Présenté par :

* AMZIR ADAM
* SENNOUNI FADWA

**Système de Détection des Intrusions utilisant le Machine Learning**

**Licence Fondamentale en Sciences Mathématiques et Informatiques (SMI)**

**Project de Fin d’Etudes**

Sous le thème

Encadré par :

* PR. ALAMI CHENTOUFI JIHANE
* PR. LAASSIRI JALAL   EXAMINATEUR
* PR. ENNEYA NOURDDINE   EXAMINATEUR

**Dédicace**

En premier lieu, nous levons nos pensées vers le ciel, où les étoiles sont les mots silencieux de la grandeur divine, pour exprimer notre gratitude infinie. Comme Dostoïevski peignant l’humanité dans toute sa complexité, nous reconnaissons que ce travail est le fruit d’un jardin cultivé par bien d’autres que nous.

Nous remercions nos familles, ces phares inébranlables dans l’océan tumultueux de notre vie, guidant nos navires de doutes vers des rivages de certitude. Leur amour et leur soutien sont les fondations sur lesquelles nous construisons nos rêves.

Et à nos amis, ces miroirs de notre âme, qui reflètent la vérité avec bienveillance et encouragent chaque pas que nous faisons. Votre amitié est un livre ouvert où chaque page est un chapitre de joie et de soutien.

Ce projet n’est pas seulement le nôtre, mais un morceau de chacun de vous, une symphonie où chaque note est essentielle à l’harmonie de l’ensemble. Merci du fond du cœur.

**Remerciements**

Au terme de cette quête académique, qui fut à la fois un voyage intellectuel et une introspection personnelle, nous tenons à exprimer notre profonde gratitude à notre encadrante, Madame Alami Chentoufi Jihane, enseignante émérite à la faculté des sciences IBN TOFAÏL. Votre guidance, comparable à une lumière dans les ténèbres de l’ignorance, a été notre phare, nous orientant vers les rivages de la connaissance et de la sagesse. Votre patience inébranlable, votre disponibilité constante et vos conseils éclairés ont été les fondations sur lesquelles nous avons construit notre projet.

Nous tenons également à remercier sincèrement tous les professeurs du département d’informatique. Votre passion pour l'enseignement et votre dévouement envers vos étudiants ont grandement contribué à notre formation académique et personnelle. Chacun d'entre vous a apporté une contribution unique à notre parcours universitaire, en partageant vos connaissances approfondies, en offrant des conseils précieux et en créant un environnement propice à l'apprentissage.

Nous exprimons également notre gratitude aux membres du jury, qui nous ont honorés en acceptant de juger ce travail.

Enfin, nous sommes reconnaissants envers tous ceux qui, de près ou de loin, ont participé à cette aventure.

**Table des matières**

[Introduction générale 1](#_Toc169625810)

[Chapitre I : Sécurité des Réseaux et Systèmes de Détection d’Intrusion (IDS) 3](#_Toc169625811)

[1. Introduction 3](#_Toc169625812)

[2. Sécurité des réseaux 3](#_Toc169625813)

[2.1 Evaluation de la sécurité d'un réseau 3](#_Toc169625814)

[3. Les attaques informatiques 4](#_Toc169625815)

[3.1 Les Ransomwares 4](#_Toc169625816)

[3.2 Le Phishing ou Hameçonnage 5](#_Toc169625817)

[3.3 La Compromission de Données 6](#_Toc169625818)

[3.4 Le Sniffing ou Reniflement de Réseau 6](#_Toc169625819)

[3.5 Les attaques par Déni de Service DDoS (Distributed Denial of Service Attack) 7](#_Toc169625820)

[3.6 IP Spoofing 8](#_Toc169625821)

[4. Méthodes de défense 9](#_Toc169625822)

[5. Définition des IDS 10](#_Toc169625823)

[6. Les types des IDS 11](#_Toc169625824)

[6.1 NIDS 11](#_Toc169625825)

[6.2 HIDS 12](#_Toc169625826)

[6.3 PIDS 12](#_Toc169625827)

[6.4 APIDS 13](#_Toc169625828)

[6.5 Hybrid Intrusion Détection System 13](#_Toc169625829)

[7. Composants d'un IDS 13](#_Toc169625830)

[8. Différences entre Système de Détection d’Intrusion (IDS) et Système de Prévention des Intrusions (IPS) 14](#_Toc169625831)

[9. Scénarios de déploiement des IDS 15](#_Toc169625832)

[10. Réponses des IDS contre les attaques 17](#_Toc169625833)

[11. Méthodes de détection d'intrusions existantes 17](#_Toc169625834)

[12. Comment l’analyse de réseau collecte-t-elle les données 18](#_Toc169625835)

[13. Conclusion 19](#_Toc169625836)

[Chapitre II : Machine Learning 20](#_Toc169625837)

[1. Introduction 20](#_Toc169625838)

[2. Types de système d’apprentissage automatique 21](#_Toc169625839)

[2.1 Apprentissage supervisé : 21](#_Toc169625840)

[2.2 Apprentissage non supervisé : 22](#_Toc169625841)

[2.3 Apprentissage semi-supervisé : 24](#_Toc169625842)

[2.4 Machine Learning par renforcement : 24](#_Toc169625843)

[3. Approximation de fonction, classification et régression 25](#_Toc169625844)

[3.1 Approximation de fonction 25](#_Toc169625845)

[3.2 Classification 26](#_Toc169625846)

[3.3 Régression 26](#_Toc169625847)

[4. Quelques Algorithmes utilisés en machine Learning 27](#_Toc169625848)

[4.1 K plus proches voisins (K-NN) : 27](#_Toc169625849)

[4.2 Algorithmes K-Means : 27](#_Toc169625850)

[4.3 Arbre de décision (Decision Tree) : 28](#_Toc169625851)

[4.4 SVM (support à vecteurs machine) : 28](#_Toc169625852)

[5. Comment fonctionne le Machine Learning ? 29](#_Toc169625853)

[6. Conclusion 30](#_Toc169625854)

[Chapitre III : L'algorithme K-NN 31](#_Toc169625855)

[1. Introduction 31](#_Toc169625856)

[2. Qu’est-ce que L’algorithme K-NN ? 31](#_Toc169625857)

[3. Fonctionnement de l’algorithme K-NN 32](#_Toc169625858)

[4. Comment choisir la valeur K ? 36](#_Toc169625859)

[5. Exemple d’utilisation de k-NN 38](#_Toc169625860)

[6. Conclusion 38](#_Toc169625861)

[Chapitre IV : Conception et réalisation 39](#_Toc169625862)

[1. Introduction 39](#_Toc169625863)

[2. Outils et environnement d’exécution 39](#_Toc169625864)

[2.1 L’environnement d’exécution 39](#_Toc169625865)

[2.2 Les outils et langage de programmation utilisés 40](#_Toc169625866)

[2.3 Le jeu de données exploité (DataSet) 40](#_Toc169625867)

[3. Implémentation et développement 41](#_Toc169625868)

[3.1 Prétraitement des données 42](#_Toc169625869)

[3.2 Entrainement du modèle 42](#_Toc169625870)

[4. Résultats et discussion 44](#_Toc169625871)

[5. Implémentation d’une attaque par déni de service (DoS) 45](#_Toc169625872)

[6. Conclusion 46](#_Toc169625873)

[Conclusion 47](#_Toc169625874)

**Liste des figures**

[Figure 1: Malware ‘WannaCry’ 5](#_Toc169622888)

[Figure 2: Phishing 6](#_Toc169622889)

[Figure 3: Sniffing 7](#_Toc169622890)

[Figure 4: DDoS Attaque 8](#_Toc169622891)

[Figure 5: IP Spoofing 9](#_Toc169622892)

[Figure 6: Système de détection d'intrusion 11](#_Toc169622893)

[Figure 7: Système de détection d'intrusion réseau (NIDS) 11](#_Toc169622894)

[Figure 8: Système de détection d'intrusion sur l'hôte (HIDS) 12](#_Toc169622895)

[Figure 9: Système de détection d'intrusion hybride 13](#_Toc169622896)

[Figure 10: La différence entre IDS et IPS 15](#_Toc169622897)

[Figure 11: Exemple d'un IDS avant le pare-feu 15](#_Toc169622898)

[Figure 12: un IDS Dans la zone démilitarisée (DMZ) 16](#_Toc169622899)

[Figure 13: un IDS à l’intérieur du réseau d’une entreprise 16](#_Toc169622900)

[Figure 14: Les catégories du Machine Learning 21](#_Toc169622901)

[Figure 15: L'apprentissage supervisé 22](#_Toc169622902)

[Figure 16: L'apprentissage non supervisé 23](#_Toc169622903)

[Figure 17: L'apprentissage semi-supervisé 24](#_Toc169622904)

[Figure 18: L'apprentissage par renforcement 25](#_Toc169622905)

[Figure 19: Classification 26](#_Toc169622906)

[Figure 20: Régression 27](#_Toc169622907)

[Figure 21: L'algorithme kNN 32](#_Toc169622908)

[Figure 22: Un nouveau point de données à catégorisé à l'aide de l’algorithme kNN 34](#_Toc169622909)

[Figure 23: La distance Euclidienne entre deux voisins 35](#_Toc169622910)

[Figure 24: Obtention des plus proches voisins et catégorisation du nouveau point 35](#_Toc169622911)

[Figure 25:presentation du overfitting, underfitting et good fit 36](#_Toc169622912)

[Figure 26:le taux d'erreur par rapport au paramètre k 37](#_Toc169622913)

[Figure 27:Les composantes du Dataset 41](#_Toc169622914)

[Figure 28:Division des Dataset en données d'entrainement et données de test 42](#_Toc169622915)

[Figure 29: Choix du paramètre k 42](#_Toc169622916)

[Figure 30: Les scores de précision et rappel du modèle 44](#_Toc169622917)

[Figure 31: Le pourcentage du F1-Score 45](#_Toc169622918)

[Figure 32: Screenshot de DoS code 45](#_Toc169622919)

[Figure 33: Score de prédiction d’une attaque de type DoS 46](#_Toc169622920)

**Liste des tableaux**

[Tableau 1: Score d’entrainement du modèle K-NN 44](#_Toc169553420)

[Tableau 2: Score de la validation du modèle K-NN 44](#_Toc169553421)

**Liste des abréviations**

|  |  |
| --- | --- |
| Abréviation | Signification |
| IDS | Système de détection des intrusions |
| IoT | Internet of Things |
| DoS | Denial of Service |
| DDoS | Distributed Denial of Service |
| NIDS | Système de détection des intrusions réseau |
| HIDS | Système de détection des intrusions sur l’hôte |
| PIDS | Système de détection des intrusions sur protocole |
| APIDS | Système de détection des intrusions hybride |
| FTP | File Transfer Protocol |
| DNS | Domain Name System |
| HTTP | HyperText Transfer Protocol |
| HTTPS | HyperText Transfer Protocol |
| AI | Artificial Intelligence |
| K-NN | K-Nearest Neighbors |
| SVM | Support Vector Machine |
| OCR | Optical Character Recognizer |

Introduction générale

Le développement fulgurant d'Internet a révolutionné la communication, le travail et les interactions avec le monde qui nous entoure, offrant des opportunités et des avantages infinis et sans précédent.

Cependant, en parallèle à cette évolution, les vulnérabilités et les failles des systèmes informatiques sont constamment découvertes, les exposant à de grands risques d’attaques.

Des menaces en formes de logiciels malveillants, des attaques de phishing... ne cessent de se développer défiant les moyens traditionnels de défense.

Pour cette raison les systèmes de détection d'intrusions (IDS) ont été développés. L'un des outils les plus importants pour prévenir les intrusions, surveiller en continu le trafic réseau, détecter les comportements suspects et signaler les éventuelles intrusions, offrant une meilleure sécurité aux systèmes informatiques.

Mais avec la nature évolutive des réseaux, les IDS sont toujours défiés pour la maintenance de leur efficacité.

Face à cette réalité, la question qui se pose est la suivante :

Comment concevoir des IDS qui peuvent évoluer pour répondre aux nouvelles formes d’attaques, garantir la sécurité aux réseaux informatiques et être à jour à un environnement en évolution continue ?

Ce projet contribuera à renforcer la sécurité des réseaux informatiques et à protéger les données sensibles contre les cyberattaques d’une manière évolutive. En évaluant les technologies et les méthodes actuelles.

Ainsi la conception d’un IDS capable de garantir une meilleure protection en explorant les techniques avancées tel que l'apprentissage automatique. Ce dernier va permettre d'être à jour en termes de précision de la détection des menaces et mêmes celles éventuelles, ce qui va renforcer les barrières des systèmes informatiques.

Ce projet est composé de quatre chapitres organisés de la manière suivante :

-Le premier chapitre met en lumière l’importance de la sécurité informatique et les méthodes de défense contre les différentes attaques. Ainsi, il présentera une architecture globale des IDS, ses types et son mode de fonctionnement.

-Le deuxième chapitre est consacré à présenter le Machine Learning, ses types et son fonctionnement ainsi que les algorithmes connus.

-Le troisième chapitre vise à indiquer l’algorithme utilisé dans ce projet et par la suite bien expliquer son fonctionnent.

-Et enfin, le quatrième chapitre se concentrera sur la conception et l’implémentation du modèle de l’IDS, ainsi que sur la partie résultat et discussion.

1. Sécurité des Réseaux et Systèmes de Détection d’Intrusion (IDS)

# Introduction

Cette dernière décennie a été marqué par la grande migration de toutes les activités, soit personnelles ou professionnelles vers un environnement numérique. Dès lors, chaque personne, entreprise, ou organisation se retrouve exposés aux nombreux risques d’attaques, menaçant ses critères de sécurité. En visant les informations sensibles telles que les données personnelles des utilisateurs, les données de l’entreprise, les détails de paiement, et les documents personnels…pour des buts malveillants.

Les impacts de ces attaques peuvent inclure des interruptions de services, une atteinte à la réputation, des vols, ainsi des grandes pertes financières. C’est pour cela que la sécurité informatique est cruciale, afin de protéger ces données contre les accès non autorisés et d’éviter les pertes engendrées.

# Sécurité des réseaux

La sécurisation d’un réseau informatique implique l’application de mesures techniques, organisationnelles, légales et humaines pour maintenir, restaurer et assurer la protection du réseau. Habituellement, la sécurité d’un réseau inclut aussi celle du système informatique qui le supporte. Mais, comment la sécurité d’un réseau est évaluée ?

## Evaluation de la sécurité d'un réseau

L’évaluation de la sécurité d’un réseau se fait selon plusieurs critères de sécurité essentiels. Trois critères principaux sont souvent mis en avant pour la sécurité de l’information :

* **La confidentialité :** Les messages peuvent être interceptés par des attaquants lors de leur transmission, compromettant ainsi la confidentialité et l’intégrité du contenu. Il est donc crucial d’assurer la transmission sécurisée des messages, particulièrement dans l’Internet des Objets (Internet of Things [IoT]).
* **Intégrité :** Il est impératif que les messages ne soient pas modifiés durant leur acheminement et qu’ils arrivent intacts au destinataire, comme ils ont été envoyés par l’émetteur. L’intégrité assure que le contenu n’a pas été altéré par des tiers non autorisés.
* **Disponibilité :** Les données et ressources doivent être accessibles lorsque nécessaires. Des attaques malveillantes telles que les dénis de service (DoS), les inondations de réseau, les trous noirs et le brouillage peuvent compromettre la disponibilité.

Quelles sont les attaques qui peuvent menacer la sécurité des réseaux ?

# Les attaques informatiques

Une « attaque » est l'exploitation d'une faille (ou vulnérabilité) d'un système informatique (système d'exploitation, logiciel ou bien même de l'utilisateur) pour nuire à ce dernier.

Tous les systèmes informatiques présentent des vulnérabilités, indépendamment de leur niveau de sécurité. Une vulnérabilité est une faiblesse ou un défaut qui peut être exploité par des menaces pour compromettre le système. De nouvelles vulnérabilités sont régulièrement découvertes, et leur exploitation peut avoir des conséquences graves.

Alors, quelles sont les types d’attaques qu’un système informatique peut contourner ?

## Les Ransomwares

Ces malwares sont actuellement parmi les plus répandus. Ils représentent une forme de chantage numérique : les données d’une entreprise sont cryptées par un cybercriminel. Le processus d’infection est le suivant : dissimulé dans une pièce jointe, un logiciel malveillant crypte les fichiers de l’ordinateur initialement infecté. En quelques secondes, tous les fichiers partagés au sein de l’entreprise sont également verrouillés. Ils deviennent inaccessibles, à moins de payer une rançon dont le montant augmente progressivement pour intensifier la pression. La figure 1 illustre l’interface du ransomware connu sous le nom de « WannaCry ».

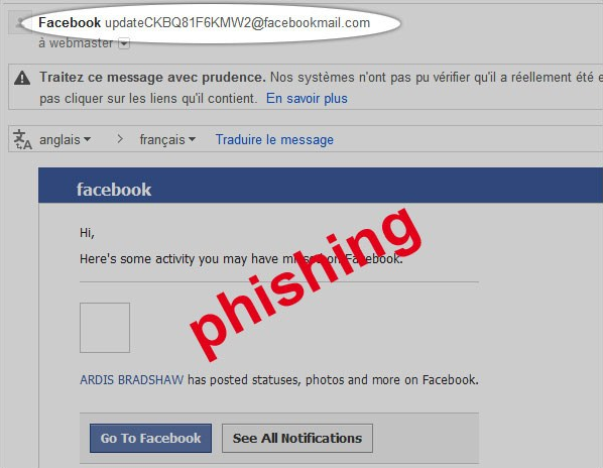


**Figure 1:** Malware ‘WannaCry’

## Le Phishing ou Hameçonnage

Cette menace persistante est bien connue sous le nom de phishing ou hameçonnage. Ces courriels frauduleux imitent souvent l’identité d’organisations privées ou d’entités gouvernementales. Les cybercriminels font preuve d’une grande créativité et d’un réel savoir-faire pour élaborer des messages trompeurs arborant des logos officiels, rendant le piège d’autant plus convaincant. Les conséquences peuvent être lourdes, allant de la récupération de mots de passe à l’installation de logiciels malveillants avec le “consentement” de l’utilisateur piégé. Selon une étude récente de Google, le phishing est considéré comme l’une des techniques les plus lucratives pour les cybercriminels, avec un taux de réussite estimé entre **12 et 25%.**

La figure 2, montre un email envoyer par Facebook pour se connecter, mais c’est un email faux.



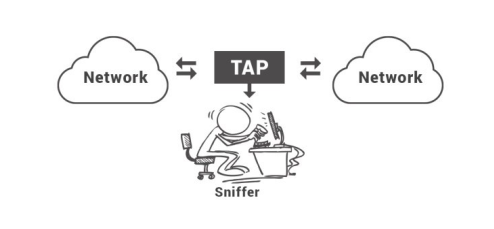
**Figure 2:** Phishing

## La Compromission de Données

La compromission de données peut survenir suite à une **intrusion malveillante** dans le réseau informatique par un hacker. Cependant, elle peut également être le fait d’un **employé de l’entreprise**. Historiquement, les employés représentent un risque significatif en raison de leur accès privilégié, de leur connaissance du système et du temps disponible pour exploiter ces avantages. Ainsi, la fuite de données peut être délibérée : un employé pourrait copier des informations sensibles sur une clé USB ou les transférer à sa messagerie personnelle dans le but de les vendre à des entités concurrentes. Par ailleurs, la fuite de données peut aussi résulter de la **négligence** des collaborateurs, comme lorsqu’ils visitent des sites web non sécurisés, laissent leur session active sans surveillance, ou utilisent des mots de passe insuffisamment robustes.

## Le Sniffing ou Reniflement de Réseau

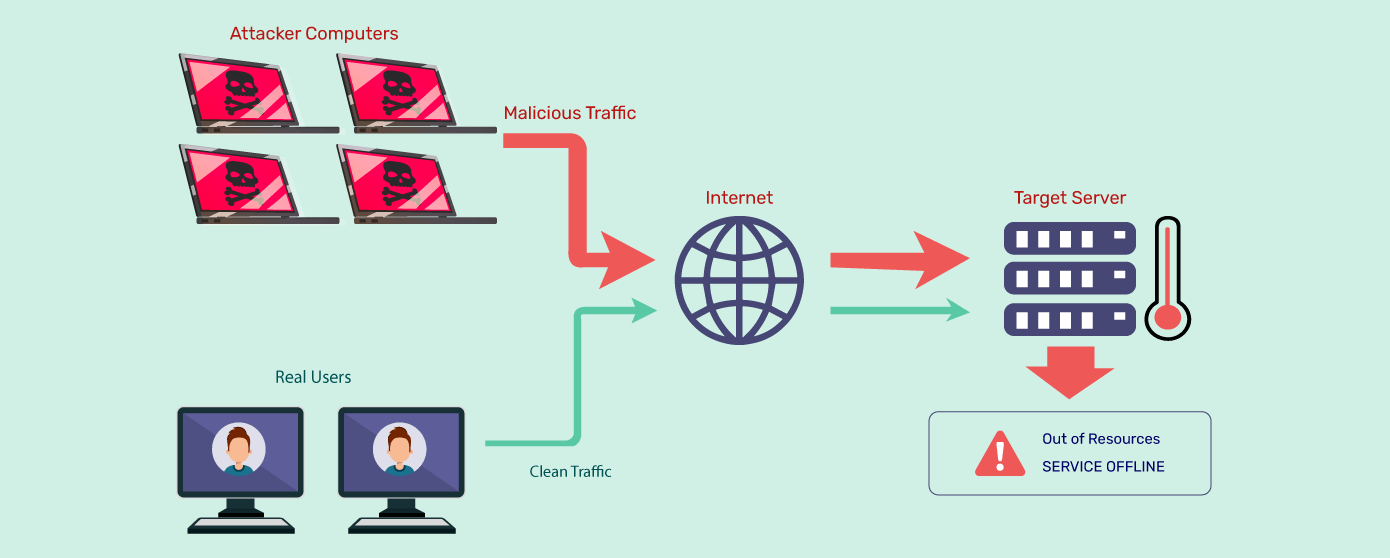
Un **sniffer** est un outil de surveillance réseau utilisé pour capturer les paquets de données en transit. Il permet à un pirate informatique d’inspecter le contenu des paquets qui ne lui sont pas destinés. En tant qu’outil de reniflement, il peut saisir toutes sortes d’informations circulant sur le réseau, révélant ainsi l’identité des utilisateurs ainsi que leurs mots de passe, particulièrement si ces données transitent via des protocoles insuffisamment sécurisés tels que le FTP (File Transfer Protocol), le DNS (Domain Name System) ou le HTTP (HyperText Transfer Protocol). Ainsi, lorsque les données ne sont pas chiffrées et qu’elles traversent l’interface réseau de l’ordinateur sur lequel le sniffer est exécuté, elles peuvent être interceptées sans difficulté par cet appareil. La figure 3, montre un criminel, que ce met entre 2 réseaux pour capturer le trafique de ces dernier pour des objectifs personnels.



**Figure 3:** Sniffing

## Les attaques par Déni de Service DDoS (Distributed Denial of Service Attack)

Une attaque DDoS consiste à mobiliser un réseau de machines compromises pour qu’elles envoient simultanément des requêtes de connexion ou des flux de données vers une cible spécifique. Submergée par un volume massif de requêtes ou par un déluge de données mesuré en gigabits par seconde, la cible devient rapidement surchargée et inopérante, rendant le service inaccessible aux utilisateurs légitimes. Les cybercriminels peuvent alors exiger une rançon de l’opérateur du site pour cesser l’attaque ou profiter de la situation pour franchir des défenses de sécurité surchargées et incapables de répondre efficacement. La figure 4, montre quatre stations qui émettent des paquets ICMP pour saturer le serveur.

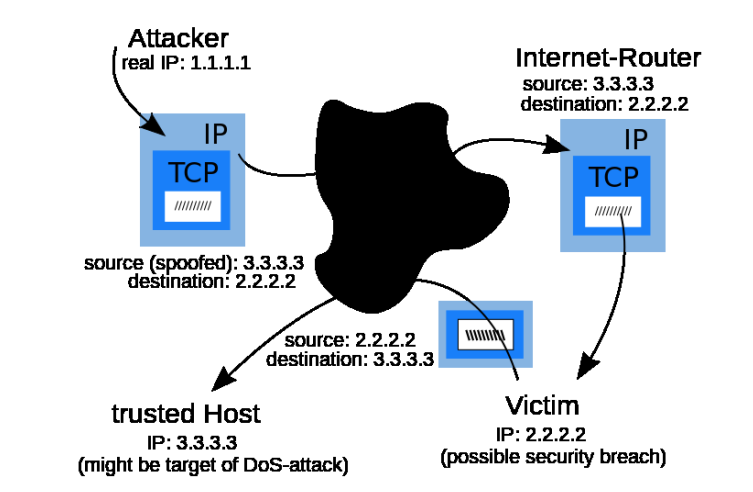


**Figure 4:** DDoS Attaque

## IP Spoofing

L’usurpation d’adresse IP est une technique utilisée pour masquer l’identité réelle d’un appareil en se faisant passer pour une machine de confiance au sein du réseau ciblé. Cette méthode implique la génération de paquets IP avec une adresse source modifiée. Toutefois, des mécanismes supplémentaires sont nécessaires pour que les réponses aux paquets falsifiés ne soient pas renvoyées à l’émetteur réel, mais plutôt à la machine dont l’adresse a été usurpée.

Cette approche est particulièrement utile pour contourner les systèmes d’authentification qui se basent sur les adresses IP. Des outils comme “hping2” permettent de modifier ou de créer des paquets IP personnalisés, offrant ainsi la possibilité de spécifier une adresse IP différente de la sienne et de se faire passer pour un autre appareil sur le réseau. La figure 5, montre comment un criminel change ses informations pour accéder à un internet-router pour s’infiltrer, modifier ou voler les données d’un utilisateur.



**Figure 5:** IP Spoofing

Après avoir établi les critères fondamentaux de la sécurité réseau et lister les types d’attaques, il est essentiel de mettre en œuvre des dispositifs de protection efficaces.

# Méthodes de défense

Les stratégies de défense contre les cyberattaques varient en efficacité, certaines prévenant activement les incidents tandis que d’autres ne font que détecter les attaques après qu’elles se soient produites.

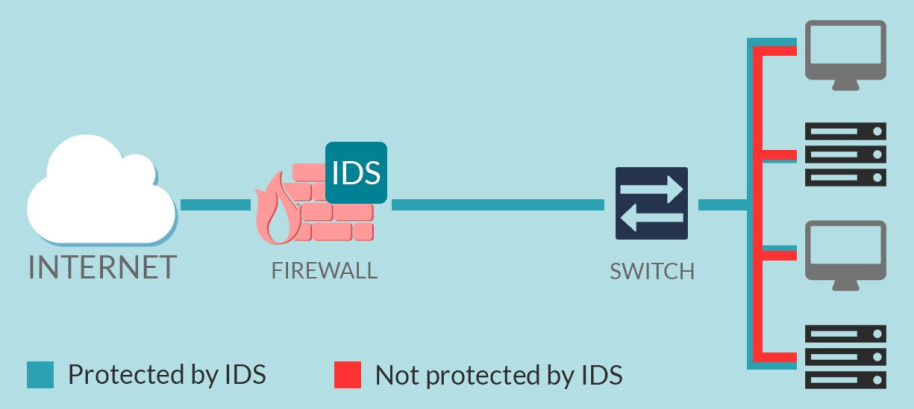
* **Le Chiffrement/Cryptage :** Le chiffrement transforme les données en un format illisible sans la clé appropriée. Ce processus cryptographique assure que seul quelqu’un possédant la clé de déchiffrement peut comprendre le contenu d’un document. Le chiffrement offre une protection contre l’interception et la modification des données. Il contribue également à l’intégrité des données, car il est difficile de modifier de manière cohérente des informations qui n’ont pas de sens à la lecture. Bien que le chiffrement soit un outil crucial en sécurité informatique, il ne résout pas tous les problèmes de sécurité.
* **Contrôle Logiciel/Software :** Pour contrer les attaques externes, les logiciels doivent être sécurisés dès leur conception. Le contrôle logiciel est intégré pendant la phase de développement et mis en œuvre pendant l’exploitation ou comme une fonction restrictive du logiciel.
* **Contrôle Matériel/Hardware :** De nombreux dispositifs contribuent à la sécurité informatique, tels que les cartes d’implémentation de chiffrement, les contrôleurs d’accès aux disques et les dispositifs de vérification d’identité.
* **Aspect Juridique :** La législation relative aux cybercrimes reste vague et évolue lentement. La communauté informatique n’a pas encore établi de normes universelles concernant l’éthique, bien que certaines organisations encouragent le développement de telles normes.
* **Les pares-feux (Firewalls)** représentent une de ces mesures cruciales. Ils agissent comme des gardiens qui contrôlent le trafic entrant et sortant selon des règles établies, empêchant ainsi les accès non autorisés et les attaques connues. Néanmoins, ils présentent des vulnérabilités :
* ***Contournement des Règles :***Des attaquants chevronnés peuvent esquiver les règles des pares-feux en recourant à des techniques de masquage, de fragmentation de paquets ou en exploitant des protocoles atypiques.
* ***Attaques Zéro-Day :***Les pares-feux ne sont pas capables de bloquer les attaques non identifiées ou les vulnérabilités encore inconnues.
* ***Manque de Visibilité :*** Ils ne parviennent pas à détecter les attaques qui tirent parti de failles logicielles ou de configurations erronées sur les appareils internes.

C’est dans ce contexte que les systèmes de détection d’intrusion (IDS) deviennent indispensables.

# Définition des IDS

Un IDS est une sonde placée sur un réseau ou un système, qui détecte et répond à des activités douteuses ou anormales en temps réel ou hors ligne. Il existe plusieurs types d’IDS, tels que les NIDS (Network Intrusion Détection System), qui se basent sur des analyses de réseau, les HIDS (Host Intrusion Détection System), qui assurent la sécurité d’un hôte, et les IDS hybrides, qui combinent à la fois les HIDS et les NIDS.

La figure 6, montre un exemple de déploiement d’un IDS.

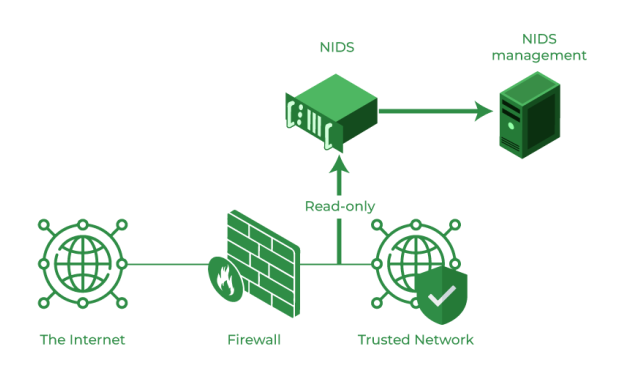


**Figure 6:** Système de détection d'intrusion

# Les types des IDS

## NIDS

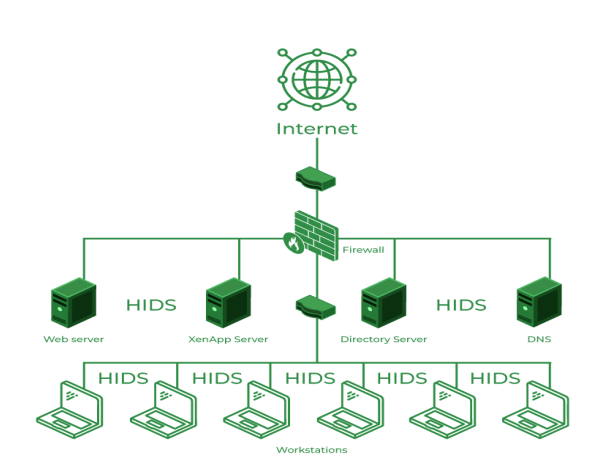
Système de détection d'intrusion réseau (NIDS) : Ils sont configurés à un point planifié du réseau pour examiner le trafic de tous les périphériques du réseau. Il effectue une observation du trafic passant sur l'ensemble du sous-réseau et fait correspondre le trafic qui est transmis sur les sous-réseaux à la collection d'attaques connues. Une fois qu'une attaque est identifiée ou qu'un comportement anormal est observé, l'alerte peut être envoyée à l'administrateur. Un exemple de NIDS est de l'installer sur le sous-réseau où se trouvent les pares-feux afin de voir si quelqu’un essaie de casser le pare-feu ; ce que le montre la figure 7.



**Figure 7:** Système de détection d'intrusion réseau (NIDS)

## HIDS

Système de détection des intrusions sur l'hôte (HIDS) : Ils s'exécutent sur des hôtes ou des périphériques indépendants sur le réseau. Un HIDS surveille uniquement les paquets entrants et sortants de l'appareil et alerte l'administrateur si une activité suspecte ou malveillante est détectée. Il prend un instantané des fichiers système existants et le compare avec l'instantané précédent. Si les fichiers du système d'analyse ont été modifiés ou supprimés, une alerte est envoyée à l'administrateur pour enquête. Un exemple d'utilisation de HIDS peut être vu sur des machines critiques, qui ne devraient pas changer leur disposition ; ce que le montre la figure 8 :



**Figure 8:** Système de détection d'intrusion sur l'hôte (HIDS)

## PIDS

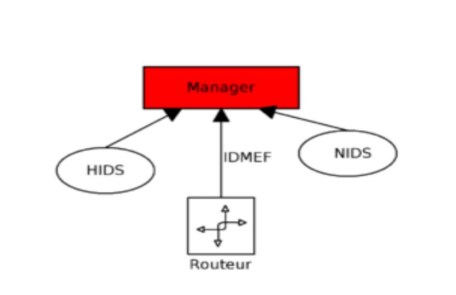
Système de détection d'intrusion basé sur protocole (PIDS) : Il comprend un système ou un agent qui résiderait systématiquement à l'avant d'un serveur, contrôlant et interprétant le protocole entre un utilisateur/appareil et le serveur. Il essaie de sécuriser le serveur Web en surveillant régulièrement le flux de protocole HTTPS et en acceptant le protocole HTTP associé. Comme HTTPS n'est pas crypté et avant d'entrer instantanément dans sa couche de présentation Web, ce système devrait résider dans cette interface, entre pour utiliser le HTTPS.

## APIDS

Système de détection d'intrusion basé sur le protocole d'application (APIDS) : C’est un système ou un agent qui réside généralement dans un groupe de serveurs. Il identifie les intrusions en surveillant et en interprétant la communication sur des protocoles spécifiques à l'application. Par exemple, cela surveillerait explicitement le protocole SQL vers le middleware lors de ses transactions avec la base de données sur le serveur Web.

## Hybrid Intrusion Détection System

Système de détection d'intrusion hybride : Il est constitué par la combinaison de deux ou plusieurs approches du système de détection d'intrusion. Dans le système de détection d'intrusion hybride, l'agent hôte ou les données du système sont combinés avec des informations de réseau pour développer une vue complète du système de réseau. Le système de détection d'intrusion hybride est plus efficace par rapport à l'autre système de détection d'intrusion. Prélude est un exemple d'IDS hybride. La figure 9 montre un exemple de déploiement de IDS hybride.



**Figure 9:** Système de détection d'intrusion hybride

# Composants d'un IDS

Les systèmes de détection d’intrusion (IDS) se composent de trois éléments clés essentiels à leur fonctionnement :

* **Composant de Surveillance :** Ce module est chargé de l’observation active des systèmes d’information pour détecter toute activité anormale ou suspecte.
* **Composant de Monitoring :** Il a pour rôle de compiler et d’analyser les informations recueillies par le composant de surveillance afin de les rendre exploitables.
* **Composant de Décision :** Basé sur les données analysées, ce composant prend les décisions nécessaires pour répondre aux incidents détectés, que ce soit par des alertes ou des actions correctives.

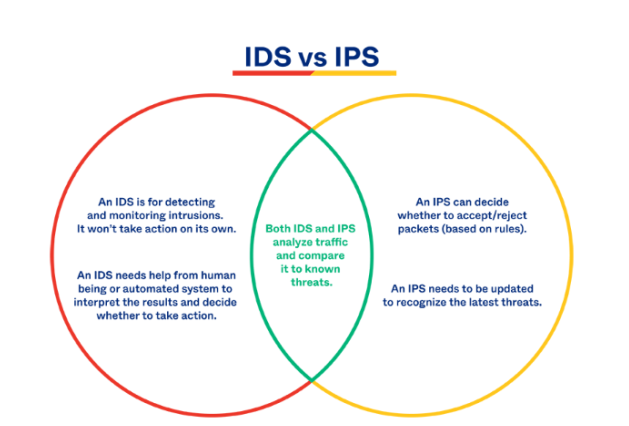
# Différences entre Système de Détection d’Intrusion (IDS) et Système de Prévention des Intrusions (IPS)

À l’inverse d’IDS, un **Système de Prévention des Intrusions (IPS)** va au-delà de la simple détection. Non seulement il détecte les intrusions, mais il prend également des mesures pour bloquer ou prévenir les menaces identifiées avant qu’elles n’affectent le réseau.

Voici les principales différences entre les deux :

* **Détection vs Prévention :** L’IDS détecte et alerte, tandis que l’IPS empêche activement les attaques.
* **Positionnement dans le réseau :**Les IDS sont généralement placés en mode écoute sur le réseau, alors que les IPS sont positionnés de manière à pouvoir bloquer le trafic, agissant comme un contrôle d’accès.
* **Réponse aux incidents :** L’IDS génère des alertes lorsqu’une activité suspecte est détectée, mais ne prend pas de mesures correctives. L’IPS, quant à lui, prend des mesures immédiates pour arrêter les attaques en cours.
* **Impact sur le trafic réseau :**Les IDS ont un impact minimal sur le trafic réseau car ils ne modifient pas le flux de données. Les IPS peuvent introduire des latences en raison de l’analyse et de l’action en temps réel sur le trafic.

La figure 10, répertorie les principales différences et similitudes entre IDS et IPS :

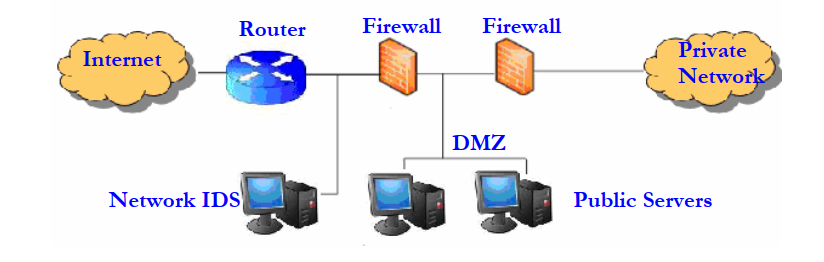


**Figure 10:** La différence entre IDS et IPS

# Scénarios de déploiement des IDS

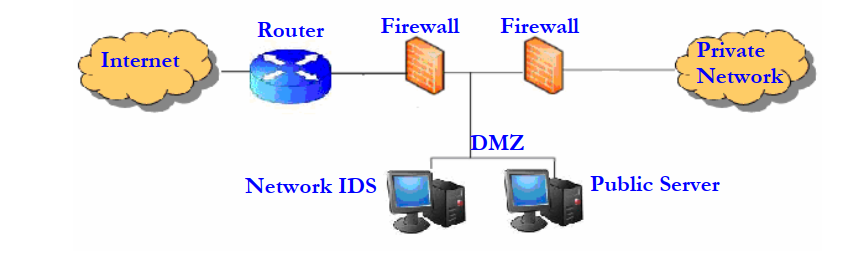
Il existe trois emplacements stratégiques où les NIDS peuvent être installés sur le réseau pour une surveillance efficace, comme illustré dans les diagrammes ci-dessous :

* **Avant le Pare-feu de la Passerelle :** En plaçant un NIDS avant le pare-feu, il est possible de surveiller l’ensemble du trafic, y compris les attaques qui pourraient être bloquées ultérieurement par le pare-feu (comme le montre la figure 11). Cette position exige un traitement rapide du trafic et une configuration minutieuse pour limiter les fausses alertes.



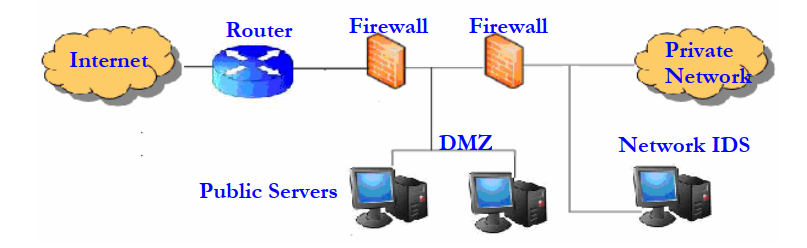
**Figure 11:** Exemple d'un IDS avant le pare-feu

* **Dans la DMZ (Zone Démilitarisée) :** Un NIDS situé dans la DMZ analyse le trafic qui a déjà été filtré en partie par le pare-feu (comme le montre la figure 12). Cela allège la charge de travail du NIDS mais réduit aussi sa capacité à observer le trafic global.



**Figure 12:** un IDS Dans la zone démilitarisée (DMZ)

* **À l’Intérieur du Réseau d’Entreprise Privé :** Un NIDS déployé à l’intérieur du réseau d’entreprise se concentre sur la surveillance des menaces internes et celles qui ont traversé le pare-feu (comme le montre la figure 13). Avec moins d’attaques potentielles à cet endroit, le NIDS a tendance de générer moins de fausses alertes et sa visibilité est confinée au réseau interne.



**Figure 13:** un IDS à l’intérieur du réseau d’une entreprise

Il est recommandé d’installer les NIDS sur des systèmes séparés des pare-feu pour éviter que les attaques ne surchargent les ressources du système et n’impactent la performance du pare-feu.

# Réponses des IDS contre les attaques

Les Systèmes de Détection d’Intrusion (IDS) disposent de plusieurs stratégies de réponse lorsqu’ils détectent des activités suspectes ou des attaques. Voici une explication détaillée des types de réponses :

* **Réponse Active :** Bien qu’un IDS ne puisse pas directement bloquer les attaques, il peut initier des actions qui contribuent à leur arrêt. Par exemple, il peut envoyer des paquets TCP reset pour interrompre les connexions suspectes ou reconfigurer le routeur/pare-feu pour bloquer les sources malveillantes. Dans des situations critiques, l’IDS peut même suspendre tout le trafic réseau pour protéger les actifs de l’entreprise.
* **Réponse Passive :** Les réponses passives consistent à informer l’administrateur de l’IDS sur les incidents détectés sans prendre de mesures directes. Ces systèmes se basent sur l’envoi d’alertes et de notifications, qui peuvent être transmises par e-mail, téléphone ou messages SNMP, laissant à l’administrateur le soin de décider des actions à entreprendre.
* **Réponse Mixte :** Les réponses mixtes allient les approches actives et passives. Selon la nature et la gravité de l’attaque, l’IDS peut automatiquement prendre certaines mesures tout en alertant l’administrateur pour une intervention plus poussée si nécessaire.

Chaque type de réponse est conçu pour s’adapter aux politiques de sécurité spécifiques d’une organisation et à la nature des menaces auxquelles elle est confrontée.

# Méthodes de détection d'intrusions existantes

* **Détection par Signatures :**
* ***Principe :*** Comparaison avec une base de données de signatures d’attaques connues.
* ***Avantages :*** Précision pour les attaques connues, simplicité d’implémentation.
* ***Inconvénients* :** Limité aux attaques connues, nécessité de mises à jour fréquentes, risque de faux positifs.
* **Détection par Anomalies :**
* ***Principe :*** Identification de comportements anormaux.
* ***Avantages :*** Capacité à détecter des attaques inconnues, pas de base de signatures requise.
* ***Inconvénients :*** Configuration et gestion complexes, risque de faux positifs.
* **Détection Basée sur le Réseau :**
* ***Principe :*** Surveillance du trafic réseau pour identifier des activités suspectes.
* ***Avantages :***Protection étendue du réseau, utile pour détecter des attaques actives.
* ***Inconvénients :*** Sensibilité aux variations normales du trafic, inefficace pour les attaques hors réseau.
* **Détection Basée sur l’Hôte :**
* ***Principe :*** Surveillance des activités systèmes pour détecter des intrusions.
* ***Avantages :*** Protection détaillée, détection des attaques exploitant des vulnérabilités systèmes.
* ***Inconvénients :***Nécessité d’agents sur chaque hôte, impact potentiel sur les performances.
* **Détection Basée sur l’Apprentissage Automatique :**
* ***Principe :*** Utilisation de l’apprentissage automatique pour reconnaître des modèles d’attaque.
* ***Avantages :*** Détection proactive et précise, adaptabilité aux nouvelles menaces.
* ***Inconvénients :*** Besoin de grandes quantités de données d’entraînement, complexité de mise en œuvre et de gestion.

# Comment l’analyse de réseau collecte-t-elle les données

L’analyse de réseau est un processus qui implique la collecte de données à partir de différentes sources pour surveiller, maintenir et optimiser les réseaux. Voici comment elle fonctionne :

* **Sources de Données :**
* ***Serveurs :*** Les serveurs comme DHCP, Active Directory, RADIUS, DNS et Syslog fournissent des informations sur les configurations, les connexions des utilisateurs, les requêtes de noms de domaine et les événements système.
* ***Trafic Réseau :*** Des protocoles comme NetFlow, Traceroute et SNMP capturent des informations sur le flux de trafic, les chemins de réseau et l’état des appareils réseau.
* **Outils de Capture et d’Analyse :**
* ***Tcpdump :*** Un outil en ligne de commande qui permet de capturer des paquets réseau transitant par une interface spécifique.
* ***Wireshark :*** Une interface graphique qui offre des capacités avancées pour capturer et analyser les paquets réseau en détail.

Ces outils et sources de données permettent aux administrateurs réseau de détecter des problèmes, d’analyser la performance du réseau et de sécuriser le réseau contre des activités malveillantes.

# Conclusion

Malgré leurs limites, les IDS jouent un rôle essentiel dans la stratégie globale de sécurité des réseaux. Lorsqu’ils sont intégrés avec d’autres dispositifs de sécurité, comme les pares-feux, les systèmes de prévention des intrusions (IPS), et les protocoles de chiffrement, ils contribuent à créer une défense en profondeur. Une approche proactive, qui inclut la formation des employés, la mise à jour régulière des systèmes et la surveillance continue, permet aux organisations de rester un pas devant les menaces et de minimiser les risques de sécurité.

1. Machine Learning

# Introduction

Machine Learning ou apprentissage automatique est un domaine scientifique et plus particulièrement une sous-catégorie de l’intelligence artificielle **(AI)**.

Le machine Learning est une technique de programmation informatique qui utilise des probabilités statistiques pour donner aux ordinateurs la capacité d'apprendre par eux-mêmes sans programmation explicite. L’objectif de base du machine Learning est ***"d’apprendre à apprendre"*** aux ordinateurs -et par la suite, à agir et réagir - comme le font les humains, en améliorant leur mode d'apprentissage et leurs connaissances de façon autonome sur la durée. L'objectif ultime serait que les ordinateurs agissent et réagissent sans être explicitement programmés pour ces actions et réactions.

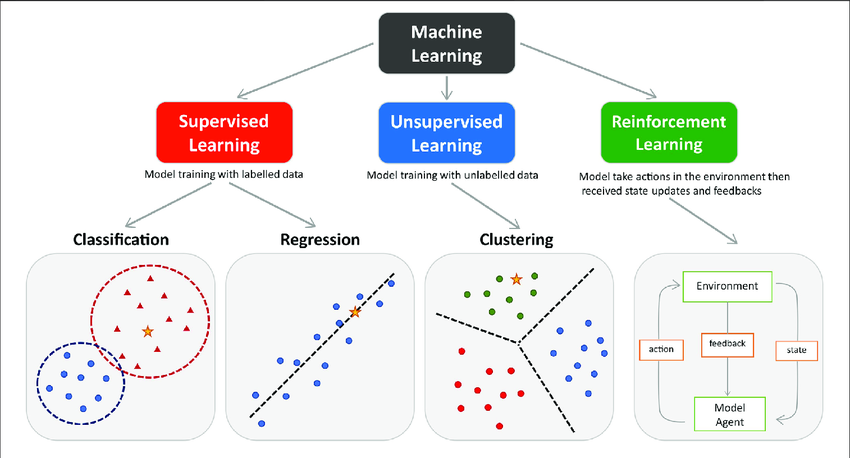
Il consiste à laisser des algorithmes découvrir des « patterns », à savoir des motifs récurrents, dans les ensembles de données. Ces données peuvent être des chiffres, des mots, des images, des statistiques…Tout ce qui peut être stocké numériquement peut servir de données pour le Machine Learning. En décelant les patterns dans ces données, les algorithmes apprennent et améliorent leurs performances dans l’exécution d’une tâche spécifique.

Pour résumer, les algorithmes du Machine Learning apprennent de manière autonome à effectuer une tâche ou à réaliser des prédictions à partir de données et améliorent leurs performances au fil du temps. Une fois entraîné, l’algorithme pourra retrouver les patterns dans de nouvelles données.

Le machine Learning utilise des programmes de développement qui s'ajustent chaque fois qu'ils sont exposés à différents types de données en entrée.

# Types de système d’apprentissage automatique

Selon la nature du problème traité, le type ainsi le volume de données on distingue trois grandes catégories d’apprentissage automatique, qui sont présentes sur la figure 14 :



**Figure 14:** Les catégories du Machine Learning

## Apprentissage supervisé :

***Supervised Learning***, également connu sous le nom d'apprentissage automatique supervisé, est défini par son utilisation d'ensembles de données étiquetés pour entraîner des algorithmes afin de classer les données ou de prédire les résultats avec précision.

Les opérateurs présentent à l'ordinateur des exemples d'entrées et les sorties souhaitées, et l'ordinateur recherche des solutions pour obtenir ces sorties en fonction de ces entrées. Le but est que l'ordinateur apprenne la règle générale qui mappe les entrées et les sorties. Un schéma présente le déroulement de cet apprentissage sur la figure 15.

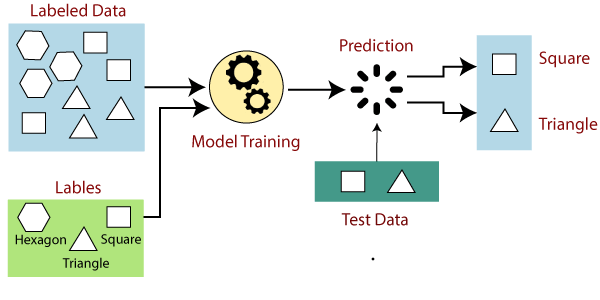
L’apprentissage supervisé peut être utilisé pour faire des prédictions sur des données indisponibles ou futures (on parle alors de ***"modélisation prédictive"***). L'algorithme essaie de développer une fonction qui prédit avec précision la sortie à partir des variables d'entrée – par exemple, prédire la valeur d'un bien immobilier (sortie) à partir d'entrées telles que le nombre de pièces, l'année de construction, la surface du terrain, l'emplacement, etc.

Cette méthode nécessite moins de données d’entraînement que les autres, et facilite le processus d’entraînement puisque les résultats du modèle peuvent être comparés avec les données déjà étiquetées. Cependant, l’étiquetage des données peut se révéler onéreux. Un modèle peut aussi être biaisé à cause des données d’entraînement, ce qui impactera ses performances par la suite lors du traitement de nouvelles données.

L’apprentissage supervisé peut se subdiviser en deux types :

* ***Classification :*** la variable de sortie est une catégorie.
* ***Régression :*** la variable de sortie est une valeur spécifique.

Les principaux algorithmes du Machine Learning supervisé sont les suivants : forêts aléatoires, arbres de décision, algorithme K-NN (k-Nearest Neighbors), régression linéaire, algorithme de Naïve Bayes, machine à vecteurs de support (SVM)...



**Figure 15:** L'apprentissage supervisé

## Apprentissage non supervisé :

Dans le cadre du l’apprentissage non supervisé, l'algorithme détermine lui-même la structure de l'entrée (aucune étiquette n'est appliquée à l'algorithme). Cette approche peut être un but en soi (qui permet de découvrir des structures enfouies dans les données) ou un moyen d'atteindre un certain but. Cette approche est également appelée « apprentissage des caractéristiques » (***Feature Learning***). Ce qui est présenté sur la figure 16.

La machine se contente d’explorer les données à la recherche d’éventuelles patterns. Elle ingère de vastes quantités de données, et utilise des algorithmes pour en extraire des caractéristiques pertinentes requises pour étiqueter, trier et classifier les données en temps réel sans intervention humaine.

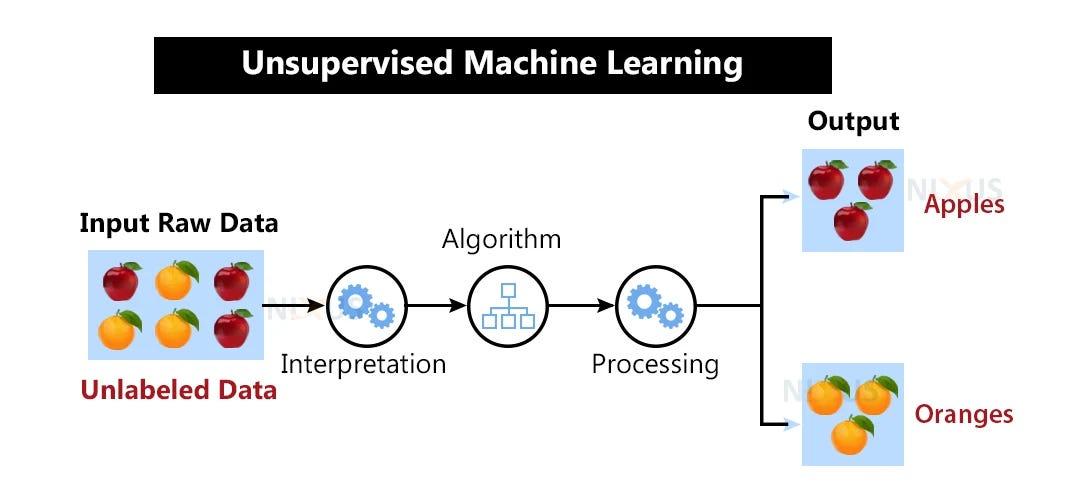
Plutôt que d’automatiser les décisions et les prédictions, cette approche permet d’identifier les patterns et les relations que les humains risquent de ne pas identifier dans les données. Cette technique n’est pas très populaire, car moins simple à appliquer. Elle est toutefois de plus en plus populaire dans le domaine de la cybersécurité.

Un exemple de Machine Learning non supervisé est l'algorithme de reconnaissance faciale prédictive de Facebook, qui identifie les personnes sur les photos publiées par les utilisateurs.

Il existe deux types de Machine Learning non supervisé :

* ***Clustering :*** l'objectif consiste à trouver des regroupements dans les données.
* ***Association :*** l'objectif consiste à identifier les règles qui permettront de définir de grands groupes de données.

Les principaux algorithmes du Machine Learning non supervisé sont les suivants : K-Means, clustering/regroupement hiérarchique et réduction de la dimensionnalité.



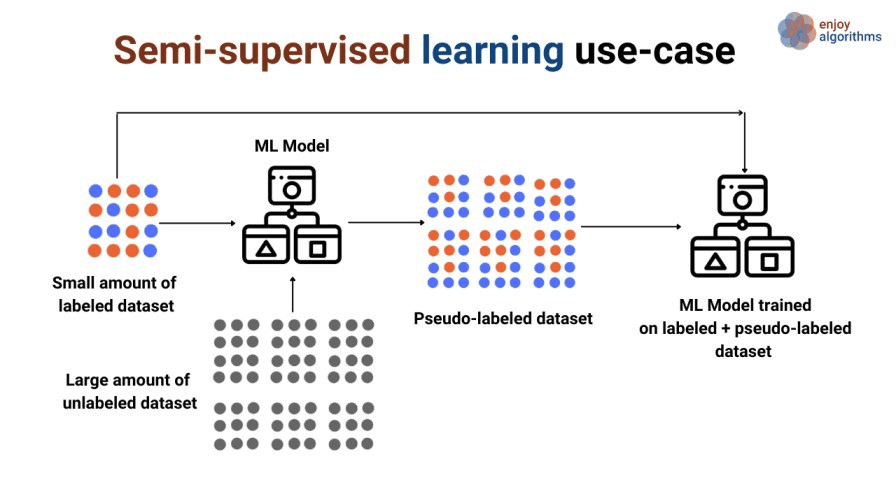
**Figure 16:** L'apprentissage non supervisé

## Apprentissage semi-supervisé :

L’apprentissage semi-supervisé vise à résoudre les problèmes avec des données non étiquetées à l'aide de l'ensemble d'informations étiquetées.

Il offre un compromis entre apprentissage supervisé et non supervisé. Pendant l’entraînement, un ensemble de données étiqueté de moindre envergure est utilisé pour guider la classification et l’extraction de caractéristiques à partir d’un ensemble plus large de données non étiquetées.

Cette approche s’avère utile dans les situations où le nombre de données étiquetées est insuffisant pour l’entraînement d’un algorithme supervisé. Elle permet de contourner le problème. Comme ce qui est montré sur la figure 17.



**Figure 17:** L'apprentissage semi-supervisé

## Machine Learning par renforcement :

Dans le Machine Learning par renforcement, un programme informatique interagit avec un environnement dynamique dans lequel il doit atteindre un certain but, par exemple conduire un véhicule ou affronter un adversaire dans un jeu.

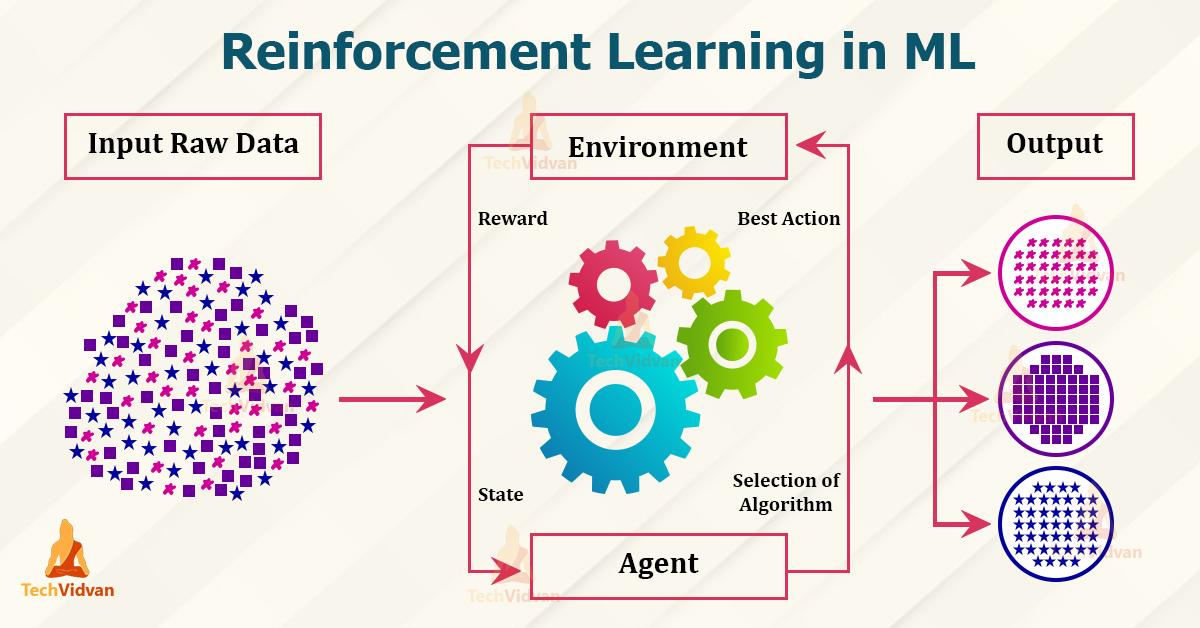
Cette technique consiste à laisser un algorithme apprendre de ses erreurs pour atteindre un objectif. L’algorithme essayera de nombreuses approches différentes pour tenter d’atteindre son but.

En fonction de ses performances, il sera récompensé ou pénalisé pour l’inciter à poursuivre dans une voie ou à changer d’approche jusqu'à ce qu'il apprenne à identifier le comportement le plus efficace dans le contexte considéré. Figure 18.

Il existe deux types de Machine Learning par renforcement :

* ***Monte Carlo :*** le programme reçoit ses récompenses à la fin de l'état « terminal ».
* ***Machine Learning par différence temporelle (TD) :*** les récompenses sont évaluées et accordées à chaque étape.

Les principaux algorithmes du Machine Learning par renforcement sont les suivants : Q-Learning, Deep Q Network (DQN) et SARSA (State-Action-Reward-State-Action).



**Figure 18:** L'apprentissage par renforcement

# Approximation de fonction, classification et régression

## Approximation de fonction

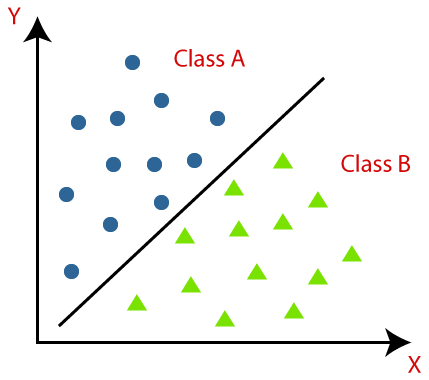
Le terme « modélisation prédictive » désigne un ensemble de méthodes qui permettent d’analyser et d’interpréter des données définies afin d’effectuer une prédiction sur de futurs données. La modélisation prédictive peut être décrite comme un problème mathématique consistant à approximer une fonction de mappage f entre les variables prédictives en entrée X et la variable à prédire Y. C'est ce qu'on appelle un problème d'approximation de fonction. En règle générale, toutes les tâches d'approximation de fonctions peuvent être divisées en tâches de classification et en tâches de régression

## Classification

La classification est une technique d'apprentissage automatique qui consiste à former un modèle pour attribuer une étiquette de classe à une entrée donnée. Elle permet de prédire le résultat d'un échantillon donné pour les variables de sortie qui se présentent sous forme de catégorie.

Il s'agit d'une tâche d'apprentissage supervisée, ce qui signifie que le modèle est formé sur un ensemble de données étiqueté qui comprend des exemples de données d'entrée et les étiquettes de classe correspondantes.

Le modèle vise à apprendre la relation entre les données d'entrée et les étiquettes de classe pour prédire l'étiquette de classe pour une nouvelle entrée invisible.



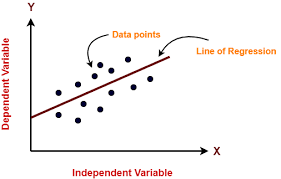
**Figure 19:** Classification

## Régression

Dans l'apprentissage automatique, la régression est un type d'apprentissage supervisé dont le but est de prédire une variable dépendante basée sur une ou plusieurs caractéristiques d'entrée (également appelées prédicteurs ou variables indépendantes).

La régression sert à trouver la relation d’une variable par rapport à une ou plusieurs autres. Elle est utilisée lorsque la variable à prédire Y est continue.

Le but de la régression est d’estimer une valeur (numérique) de sortie à partir des valeurs d’un ensemble de caractéristiques en entrée. Par exemple, estimer le prix d’une maison en se basant sur sa surface, le nombre d'étages, son emplacement, etc. Donc, le problème revient à estimer une fonction de calcul en se basant sur des données d'entraînement.



**Figure 20:** Régression

# Quelques Algorithmes utilisés en machine Learning

Il existe une large variété d’algorithmes de Machine Learning. Certains sont toutefois plus couramment utilisés que d’autres.

## K plus proches voisins (K-NN) :

L’algorithme des K plus proches voisins ou K-Nearest Neighbors (kNN) est un algorithme de Machine Learning qui appartient à la classe des algorithmes d’apprentissage supervisé simple et facile à mettre en œuvre qui peut être utilisé pour résoudre les problèmes de classification et de régression.

Il examine les étiquettes d'un nombre choisi de points de données entourant un point de données cible, afin de faire une prédiction sur la classe à laquelle appartient le point de données. KNN est un algorithme conceptuellement simple mais très puissant, et pour ces raisons, il est l'un des algorithmes d'apprentissage automatique le plus populaires.

## Algorithmes K-Means :

Les K-Means sont des algorithmes de Machine Learning sans supervision qui sont utilisés pour résoudre des problèmes de clustering(regroupement). Ils divisent et classent un ensemble de points de données non affectés d'un label (sans classification externe) en un groupe appelé « cluster » (sans rapport avec les clusters de serveurs). Chaque itération de l'algorithme assigne chaque point à un groupe présentant des caractéristiques similaires. Les points de données peuvent être suivis dans le temps pour détecter les changements qui se produisent dans les clusters.

Les algorithmes K-Means peuvent confirmer des hypothèses sur les types de groupes qui existent dans un Dataset spécifique, ou être utilisés pour découvrir des clusters inconnus. Parmi les cas d'usage commerciaux, citons le regroupement de l'inventaire par activité commerciale et la détection d'anomalies dans des données – par exemple, les données brutes collectées par un bot (robot logiciel).

## Arbre de décision (Decision Tree) :

Les arbres de décision font partie de la catégorie des algorithmes supervisés, elles permettent de prédire une valeur (prédiction) ou une catégorie (classification).

Un arbre de décision est représenté sous forme d’un arbre, chaque nœud de l’arbre représente un test (condition) sur une variable, et chacun de ces enfants correspond à une réponse possible. Les feuilles correspondent à une étiquette.

C’est un type d'algorithme qui peut être appliqué à de nombreux contextes : grande distribution, finance, produits pharmaceutiques, etc. La machine se contente d’établir l’arborescence de divers résultats qui peuvent ou ne peuvent pas se produire, et suit chaque événement jusqu'à sa conclusion naturelle tout en calculant toutes les probabilités des événements pouvant se produire.

Par exemple, une banque pourra utiliser des algorithmes d'arbre de décision pour décider si elle finance tel ou tel emprunt. Les compagnies pharmaceutiques utilisent ces algorithmes pendant leurs tests pour calculer la probabilité des effets secondaires et le coût moyen du traitement.

## SVM (support à vecteurs machine) :

SVM (Support Vector Machine ou Machine à vecteurs de support) est un algorithme d’apprentissage automatique supervisé qui peut être utilisé pour les problèmes de classification ou de régression. Toutefois, il est surtout utilisé dans les problèmes de classification.

Il était extrêmement populaire à l’époque où il a été développé, dans les années 1990, et continue de l'être car il produit une précision significative avec un minimum de puissance de calcul.

Le SVM joue aussi un rôle important dans la reconnaissance des modèles qui est l'un des domaines de recherche les plus populaires et actifs de nos jours.

Le principe des SVM consiste à ramener un problème de classification ou de discrimination à un hyperplan (Feature Space) dans lequel les données sont séparées en plusieurs classes dont la frontière est la plus éloignée possible des points de données (ou marge maximale).

# Comment fonctionne le Machine Learning ?

Le développement d’un modèle de Machine Learning repose sur quatre étapes principales.

La première étape consiste à sélectionner et à préparer un ensemble de données d’entraînement (Dataset). Ces données seront utilisées pour nourrir le modèle de Machine Learning pour apprendre à résoudre le problème pour lequel il est conçu.

Les données peuvent être étiquetées, afin d’indiquer au modèle les caractéristiques qu’il devra identifier. Elles peuvent aussi être non étiquetées, et le modèle devra repérer et extraire les caractéristiques récurrentes de lui-même.

Dans les deux cas, les données doivent être soigneusement préparées, organisées et nettoyées (Noise Free). Dans le cas contraire, l’entraînement du modèle de Machine Learning risque d’être biaisé. Les résultats de ses futures prédictions seront directement impactés.

La deuxième étape consiste à sélectionner un algorithme à exécuter sur l’ensemble de données d’entraînement. Le type d’algorithme à utiliser dépend du type et du volume de données d’entraînement et du type de problème à résoudre.

La troisième étape est l’entraînement de l’algorithme. Il s’agit d’un processus itératif. Des variables sont exécutées à travers l’algorithme, et les résultats sont comparés avec ceux qu’il aurait dû produire. Les » poids » et le biais peuvent ensuite être ajustés pour accroître la précision du résultat.

On exécute ensuite de nouveau les variables jusqu’à ce que l’algorithme produise le résultat correct la plupart du temps. L’algorithme, ainsi entraîné, est le modèle de Machine Learning.

La quatrième et dernière étape est l’utilisation et l’amélioration du modèle. On utilise le modèle sur de nouvelles données, dont la provenance dépend du problème à résoudre. Par exemple, un modèle de Machine Learning conçu pour détecter les spams sera utilisé sur des emails.

De son côté, le modèle de Machine Learning d’un aspirateur robot ingère des données résultant de l’interaction avec le monde réel comme le déplacement de meubles ou l’ajout de nouveaux objets dans la pièce. L’efficacité et la précision peuvent également s’accroître au fil du temps.

# Conclusion

Ce chapitre a présenté la notion de Machine Learning et ses bases pour réussir à résoudre le problème, ainsi les différents algorithmes d’apprentissage utilisés.

Les machines peuvent être entraînées à effectuer des activités humaines dans plusieurs domaines et peuvent aider les humains à vivre mieux. L’apprentissage automatique peut être supervisé ou non supervisé. Si vous avez moins de données et des données clairement étiquetées pour la formation, l’apprentissage non supervisé donne généralement de meilleures performances et de meilleurs résultats pour les grands ensembles de données.

Ce qui ramène à la question suivante : quel algorithme sera utilisé dans ce projet ?

1. L'algorithme K-NN

# Introduction

La performance des algorithmes varie en fonction des caractéristiques spécifiques telles que la taille, la distribution et la nature des données. Ainsi le choix d’un algorithme dépend de plusieurs facteurs, principalement la nature des données, la complexité du problème qu’on essaye de résoudre et les contraintes propres au projet.

Alors quel est l’algorithme idéal qui peut être implémenté dans ce projet ?

# Qu’est-ce que L’algorithme K-NN ?

L'algorithme des k plus proches voisins, aussi connu sous le nom de KNN (K-Nearest Neighbors), est un algorithme d’apprentissage automatique supervisé simple et facile à mettre en œuvre.

Ce dernier est un discriminant d'apprentissage supervisé non paramétrique, qui utilise la proximité pour effectuer des classifications ou des prédictions sur le regroupement d'un point de données individuel. Bien qu'il puisse être utilisé pour des problèmes de régression ou de classification, il est généralement utilisé comme algorithme de classification, en partant de l'hypothèse que des points similaires peuvent être trouvés les uns à côté des autres.

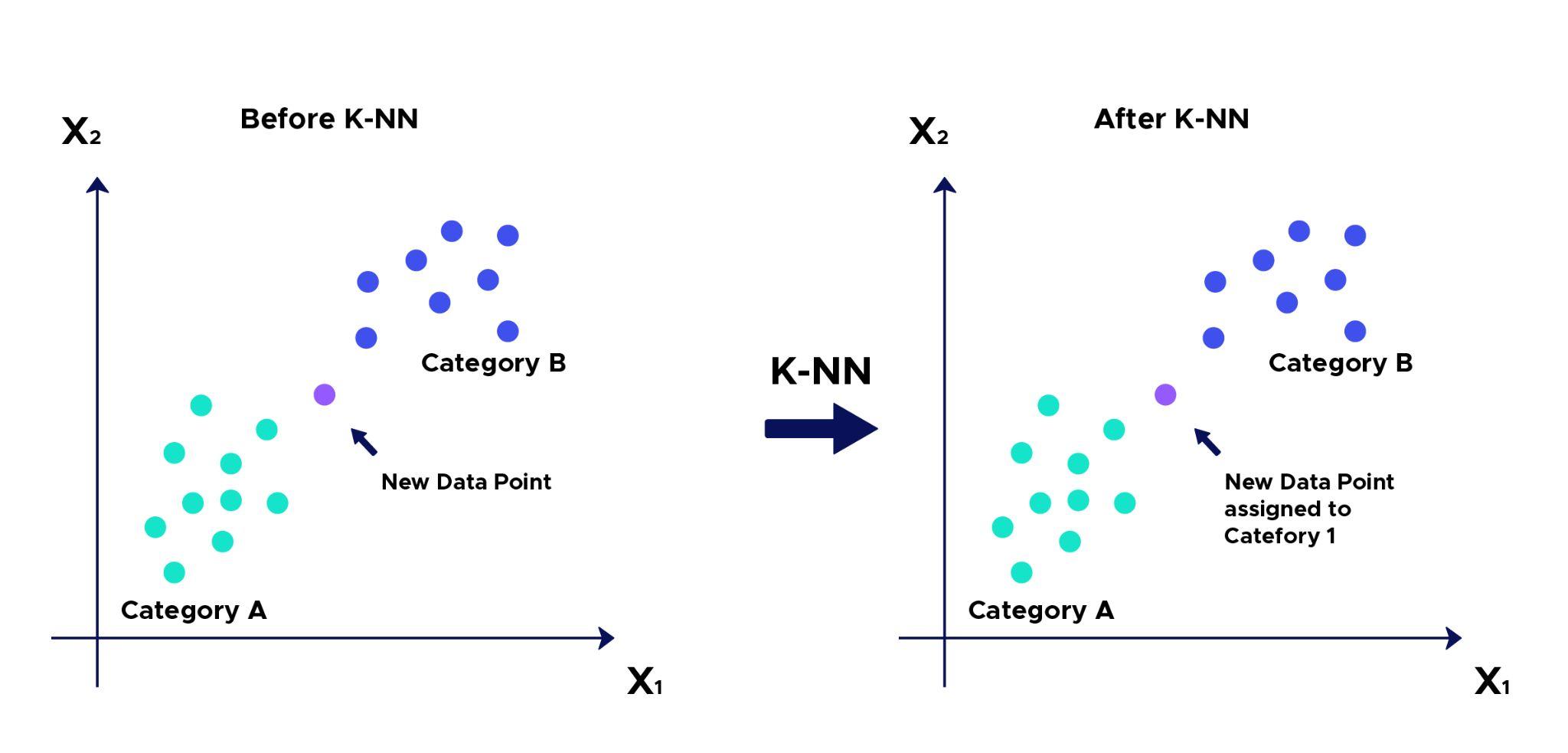
En se basant sur des données d'entrée étiquetées qu'il sera en mesure de proposer une sortie appropriée en partant maintenant de nouvelles données non étiquetées.

Dans la pratique, KNN n'aura pas besoin de passer par la construction d'un modèle prédictif (Training Set) pour réaliser des prédictions. L'algorithme classe les nouvelles données en se basant sur le jeu de données précédant pour fournir des résultats.

En effet, cet algorithme est qualifié comme paresseux (Lazy Learning) car il n’apprend rien pendant la phase d’entraînement. Pour prédire la classe d’une nouvelle donnée d’entrée, il va chercher ses K voisins les plus proches (en utilisant la distance euclidienne, ou autres) et choisira la classe des voisins majoritaires.

Son principe se décrit par cette affirmation : « *Dis-moi qui sont tes voisins, je te dirais qui tu es !* ».

Dans l’exemple ci-dessous (figure 21), les caractéristiques sont x1 et x2 et les labels sont A et B



**Figure 21:** L'algorithme kNN

On ajoute un nouveau point dont on ne connaît pas le label… L’algorithme kNN va lui attribuer un label, en utilisant les données précédentes. Le point se voit attribuer à la catégorie A.

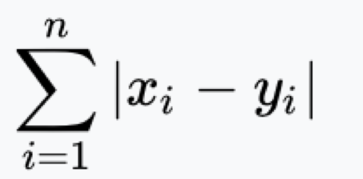
# Fonctionnement de l’algorithme K-NN

Pour effectuer une prédiction, l'algorithme va chercher les K instances du jeu de données les plus proches de l’objectif. Ensuite pour ses K voisins, l’algorithme se basera sur leurs variables de sortie (output variable) pour calculer la valeur de la variable K qu’on souhaite prédire.   
Il existe plusieurs fonctions de calcul de distance, notamment, la distance euclidienne, la distance de Manhattan, la distance de Hamming, la distance de Minkowski…etc.

On choisit la fonction de distance en fonction des [types de données](https://mrmint.fr/types-de-donnees-machine-learning) qu’on manipule. Pour les données quantitatives (exemple : poids, salaires, taille, montant de panier électronique etc…) et du même type, la distance euclidienne est un bon candidat. Quant à la distance de Manhattan, elle est une bonne mesure à utiliser quand les données (input variables) ne sont pas du même type (exemple : âge, sexe, longueur, poids etc…).

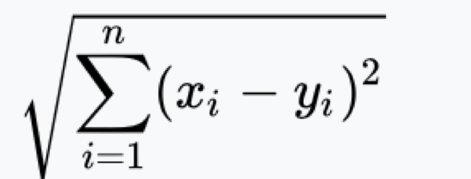
* **Distance euclidienne :**

Calcule la racine carrée de la somme des différences carrées entre les coordonnées de deux points :



* **Distance Manhattan :**

Calcule la somme des valeurs absolues des différences entre les coordonnées de deux points :



Pour appliquer cette méthode, les étapes à suivre sont les suivantes :

***Étape - 1 :***

Sélectionnez le nombre K des voisins

***Étape - 2 :***

Calculez la distance euclidienne de chaque point par rapport au point cible.

***Étape - 3 :***

Prenez les K voisins les plus proches selon la distance euclidienne calculée.

***Étape - 4 :***

Parmi ses voisins, comptez le nombre de points de données dans chaque catégorie.

***Étape - 5 :***

Attribuez les nouveaux points de données à la catégorie pour laquelle le nombre de voisins est maximum.

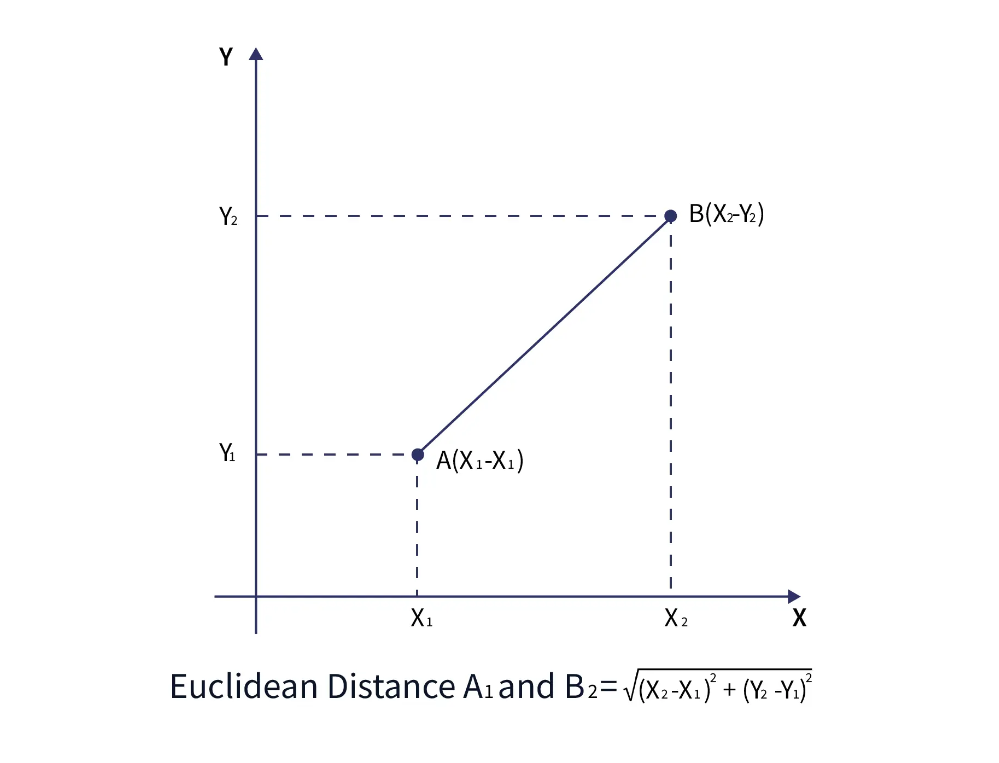
Supposons que nous ayons un nouveau point de données et que nous devons le placer dans la catégorie requise. Considérez l'image ci-dessous :



**Figure 22:** Un nouveau point de données à catégorisé à l'aide de l’algorithme kNN

Tout d’abord, nous choisirons le nombre de voisins tel que k=5.

Ensuite, nous calculerons la distance euclidienne entre les points de données. La distance euclidienne est la distance entre deux points, que nous avons déjà étudiée en géométrie. Il peut être calculé comme suit sur la figure 23 :

****

**Figure 23:** La distance Euclidienne entre deux voisins

En calculant la distance euclidienne, les voisins les plus proches sont obtenus, soit trois voisins les plus proches dans la catégorie A et deux voisins les plus proches dans la catégorie B. Considérez l'image ci-dessous figure 24 :



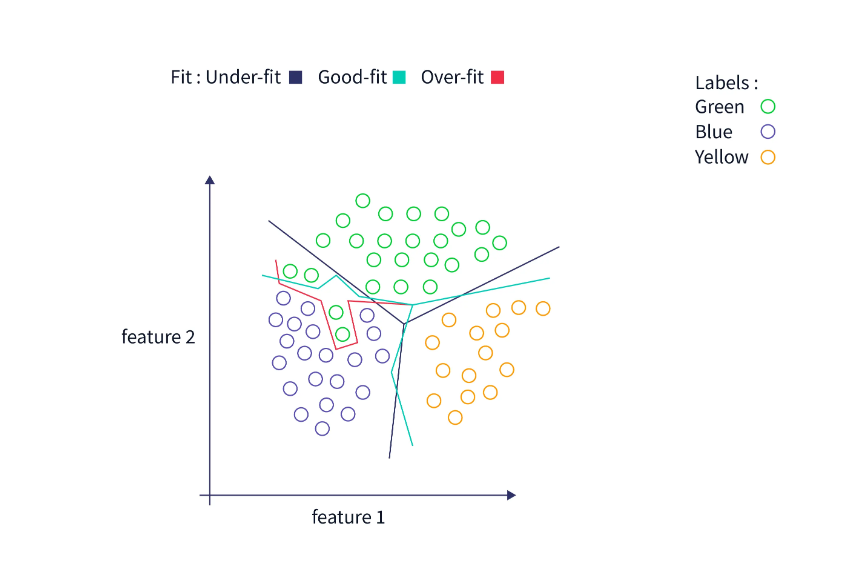
**Figure 24:** Obtention des plus proches voisins et catégorisation du nouveau point

Comme on peut le constater, les trois voisins les plus proches sont de catégorie A. Ce nouveau point de données doit donc appartenir à la catégorie A.

Mais, comment choisit-on ce paramètre ‘k’ lors de l’implémentation de l’algorithme ?

# Comment choisir la valeur K ?

Le choix de la valeur K à utiliser est crucial pour effectuer une meilleure prédiction avec K-NN. En règle générale, moins on utilisera de voisins (un nombre K petit) plus on sera sujette au sous-apprentissage (**underfitting**). Par ailleurs, plus on utilise de voisins (un nombre K grand) plus on sera fiable dans notre prédiction. Toutefois, si on utilise K nombre de voisins avec K=N et N étant le nombre d’observations, on risque d’avoir du ([**overfitting**](https://mrmint.fr/overfitting-et-underfitting-quand-vos-algorithmes-de-machine-learning-derapent))et par conséquent un modèle qui se généralise mal sur des observations qu’il n’a pas encore vues.



**Figure 25:**presentation du overfitting, underfitting et good fit

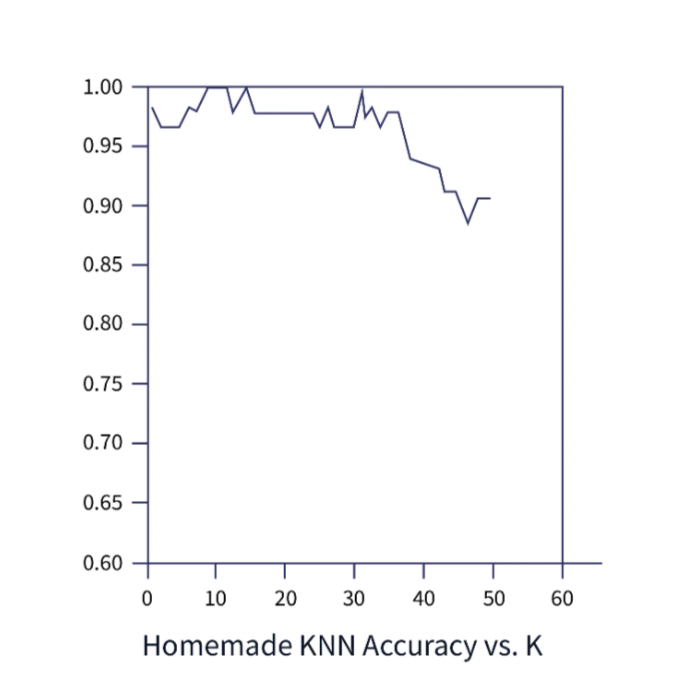
Pour sélectionner la valeur de k qui convient à vos données, nous exécutons plusieurs fois l’algorithme KNN avec différentes valeurs de k. Puis nous choisissons le k qui réduit le nombre d’erreurs rencontrées tout en maintenant la capacité de l’algorithme à effectuer des prédictions avec précision lorsqu’il reçoit des données nouvelles (non vues auparavant).

Voici quelques points à garder en tête :

-Définissez K comme un nombre impair, nous avons donc un point supplémentaire pour régler les départages dans les cas extrêmes.

-Une valeur décente de k peut être la racine carrée du nombre de points de données impliqués dans l'ensemble de données d'entraînement. Bien entendu, cette méthode peut ne pas fonctionner dans tous les cas.

-Tracez le graphique du taux d'erreur par rapport au K :



**Figure 26:**le taux d'erreur par rapport au paramètre k

Tout d’abord, divisez l’ensemble de données de surveillance en deux catégories : Formation et Données de test.

Les données de formation devraient être plus que ces dernières. Commencez par k = 1 et prédisez avec KNN pour chaque point de données dans les données de test. Comparez ces prédictions avec les étiquettes réelles et déterminez la précision du modèle. Trouvez le taux d'erreur.

Répétez ce processus pour différentes valeurs de k et tracez le taux d'erreur par rapport au graphique k, puis choisissez la valeur de k qui a le taux d'erreur le plus faible.

Enfin le modèle est prêt pour être utilisé.

# Exemple d’utilisation de k-NN

Il peut être utilisé dans des technologies comme l’OCR (Optical Character Recognizer), qui tente de détecter l’écriture manuscrite, les images et même les vidéos.

Il peut être utilisé dans le domaine des notations de crédit. Il essaie de faire correspondre les caractéristiques d’un individu avec le groupe de personnes existantes afin de lui attribuer la cote de crédit. Il se verra attribuer la même note que celle accordée aux personnes correspondant à ses caractéristiques.

Il est utilisé pour prédire si la banque doit accorder un prêt à un particulier. Il tentera d’évaluer si l’individu donné correspond aux critères des personnes qui avaient précédemment fait défaut ou ne fera pas défaut à son prêt.

# Conclusion

Contrairement à de nombreux algorithmes plus complexes, L’algorithme des k plus proches voisins (**kNN**) est l’un des algorithmes de Machine Learning supervisé les plus simple à mettre en œuvre qui peut être utilisé pour résoudre des problèmes de classification et de régression. L’algorithme est polyvalent.

La phase d’apprentissage est immédiate ; Il n’est pas nécessaire de construire un modèle, d’ajuster plusieurs paramètres ou de faire des hypothèses supplémentaires. Il suffit de découper notre jeu de données initiales (labellisées) en deux lots. Le premier servira à apprendre le second à tester.

La phase de test permet de mesurer la qualité des prédictions contre des données pour lesquelles on connaît déjà la classe.

En conclusion, l'algorithme **KNN** (K-Nearest Neighbors) est un choix prometteur pour ce projet de détection d'intrusions. Sa facilité d'implémentation, sa capacité à s'adapter à différents types de données et sa polyvalence en font un candidat attrayant, surtout dans un domaine aussi complexe que la sécurité informatique.

1. Conception et réalisation

# Introduction

Dans le contexte de la protection des systèmes informatiques contre les intrusions et les attaques, le système de détection des intrusions (**IDS**) joue un rôle crucial en identifiant les activités suspectes. De plus l’intégration de l’apprentissage automatique renforcera d’avantage les barreaux de la sécurité.

Parmi les méthodes d’apprentissage automatique connues, l’algorithme supervisé **K-NN** a été le meilleur choix pour ce projet grâce à sa simplicité et son efficacité en termes de classification des anomalies.

Le développement d’un tel système efficace nécessite une étude profonde et un choix bien réfléchi.  
Ce chapitre mettra en lumière les outils et les méthodes utilisés, le jeu de données (Dataset) exploité, ainsi l’implémentation et l’apprentissage du modèle accompagné d’une analyse des résultats obtenues.

# Outils et environnement d’exécution

## L’environnement d’exécution

Le domaine du Machine Learning nécessite des ressources matérielles, notamment des processeurs graphiques (GPU) puissants, pour effectuer des calculs complexes. Initialement, pour un environnement cloud offrant des capacités de calcul et de mémoire supérieures à celles d’un ordinateur personnel, **Google Colab,** l’un des outils cloud les plus prisés pour le Machine Learning a été choisi pour ce projet.

**Google Colaboratory** est un service cloud basé sur les Jupyter Notebooks qui facilite le développement d’applications de Deep Learning en Python. Il met à disposition gratuitement un processeur GPU, 12 Go de RAM et plus de 100 Go d’espace de stockage. L’accès à ce service nécessite simplement un compte Google et une connexion stable.

## Les outils et langage de programmation utilisés

**Python** a été retenu comme langage de développement en raison de son interprétabilité, de sa polyvalence et de sa popularité dans les domaines de l’apprentissage automatique et de l’intelligence artificielle. Python se distingue également par sa vaste gamme de bibliothèques logicielles open source. Celles utilisées :

**Scikit-learn**, une bibliothèque Python d’apprentissage automatique dédiée au calcul scientifique, qui propose divers algorithmes pour la régression, la classification et le partitionnement de données.

Le **NumPy**, une bibliothèque Python de traitement de données qui permet de manipuler des tableaux et des matrices multidimensionnels de grande taille. Elle fournit également un ensemble de fonctions mathématiques de haut niveau pour effectuer des opérations sur ces tableaux.

**Le Pandas**, une autre bibliothèque Python de traitement de données utilisée pour la manipulation et l'analyse de données. Elle travaille avec des structures de données appelées trame de données « DataFrames ». Pandas offre de nombreux outils intégrés pour une multitude de tâches, dont le nettoyage des données, qui est une étape cruciale. Selon IBM Analytics, environ 80 % du temps consacré à un projet de Machine Learning est consacré au nettoyage des données. De nombreux ensembles de données contiennent des champs vides, ce qui peut nuire considérablement à la performance d'un modèle. La fonction **isnull()** de Pandas est un exemple d'outil permettant de gérer ces valeurs manquantes.

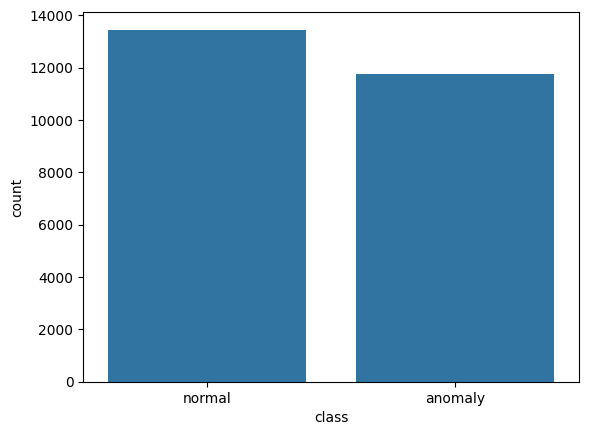
**Le Seaborn** est aussi une bibliothèque Python, pour la visualisation de données basée sur **Matplotlib** qui fournit une interface de haut niveau permettant la visualisation des données en créant des graphiques statistiques attrayants et informatifs.

## Le jeu de données exploité (DataSet)

La réussite des **IDS** qui emploient l’apprentissage profond est étroitement liée au jeu de données choisi. Dans ce travail, l’ensemble de données (Dataset) choisi, est appelé « **Network Intrusion Detection** », contenant une grande variété d'intrusions simulées dans un environnement de réseau militaire. Il a permis de créer un environnement permettant d'acquérir des données brutes de vidage **TCP/IP** pour un réseau en simulant un réseau local typique de l'**US Air Force**. Le réseau local a été ciblé comme un environnement réel et a subi de multiples attaques.

Ce choix est fait en se basant sur plusieurs points. En premier, ce Dataset présente une richesse de scenarios d’intrusion, permettant de tester l’efficacité du modèle face à différents types d’attaques. En plus, il reflète des scenarios réels et un étiquetage bien précis, ce qui améliore d’avantage la capacité et l’efficience du modèle.

Le modèle proposé est un modèle de Machine Learning (**K-NN**) en matière de cybersécurité permettant à l’IDS de faire face aux types d’attaques du réseau afin d’identifier les activités malveillantes dans un trafic de réseau réel. La figure 27 affiche le pourcentage de normalité et d’anomalie dans le Dataset proposé.



**Figure 27:**Les composantes du Dataset

Alors, comment adapter les caractéristiques de ces données à l’implémentation de l’algorithme d’apprentissage choisi ?

# Implémentation et développement

L’implémentation du modèle de système de détection des intrusions se fait en plusieurs étapes. L’importation des bibliothèques et de données, le prétraitement de données et l’entrainement de l’algorithme font une partie importante du processus de préparation de l’environnement de travail.

Tout d’abord, il faut importer toutes les bibliothèques nécessaires : Pandas, Numpy, Scikit-learn, Matplotlib et Seaborn. Et ensuite charger les données.

## Prétraitement des données

Les données collectées peuvent contenir des valeurs manquantes, des doublons ou des valeurs aberrantes affectant par la suite l’efficacité de la performance du modèle.

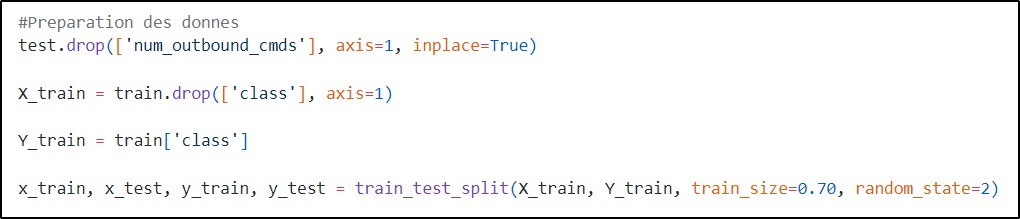
Le prétraitement a pour but de nettoyer et transformer les données pour les rendre meilleures pour l’entrainement, ainsi renforcer l’algorithme utilisé.

Le nettoyage consiste à supprimer les valeurs nulles et les doublons, et la normalisation des données consiste à transformer les valeurs des variables pour les ramener à une échelle commune.

## Entrainement du modèle

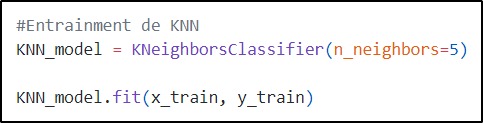
L’entrainement du modèle est la partie la plus importante où le modèle apprend à partir des données prétraitées auparavant. Cette phase contient aussi des étapes, comme suit :

* La division des données en un ensemble d’entrainement et un ensemble de test, ce que montre la figure 28.



**Figure 28:**Division des Dataset en données d'entrainement et données de test

* Le choix du paramètre (**k**) pour l’entrainement de k-NN, figure 29.



**Figure 29:** Choix du paramètre k

* Et l’évaluation de la performance du modèle à l’aide de métriques telles que la précision, le rappel, F1-score...

Les métriques utilisées pour évaluer les algorithmes implémentés, sont :

* **True Positive (TP) :** indique que le modèle a prédit correctement un résultat.
* **False Positive (FP) :** indique que le modèle a prédit incorrectement un résultat.
* **True Negative (TN) :** indique que le modèle a prédit correctement un résultat négatif.
* **False Negative (FN) :** indique que le modèle a prédit incorrectement un résultat négatif.
* **Accuracy :** est simplement la mesure de la capacité d'un modèle à prédire la bonne classe pour une donnée.

**Accuracy = (TP + TN) / (TP + TN + FP + FN)**

* **Precision :** représente la proportion de vrais positifs par rapport au nombre total de positifs.

**Precision = TP / (TP + TN)**

* **Recall :** représente la proportion de cas positifs correctement identifiés.

**Recall = TP / (TP + FN)**

* **F1-Score :** est similaire à la ‘Précision’, mais il s'agit d'un meilleur indicateur car il cherche à créer un équilibre entre la précision et le ‘Recall’.

**F1-Score = 2 \* (Precision \* Recall) / (Precision + Recall)**

Ces étapes garantissent un modèle k-NN opérationnel, optimisé et prêt à détecter les intrusions.

# Résultats et discussion

La table 1 montre les scores après l’entrainement du modèle k-NN :

**Tableau 1: Score d’entrainement du modèle K-NN**

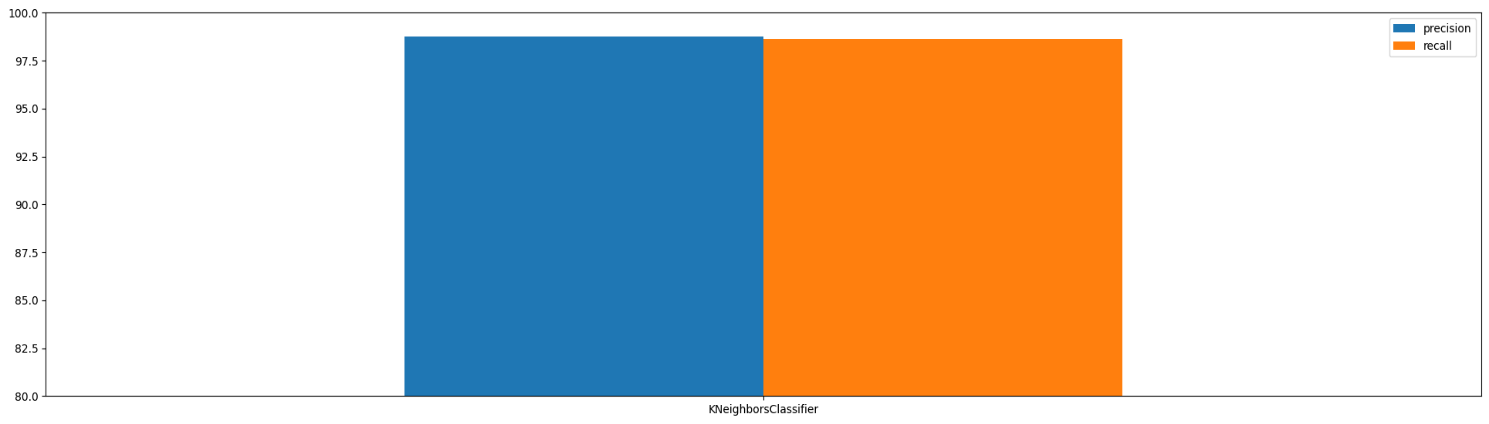
|  |  |  |  |
| --- | --- | --- | --- |
| Classe | Precision | Recall | F1-Score |
| Normale | 0.99 | 0.99 | 0.99 |
| Anomalie | 0.99 | 0.99 | 0.99 |

La table 2 affiche le score du modèle k-NN :

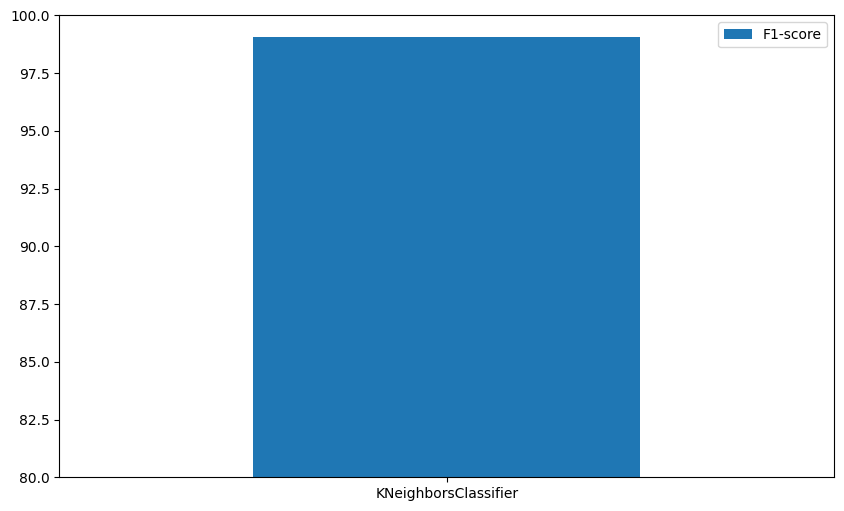
**Tableau 2: Score de la validation du modèle K-NN**

|  |  |  |
| --- | --- | --- |
|  | Precision | Recall |
| K-NN | 98.77± 0.31 | 98.62 ± 0.26 |

Les figures 30 et 31 résument les scores du modèle, la précision et le rappel, ainsi que le F1-score sur des graphes :



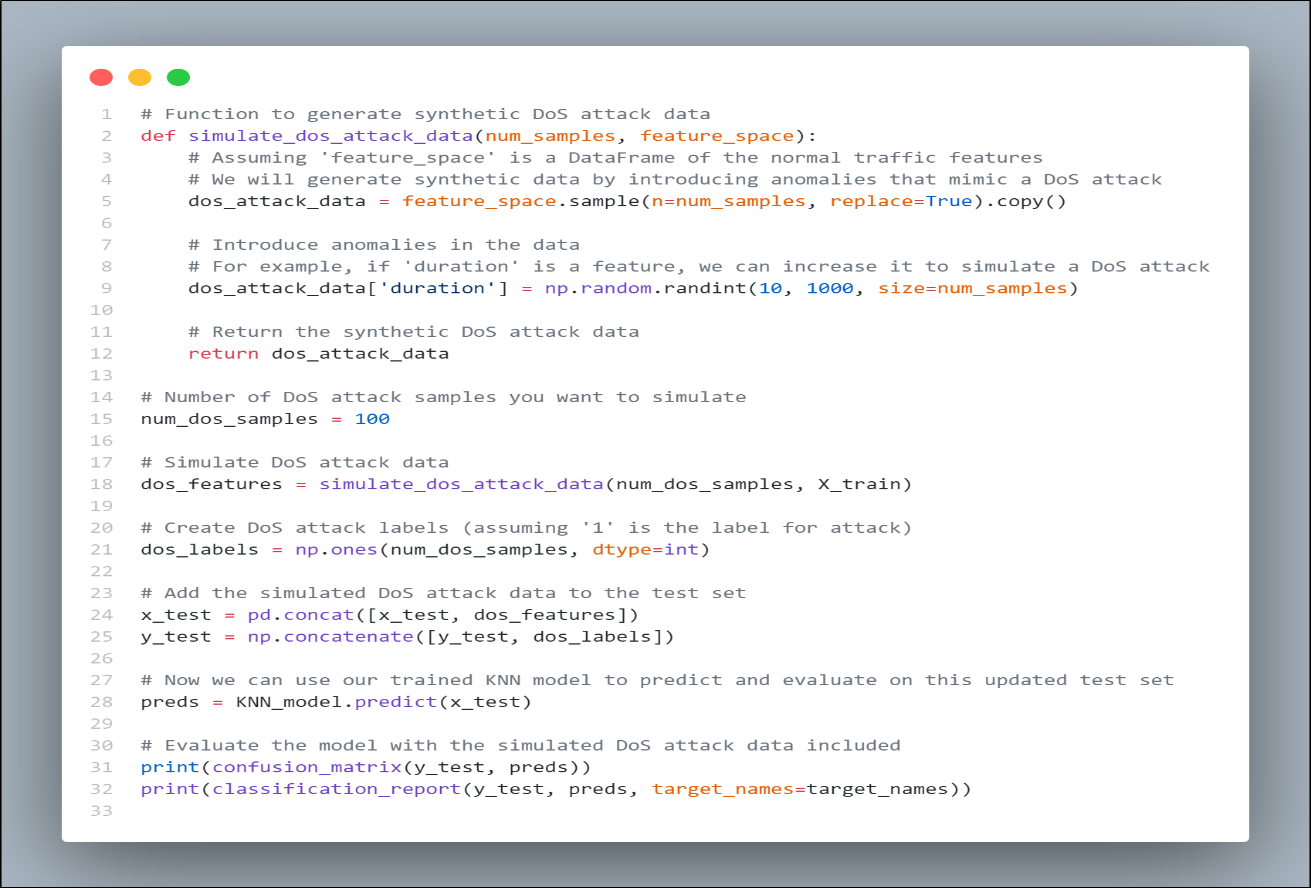
**Figure 30:** Les scores de précision et rappel du modèle



**Figure 31:** Le pourcentage du F1-Score

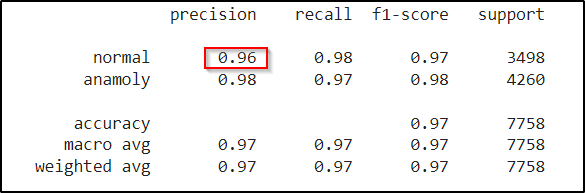
# Implémentation d’une attaque par déni de service (DoS)

Une implémentation d’une attaque par déni de service (**DoS**), introduit dans la figure 32,est effectuée pour la classification de ce type d’attaque réseau, afin d’améliorer le taux de détection. La performance des approches de détection proposées a été évalué pour prendre en compte les différentes mesures d’évaluation des algorithmes d’apprentissage à savoir, la précision, le rappelle, le score F1, le taux de détection et le taux de fausses alarmes.



**Figure 32:** Screenshot de DoS code

Elle donne le score suivant dans la figure 33 :



**Figure 33:** Score de prédiction d’une attaque de type DoS

Après l’ajout de l’attaque DoS, le score de la classe « Normal » est descendu à 96%, ce qui implique la présence de plusieurs false positive (comme introduit dans les paragraphes précédentes).

# Conclusion

Dans ce chapitre la conception et la mise en œuvre d'un système de détection d'intrusion (**IDS**) utilisant le modèle **K-NN** ont été présentées.

Les résultats montrent une haute précision et un bon rappel, malgré une légère augmentation des faux positifs avec l'intégration des attaques **DoS**. L'utilisation d'outils cloud et de bibliothèques Python a été essentielle pour le succès de ce projet.

Conclusion

Ce projet de fin d’études a démontré la pertinence et l’efficacité des systèmes de détection d'intrusions (**IDS**) basés sur l'apprentissage automatique, notamment à travers l'algorithme **K-NN**. En explorant les concepts fondamentaux de la sécurité des réseaux et des **IDS**, nous avons mis en lumière les défis et les vulnérabilités auxquels sont confrontés les systèmes informatiques modernes. L'application du Machine Learning dans ce contexte a permis d'améliorer significativement la capacité de détection des intrusions, en offrant une réponse adaptable et évolutive face aux menaces en constante évolution. Les résultats obtenus montrent que l'intégration de techniques avancées d'analyse de données peut renforcer la protection des réseaux contre les cyberattaques, ouvrant ainsi la voie à des systèmes de sécurité plus robustes et intelligents. Cette étude propose des bases solides pour des recherches futures visant à perfectionner ces technologies et à les adapter aux besoins spécifiques des environnements réseau contemporains.

**Bibliographie et références**

* Ilemona S. Atawodi, « A Machine Learning Approach to Network Intrusion Detection System Using K Nearest Neighbor and Random Forest » Memoire de Master. University of Southern Mississippi (2019).
* Adil Chouieb et Mohammed islem Chetouane, « Evaluation de la sécurité d’un réseau informatique Mécanisme test d’intrusion » Mémoire de fin de formation. I.N.S.F.P chabouni idriss, JIJEL.
* Hamouda Djallel, « Un système de détection d’intrusion pour la cybersécurité » Mémoire de fin d’études Master. Université de Guelma, Faculté des Mathématiques, d’Informatique et des Science de la matière. (2020)
* Tighzer Yanis et Mouzaia Raouf, « Apprentissage automatique pour la détection d’intrusion dans un système informatique » Mémoire de fin de cycle master. Université A. Mira de Bejaia, Faculté des Sciences Exactes.
* M.Sefraoui Omar, « Construction d'un système de détection d'intrusion réseau (NIDS) à l'aide de Suricata, Zeek, Filebeat et ELK » Mémoire de projet de fin d’année. Université Mohammed Premier, Ecole Nationale des Sciences Appliquées Oujda. (2023)
* <https://www.indusface.com/learning/what-is-a-ddos-attack/>
* <https://blent.ai/blog/a/svm-support-vector-machine>
* <https://datascientest.com/>
* <https://geekflare.com/fr/regression-vs-classification/>
* <https://www.jedha.co/formation-ia/algorithmes-machine-learning#:~:text=La%20classification%20permet%20de%20pr%C3%A9dire,est%20une%20valeur%20r%C3%A9elle%20sp%C3%A9cifique>
* <https://www.ibm.com/topics/knn#:~:text=The%20k%2Dnearest%20neighbors%20(KNN)%20algorithm%20is%20a%20non,used%20in%20machine%20learning%20today>.
* <https://mrmint.fr/introduction-k-nearest-neighbors>
* <https://www.scaler.com/topics/machine-learning/knn-algorithm-in-machine-learning/>
* <https://www.researchgate.net/figure/The-main-types-of-machine-learning-Main-approaches-include-classification-and-regression_fig1_354960266>