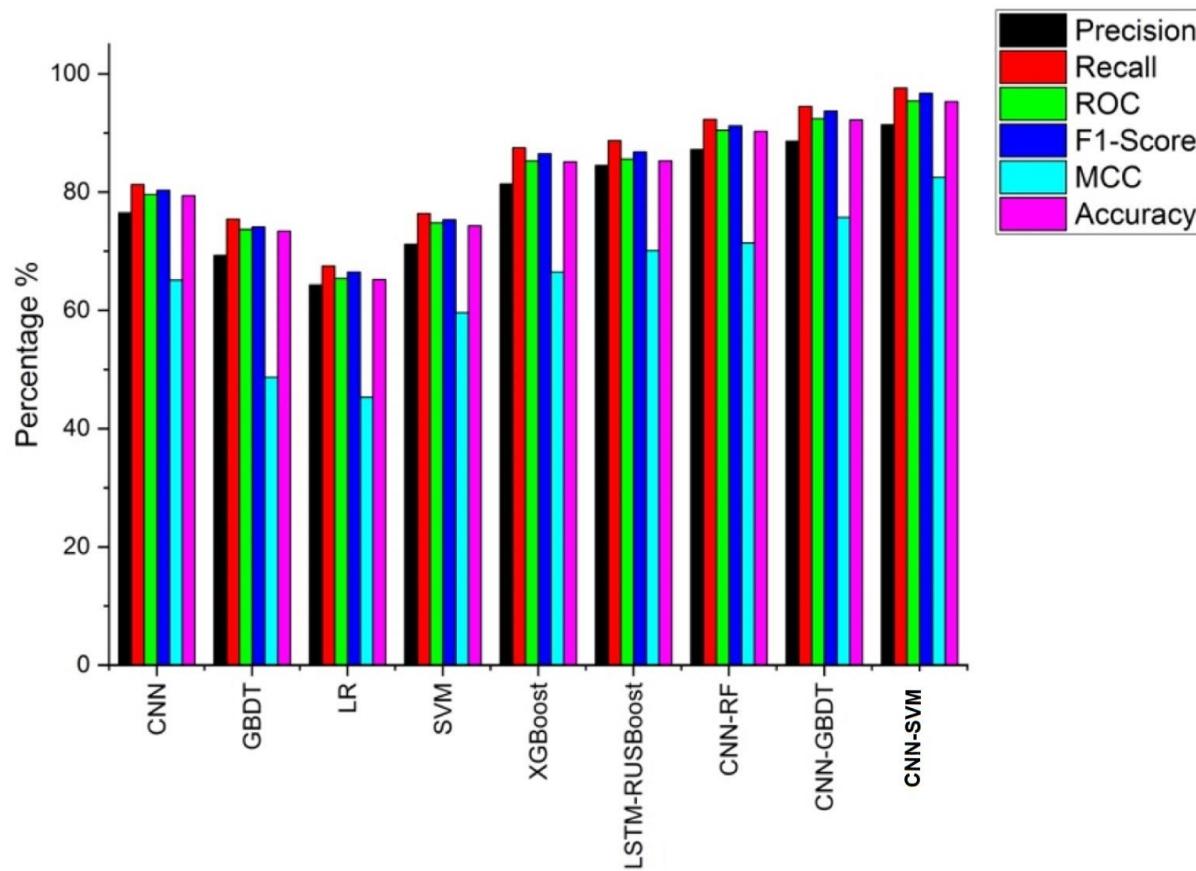
Un concentrateur de donnée est membre de la blockchain de son réseau FAN/WAN, et aussi de la blockchain de son réseau NAN. De façon à ce que dans cette dernière, il peut uniquement décrypter les données à l'aide de sa clé privée pour qu'il puisse utiliser par la suite le système de détection de vol d'énergie basé deep learning sur les mesures des différents smart meter.

#### **1. Choix de Dataset**

L'analyse menée dans notre étude est appliquée aux données recueillies lors des essais de technologie de comptage intelligent de l'électricité effectués par les réseaux de l'Electricity Supply Board (ESB) dans le cadre du projet CER Smart Metering en Irlande, qui sont accessibles au public sur [45]. Les données comprennent plus de 6 000 compteurs intelligents pour les résidences, les PME et d'autres emplacements. Les données de charge électrique ont été enregistrées par les compteurs intelligents toutes les demi-heures pendant l'essai sur 18 mois. Chaque fichier d'utilisation des données des compteurs intelligents est composé de trois colonnes : i) identifiant unique du compteur domestique, ii) horodatage et iii) relevés d'électricité, pour des intervalles de 30 minutes en kWh [46].

## 2. Choix du modèle IA utilisé

Lorsque l'on compare les résultats de différentes approches utilisées pour l'identification du vol électrique, les approches deep Learning qui utilisent CNN surpassent les techniques Machine Learning telles que la régression logistique, la machine à vecteurs de support et la forêt aléatoire. La figure 37 illustre les hautes performances des modèles DL par rapport au modèles ML dans la détection de vol d'énergie.



*Figure 37 : Comparaison de performance entre plusieurs modèles de détection de vol d'énergie.*

Ces résultats sont attribués à la capacité du réseau neuronal convolutif à apprendre des caractéristiques à partir d'une quantité substantielle de modèles de consommation d'énergie, ce qui améliore la précision de l'identification du vol d'électricité. Nous devrions considérer qu'il s'agit essentiellement d'un problème basé sur les anomalies puisque la détection des utilisateurs de vol d'énergie se fait en identifiant les habitudes de consommation anormales des consommateurs suspects.

Le modèle sur qui on se basera sera le modèle CNN (Convolutional Neural Network).

La détection de vol d'électricité en utilisant ce modèle est divisée en trois étapes principales comme suit :

## 2.1. Analyse des données et prétraitement [44]

Pour expliquer la raison de l'application d'un CNN pour l'extraction de caractéristiques, une analyse sera faite d'abord sur les facteurs qui affectent les comportements des consommateurs d'électricité. Pour le prétraitement des données, plusieurs tâches sont considérées telles que le nettoyage des données (résolution des valeurs aberrantes), l'imputation des valeurs manquantes et la transformation des données.

### ➤ Nettoyage des données

Il existe des valeurs erronées qui correspondent au pic électrique causé par des activités de consommation élevées pendant les vacances ou des occasions telles que les anniversaires et les célébrations. Ici, la « règle empirique des trois sigma » est utilisée pour restituer les valeurs aberrantes selon la formule suivante :

$$F(X_{i,t}) = \begin{cases} \text{avg}(X_{i,t}) + 2\sigma(X_{i,t}), & X_{i,t} > X'_{i,t} \\ X_{i,t}, & \text{Else} \end{cases} \quad (1)$$

Où  $X'_{i,t}$  est calculé par la moyenne  $\text{avg}(\cdot)$  et l'écart type  $\sigma$  pour chaque intervalle de temps comprenant couple jour de la semaine/heure pour chaque mois.

### ➤ Imputation des valeurs manquantes

Pour différentes raisons, tels que les problèmes de stockage et la défaillance des compteurs intelligents, il peut y avoir un manque de valeurs dans les données de consommation d'électricité.

Avec l'analyse des données originales, il sait avérer qu'il y a deux types de données manquantes : l'une est le manque continu de plusieurs données, et la solution consiste à supprimer les utilisateurs lorsque le nombre de valeurs manquantes dépasse 10, l'autre manque de données est les données uniques, ce qui est traité par la formule (2), n'empêche les données peuvent être récupéré par la formule suivante :

$$F(X_{i,t}) = \begin{cases} \frac{X_{i,t-1} + X_{i,t+1}}{2}, & X_{i,t} \in NaN \\ X_{i,t}, & \text{Else} \end{cases} \quad (2)$$

Où  $x_{i,t}$  représente la consommation d'électricité du consommateur i sur une période (par exemple, une heure); si  $x_i, t$  est nul, on le représente comme *null*.

### ➤ Normalisation des données

Les données doivent être normalisées car le réseau de neurones est sensible à la diversité de données. L'une des méthodes courantes pour cela est la normalisation min-max, elle est calculée selon la formule 3 ci-dessous :

$$F(X_{i,t}) = \frac{X_{i,t} - \min(X_{i,T})}{\max(X_{i,T}) - \min(X_{i,T})}, \quad (3)$$

Où  $\min(\cdot)$  et  $\max(\cdot)$  représentent les valeurs min et max sur une journée, respectivement.

## 2.2. Génération d'ensembles de données d'entraînement et de test [44]

Pour évaluer les performances de la méthodologie, l'ensemble de données prétraité est divisé en un ensemble de données d'entraînement et un ensemble de données de test par l'algorithme de validation croisée.

- a. L'ensemble de données d'entraînement est utilisé pour former les paramètres de notre modèle.
- b. Tandis que l'ensemble de données de test est utilisé pour évaluer dans quelle mesure le modèle se généralise à de nouveaux échantillons de clients invisibles. Étant donné que les consommateurs de vol d'électricité sont nettement plus nombreux que les consommateurs non frauduleux, la nature déséquilibrée de l'ensemble de données peut avoir un impact négatif majeur sur les performances des méthodes d'apprentissage automatique supervisé. Pour réduire ce biais, l'algorithme de technique de sur-échantillonnage minoritaire synthétique (SMOTE) est utilisé pour rendre le nombre de vols d'électricité et de consommateurs non frauduleux égal dans l'ensemble de données d'entraînement.

## 2.3. Classification à l'aide du modèle CNN

Dans le modèle CNN proposé, un réseau neuronal convolutif (CNN) est d'abord conçu pour apprendre les caractéristiques entre les différentes heures de la journée, à partir des données massives et variables des compteurs intelligents à l'aide des opérations de convolution et de sous-échantillonnage.

Ensuite, un classificateur de comportement est formé sur la base des caractéristiques obtenues pour détecter si le consommateur vole de l'électricité. Enfin, la matrice de confusion et le rapport de classification sont utilisés pour évaluer la précision du modèle CNN sur l'ensemble de données de test.

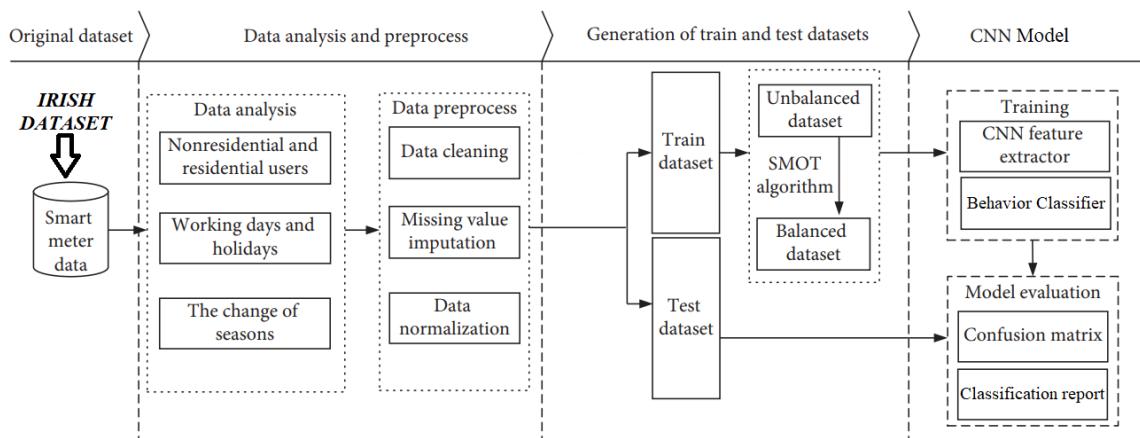


Figure 38 : Flux de détection de vol d'énergie avec CNN [44]

#### Etape 4 : Calcule de la nouvelle réputation

Le comportement du client (participation du client dans le consensus) et la réputation générée par le modèle IA (concerne le vol d'énergie) sont complémentaires. En effet, si un client vole de l'énergie, son comportement ne peut qu'être suspect.

Chaque jour, avec le modèle AI on met à jour deux réputations :

$$R_{theft} = a * R_{theftp} + (1 - a) * R_{theftc}$$

- $R_{theft}$  : Nouvelle valeur de réputation où **0** est la plus mauvaise réputation et **100** la meilleure.
- $R_{theftp}$  : Réputation précédente.
- $R_{theftc}$  : Réputation courante.

Et enfin on calcul la réputation  $R_{agrégé}$  qui est l'agrégation des deux réputations  $R_{theft}$  et  $R_n$  via la formule suivante :

$$R_{agrégé} = a * R_{agrégép} + \frac{1-a}{2} * R_{theft} + \frac{1-a}{2} * R_n$$

---

#### Algorithme 8 : Calcul de la nouvelle réputation

---

##### Les entrées :

*ED:* Energie distribué

*ER:* Energie récolté

*mesures : Mesures récoltées.*

*List\_rep\_bloc\_NAN:* La liste des réputations stockée au niveau du dernier bloc de la blockchain de niveau NAN

*List\_Rpia\_prec :* La liste des réputations de tour précédent calculé par le modèle IA

*List\_agrégé:* Copie de *List\_agrégé\_bloc* (stockée au niveau du dernier bloc de la blockchain)

*List\_theft:* Copie de *List\_theft\_bloc* (stockée au niveau du dernier bloc de la blockchain)

clé\_publique : Liste des clé publique des Nœuds de chaque région

**Paramètre:**

Nœud i ∈ (1, ..., n).

détecer\_vol: Déetecte de vol par le modèle IA

Rtheft: Output de détection de vol d'énergie

Rn: Réputation calculé au niveau NAN

**Les sorties :**

La liste des réputations mise à jour (List\_rep).

```

1: if (ER > 1.04 ED) then //vérifier si pertes ne dépasse pas les 4%
2:   for noeud i ∈ M do
3:     Rtheft = détecter_vol (mesure [i]) //appliquer modèle ia pour détecter noeud
malveillants
      List_theft[i] = (α * List_theft[i]) + (1-α )Rtheft
4:     Rn = List_rep_bloc_NAN[i] //récupérer la réputation de niveau NAN
5:     List_agrégré [i] = (α * List_agrégré [i] ) + (1-α)/2Rtheft + (1-α)/2 Rn
6:     if (List_agrégré [i] <= Rep_min ) then
7:       SendAlertToControlCenter(i)
8:     end if
9:   done
10: else for noeud i ∈ M do
11:   Rn = List_rep_bloc_NAN[i]
12:   Rtheftp = List_theftp[i]//dans le cas où les pertes ne dépasse pas les 4% on prend
ancien réputation calculé par le modèle IA
13:   List_agrégré [i] = (α * List_agrégré [i]) + (1-α)/2Rtheftp + (1-α)/2 Rn
14:   if (List_agrégré [i] <= Rep_min ) then
15:     SendAlertToControlCenter(i)
16:   end if
17: done

```

---

## **Etape 5 : Blockchain niveau WAN**

L'étape suivante concerne le niveau WAN de la blockchain Ethereum, la procédure est comme suit :

À ce niveau, comme déjà vu dans l'architecture proposée, les membres d'une blockchain sont tous les concentrateurs de données (DCs) d'un réseau WAN/FAN.

Chaque coordinateur (DC) d'un réseau NAN est chargé de transmettre les données contenues dans les blocs de la blockchain de son réseau NAN à la blockchain de son groupe dans le réseau FAN/WAN après avoir analysé les données et mis à jour la réputation des smart meter en détectant les nœuds qui volent de l'énergie .

Le centre de contrôle est ensuite autorisé à accéder aux blockchains des zones étendues par le biais des DCs, afin de lire les données agrégées des compteurs intelligents régionaux.

À ce niveau puisqu'on a supposé que les Dcs sont fiables, un DC leader sera sélectionné à tour de rôle entre les DCs. A chaque période t, on choisit un DC dont  $ID = t \text{ modulo } N$ , où N est le nombre de DCs, et  $t=t+1$  à chaque nouvelle période.

## 1. Construction de donnée D

Le DC suit presque le même processus que les smart meter pour construire une donnée D, ce processus est détaillé comme suit :

1. Le DC calcule le chiffré (C) des mesures et la liste des réputations des smart meter en utilisant la clé publique du centre de contrôle :

$$C = CH_{CléPubliqueDecentredecontrôle}(mesures \parallel listedesréputations)$$

La concaténation des mesures et des réputations se fait selon une structure de donnée appropriée pour affecter pour chaque client : ses mesures ainsi que sa réputation. Cette structure de donnée peut être par exemple une matrice à trois colonnes comportant les listes de réputations, les mesures et la clé publique du smart meter.

2. Le DC ajoute son identifiant (sa clé publique) au chiffré (C) pour construire un message (m).

$$m = clépublique \parallel C$$

3. Ensuite, il hache le message (m) avec une fonction de hachage.
4. Une fois le hach H(m) prêt, il construit la signature (S) de ce dernier avec sa clé privée.

$$S = CH_{cléprivé}(H(m))$$

5. A la fin, Il construit la donnée, qui contient deux informations, le message (m) et la signature (S).

$$D = m \parallel S$$

A la réception d'une donnée D, le DC Leader vérifie son intégrité comme ceci : (1) il prend la signature S, (2) il la déchiffre avec la clé publique DC émetteur pour obtenir un hach H'.

La mémoire locale de chaque nœud contient des données qui n'ont pas été enregistrées dans un bloc et qui ont passé le processus de vérification, arrivé là, ils seront enregistrés dans un bloc

## 2. Structure de bloc

Un bloc de ce niveau comporte cinq éléments : le numéro du bloc (id), le hachage du bloc précédent, le timestamp, le corps de données, et le hachage du bloc comme montré dans la figure 39.

Bloc i
Timestamp
Data : <ul style="list-style-type: none"> <li>• Clé publique SM1, Mesure1, Réputation1 : --</li> <li>• Clé publique SM2, Mesure2, Réputation2 : --</li> <li>• Clé publique SM3, Mesure3, Réputation3 : --</li> <li>...</li> </ul>
Hash précédent
Hash (Bloc i)

Figure 39 : Structure d'un bloc niveau WAN

## 4. Discussion

Ci-dessous nous indiquons comment notre solution contourne les attaques citées précédemment :

Technique d'attaque	Manière de défense
Déconnecter le compteur	Si un client malveillant déconnecte son compteur, sa réputation va diminuer et sera détecté par la blockchain.
Contourner le compteur pour supprimer les mesures	Si un compteur n'envoie pas ou contourne ses mesures, sa réputation va diminuer et sera détecté soit par la blockchain dans le cas du non envoi de mesures ou bien par l'IA dans le cas du contournement des mesures.
Intercepter/altérerer les communications	Pour que les données arrivent au DC, un SM chiffre et signe ses mesures et les transmet

	aux autres nœuds du réseau sur plusieurs canaux de communication (consensus), donc la falsification de l'information sera difficile car l'attaquant doit détourner plusieurs canaux ainsi que la clé privée du SM.
Arrêter de rapporter toute la consommation	Si un client malveillant ne rapporte pas toute sa consommation, il va être détecté par le modèle IA, ce qui affectera négativement sa réputation.
Modifier le profil de charge de l'appareil pour masquer les charges importantes	Si un client malveillant ne rapporte pas toute sa consommation, il va être détecté par le modèle IA ce qui affectera négativement sa réputation.
Déclarer zéro consommation	Si un client malveillant ne rapporte pas toute sa consommation, il va être détecté par le modèle IA ce qui affectera négativement sa réputation.
Intercepter et falsifier des paquets de données lors de la transmission des DCs au centre de contrôle dans les systèmes électriques existants ou entre les DCs.	Pour que les données arrivent au centre de contrôle, un DC chiffre et signe ses mesures et les transmet aux autres nœuds du réseau sur plusieurs canaux de communication (consensus), donc la falsification de l'information sera difficile car l'attaquant doit détourner plusieurs canaux ainsi que la clé privée du DC.
Envahir la base de données dans le centre de contrôle des systèmes électriques existants.	Les données dans le centre sont décentralisées sur plusieurs serveurs au lieu d'être centralisé dans un seul endroit et de plus, ils sont cryptés par clé publique de centre de contrôle.

**Tableau 14 :** Explication du détournement des attaques.

#### 4.1. Buts atteints

- La solution blockchain basée sur la réputation peut protéger le SG contre la FDIA extérieure où les nœuds défectueux et les paquets falsifiés peuvent être détectés
- La solution basée sur deep learning peut détecter les attaques de vol d'énergie et assure que les mesures envoyées par les smarts meter ne sont pas falsifiées.
- La solution combinée est capable de protéger le Smart Grid contre les deux attaques interne et externe et aussi d'identifier l'origine de l'attaque.

#### Conclusion

Dans ce chapitre de conception, nous avons présenté l'architecture générale AMI dans les réseaux SG, les vecteurs d'attaques qui se basent sur l'attaque false data injection (FDIA) dans le système de comptage intelligent. Nous avons aussi proposé une architecture et un cadre de protection des données énergétique en combinant la technologie blockchain et une technique deep learning en décrivant les principes de fonctionnement des deux méthodes et une collaboration pour renforcer la sécurité des données du système électrique moderne contre l'attaque FDIA et le vol d'énergie. Enfin on a expliqué comment notre solution répond aux attaques FDIA et les objectifs atteints.

Dans le chapitre suivant, nous allons nous consacrer à l'implémentation de notre solution via différents outils existant, où par la suite on atteindra les objectifs fixés préalablement concernant les menaces FDIA ainsi que le vol d'énergie.

# Chapitre 4

## Réalisation et Simulation

## Introduction

Dans ce dernier chapitre, nous allons présenter l'architecture finale de notre solution, énumérer les différents outils nécessaires à sa réalisation. Ensuite nous allons faire une analyse de performances et de sécurité en utilisant divers métriques pour l'intelligence artificielle d'une part et la blockchain de l'autre.

### 1. Architecture de Simulation

L'architecture de simulation implémentée est constituée de deux réseaux NAN, un réseau WAN, Un DC qui est installé pour chaque région et un compte administrateur pour déployer les contrats et gérer l'ensemble du réseau. Un tour de consensus dans la blockchain NAN est effectué chaque 30 minute, ce qui permet d'enregistrer les mesures de chaque NAN et de mettre à jour la valeur **Rcons** (Réputation du consensus) de chaque nœud.

À la fin de la journée, les 48 mesures collectés sont fournis au modèle deep learning installé sur le DC pour la détection d'un quelconque vol d'énergie pour que la Réputation **Rtheft** soit mise à jour.

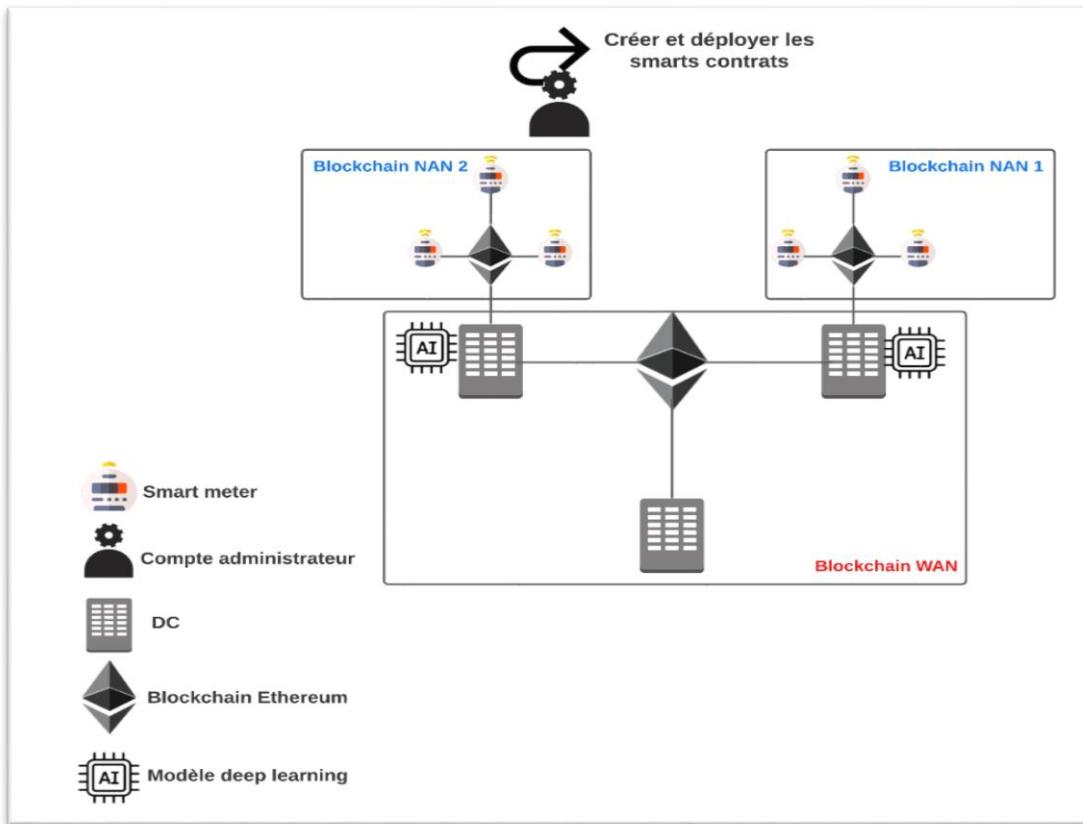
La réputation agrégée est à son tour mise à jour dans deux cas :

- Si **Rcons** est supérieur à la réputation minimale dans le réseau NAN (chaque 30 minute)
- Si **Rtheft** est supérieure à la réputation minimale dans le DC (chaque 24 heures, cela est dû au fait que le modèle IA a besoin de 48 mesures pour le calcul de **Rtheft**).

Après la mise des réputations, celles-ci sont enregistrées sur la blockchain WAN avec les mesures.

#### Remarque :

La réputation minimale est prédéfini dans notre réalisation à 40, la raison est expliquée un peu plus en bas dans la section « définition des paramètres »



*Figure 40 : Architecture de simulation*

## 2. Environnement de travail

### Outils d'implémentations

#### Partie Intelligence Artificielle

##### Python



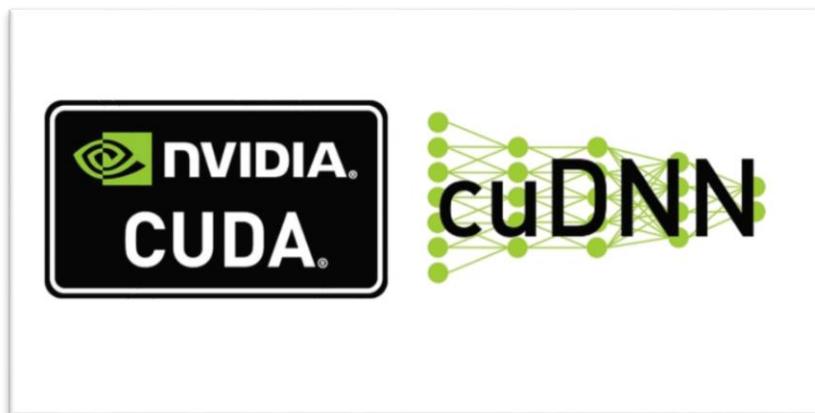
Python est un langage de programmation interprété, multi-paradigme et multiplateformes. Il favorise la programmation impérative structurée, fonctionnelle et orientée objet. Il est considéré comme étant le langage le plus populaire dans le monde de l'intelligence artificielle [Net 12].

## Tensorflow



TensorFlow est un framework de Machine Learning créé par Google et disponible en open source. Il est compatible avec python et simplifie le processus d'acquisition de données, d'entraînement des modèles de Machine Learning/deep learning, de génération de prédictions et de raffinement des résultats futurs. L'API front-end pratique de python aide à créer des applications et modèle IA plus facilement et de manière plus rapide [Net 13].

## cuDNN (NVIDIA CUDA Deep Neural Network)



La bibliothèque NVIDIA CUDA® Deep Neural Network (cuDNN) est une bibliothèque accélérée par GPU de primitives pour les réseaux de neurones profonds. Elle fournit des implémentations hautement optimisées pour la routine standard telle que la convolution, la mise en commun, la normalisation et les couches d'activation. cuDNN accélère les frameworks d'apprentissage en profondeur largement utilisés, notamment TensorFlow, Keras, MATLAB, PyTorch..etc.

Cette bibliothèque est largement utilisée par les chercheurs en apprentissage profond et les développeurs de frameworks du monde entier qui veulent profiter d'une accélération GPU hautes performances [Net 14].

## Pandas



Pandas est une bibliothèque logicielle écrite pour le langage de programmation Python pour la manipulation et l'analyse de données. En particulier, il propose des structures de données et des opérations de manipulation de tableaux numériques et de séries chronologiques [Net 15].

## Scikit-learn



Scikit-learn (connu sous le nom de sklearn) est une bibliothèque de machine learning pour le langage de programmation Python. Elle comporte divers algorithmes de classification, de régression et de clustering, et est conçue pour interagir avec les bibliothèques numériques et scientifiques Python NumPy et SciPy [Net 16].

## Partie Blockchain

### Ganache gui



Ganache gui est une blockchain personnelle pour le développement Ethereum utilisée pour déployer des contrats, développer des applications et exécuter des tests. Elle est disponible à la fois comme application de bureau et comme outil de ligne de commande (anciennement connu sous le nom de TestRPC). Ganache est disponible pour Windows, Mac et Linux [Net 17].

### Brownie



Brownie est un framework basé sur python pour les contrats intelligents ciblant la machine virtuelle Ethereum. Parmis les dépendances de ganache on trouve python et ganache [Net 18].

### 3. Evaluation

#### 3.1. Modèle IA :

Toutes les expériences sont implémentées en Python 3.8 sur un PC standard avec un processeur Intel Core i5-8300H fonctionnant à 2,30 GHz – 4 GHz et avec 8,0 Go de RAM DDR4. L'architecture et l'interface CNN est construite sur la base de TensorFlow et scikitlearn. Pour l'apprentissage en profondeur, les coeurs CUDA de la Nvidia Geforce GTX1050 4GB GDDR5 sont utilisés car ils sont spécifiquement conçus pour des tâches telles que le traitement parallèle, la mise à l'échelle en temps réel et l'accélération de l'apprentissage.

##### 3.1.1. Les paramètres :

Dans un modèle CNN, chaque couche a deux types de paramètres : les poids (pondérations) et les biais. Le nombre total de paramètres n'est que la somme de tous les poids et biais.

- **Weights :** Les poids sont les coefficients de l'équation qu'on essaye de résoudre. Lorsqu'un réseau de neurones est formé sur l'ensemble d'apprentissage, il est initialisé avec un ensemble de poids. Ces poids sont ensuite optimisés pendant la période d'entraînement et les poids optimaux sont produits.

Les poids utilisés par notre modèle ont la forme suivante :

$$[(576,), (36864,), (6144,), (96,)]$$

- **Bias :** Le biais est simplement une valeur constante (ou un vecteur constant) qui est ajoutée au produit des entrées et des poids. Le biais est utilisé pour compenser le résultat. Le biais est utilisé pour déplacer le résultat de la fonction d'activation vers le côté positif ou négatif.

Les bias utilisés par notre modèle ont la forme suivante :

$$[(64,), (64,), (48,), (2,)]$$

La somme des deux types de paramètres est :

$$\text{Somme} = 576 + 36864 + 6144 + 96 + 64 + 64 + 48 + 2 = 43858.$$

La figure suivante reflète l'architecture générale du modèle utilisée :

Layer (type)	Output Shape	Param #
conv2d_24 (Conv2D)	(None, 4, 6, 64)	640
conv2d_25 (Conv2D)	(None, 2, 4, 64)	36928
max_pooling2d_12 (MaxPooling)	(None, 1, 2, 64)	0
dropout_12 (Dropout)	(None, 1, 2, 64)	0
flatten_12 (Flatten)	(None, 128)	0
dense_24 (Dense)	(None, 48)	6192
dense_25 (Dense)	(None, 2)	98
Total params:	43,858	
Trainable params:	43,858	
Non-trainable params:	0	

Figure 41: Architecture modèle CNN utilisé

- Choix des hyperparamètres du modèle.
  - Optimiser utilisé : **Adam**, avec taux d'apprentissage (Learning rate) = **0.001**

En utilisant de grands modèles et ensembles de données, il a été démontré qu'Adam peut résoudre plus efficacement des problèmes pratiques d'apprentissage en profondeur [50] comme le montre la figure suivante :

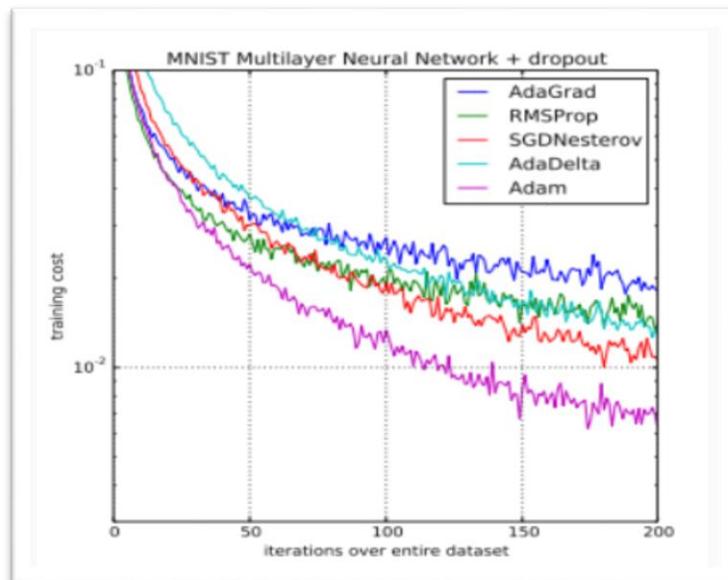


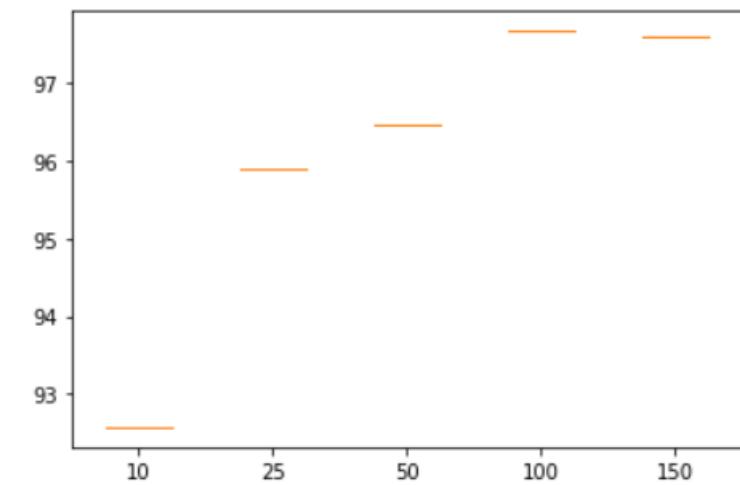
Figure 42 : Comparaison d'Adam avec d'autres algorithmes d'optimisation [50]

- Fonction de perte : Categorical crossentropy
- Fonction d'activations : Relu (Rectified Linear Unit) et **Softmax** en dernier pour convertir le score en probabilité vu le contexte de classification multi-classe (2 classes : vol et pas vol).
- Epoques d'entraînement : 100

Vu que le taux de détection du modèle CNN peut varier selon la période de l'entraînement (nombre d'époques), une expérimentation a été faite pour visualiser le comportement du modèle en variant le nombre d'époques d'entraînement.

Avec un nombre d'époque égal à 100, notre modèle a une plus grande précision comme le montre la figure 43.

```
epoches=10: 92.574% (+/-0.000)
epoches=25: 95.898% (+/-0.000)
epoches=50: 96.464% (+/-0.000)
epoches=100: 97.666% (+/-0.000)
epoches=150: 97.595% (+/-0.000)
```



*Figure 43 : Taux de prédiction en fonction de l'époque d'entraînement*

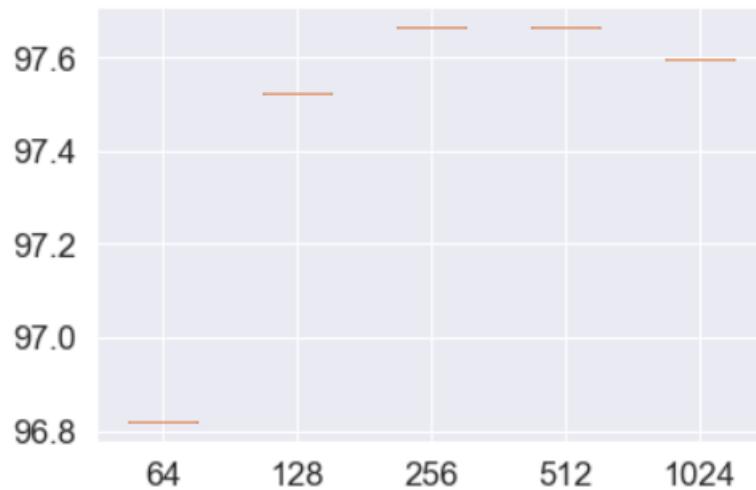
Au-delà de 100, le modèle pourrait converger vers du overfitting (sur apprentissage) qui peut s'avérer fatidique pour les performances du modèle.

- Taille du Batch : 256

La taille du lot dépend de la taille du jeu de donnée et du système. On peut sélectionner la taille du lot autant que notre RAM GPU peut contenir. De plus, le nombre de tailles de lot doit être choisi pas trop haut et pas très bas.

➤ Nombre de filtres dans les couches convolutives ; 256

Une expérimentation a été faite sur la base du nombre de filtres à initialiser, le résultat est illustré comme ceci :



*Figure 44 : Taux de prédiction en fonction du nombre de filtres*

La figure 44 montre une certaine convergence du taux de prédiction selon le nombre de filtres implémenté. En effet, on peut dire que le nombre de filtres n'influe pas beaucoup sur l'efficacité du modèle car le taux de prédiction est approximativement le même, toutefois il est meilleur lorsque  $N_{filter} = 256$  ou  $N_{filter} = 512$ .

### 3.1.2. Les métriques :

**Rapport de classification :** Un rapport de classification est utilisé pour mesurer la qualité des prédictions à partir d'un algorithme de classification. Combien de prédictions sont vraies et combien sont fausses.

	Precision	Recall	F1-Score
Classe 0 (Non vol)	0.99	0.99	0.99
Classe 1 (Vol)	0.99	0.99	0.99
Moyenne / Total	0.99	0.99	0.99

*Tableau 15 : Score de classification pour le modèle CNN généré*

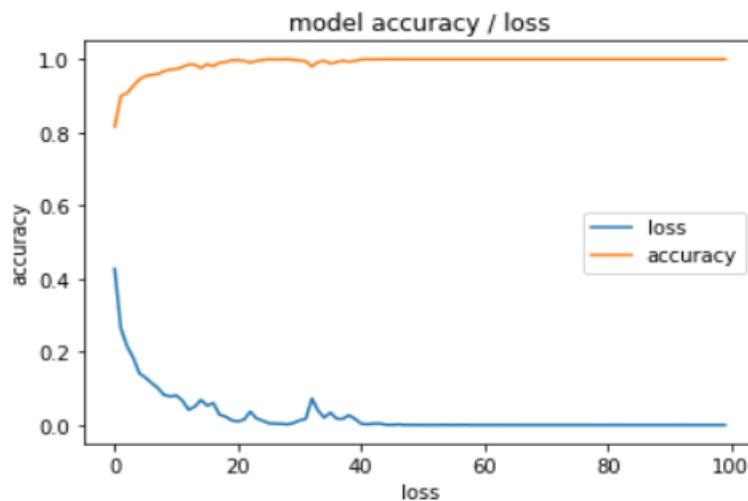


Figure 45 : Prédiction/perte en fonction des époques d'entraînement

Le Taux de prédiction du modèle est de 99%.

## Chargement du meilleur modèle entraîné

```
Entrée [63]: from keras.models import Sequential, load_model
model = tf.keras.models.load_model('my_best_model.epoch96-loss0.08.hdf5')

accuracy = model.evaluate(X_test, Y_test)

print(accuracy)

45/45 [=====] - 0s 4ms/step - loss: 0.0580 - accuracy: 0.9873
[0.057961318641901016, 0.987270176410675]
```

Figure 46 : Evaluation du modèle entraîné

- **Matrice de confusion** : Une matrice de confusion contient les informations sur les classifications réelles et prévues effectuées par une approche de classification comme le montre le tableau suivant.

		Valeur Prédit	
		Négative	Positive
Valeur Réel	Négative	Vrai négatif (TN)	Faux positive (FP)
	Positive	Faux négatif (FN)	Vrai positive (TP)

TP, FP, FN et TN sont définis comme suit :

- True Negative (TN) : est le nombre de classification correcte des noeuds normaux.
- False Negative (FN) : est le nombre de classification incorrecte des noeuds d'attaques en tant que noeuds normaux.
- False Positive (FP) : est le nombre de classification incorrecte des noeuds normaux en tant que noeuds d'attaque.
- True Positive (TP) : est le nombre de classification correcte des noeuds d'attaques.

- **Le taux de faux positifs (FPR)** : est défini comme la proportion de noeuds normaux, qui sont incorrectement classés c-à-d classés comme noeuds d'attaque par le classifieur. Plus la valeur FPR est petite, plus l'efficacité est grande.

$$FPR = FP / (FP + TN)$$

- **Le taux de faux négatifs (FNR)** : est défini comme la proportion de noeuds d'attaque, qui sont incorrectement classés comme noeuds normaux par le classifieur. Plus la valeur FNR est petite, plus l'efficacité de la classification est grande.

$$FNR = FN / (TP + FN)$$

- **L'exactitude** : est définie comme le rapport entre le nombre de résultats correctement classés et le nombre total de résultats classés. Une valeur d'exactitude élevée indique une classification efficace.

$$Exactitude = (TP + TN) / (TP + TN + FP + FN)$$

La matrice de confusion de notre modèle est comme suit :

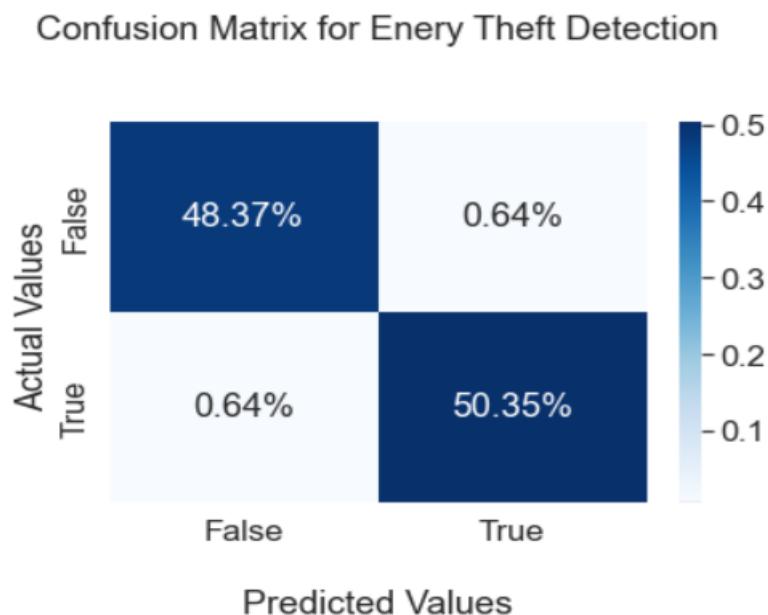


Figure 47 : Matrice de confusion du modèle généré

➤ **Le temps d'exécution**

Le temps d'exécution de la prédiction pour la toute première mesure peut être élevé à cause du temps nécessaire pour charger les modules essentiels à la prédiction sur le système (Tensorflow, cuDNN). Ce temps est approximativement de 20 secondes à 40 secondes. Puis les prédictions suivantes prennent pas plus de 100 millisecondes (0.1 secondes) pour être calculés.

### Exemple d'application :

Prenons le scénario suivant : Au niveau du DC, l'appel à l'IA se fait sur 5 smarts meter (SM1-SM5), les figures suivantes reflètent le temps d'exécution et le résultat des cinq prédictions du modèle IA sur les smarts meter précédents :

```
Traitements des fichiers de mesures..
SM1.csv
Lecture des mesures du fichier SM1.csv..
Traitement des Mesures du Smart Meter 1000
Prétraitement des mesures..
Prédiction..
2022-06-11 12:09:37.172051: I tensorflow/compiler/mlir/mlir_graph_optimization_pass.cc:185] None of the MLIR Optimization Passes are enabled (registered 2)
2022-06-11 12:09:44.360587: I tensorflow/stream_executor/cuda/cuda_dnn.cc:369] Loaded cuDNN version 8100
Probabilités des deux classes est : [[0.85779774 0.14220233]]
Probabilité de Non Vol du Smart Meter 1000 est de : 0.85779774 ==> 86%
-----Fin de la prédiction des mesures du Smart Meter 1000-----
Le temps d'exécution de la prédiction pour le Smart Meter 1000 est : 23.4689226151 secondes
```

*Figure 48 : Temps d'exécution + résultat de la Prédiction Numéro 1*

```
SM2.csv
Lecture des mesures du fichier SM2.csv..
Traitement des Mesures du Smart Meter 1001
Prétraitement des mesures..
Prédiction..
Probabilités des deux classes est : [[3.3415803e-15 1.0000000e+00]]
Probabilité de Vol du Smart Meter 1001 est de : 1.0 ==> 100%
-----Fin de la prédiction des mesures du Smart Meter 1001-----
Le temps d'exécution de la prédiction pour le Smart Meter 1001 est : 0.0781214237 secondes
```

*Figure 49 : Temps d'exécution + résultat de la Prédiction Numéro 2*

```
SM3.csv
Lecture des mesures du fichier SM3.csv..
Traitement des Mesures du Smart Meter 1002
Prétraitement des mesures..
Prédiction..
Probabilités des deux classes est : [[0.7038552 0.29614472]]
Probabilité de Non Vol du Smart Meter 1002 est de : 0.7038552 ==> 70%
-----Fin de la prédiction des mesures du Smart Meter 1002-----
Le temps d'exécution de la prédiction pour le Smart Meter 1002 est : 0.0624637604 secondes
```

*Figure 50 : Temps d'exécution + résultat de la Prédiction Numéro 3*

```
SM4.csv
Lecture des mesures du fichier SM4.csv..
Traitement des Mesures du Smart Meter 1003
Prétraitement des mesures..
Prédiction..
Probabilités des deux classes est : [[9.9987769e-01 1.2225068e-04]]
Probabilité de Non Vol du Smart Meter 1003 est de : 0.9998777 ==> 100%
-----Fin de la prédiction des mesures du Smart Meter 1003-----
Le temps d'exécution de la prédiction pour le Smart Meter 1003 est : 0.0623693466 secondes
```

*Figure 51 : Temps d'exécution + résultat de la Prédiction Numéro 4*

```

SM5.CSV
Lecture des mesures du fichier SM5.CSV..
Traitement des Mesures du Smart Meter 1004
Prétraitement des mesures..
Prédiction..
Probabilités des deux classes est : [[0.99358165 0.00641842]]
Probabilité de Non Vol du Smart Meter 1004 est de : 0.99358165 ==> 99%
-----Fin de la prédiction des mesures du Smart Meter 1004-----
Le temps d'exécution de la prédiction pour le Smart Meter 1004 est : 0.0937473774 secondes

```

Figure 52 : Temps d'exécution + résultat de la Prédiction Numéro 5

	Exactitude	FPR	Latence (s)
IA	<b>98.72%</b>	<b>0.64%</b>	<b>1ère prédiction ≈ 20</b> <b>Les suivantes : &lt; 0.1</b>

Tableau 16 : Résultat de la solution de l'apprentissage automatique contre les attaques internes.

### 3.2.Blockchain :

#### 3.2.1. Métriques

Afin d'évaluer le protocole de consensus proposé, nous effectuons des simulations pour vérifier sa précision et son efficacité. Pour ce faire, nous décrivons les métriques à analyser présentées ci-dessous :

- **Détection des nœuds malicieux (NM)** : est la détection des nœuds nommés malveillants, c'est-à-dire ont une réputation inférieur à la valeur '*Repminimale*' aux cours des cycles d'exécutions.
- **Coût de communication** : est le nombre de messages échangés dans un tour de consensus.
- **Coût des transactions** : Le Gas (une unité de mesure utilisée pour mesurer le travail effectué par Ethereum pour effectuer des transactions ou toute interaction au sein du réseau), dans notre cas on se charge d'étudier le gas consommé par l'ensemble des transactions dans chaque tour de consensus Nan ou Wan
- **Latence par cycle d'exécution**: est le temps nécessaire pour accomplir un cycle d'exécution. Dans notre cas, on calcule *la latence et le coût de communication* pour les deux blockchains NAN et WAN.
- **Taux des nœuds validateurs suspicieux** : est la détection des nœuds qui ont un dysfonctionnement (soit malveillant ou défaillant) pendant un cycle d'exécution.

### 3.2.2. Les paramètres

Le tableau suivant énumère les paramètres utilisés pour nos expériences :

Paramètre	Valeurs	Description
<b>N</b>	[9, 30, 60, 90]	Taille du réseau global : $N = N_{SM} + N_{DC}$
<b>N<sub>SM</sub></b>	[6, 20, 40, 60]	Nombre de smart meter
<b>N<sub>DC</sub></b>	[3, 10, 20, 30]	Nombre de DC dans le réseau WAN
<b>N<sub>SM_NAN</sub></b>	[3, 10, 20, 30]	Nombre de smart meter dans chaque réseau NAN
<b>P</b>	[0.2, 0.5]	Pourcentage des nœuds SM validateurs
<b>Repmin</b>	40	La valeur minimale de la réputation pour laquelle les nœuds malveillants commencent à être signalés.
<b>NM</b>	[20%, 30%, 40%]	Pourcentages des nœuds malveillants utilisés pour les tests
<b>NV<sub>M</sub></b>	[20%, 40%]	Le taux de nœuds validateurs malicieux
<b>Alpha</b>	[0.2, 0.5, 0.8]	Paramètre utilisé pour calculer la réputation de comportement du nœud dans le consensus

*Tableau 17 : Les paramètres utilisés dans l'évaluation de la blockchain*

### 3.2.3. Simulations, résultats, et analyses :

#### ➤ Fixation des paramètres et détection des nœuds malveillants

Un nœud peut être déclaré comme malicieux (NM) si sa valeur de réputation est inférieure à la valeur minimale de réputation précisée (Rep minimale). L'évaluation de Cette métrique montre l'impact du paramètre alpha ( $\alpha$ ) utilisé dans la méthode EWMA pour la détection des nœuds malveillants comme montré dans la figure 56. Lorsqu'on maximise alpha dans la méthode EWMA, La valeur de réputation mise à jour d'un nœud fautif est faiblement diminuée, ce qui nécessite des nombreuses chances à un nœud avant d'être déclaré malveillant, tel que selon la formule suivante de EWMA :

$$R_n = \alpha * R_p + (1 - \alpha) * R_c$$

Si  $\alpha$  est maximisé, on donne une importance à la réputation précédente  $R_p$  plus que la courante, ce qui fait que la diminution n'est pas très importante, si  $\alpha$  est minimisé on donne importance à  $R_c$  (réputation courante), C'est pourquoi dans notre cas nous avons choisi une

valeur  $\alpha$  minimale afin que la réputation soit diminuée de manière à ce que le nœud suspectif soit déclaré le plus rapidement possible.

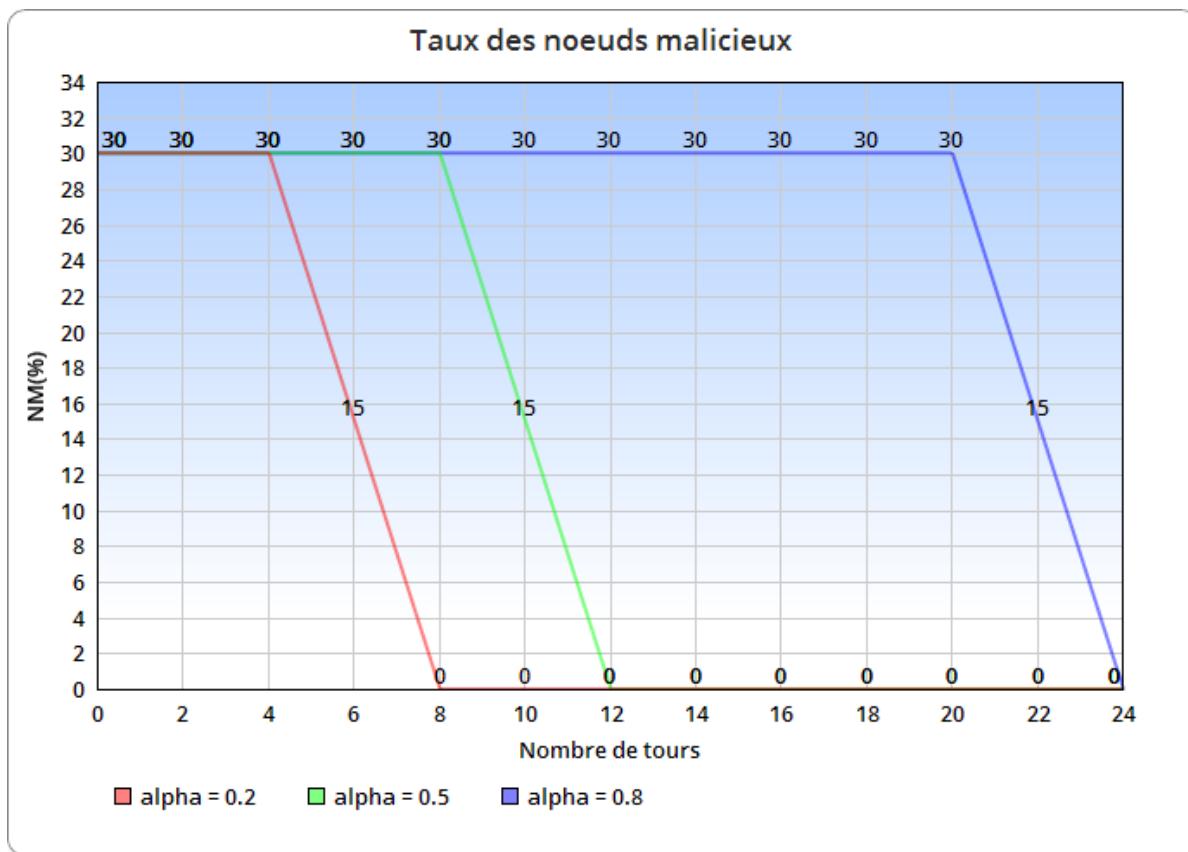


Figure 53 : Temps de détections des nœuds malicieux par rapport à alpha

#### Remarque :

La réputation minimale influence sur le temps de détection des nœuds malveillants. En effet, son initialisation avec une grande valeur peut induire le système en erreur en signalant une attaque dès sa détection alors qu'elle peut être une fausse attaque (False positive). Par contre, définir la réputation minimale avec une valeur minime permet au système de se donner plusieurs chances (plusieurs opportunités) pour la détection d'une attaque, ce qui réduit considérablement une détection d'attaque false positive.

#### ➤ Coût de communication

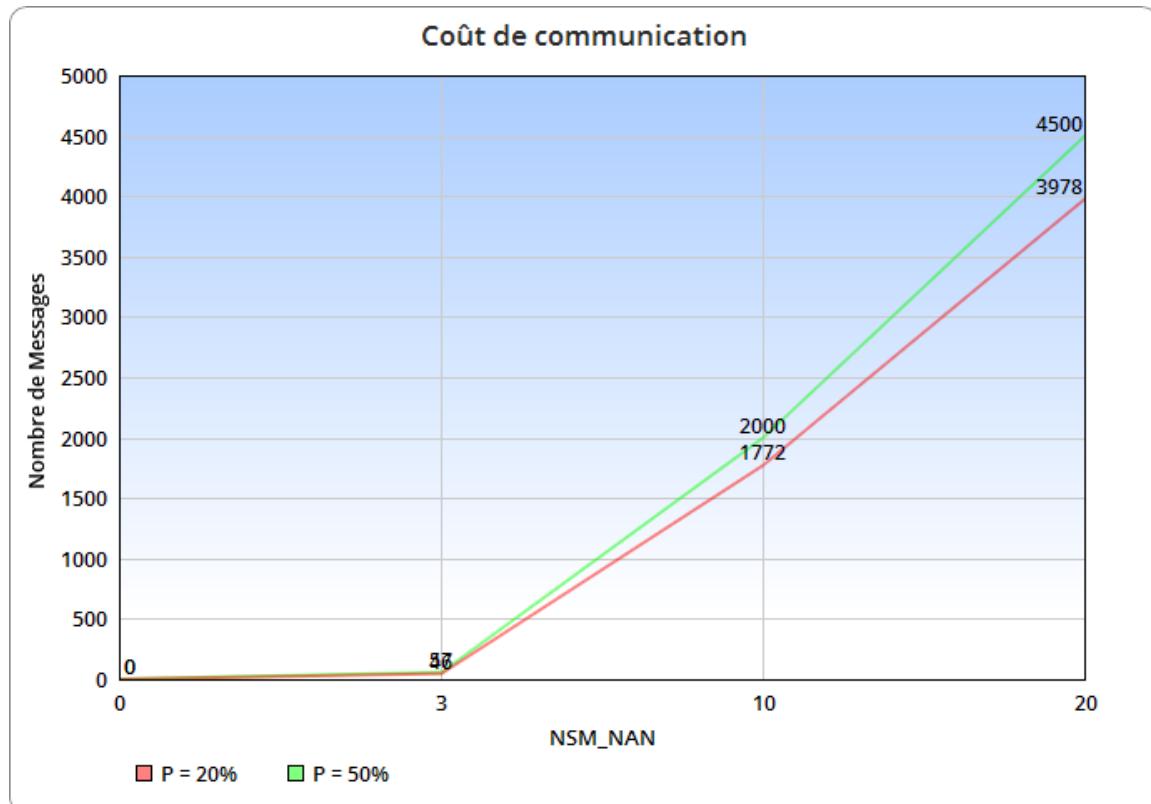
Pour calculer les messages échangés entre les nœuds à chaque tour de consensus, on utilise une variable « count » initialisée à 0 au début de tour de consensus et incrémentée à chaque envoi de message entre les nœuds. En supposant que la taille du réseau NAN est  $N_{SM\_NAN}$ , et que le groupe de validateurs est ‘P’ (représente le % des nœuds qui valident le bloc).

Les nœuds communiquent entre eux dans les cas suivants :

- Etablissement de connexion

- Echanges des mesures entre les nœuds
- Vérification de block B1 (par les nœuds validateurs)
- Echanges de votes.
- Vérification de block B2 (par les nœuds validateurs)

Les résultats de simulation sont présentés dans graphe suivant :



*Figure 54 : Coût de communication*

Comme montré dans la figure 54, le nombre de messages augmente en augmentant le nombre des nœuds, parce qu'il aura plus de messages dans la phase d'établissement de connexion et l'échanges des mesures entre les nœuds. Cette augmentation est présente aussi dans le cas d'une augmentation des nœuds validateurs  $P$  et cela est dû aux votes échangés dans la phase de validation de B1 et B2

#### ➤ Coût de Transactions

Chaque transaction effectuée en Ethereum est mesurée par le Gas, qui est l'unité d'évaluation du travail effectué dans le réseau. Lorsqu'un contrat est exécuté par la machine virtuelle l'Ethereum, il consomme du gas. Lorsqu'il ne dispose plus de gas, il arrête de fonctionner, donc ce dernier est un facteur important pour évaluer la performance de notre système. Pour ce faire, on calcule les coûts des transactions à chaque tour de consensus selon le nombre des nœuds et  $P$  (pourcentage des noeuds validateur) au :

### 1. déploiement :

Au déploiement, le smart contrat Solidity est ajouté à la blockchain ainsi que les smarts meter de chaque NAN et Les DCs de réseau WAN, en variant le nombre des nœuds. On peut voir la croissance du GAS nécessaire pour le déploiement en fonction de la taille du réseau, ce croissant est bilatéral comme montré dans la figure 55

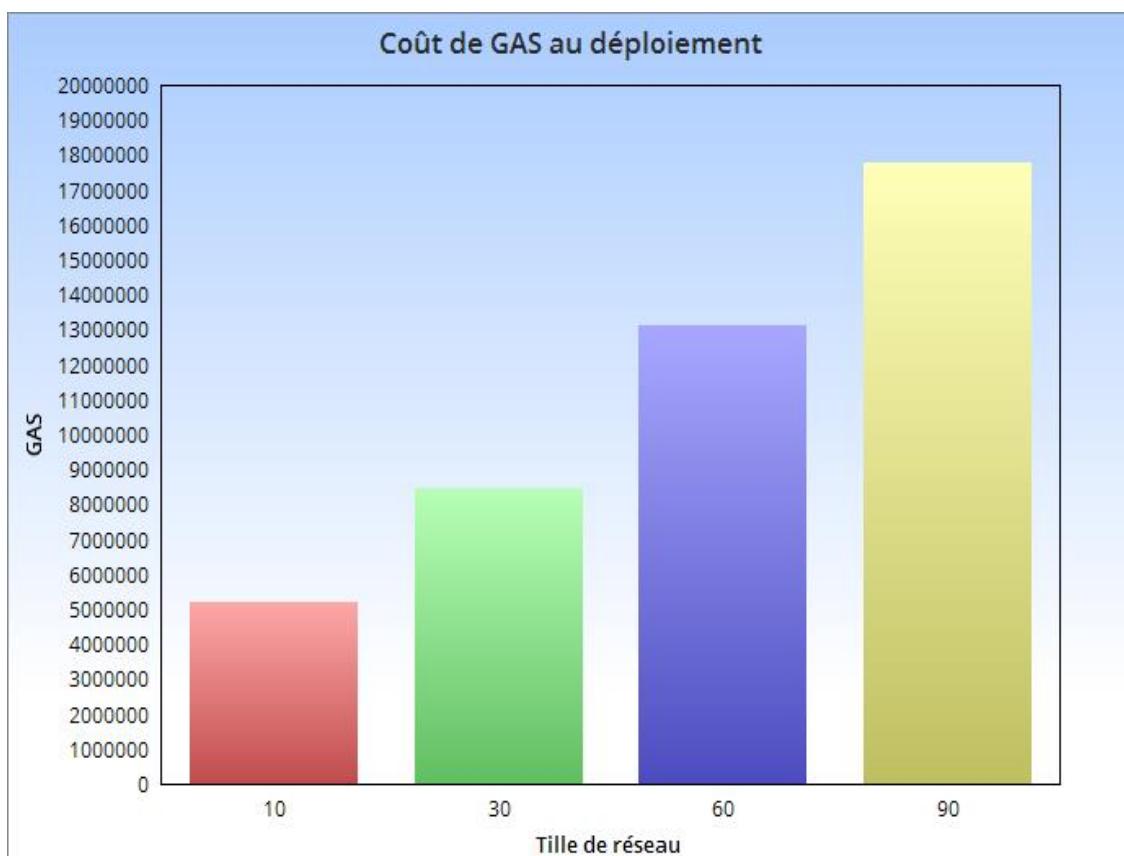
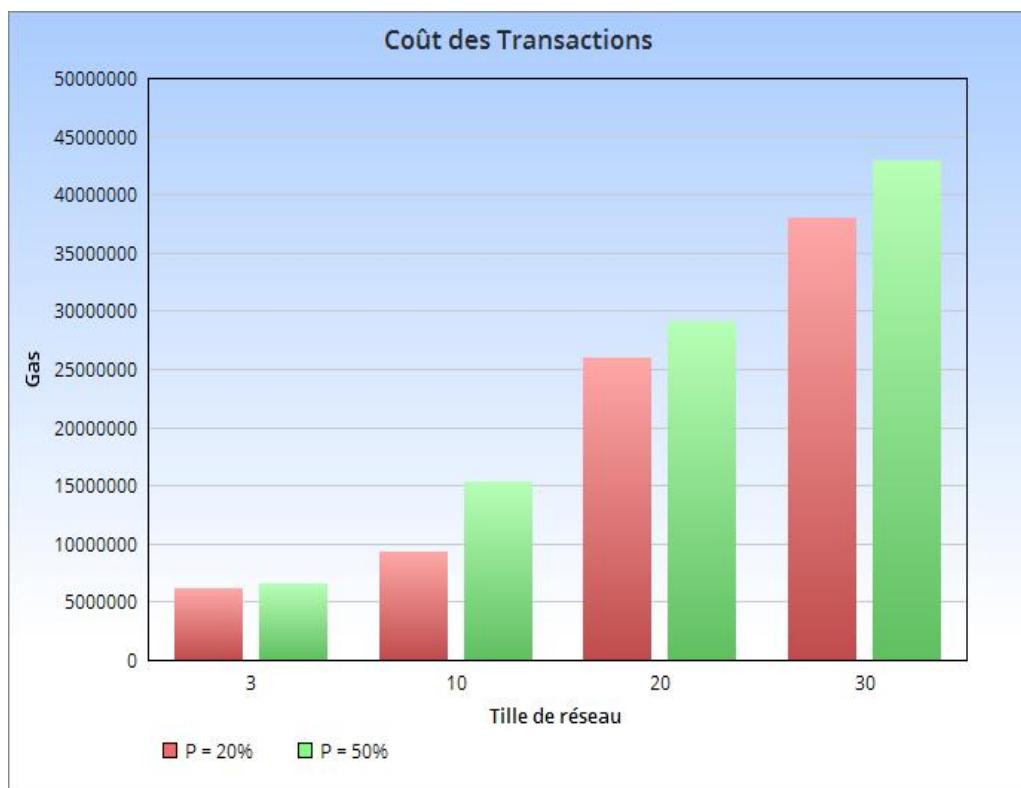


Figure 55 : Coût de Transactions au déploiement selon la taille de réseau

### 2. Chaque transaction :

À chaque transaction, divers traitements sont effectués sur la blockchain (ajout du block B1, vote, validation de B1, mise à jour des réputations, etc..), en observant le GAS nécessaire pour les transactions effectués durant chaque tour de consensus, on remarque qu'en augmentant la taille de réseau NAN et le nombre de nœuds validateurs, le GAS nécessaire augmente à son tour. Cela est dû à l'augmentation des interactions et des traitements effectués sur la blockchain comme le montre la figure 56.



*Figure 56 : Cout des transactions selon la taille du NAN*

#### ➤ Latence par cycle d'exécution

La latence est un indicateur important pour évaluer la performance d'un système distribué. Après de nombreuses simulations, nous avons pu obtenir une variété de temps de consensus selon le nombre de nœuds validateurs, et que le temps nécessaire pour atteindre ce consensus PoT augmente principalement avec la croissance du nombre de nœuds de validation (P), car ce sont eux qui valident le bloc proposé et diffusent leurs votes à tous les nœuds du réseau.

Le temps d'exécution du cycle comprend également le temps d'échange des mesures, où chaque nœud envoie sa propre consommation d'électricité générée à tous les nœuds du réseau, cela rallonge le temps d'exécution. Les résultats de simulations pour chaque réseau sont illustrés comme suit :

➤ **Chaque transaction NAN :**

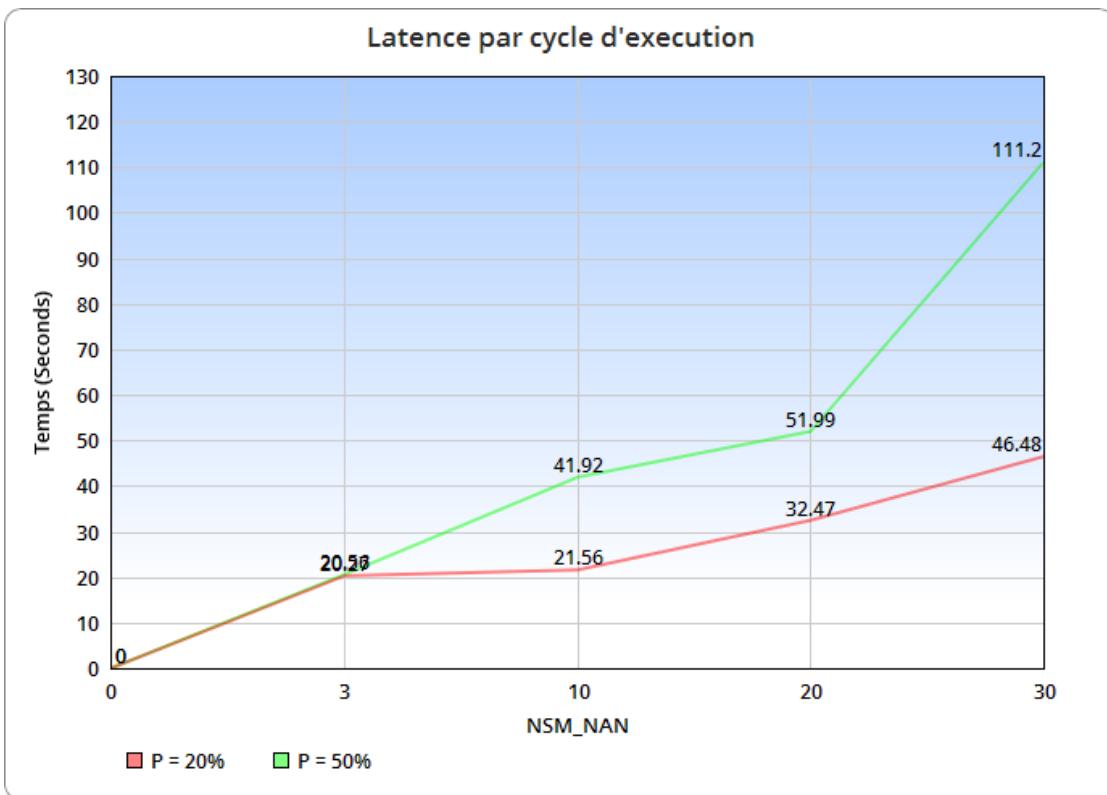


Figure 57 : Latence par cycle de consensus selon taille de réseau du NAN

➤ **Chaque transaction WAN :**

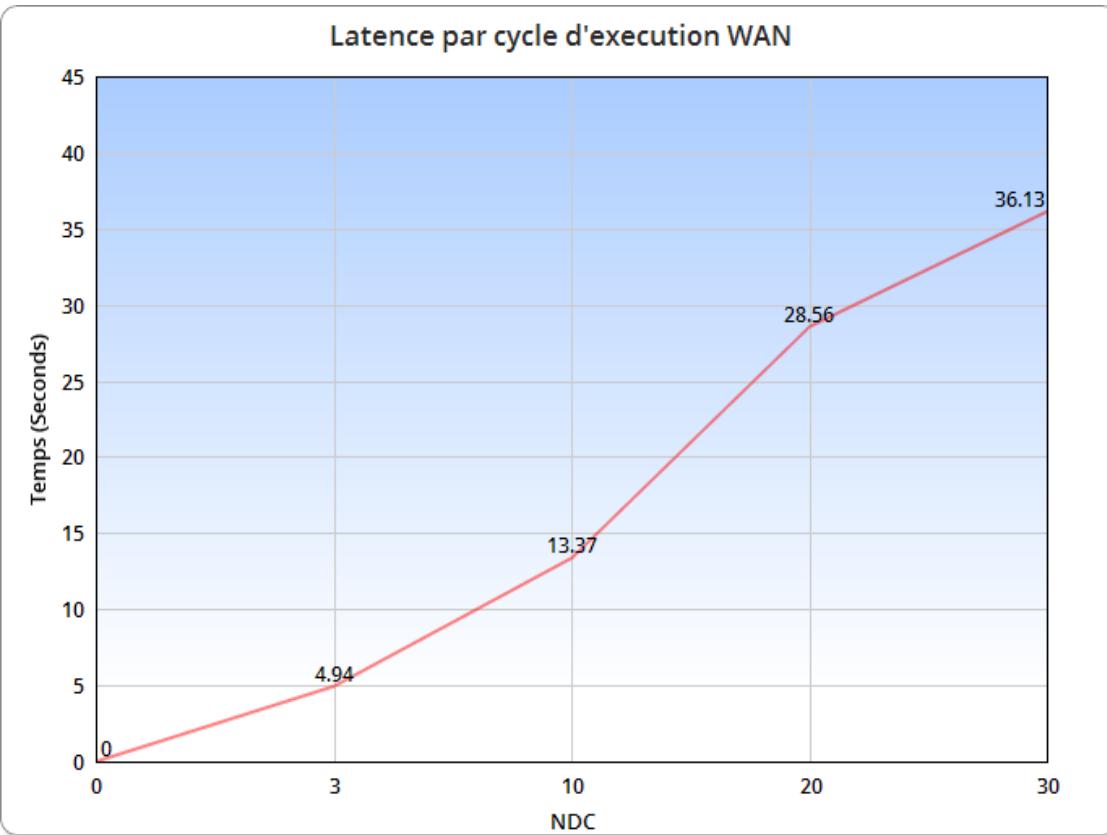
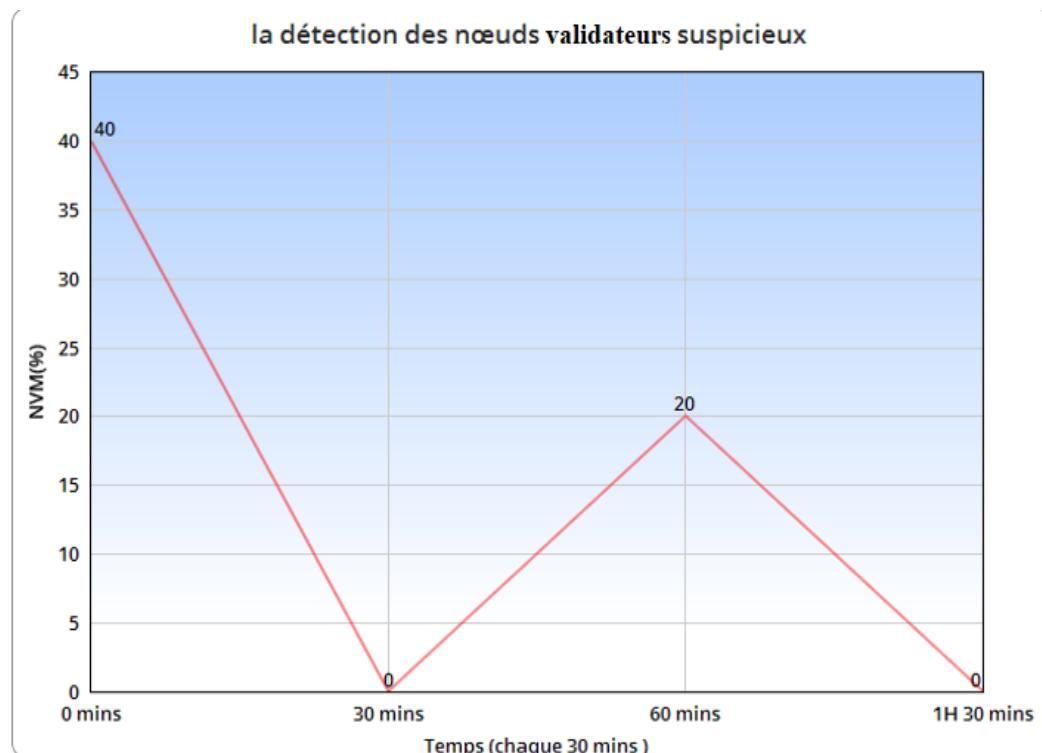


Figure 58 : Latence par cycle d'exécution dans le réseau WAN

### ➤ Taux des nœuds validateurs suspicieux

Pour cette métrique, nous allons évaluer le taux des nœuds validateurs dans POT qui peut identifier et supprimer les nœuds suspicieux des nœuds validateurs de consensus grâce au système de réputation appliqués, qui permet de sanctionner ces nœuds qui ont une réputation minime et sélectionner d'autres bien classés comme montré dans le scénario suivant:

1. Premier tour de consensus les nœuds validateurs sont [**‘1000’, ‘1001’, ‘1002’, ‘1003’, ‘1004’**]
2. Les nœuds **‘1000’** et **‘1001’** vont générer des fausses données, ils seront éliminés durant le deuxième tour de consensus.
3. Pour le dernier tour, le nœud **‘1003’** va générer des fausses données, il sera donc éliminé de la liste de validateurs.



*Figure 59 : La détection et l'élimination des nœuds validateurs malicieux*

La figure 59 prouve que notre système réduit le taux des nœuds validateurs suspicieux assurant ainsi la sécurité et la disponibilité des données.

### 3.3. Combinaison Blockchain et IA :

#### 3.3.1. Scénarios de simulation

Pour l'étape de validation, On fixe les paramètres pour les deux scénarios liés à :

##### ➤ L'attaque interne

Altération de la mesure dans le SM avant son envoi, en génère des mesures d'attaques pour les deux smartmeter ayant ID ‘1008’ et ‘1009’, et on observe la réputation  $R_{theft}$  et  $R_{cons}$  des deux smartmeter,

##### ➤ L'attaque externe

Falsification de la mesure lors de sa transmission depuis les deux smarts meter ayant l'id ‘1008’ et ‘1009’ lors de leurs envoie, et on observe la réputation  $R_{Theft}$  et  $R_{cons}$  des deux smarts meter

##### ➤ L'attaque interne et externe

Pour cette attaque, on va effectuer une simulation à la fois interne et externe et observer les deux réputations  $R_{theft}$ ,  $R_{cons}$  ainsi que  $R_{Agrégé}$ .

#### 3.3.2. Simulations, résultats, et analyses

##### ➤ La solution basée blockchain

En observant les réputations des nœuds, on remarque que  $R_{Theft}$  reste stable et donc l'attaque interne n'a pas été détectée. Par contre  $R_{cons}$  diminue à chaque tour de consensus, donc la blockchain diminue la réputation du nœud, Celui-ci sera déclaré dès qu'il atteint  $Rep_{MIN}$ , les résultats de simulation sont illustrés dans la figure 60.



Figure 60 : La diminution de réputation dans l'attaque externe

➤ **La solution basée sur l'apprentissage automatique**

En observant les réputations des nœuds, on remarque que  $R_{cons}$  reste stable et donc l'attaque externe n'a pas été détectée. Par contre  $R_{theft}$  diminue dès que le modèle IA est lancé (chaque 24H), les résultats de simulation sont illustrés dans la figure 61.

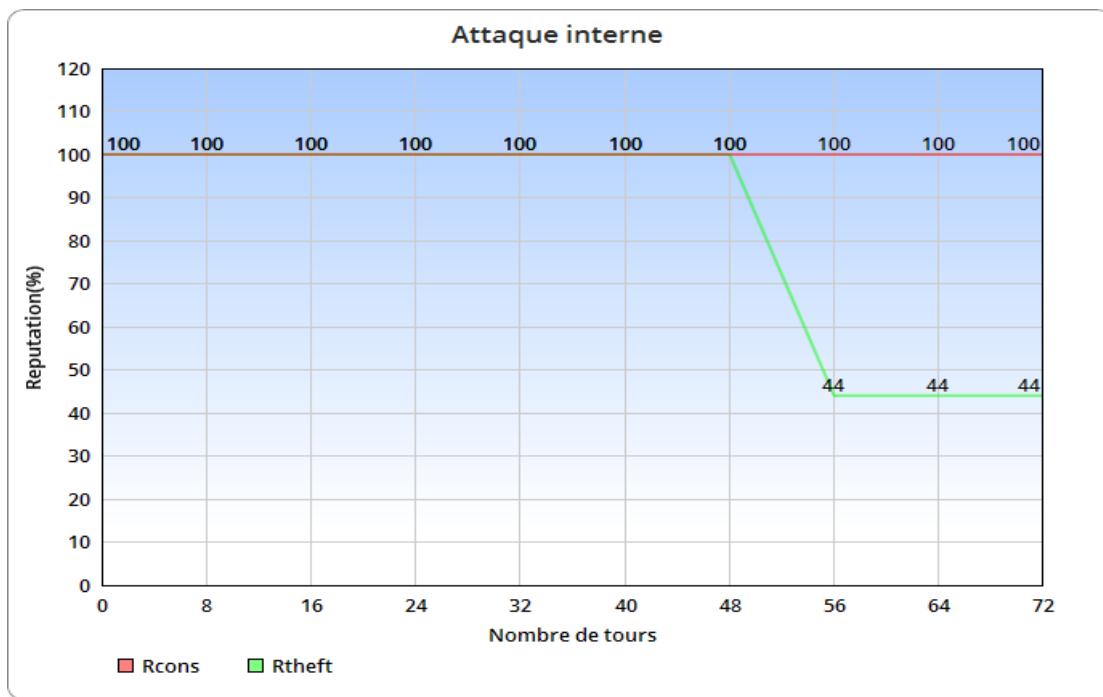


Figure 61 : La diminution de réputation dans l'attaque interne

➤ **Evolution de la réputation de la solution combinée blockchain et apprentissage automatique**

La solution blockchain et IA détecte les deux attaques interne et externe tel que la réputation  $R_{agrégé}$  sera diminuée dans les deux attaques, comme montré dans la figure 62 tel que :

[0, 40] : Aucune attaque n'a été lancé ni détecté

[40,48] : Attaque externe a été lancée par 1008 et 1009, détecté par les deux systèmes, blockchain et le système combiné

[48, 56] : Attaque interne a été détecté par les deux systèmes, IA et le système combiné après l'écoulement des 24 heures nécessaire à la récolte des 48 mesures.

[56, 64] : Aucune attaque n'a été lancée ni détecté

[64, 72] : Attaque externe a été lancée par 1008 et 1009, détecté par les deux systèmes, blockchain et le système combiné

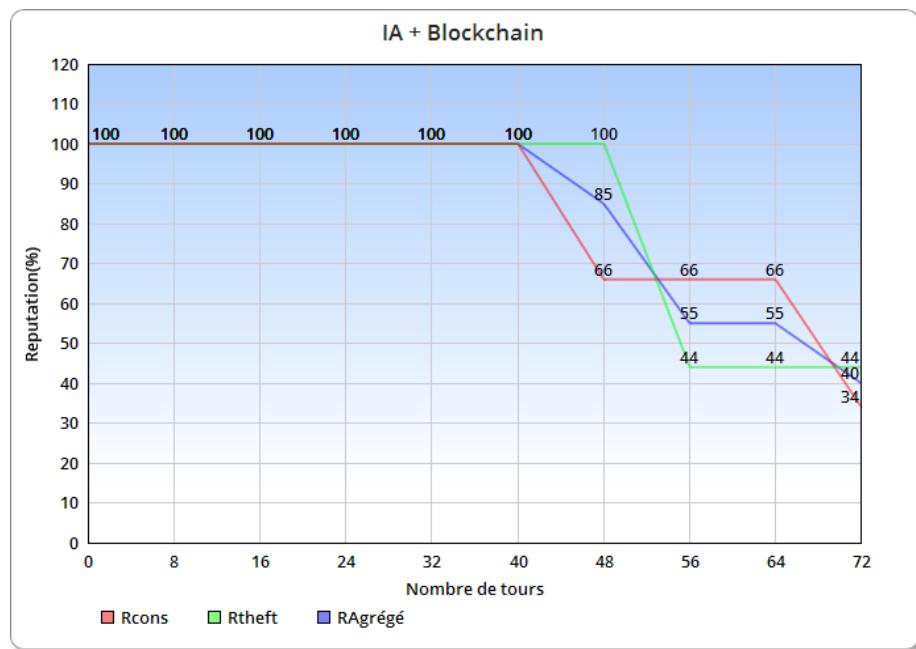


Figure 62 : La diminution de réputation dans l'attaque interne et externe

- **Taux de détection des NVm par la solution combinée**

Afin d'évaluer notre solution combinée et voir l'avantage ajouté, on va étudier le taux de détection des nœuds malicieux depuis l'instant t0 où la solution blockchain ainsi que la solution deep learning ont toutes les deux été ajoutées. Lors de ce test, 4 nœuds ont été utilisés : deux d'entre eux génèrent des attaques internes, et deux autres génèrent des attaques externes.

Les résultats de la simulation sont représentés dans la figure 63.

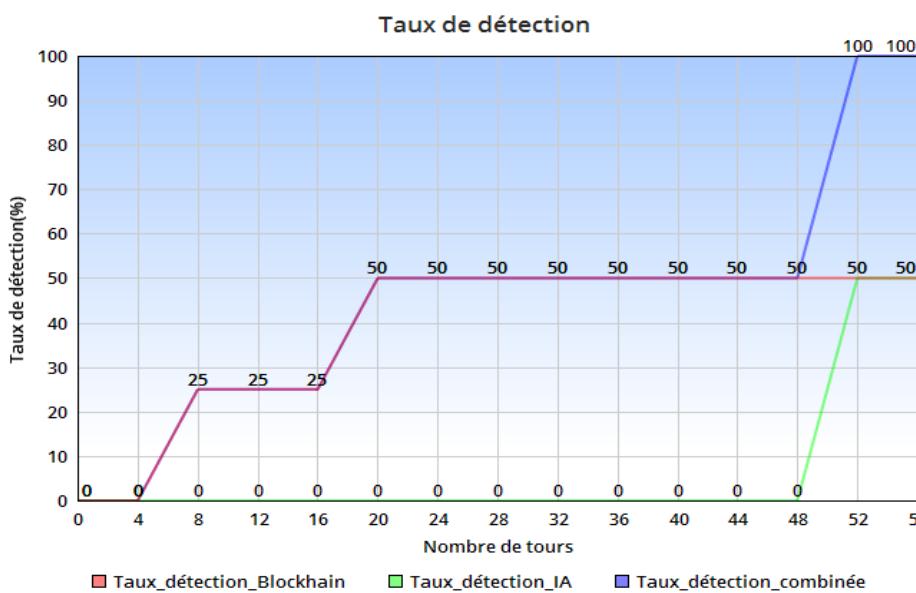


Figure 63 : Taux de détection de chaque solution

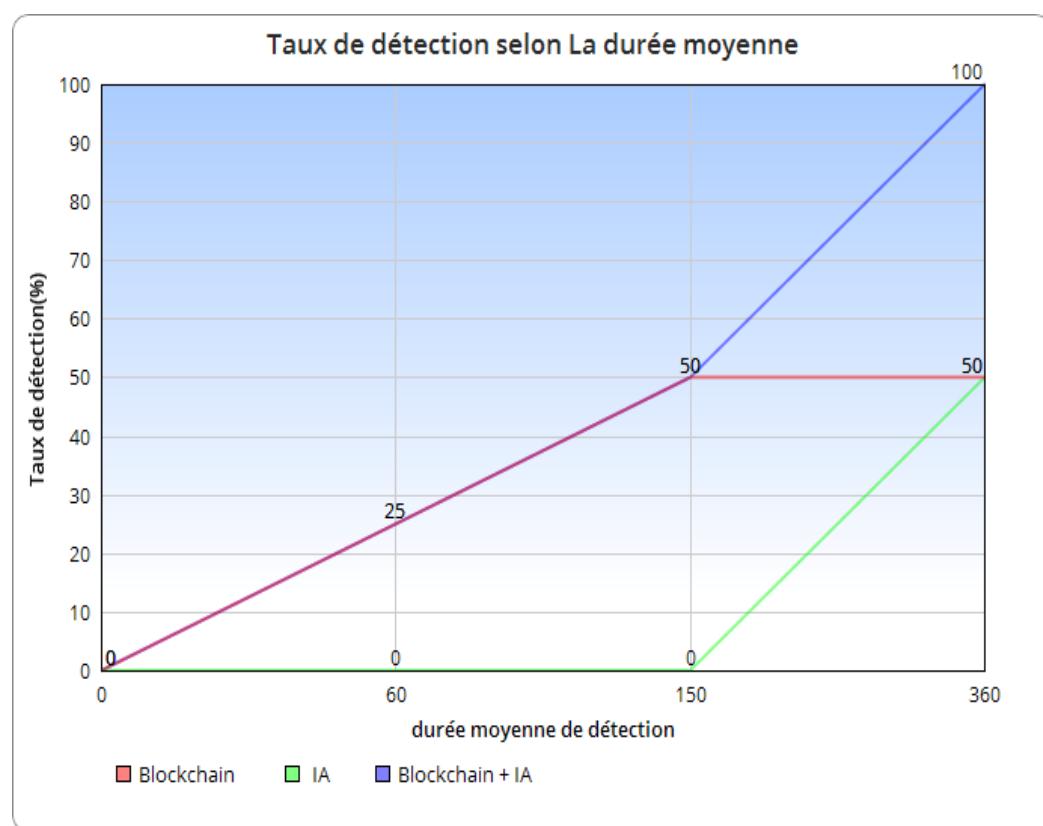
- De tour = 0 jusqu'à tour = 8 => 25% des nœuds sont détecté par la blockchain.
- De tour = 8 jusqu'à tour = 20 => 25% supplémentaires de nœuds sont détecté comme étant des nœuds malicieux par la blockchain.
- De tour = 48 jusqu'à tour = 52 => 50% des nœuds sont détecté par l'IA, l'attente jusqu'au 48ème tour est dû au fait que le modèle IA nécessite les 48 mesures de la journée.

Donc de tour = 4 jusqu'à tour = 52 => les nœuds malicieux ont tous été détectés, 50% par la blockchain et 50% par l'IA.

- **Durée moyenne de détection de chaque nœud malicieux**

Maintenant, on calcule la durée moyenne de détection qui représente le temps de détection de chaque nœud malicieux sur le nombre de nœuds malicieux. Et ceci en répartissant les taux de détection de chaque solution implémentée.

Les résultats de simulation sont présentés dans la figure 64.



*Figure 64 : Taux de détection selon la durée moyenne*

La figure 64 montre que :

1. La durée moyenne nécessaire à la détection de 50% des attaques externes est de 150 minutes.
2. La durée moyenne nécessaire à la détection de 50% des attaques internes est de 360 minutes et cela est dû à l'attente des 48 mesures nécessaires au modèle IA pour générer la prédiction.
3. La durée moyenne nécessaire à la détection de 100% des attaques internes et externes est de 360 minutes.

Enfin les résultats des différents cas de simulations confirment que :

1. La solution basée sur la blockchain peut protéger SG contre la FDIA externe mais ne parvient pas à détecter la FDIA interne.
2. La solution basée sur l'apprentissage automatique peut détecter les deux attaques, mais ne peut pas identifier leur type (interne ou externe) et donc la source de l'attaque.
3. La solution combinée est capable de détecter les deux attaques et réussit à identifier l'origine de l'attaque.

## Conclusion

Nous avons présenté dans ce chapitre l'architecture finale de notre solution, les outils utilisés dans la réalisation et les métriques analysées afin de mettre en œuvre et évaluer le modèle deep learning créé et la blockchain fournie, puis nous avons pu réaliser de nombreuses expériences, dont les résultats démontrent que d'un côté, grâce au modèle CNN conçu, le vol d'énergie est détecté à 99% et de l'autre, la blockchain permet d'en venir à bout de l'attaque FDIA.

Cela nous amène à conclure qu'avec l'intégration et la combinaison de ces deux technologies de l'IA et de la blockchain, la sécurité et la fiabilité d'un réseau intelligent est assurée de manière fiable, efficace et continue.

## Conclusion générale et perspectives

L'objectif des travaux menés dans le cadre de notre projet est de proposer une solution pour sécuriser les réseaux intelligents des attaques de vols d'énergie et de l'injection de fausses (FDIA). Cette solution est basée sur la combinaison des deux technologies de l'intelligence artificielle et la blockchain.

Le réseau intelligent est un système composé de divers composants distribués dont l'objectif principal est de fournir de l'électricité de manière intelligente, tout en permettant l'intégration facile de nouvelles fonctionnalités et mesures dans le réseau traditionnel. La cybersécurité dans le réseau intelligent est un domaine de recherche relativement nouveau et durant notre recherche, nous avons illustré le concept de SG en détail et fourni de nombreuses informations sur ses technologies conceptuelles, comme on a présenté les exigences et les défis en matière de sécurité dans ces réseaux intelligents. Cela a été suivi d'une discussion sur les opportunités et les techniques d'atténuation d'attaques basées sur technologies de la blockchain et l'IA déjà faite auparavant, celle-ci ont montré que la combinaison de ces deux technologies peut représenter un atout de force et de lutte contre différentes anomalies en terme de sécurité informatique dans les smarts grids.

Les récentes avancées de la Blockchain et de l'intelligence artificielle ont fait de ces technologies, des technologies révolutionnaires dans pratiquement tous les domaines, et en particulier dans la poursuite de la construction d'un réseau intelligent sécurisé et résilient.

Les techniques associées à l'IA comprennent les réseaux de neurones artificiels, la détection d'une consommation anormale de l'énergie, la quantité de l'énergie volée, et la vitesse de prédiction.

Les services de la blockchain permettent de leur cotés le stockage sécurisé et immuable des données, les transactions numériquement sécurisées et transparentes, la traçabilité des données, la confidentialité de données grâce à différentes techniques de cryptographie.

Durant notre projet, nous avons proposé une solution de sécurité basée sur la technologie Blockchain et l'intelligence artificielle pour éviter d'un côté l'attaque par fausse injection de données ainsi que les différentes attaques externes qui peuvent être menées sur l'architecture AMI. Et de l'autre, être apte à détecter un quelconque vol ou déroutement de l'énergie au niveau des smarts meters.

Afin de pouvoir démontrer l'efficacité et la fiabilité de la solution proposée, plusieurs métriques d'évaluations ont été faite. Les résultats démontrent que l'utilisation du modèle IA sur les DCs permet de détecté jusqu'à 99% des cas de vol d'énergie et l'utilisation d'une

blockchain privée garantir l'authenticité, la confidentialité et la fiabilité des données émises par les smart meters dans le réseau intelligent.

Pour les perspectives, Il ne fait aucun doute que l'avenir de l'énergie évolue vers un approvisionnement en énergie plus décentralisé, flexible et durable. L'intégration des TIC et des infrastructures énergétiques est l'un des facteurs les plus critiques pour le succès de l'évolution des réseaux intelligents. L'interopérabilité basée sur des normes est identifiée comme la base du processus d'intégration, sur laquelle une architecture de réseau décentralisée peut être mise en œuvre pour tenir compte de la nature de plus en plus dynamique et distribuée des actifs (production, consommation, surveillance, contrôle et gestion).

L'utilisation de la solution proposé en haut dans un environnement réel serait très intéressante et très bénéfique pour une vision plus exacte et précise de l'avancé mondiale pour la conception d'un réseau intelligent. Celle-ci montrera la résilience du smart meter ou du concentrateur de données face à l'attaque d'injection de fausse données, et l'aptitude du DC à détecter et identifié un smart meter corrompu ou malveillant.

## Webographie

[Net 1] 5 questions pour tout savoir sur les smart grid. Consulté le 10 février 2022.

<https://particulier.edf.fr/fr/accueil/guide-energie/electricite/smartgrid-reseau-electrique-intelligent.html>.

[Net 2] Principe de fonctionnement et enjeux des smart grids dans le tertiaire et l'industrie. Consulté le 16 février 2022.

<https://idelecplus.com/principe-de-fonctionnement-et-enjeux-des-smart-grids-dans-le-tertiaire-et-lindustrie>.

[Net 3] Study of Smart Grid Communication Network Architectures and Technologies. Consulté le 16 février 2022.

<https://www.scirp.org/journal/paperinformation.aspx?paperid=91158>

[Net 4] Smart Grid Communication Network. Consulté le 17 février 2022.

[https://www.researchgate.net/figure/Smart-Grid-Communication-Network\\_fig1\\_330882408](https://www.researchgate.net/figure/Smart-Grid-Communication-Network_fig1_330882408)

[Net 5] Electric Power Risk Assessment. Consulté le 17 février 2022.

<http://www.solarstorms.org/ElectricAssessment.html>

[Net 6] The Increasing Importance of Security for the Smart Grid. Consulté le 10 février 2022.

<https://www.power-grid.com/smart-grid/the-increasing-importance-of-security-for-the-smart-grid/>

[Net 7] CONSENSUS MECHANISMS, consulté le 24 février 2022

<https://ethereum.org/en/developers/docs/consensus-mechanisms/>

[Net 8] Permissioned and Permissionless Blockchains: A Comprehensive Guide, Consulté 1 mars 2022,<https://www.blockchain-council.org/blockchain/permissioned-and-permissionless-blockchains-a-comprehensive-guide/>

[Net 9] Blockchain : comprendre la distinction On Chain / Off Chain, Consulté 1 Mars 2022, <https://www.cryptoencyclopédie.com/single-post/blockchain-comprendre-distinction-On-Chain-Off-Chain>

[Net 10] What is deep learning and how does it work ? Consulté le 27 février 2022.

<https://towardsdatascience.com/what-is-deep-learning-and-how-does-it-work-2ce44bb692ac>

[Net 11] Electric Light & Power, “The increasing importance of security for the smart grid,” 2019. [Online]. Consulté 3 avril 2022.

Available: [https://www.elp.com/articles/powergrid\\_international/print/volume-16/issue-4/features/the-increasing-importance-of-security-for-the-smart-grid.html](https://www.elp.com/articles/powergrid_international/print/volume-16/issue-4/features/the-increasing-importance-of-security-for-the-smart-grid.html)

[Net 12] Python (*langage*). Consulté le 8 juin 2022.

[https://fr.wikipedia.org/wiki/Python\\_\(langage\)](https://fr.wikipedia.org/wiki/Python_(langage))

[Net 13] TensorFlow : le framework de Machine Learning de Google. Consulté le 8 juin 2022. <https://datascientest.com/tensorflow>

[Net 14] Nvidia cuDNN. Consulté le 9 juin 2022. <https://developer.nvidia.com/cudnn>

[Net 15] Pandas (*software*). Consulté le 9 juin 2022.

[https://en.wikipedia.org/wiki/Pandas\\_\(software\)](https://en.wikipedia.org/wiki/Pandas_(software))

[Net 16] Scikit-learn. Consulté le 9 juin 2022. <https://en.wikipedia.org/wiki/Scikit-learn>

[Net 17] Sweet tools for smart contracts. Consulté le 9 juin 2022. <https://trufflesuite.com/>

[Net 18] brownie. Consulté le 9 juin 2022. <https://eth-brownie.readthedocs.io/en/stable/>

## Bibliographie

[1] M. D. H. Abdullah, M. H. Zurina, Z. Zulkarnain, and M. A. Mohamed, ‘Attacks, Vulnerabilities and Security Requirements in Smart Metering Networks’, *KSII Trans. Internet Inf. Syst.*, vol. 9, pp. 1493–1515, Apr. 2015, doi: 10.3837/tiis.2015.04.013.

[2] Y. Liu, P. Ning, and M. K. Reiter, ‘False data injection attacks against state estimation in electric power grids’, in *Proceedings of the 16th ACM conference on Computer an*

[3] S. S. S. R. Depuru, L. Wang, V. Devabhaktuni, and N. Gudi, ‘Measures and setbacks for controlling electricity theft’, in *North American Power Symposium 2010*, Sep. 2010, pp. 1–8. doi: 10.1109/NAPS.2010.5619966

[4] Dharmesh Faquir, Nestoras Chouliaras , Vlachou Sofia , Kalopoulou Olga 2 , Leandros Maglarasx, ‘Cybersecurity in smart grids, challenges and solutions’, *AIMS Electronics and Electrical Engineering* 2021, Volume 5, Issue 1: 24-37. doi: 10.3934/electreng.2021002

[5] Z. El Mrabet , H. Elghazi , N. Kaabouch , H. Elghazi STRS Lab, INPT, Rabat, Morocco, “Cyber-Security in Smart Grid: Survey and Challenges”,*Electrical Engineering Department, UND, USA*

[6] Pardeep Kumar, Yun Lin, Guangdong Bai, Andrew Paverd, “Smart Grid Metering Networks: A Survey on Security, Privacy and Open Research Issues”, February 2019 *IEEE Communications Surveys & Tutorials* PP(99):1-1 DOI:10.1109/COMST.2019.2899354

[7] S. Aoufi, A. Derhab, and M. Guerroumi, ‘Survey of false data injection in smart power grid: Attacks, countermeasures and challenges’, *J. Inf. Secur. Appl.*, vol. 54, p. 102518,

Oct. 2020, doi: [10.1016/j.jisa.2020.102518](https://doi.org/10.1016/j.jisa.2020.102518).

[8] Kang JW , Joo IY , Choi DH . False data injection attacks on contingency analysis: attack strategies and impact assessment. *IEEE Access* 2018;6:8841–51 .

[13] AHMED S. MUSLEH , GANG YAO2 , AND S. M. MUYEEN 3 , *Blockchain Applications in Smart Grid—Review and Frameworks*, National Natural Science Foundation of China under Grant 61673260 and Grant 61603246, 10.1109/ACCESS.2019.2920682

[14] Muhammad Baqer Mollah, Jun Zhao, Dusit Niyato, Kwok-Yan Lam, Xin Zhang, Amer M.Y.M. Ghias, Leong Hai Koh, and Lei Yang, *Blockchain for Future Smart Grid: A Comprehensive Survey*, *IEEE INTERNET OF THINGS JOURNAL* 2020

[15] M. N. Luke, S. J. Lee, Z. Pekarek, and A. Dimitrova. (2018). *Blockchain in Electricity: A Critical Review of Progress to Date*.

[16] S. Aoufi, A. Derhab, and M. Guerroumi, ‘Survey of false data injection in smart power grid: Attacks, countermeasures and challenges’, *J. Inf. Secur. Appl.*, vol. 54, p. 102518, Oct. 2020, doi: [10.1016/j.jisa.2020.102518](https://doi.org/10.1016/j.jisa.2020.102518).

[17] G. Liang, S. R. Weller, F. Luo, J. Zhao, and Z. Y. Dong, ‘Distributed Blockchain-Based Data Protection Framework for Modern Power Systems Against Cyber Attacks’, *IEEE Trans. Smart Grid*, vol. 10, no. 3, pp. 3162–3173, May 2019, doi: [10.1109/TSG.2018.2819663](https://doi.org/10.1109/TSG.2018.2819663).

[18] J. Obert, A. Chavez and J. Johnson, "Behavioral Based Trust Metrics and the Smart Grid," 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE), New York, NY, pp. 1490-1493, 2018.

[19] Varun Badrinath Krishna, Ravishankar K. Iyer, William H. Sanders, “ARIMA-Based Modeling and Validation of Consumption Readings in Power Grids” *Information Trust Institute, Center Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, 1308 West Main Street, Urbana, IL 61801, USA - May 2016*.

[20] Paria Jokar, Student Member, IEEE, Nasim Arianpoo, Student Member, IEEE, and Victor C. M. Leung, Fellow, IEEE, “Electricity Theft Detection in AMI Using Customers’ Consumption Patterns”

[21] Shuan Li , Yinghua Han , Xu Yao, Song Yingchen, Jinkuan Wang, and Qiang Zhao “Electricity Theft Detection in Power Grids with Deep Learning and Random Forests” Received 19 March 2019; Revised 30 July 2019; Accepted 14 August 2019; Published 3 October 2019, <https://doi.org/10.1155/2019/4136874>

[23] J. C. Olivares-Rojas, E. Reyes-Archundia, J. A. Gutiérrez-Gnechi, J. Cerdá-Jacobo, and

J. W. González-Murueta, ‘A Novel Multitier Blockchain Architecture to Protect Data in Smart Metering Systems’, *IEEE Trans. Eng. Manag.*, vol. 67, no. 4, pp. 1271–1284, Nov. 2020, doi: 10.1109/TEM.2019.2950410.

[24] X. Kong, J. Zhang, H. Wang, and J. Shu, ‘Framework of decentralized multi-chain data management for power systems’, *CSEE J. Power Energy Syst.*, vol. 6, no. 2, pp. 458–468, Jun. 2020, doi: 10.17775/CSEEJPES.2018.00820.

[25] Donghyeok Lee, Namje Park, “Blockchain based privacy preserving multimedia intelligent video surveillance using secure Merkle tree”, *Springer Science+Business Media, LLC, part of Springer Nature 2020*

[26] Hamid Malik, *Performance Analysis of Blockchain-based Smart Grid with Ethereum and Hyperledger Implementations*,MASTER’S THESIS, OULU University

[27] Zhuang Q, Chen L. *Proof of Reputation : A Reputation-based Consensus Protocol for Blockchain Based Systems*. In: IECC '19: Proceedings of the 2019 International Electronics Communication Conference. 2019. p. 131–8.

[28] Ejaz UlHaq, Jianjun Huang, HuarongXu, Kang Li, Fiaz Ahmad “A hybrid approach based on deep learning and support vector machine for the detection of electricity theft in power grids”, *The 4th International Conference on Electrical Engineering and Green Energy CEEGE 2021, 10–13 June, Munich, Germany* . <https://doi.org/10.1016/j.egyr.2021.08.038>

[29] Muhammad Shafay, Raja Wasim Ahmad, Khaled Salah, Ibrar Yaqoob, Raja Jayaraman, Mohammed Omar, “Blockchain for Deep Learning: Review and Open Challenges ”. Khalifa University of Science and Technology at Award RCII-2019-002-Research Center for Digital Supply Chain and Operations Management and CIRA-2019-001.

[30] Yann LeCun, Yoshua Bengio & Geoffrey Hinton, Deep Learning, Review, doi:10.1038/nature14539.

[31] Marwa Keshk, Benjamin Turnbull, Nour Moustafa, Dinusha Vatsalan and Kim-Kwang Raymond Choo,A Privacy-Preserving Framework based Blockchain and Deep Learning for Protecting Smart Power Networks ,DOI 10.1109/TII.2019.2957140, *IEEE Transactions on Industrial Informatics*

[32] Mohamed Amine Ferrag, Leandros Maglaras ,*DeepCoin: A Novel Deep learning and Blockchain-based Energy Exchange Framework for Smart Grids*,

[33] C. H. Liu, Q. Lin, and S. Wen, “Blockchain-enabled data collection and sharing for industrial iot with deep reinforcement learning,” *IEEE Transactions on Industrial Informatics*, pp. 1–1, 2018.

[34] S. Raje, S. Vaderia, N. Wilson, and R. Panigrahi, “Decentralised firewall for malware detection,” in *Proc. Int. Conf. Adv. Comput., Commun. Control (ICAC)*, Dec. 2017, pp. 1–5.

[35] Xuhui Chen, Changqing Luo, Jinlong Ji, Weixian Liao, When Machine Learning Meets Blockchain: A Decentralized, Privacy-preserving and Secure Design, Conference Paper · December 2018 DOI: 10.1109/BigData.2018.8622598

[36] Shailendra Rathore , Yi Pan and Jong Hyuk Park, BlockDeepNet: A Blockchain-Based Secure Deep Learning for IoT Network, *Sustainability* 2019, 11(14), 3974; <https://doi.org/10.3390/su11143974>

[37] Zhaoyang, D. O. N. G., Fengji, L. U. O., & Liang, G. (September 2018) “Blockchain: a secure, decentralized, trusted cyber infrastructure solution for future energy systems”. *Journal of Modern Power Systems and Clean Energy*, 6(5), 958-967. ISSN Information: DOI: 10.1007/s40565-018-0418-0

[38] S. Aoufi, A. Derhab, and M. Guerroumi, ‘Survey of false data injection in smart power grid: Attacks, countermeasures and challenges’, *J. Inf. Secur. Appl.*, vol. 54, p. 102518, Oct. 2020, doi: 10.1016/j.jisa.2020.102518.

[39] G. Liang, S. R. Weller, F. Luo, J. Zhao, and Z. Y. Dong, ‘Distributed Blockchain-Based Data Protection Framework for Modern Power Systems Against Cyber Attacks’, *IEEE Trans. Smart Grid*, vol. 10, no. 3, pp. 3162–3173, May 2019, doi: 10.1109/TSG.2018.2819663.

[40] S. McLaughlin, B. Holbert, A. Fawaz, R. Berthier, and S. Zonouz, ‘A Multi-Sensor Energy Theft Detection Framework for Advanced Metering Infrastructures’, *IEEE J. Sel. Areas Commun.*, vol. 31, no. 7, pp. 1319–1330, Jul. 2013, doi: 10.1109/JSAC.2013.130714.

[41] J. C. Olivares-Rojas, E. Reyes-Archundia, J. A. Gutiérrez-Gnecchi, J. Cerdá-Jacobo, and J. W. González-Murueta, ‘A Novel Multitier Blockchain Architecture to Protect Data in Smart Metering Systems’, *IEEE Trans. Eng. Manag.*, vol. 67, no. 4, pp. 1271–1284, Nov. 2020, doi: 10.1109/TEM.2019.2950410.

[42] X. Kong, J. Zhang, H. Wang, and J. Shu, ‘Framework of decentralized multi-chain data management for power systems’, *CSEE J. Power Energy Syst.*, vol. 6, no. 2, pp. 458–468, Jun. 2020, doi: 10.17775/CSEJPES.2018.00820.

[43] Ali Akbar Ghasemi, Mohsen Gitizadeh, Electrical Power and Energy Systems, Department of Electronics and Electrical Engineering, Shiraz University of Technology, Shiraz, Iran

[44] Shuan Li , Yinghua Han , Xu Yao,Song Yingchen, Jinkuan Wang, and Qiang Zhao. “Electricity Theft Detection in Power Grids with Deep Learning and Random Forests”. *Journal of Electrical and Computer Engineering Volume 2019, Article ID 4136874, 12 pages* <https://doi.org/10.1155/2019/4136874>

[45] CER Smart Metering Project—Electricity Customer Behaviour Trial 2009–2010, May 2019, [online] Available: <https://www.ucd.ie/issda/CER-electricity>.

[46] Curbelo Montañez, C. A., & Hurst, W., A Machine Learning Approach for Unemployment Detection using the Smart Metering Infrastructure, *IEEE Access*, vol. 8, pp. 22525-22536, 2020

[47] Hunter, J. S. (1986). *The exponentially weighted moving average*. *Journal of quality technology*, 18(4), 203-210.

[48] T. B. Smith, “Electricity theft- comparative analysis,” *Energy Policy*, vol. 32, pp. 2067–2076, Aug. 2003.

[49] S. S. S. R. Depuru, L. Wang, V. Devabhaktuni, and N. Gudi, ‘Measures and setbacks for controlling electricity theft’, in *North American Power Symposium 2010, Sep. 2010*, pp. 1–8. doi: 10.1109/NAPS.2010.5619966.

[50] Diederik P. Kingma, Jimmy Ba. Titre : Adam: A Method for Stochastic Optimization. Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015. Available at <https://doi.org/10.48550/arXiv.1412.6980>

[51] Hrishikesh Mohan Dabir, Aditya Suresh Kadam, Gaurav Hadge, Ayushman Singh Rathore, Prof. Shubhangi Ingale “EFFICIENT ELECTRICITY THEFT DETECTION USING MACHINE LEARNING ALGORITHMS”. M.E.S. College of Engineering,Pune. Available at : [https://ijisrt.com/assets/upload/submitted\\_files/1576332640.pdf](https://ijisrt.com/assets/upload/submitted_files/1576332640.pdf)

# Annexes

## Annexe 1: Sous-systèmes de la sécurité du système électrique en ligne

### ➤ Contingency Analysis (CA)

L'analyse des contingences est une application logicielle de l'EMS, qui vise à vérifier la stabilité de l'état normal du système face à l'occurrence d'un événement ou à l'apparition de l'un des événements inattendus prédefinis, tels qu'une panne de générateur ou de ligne de transmission. L'analyse de contingence assure le rôle d'évaluation de la sécurité comme elle vise à préparer des actions correctives rapides si un événement inattendu se produit. [7]

### ➤ Security Constrained Optimal Power Flow (SCOPF)

Cette application est déclenchée lorsque l'état du système passe à un état normal non sécurisé et prend les ajustements de contrôle appropriés pour éviter les violations de sécurité. [7]

### ➤ Security Constrained Economic Dispatch (SCED)

Le SCED est le noyau du MMS, qui existe à un niveau plus élevé que le EMS, cette fonction définit le prix de l'électricité. Elle représente une cible attrayante pour les attaquants qui cherchent à réaliser des profits financiers illégaux. Les attaquants peuvent être l'un des clients du système de transmission tels que : les producteurs d'électricité, les consommateurs industriels, les compagnies ferroviaires, et les fournisseurs qui achètent et revendent de l'électricité.

## Annexe 2 : CA Attack

Dans cette attaque l'attaquant doit:

- Avoir la topologie du système, y compris l'état des disjoncteurs et le nom de chaque ligne.
- Avoir l'état des disjoncteurs et les paramètres/limites de chaque ligne.
- Compromettre les capteurs du système SCADA en observant et en manipulant les mesures.

- Compte tenu des mesures manipulées, l'adversaire peut effectuer des SE et des BDD pour calculer l'estimation du débit de la ligne souhaitée tout en n'étant pas détecté par le processus BDD.
- Obtenir la liste des contingences. Les paires de contingences ciblées sont abandonnées ou ajoutées à la liste de contingences sans attaque. A cette fin, en utilisant l'estimation du flux de lignes manipulée dans l'étape précédente, l'adversaire doit être en mesure de calculer de nouvelles estimations du flux de lignes contingences.

D'où FDIA peut être réalisable en :

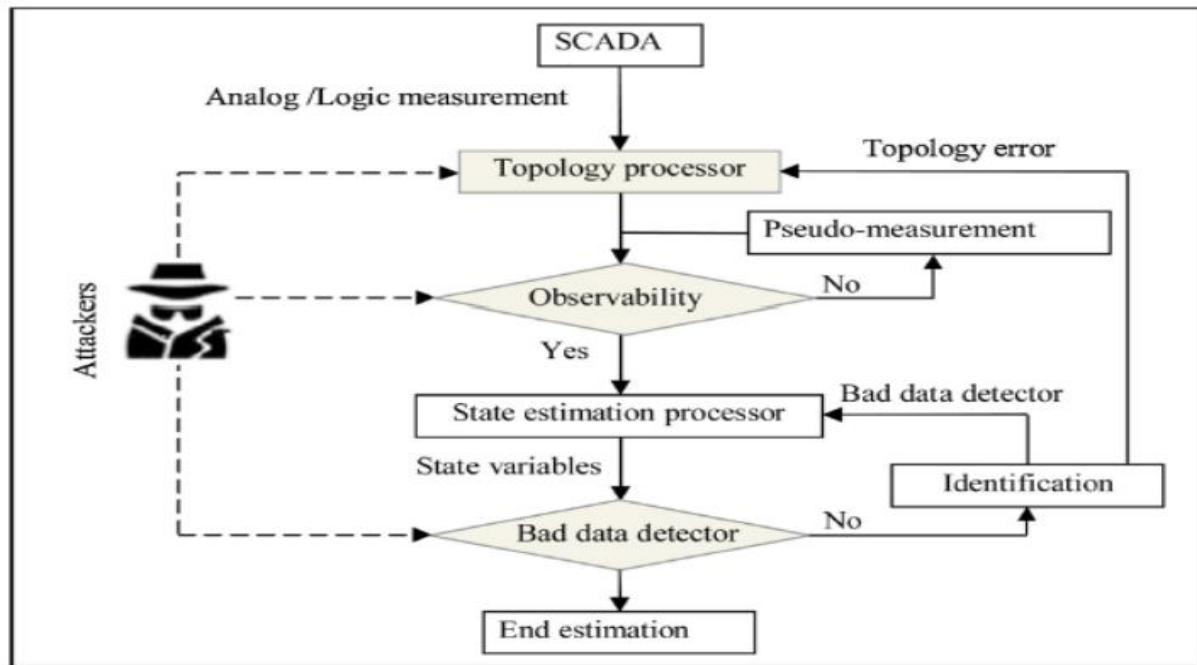
- Utilisant des mesures hors ligne ou en ligne pour estimer la topologie du système
- Injectant des données malveillantes dans les capteurs via un protocole de communication SCADA non sécurisé
- Comprenant l'architecture et les algorithmes des applications des systèmes d'énergie comme SE, CA et BDD à partir de livres et des publications de recherche.

### **Annexe 3 : Classification FDIA basée sur le système cible (Suite)**

#### **3.3.1. State estimation attack**

De mauvaises mesures peuvent être introduites pour diverses raisons telles que des défaillances de compteurs et les attaques malveillantes. Des techniques de détection de ces mauvaises mesures ont été développées comme la technique du *Bad data detection (BDD)* afin pour protéger l'estimation d'état [2], ainsi pour détecter l'injection de mauvaises données, la redondance des mesures est largement exploitée [7] et pour cela on compare la différence entre les mesures estimées  $Z'$  et les mesures réelles  $Z$  avec un seuil de tolérance en utilisant la formule :  $Z - Z' > \sigma$

Les mesures normales des capteurs donnent généralement une estimation des variables d'état proches de leurs valeurs réelles, tandis que les mesures anormales peuvent éloigner les variables d'état estimées de leurs vraies valeurs. Ainsi, il existe généralement une "incohérence" entre les bonnes et les mauvaises mesures [2].



*Figure 65 : Fonction du State Estimation.*

Sur la base de ces fonctions, nous pouvons définir trois classes d'attaques :

#### ➤ **Bypassing Bad Data Detector Attack**

Le système de détection de mauvaises données BDD compare la mesure estimée avec les mesures réelles. Si la différence dépasse un seuil prédéfini, une alarme est déclenchée. Supposent qu'un attaquant peut avoir accès à la matrice H et peut modifier n'importe quel compteur dans le réseau. Un vecteur d'attaque est construit de manière à éviter le BDD. Soit  $Z_a$  représentant les fausses mesures [7] :

$$Z_a = \mathbf{Z} + \mathbf{a} = \mathbf{H}(\mathbf{X}) + \mathbf{e} + \mathbf{a} = \mathbf{H}(\mathbf{X} + \mathbf{C})$$

Le vecteur d'attaque  $\mathbf{a}$  représente les données malveillantes qui sont ajoutées aux mesures originales, il est construit comme une combinaison linéaire des vecteurs colonnes de H. si nous trouvons  $a = H(C)$ , nous pouvons contourner le BDD [7].

#### ➤ **Observability Attack**

Vise à rendre le système inobservable en ciblant la disponibilité et l'intégrité des mesures collectées. Lorsqu'un système est inobservable, les attaquants peuvent injecter de mauvaises données pour obtenir des avantages économiques ou perturber le système d'alimentation sans être détectés [7].

### ➤ Topology Attack

Le processus de topologie (TP) construit un modèle électrique basé sur l'état des disjoncteurs et des dispositifs de commutation reçus. Ce modèle représente l'image des lignes de transmission entre les nœuds du réseau.

Topologie attack vise à lancer une erreur de topologie afin de pousser les opérateurs à modifier la topologie correcte en fonction de l'erreur de topologie conçue, ce qui conduit par conséquent à des états estimés erronés [7].

### 3.3.2. Contingency Analysis Attack (CA attack)

L'analyse des contingences (CA) est l'une des trois fonctions majeures de la sécurité des systèmes électriques, cette fonction joue le rôle d'évaluation de la sécurité [7].

L'attaquant supprime ou ajoute furtivement des paires de contingences de transmission de flux de lignes de transmission à partir d'une liste de contingence normale ou vers celle-ci en trompant le processus de CA par l'injection de fausses données. Les paires de contingences manipulées sont ensuite intégrées comme contraintes de sécurité dans les contraintes d'exploitation de la répartition économique sous contrainte de sécurité. Par conséquent, cette attaque conduit à un calcul erroné du prix marginal de localisation (LMP) sur les marchés de l'énergie en temps réel [8].

### 3.3.3. SCOPF Attack

En observant la séquence d'exécution des applications du système de sécurité de l'alimentation, la fonction OPF est automatiquement déclenchée lorsque l'AC passe à l'état normal de sécurité, dans le cas contraire, le SCOPF exécute une action préventive. L'AGC utilise le résultat de ces applications pour ajuster la puissance de sortie des différents générateurs dans les zones voisines [7] dans cette section on va énoncer les différentes attaques sur ces applications :

### ➤ Flux de puissance optimal (OPF)

L'OPF est l'application chargée de trouver la répartition optimale de la puissance active et réactive d'un réseau électrique donné. Elle optimise les fonctions objectives du système, comme le coût total de production, les pertes du système et le délestage de la charge, tout en satisfaisant les équations de flux de puissance et les limites de fonctionnement des équipements. En raison de l'importance de l'OPF, il s'agit d'une cible attrayante pour un attaquant qui souhaite obtenir des avantages financiers illégaux ou provoquer une panne de système en modifiant les mesures, ce qui génère

une surcharge. Pour lancer une telle attaque, le vecteur d'attaque doit être construit de manière à éviter l'alarme BDD et à satisfaire les contraintes OPF.

#### ➤ SCOPF

Un attaquant peut compromettre les opérations du SCOPF en utilisant un vecteur d'attaque malveillant, le système électrique sera non conforme, ce qui créera une grave défaillance en cascade.

#### ➤ AGC

Le rôle principal de l'AGC est de maintenir l'équilibre entre la demande et la production. Cet équilibre est contrôlé en calculant la fréquence du système, qui doit être maintenue dans des limites acceptables, et en comparant le flux d'énergie de la ligne de liaison avec sa valeur programmée de puissance de la ligne de raccordement. Comme l'AGC joue un rôle très important dans la stabilité du système électrique, il est considéré comme une cible attrayante pour les attaquants qui souhaitent endommager le système électrique. Les mesures de télémetrie sont utilisées directement par l'AGC sans être vérifiées par le BDD dans le schéma existant. Ainsi, les attaquants doivent donc modifier ces mesures pour altérer le signal ACE délivré.

#### **3.3.4. SCED attack**

Le Dispatch économique « *Economic Dispatch* » (ED) est défini comme l'application responsable de la détermination de la quantité d'énergie produite par chaque générateur ( $PG_i$ ). L'objectif de l'ED est de répondre à la demande avec un coût minimal sous la contrainte de l'équilibre de puissance et des limites de production d'énergie ( $PG_i$ ) [7].

SCED est exécuté périodiquement pour définir la répartition de la production avec un coût minimum. Si l'attaquant réussit à modifier certaines mesures, il peut provoquer une surcharge physique de la ligne ou une redistribution de la charge, ce qui impose des dommages physiques et des pertes économiques au système électrique.

#### **3.3.5. DSSE attack**

DSSE fournit une solution de flux de charge en temps réel, qui est utilisée à des fins de planification pour assurer une liste de fonctions liées à la sécurité physique et économique.

La fonction Volt/var control (VVC) vise à maintenir la tension du réseau dans une plage nominale. Elle est responsable de la définition de la position ou de l'état de la commutation

des condensateurs et du transformateur de changement de prise de charge (LTC), qui sont utiles pour stabiliser la charge du réseau.

Les attaques par injection de fausses données dans le système de distribution peuvent corrompre les mesures qui sont utilisées par le VVC afin de déclencher de fausses commandes vers le LTC et les condensateurs [7].

#### **Annexe 4 :L'algorithme Proof-of-Efficiency**

La description de la version 1 de l'algorithme PoEf est la suivante où:

$R_c = \text{tauxactuel}$	$L = \text{limitedelapériodeinitiale}$
$P_c = \text{Productionactuelle}$	$n = \text{nombretotaldepériodes}$
$C_c = \text{consommationdecourant}$	$R_a = \text{tauxmoyen}$
$R_l = \text{derniertaux}$	$P_{per} = \text{productiond'unepériodespécifique}$
$P_l = \text{dernièreProduction}$	$C_{per} =$
$\text{Consommationd'unepériodespécifique}$	
$C_l = \text{dernièrereconsommation}$	$R_{amax} = \text{tauxmoyenmaximum}$
$R_p = \text{taux précédent}$	$R_{cmax} = \text{tauxdecourantmaximal}$
	$per = \text{périodedetempsdelatransaction}$
	$R_{lmax} = \text{derniertauxmaximum.}$

---

#### **Algorithm PoEF version 1**

---

**Input:** Un ensemble de transactions énergétiques

**Requires :** tous les nœuds ont le même ensemble de transactions

**1:** Mettre temporairement à jour les relevés de transactions dans la base de données.

**2:** Pour tous les nœuds présents dans la transaction actuelle, alors

**3:**       $R_c = P_c - C_c$

**4:**       $R_l = P_l - C_l$

**5:**       $R_p = R_c - R_l$

**6:**      Pour  $per = L$  à  $per = n-2$  alors

**7:**               $R_a = \text{average}(P_{per} - C_{per})$

**8:**              fin pour

**9:**               $R_{amax} = \max(R_a)$

**10:** Fin pour

**11:**  $R_{cmax} = \max(R_c)$

**12:**  $R_{lmax} = \max(R_l)$

**output:** l'ensemble des nœuds ayant le rendement maximal, les taux précédents et moyens.