République Algérienne Démocratique et Populaire

Ministère de l'enseignement supérieur et de la recherche scientifique
Université 20 Aout 1955-SKIKDA

Faculté des Sciences
Département d'Informatique

Mémoire de fin d'études en vue de l'obtention du diplôme de Master professionnel en Informatique

Option : Réseaux et Systèmes Distribues (RSD)

Thème

Coopération de modèles d'IA pour la détection d'intrusions

Réalisé par :

-BOUCHAREB KENZA

-KABIR DALILA

Encadre par :

Pr. MAZOUZI SMAINE

Année Universitaire 2024-2025


بسم الله الرحمن الرحيم

Remerciement

Louange Tout d'abord, nous remercions Dieu de nous aider et nous donner la force et la volonté pour achever ce modeste travail.

*Ensuite, nous tenons à exprimer nos plus vifs remerciements notre encadreur « **Mr Mazouzi Smaine** » pour son encadrement continu.*
On le remercie également pour la confiance qu'il nous a accordée et pour la grande liberté d'idées et de travail qu'il nous a donnée.

Nous tenons à remercier également les membres des jurys pour avoir bien voulu évaluation et juger ce travail.

Nous veut aussi adresser nous sincères remerciements à tous les enseignants de département de l'informatique qui ont contribué à nous formation.

Quelques personnes ont contribué à la réalisation de ce travail et méritent des remerciements.

Enfin et surtout, nous tenons à remercier vivement toute nous familles notamment nous parents, qui nous ont toujours encouragés dans la poursuite de nous études, ainsi que pour leur aider, leur soutenir sans oublier.

Dédicace

Louange à Dieu, en premier et en dernier, autant de fois qu'Il a été loué et autant de fois qu'Il le sera. Merci à Lui pour ce qu'Il m'a donné, facilité et permis d'accomplir.

À moi-même, qui ai tenu bon, affronté les épreuves et persévéré, je dédie ce travail comme le fruit de ma patience et de mes efforts.

À mes parents bien-aimés :
À ma mère Mahbouba, source d'amour et de prières,
Et à mon père Mouloud, mon soutien et ma plus grande fierté,
Merci pour votre amour, votre présence et vos invocations constantes.

À mon encadrant, Dr. MAZOUZI SMAINE,
Je vous remercie sincèrement pour votre accompagnement, vos conseils et votre soutien tout au long de ce travail.

À mes frères Imed et Omar, mes sœurs Selma, Rahma, Roumaissa, et ma petite Ayoche,
Merci d'être là, toujours. Vous êtes ma force et ma joie.

À mes grands-parents Mohamed et Zahra,
Toute ma reconnaissance et mes prières les plus sincères.

À mes amies et compagnes de route,
Merci pour votre soutien, vos mots et vos sourires qui m'ont portée.

Une pensée particulière pour mon amie et sœur de cœur Romeissa BenHmidcha,
Tu as toujours été présente dans les moments difficiles comme dans les moments de bonheur. Que tu restes une bénédiction dans ma vie.

Merci aux familles Bouchareb et Bousselsal pour leur amour et leur appui.

À ma binôme Dalila,
Merci pour ton engagement, ton sérieux et ta présence durant tout ce parcours. Ce travail est aussi le fruit de notre collaboration.

À tous ceux qui ont marqué ma vie, ce succès vous appartient autant qu'à moi.

KENZA

Dédicace

Je dédie ce travail

*A mon père *Mohamed Salah* qui ma toujours soutenue dans les moments difficiles et pour ses sacrifices et ses encouragements dieu me le garde.*

À la mémoire de ma mère **Fatiha** *et de mes sœurs* **Nadira** *et* **Hassina**, *qui ont sacrifié leur vie pour mon bien-être, je dédie ce modeste mémoire. Je prie de tout cœur pour que leurs âmes reposent en paix.*

À mes chers frères et à ma chère sœur, pour leur soutien et leurs encouragements constants, avec une pensée toute particulière pour mon frère **Abdenour**, *qui a été mon véritable pilier dans les moments difficiles tout au long de ce parcours.*

A toute ma petite famille (mon mari ***Rabah*** *et mes enfants:* ***Mohamed***, ***Haythem*** *et* ***Rayane****) pour leur soutien tout au long de temps,*

À mon encadrant, ***Pr. Mazouzi Smaine*** *Je le remercie sincèrement pour son accompagnement, ses conseils et son soutien tout au long de ce travail.*

*****Kenza*****, qui a été mon binôme durant toute cette année*

A tous ceux qui me sont chers et que j'ai involontairement oublié A tous mes professeurs.

Merci !

Dalila

Résumé :

Avec l'augmentation continue des cyberattaques et l'évolution des techniques d'intrusion, les systèmes de détection d'intrusions (IDS) sont devenus un élément essentiel pour assurer la sécurité des réseaux. Dans ce contexte, ce travail vise à améliorer l'efficacité des systèmes de détection en adoptant une approche hybride basée sur la coopération entre un modèle de réseau de neurones profond (DNN) et un classificateur bayésien. Cette approche combine les capacités d'apprentissage profond représentatif avec la simplicité et l'efficacité des modèles statistiques, dans le but d'améliorer la précision de la classification et de réduire le taux de faux positifs.

Le système proposé a été mis en œuvre et évalué en utilisant la base de données standard NSL-KDD, largement utilisée dans les recherches en détection d'intrusions. L'évaluation a porté sur plusieurs métriques telles que la précision, le score F1 et le taux de faux positifs. Les résultats ont montré que l'intégration des deux modèles offre une performance supérieure par rapport aux systèmes traditionnels basés sur un seul modèle, soulignant ainsi l'efficacité de cette approche coopérative face aux défis croissants de la cybersécurité.

Mots-clés : détection d'intrusions, intelligence artificielle, réseaux de neurones profonds, classificateur bayésien, NSL-KDD, cybersécurité, apprentissage automatique.

ملخص:

مع التزايد المستمر للهجمات السيبرانية وتطور تقنيات الاختراق ، أصبحت أنظمة كشف التسلل (IDS) عنصرا أساسيا لضمان أمن الشبكات. في هذا السياق ، يهدف هذا العمل إلى تحسين كفاءة أنظمة الكشف من خلال اعتماد مقاربة هجينة تقوم على التعاون بين نموذج الشبكة العصبية العميقة (DNN) والمصنف البايزي . تقوم هذه المقاربة على دمج قدرات التعلم التمثيلي العميق مع بساطة وفعالية النماذج الإحصائية، بهدف تعزيز دقة التصنيف وتقليل معدل الإيجابيات الكاذبة.

لقد تم تطبيق النظام المقترح وتقييمه باستخدام قاعدة البيانات القياسية NSL-KDD ،التي تستخدم على نطاق واسع في أبحاث كشف التسلل .وقد شمل التقييم مجموعة من المقاييس مثل الدقة ، ومعامل F1، والنسبة المئوية للإيجابيات الكاذبة. أظهرت النتائج أن التكامل بين النموذجين يوفر أداء متفوقا مقارنة بالأنظمة التقليدية المعتمدة على نموذج واحد ، مما يبرز فعالية المقاربة التعاونية في مواجهة التحديات المتزايدة للأمن السيبراني.

الكلمات المفتاحية: كشف التسلل، الذكاء الاصطناعي، الشبكات العصبية العميقة، المصنف البايزي، NSL-KDD، الأمن السيبراني، التعلم الآلي.

Abstract :

With the continuous increase in cyberattacks and the evolution of intrusion techniques, Intrusion Detection Systems (IDS) have become an essential component for ensuring network security. In this context, this work aims to improve the efficiency of detection systems by adopting a hybrid approach based on the cooperation between a Deep Neural Network (DNN) model and a Bayesian classifier. This approach combines the representational learning capabilities of deep learning with the simplicity and effectiveness of statistical models, with the goal of enhancing classification accuracy and reducing the false positive rate.

The proposed system was implemented and evaluated using the standard NSL-KDD dataset, which is widely used in intrusion detection research. The evaluation involved several metrics such as accuracy, F1-score, and false positive rate. The results showed that integrating the two models provides superior performance compared to traditional systems based on a single model, thus highlighting the effectiveness of this cooperative approach in addressing the growing challenges of cybersecurity.

Keywords: Intrusion detection, Artificial intelligence, Deep neural networks, Bayesian classifier, NSL-KDD, Cybersecurity, Machine learning.

Sommaire

Chapitre 2 : *Apprentissage Automatique*

Chapitre 3 : Description du projet

Chapitre 4 : Implémentation

La Liste des Figures

La Liste des Tableaux

Introduction Générale

Introduction générale :

Avec l'évolution rapide des technologies de l'information et de la communication, la sécurité des systèmes informatiques est devenue une préoccupation majeure. La prolifération des attaques informatiques sophistiquées et leur impact potentiel sur les infrastructures critiques exigent le développement de solutions efficaces pour la détection précoce des intrusions et la protection des réseaux. Dans ce contexte, l'intelligence artificielle (IA) s'est imposée comme un outil prometteur pour améliorer la capacité des systèmes de détection d'intrusions (IDS) à identifier les comportements anormaux et les attaques malveillantes.

Ce mémoire porte sur la coopération de modèles d'intelligence artificielle dans le domaine de la détection d'intrusions, en exploitant les forces complémentaires de plusieurs algorithmes pour accroître la précision et la robustesse du système. Le travail s'articule autour de quatre grands axes.

Le premier chapitre introduit les notions fondamentales des systèmes de détection d'intrusions, en présentant leurs différentes architectures (NIDS, HIDS, hybrides), leurs méthodes de détection (signature, anomalies, hybrides), ainsi que leurs caractéristiques et les enjeux liés à leur efficacité.

Le deuxième chapitre se concentre sur le rôle de l'intelligence artificielle en cybersécurité. Il présente les concepts clés de l'IA, ses approches (IA faible et forte), ainsi que ses sous- domaines, notamment le machine learning et le deep learning, qui sont largement utilisés pour améliorer les performances des IDS modernes. Ce chapitre détaille également plusieurs algorithmes classiques et leur pertinence dans la détection d'intrusions.

Le troisième chapitre décrit l'approche innovante de coopération de modèles, basée sur l'intégration d'un réseau de neurones profond et d'un classificateur bayésien. Cette coopération vise à combiner les forces des deux modèles afin d'optimiser la détection des anomalies. Une explication détaillée des architectures, fonctions d'activation, algorithmes d'apprentissage, ainsi que des principes de la classification bayésienne est fournie.

Enfin, le quatrième chapitre présente la mise en œuvre pratique de ces modèles à travers des expériences menées sur le jeu de données NSL-KDD. Cette étape inclut la présentation des outils utilisés, comme Python et Google Colab, la description du protocole expérimental, ainsi que l'évaluation des performances des modèles à l'aide de

métriques rigoureuses telles que la précision, l'exactitude et le score F1.

Ce travail illustre ainsi l'importance de la coopération entre différents modèles d'intelligence artificielle pour relever les défis de la sécurité des réseaux et propose une contribution significative à l'amélioration des systèmes de détection d'intrusions, en vue de protéger efficacement les infrastructures critiques face aux menaces croissantes.

Chapitre 1

La sécurité informatique et détection d'intrusion

Introduction :

La sécurité informatique est l'une des pierres angulaires pour garantir l'intégrité des données et les protéger contre les menaces croissantes. Elle englobe un ensemble de mesures et de techniques visant à protéger les systèmes d'information contre les risques potentiels tels que les cyberattaques et les intrusions.

L'un des outils clés dans ce domaine est le système de détection d'intrusion (IDS), qui surveille les activités inhabituelles sur les réseaux ou les appareils et répond efficacement aux menaces. Grâce à l'intégration des technologies de l'intelligence artificielle, il est possible d'améliorer les performances de ces systèmes pour une détection précoce et réduire les faux positifs, contribuant ainsi à renforcer la sécurité de manière proactive.

Ce chapitre présente les principes fondamentaux de la sécurité informatique, l'évolution des systèmes de détection d'intrusion et l'utilisation de l'intelligence artificielle pour améliorer la protection des données.

I. Sécurité informatique :

I.1.Définition :

La sécurité informatique désigne l'ensemble des techniques mises en œuvre pour limiter les faiblesses des systèmes d'information face aux incidents, qu'ils soient intentionnels ou accidentels. Elle vise à préserver l'intégrité des systèmes, qu'ils soient internes ou externes, afin d'assurer leur fonctionnement optimal et la continuité des services.

Elle regroupe toutes les actions, méthodes et politiques permettant de sécuriser les réseaux, les systèmes informatiques et les données contre les différentes formes de menaces et d'attaques. [1]

I.2.Objectifs de la sécurité informatique:

Le système d'information se définit comme l'ensemble des données ainsi que des moyens matériels et logiciels mis à disposition par l'entreprise pour assurer la gestion, le stockage et la circulation de l'information. Il constitue un élément stratégique du patrimoine de l'entreprise, qu'il est impératif de préserver et de sécuriser.

La sécurité informatique, de manière générale, a pour but de garantir que les ressources informatiques, qu'elles soient matérielles ou logicielles, ne soient utilisées que conformément aux usages autorisés et prévus.

Elle repose principalement sur cinq objectifs fondamentaux :

- **L'intégrité** : assurer que les données sont exactes, complètes et non altérées.
- **La confidentialité** : veiller à ce que seules les personnes dûment autorisées puissent accéder aux informations sensibles.
- **La disponibilité** : garantir un accès continu et fiable au système d'information pour ses utilisateurs.
- **La non-répudiation** : permettre de prouver qu'une opération a bien été réalisée par une entité donnée, empêchant ainsi toute tentative de dénégation.
- **L'authentification** : s'assurer que toute personne accédant au système est bien celle qu'elle prétend être, en limitant l'accès aux seules personnes autorisées.[2]

I.3.Problèmes de la sécurité informatique :

La sécurité informatique est confrontée à trois types de problèmes majeurs : **les vulnérabilités**, **les menaces** et **les attaques**.

- **Les vulnérabilités** désignent des points faibles présents dans la spécification, la conception, le développement ou la configuration des systèmes informatiques. Leur exploitation peut permettre à un intrus de compromettre le système.[3]
- **Les menaces** représentent la possibilité qu'une propriété de la sécurité soit violée. Elles surviennent lorsqu'une ou plusieurs vulnérabilités sont exploitées, que ce soit de manière délibérée ou accidentelle.[4]
- **Les attaques** sont des actions malveillantes visant à tirer parti d'une faiblesse du système dans le but de porter atteinte à un ou plusieurs aspects de la sécurité.[5]

I.4.Les Attaques informatiques :

Les cyberattaques, qui consistent à tenter de compromettre des systèmes informatiques, sont réalisées par des individus ou des groupes motivés par des intérêts financiers, des idéologies politiques, ou simplement par curiosité. L'utilisation croissante des réseaux informatiques a mis en lumière diverses failles de sécurité, lesquelles sont souvent liées à des ressources budgétaires limitées, à un manque de temps, à une pénurie de personnel qualifié ou encore à des politiques de sécurité insuffisantes. Les conséquences financières engendrées par de telles attaques peuvent être importantes, ce qui pousse les organisations à renforcer leurs dispositifs de sécurité informatique.

Il est donc plus pertinent d'adopter une approche basée sur la prévention, la détection et la réaction face aux attaques, plutôt que de rechercher une sécurité absolue, souvent inatteignable. La prévention repose sur des méthodes telles que l'authentification, le chiffrement ou encore la dissimulation, afin de rendre les attaques plus difficiles. Toutefois, puisque la prévention ne peut pas tout empêcher, il est essentiel de mettre en place des méanismes de détection permettant d'identifier toute activité suspecte allant à l'encontre des politiques de sécurité établies.[6]

I.4.1.Schéma d'une attaque :

Une attaque informatique se définit comme une tentative de compromettre l'un des objectifs de la sécurité des systèmes d'information, tandis qu'une intrusion correspond à une attaque ayant abouti. Le déroulement d'une attaque peut être résumé en six étapes essentielles :[7]

- **Collecte d'informations** concernant le système ciblé.
- **Accès non autorisé au système** en exploitant les données précédemment recueillies.
- **Mise en place d'un dispositif** permettant un accès ultérieur, tel que l'ajout de code malveillant dans la mémoire EEPROM.
- **Propagation de l'intrusion** vers d'autres systèmes afin de faciliter des attaques de type distribué.
- **Mise hors service du système**, rendant son fonctionnement impossible ou fortement perturbé.
- **Suppression des traces laissées par l'attaquant** pour éviter toute détection.

Cette structure permet de mieux comprendre le processus d'une attaque et les mesures nécessaires pour y faire face.

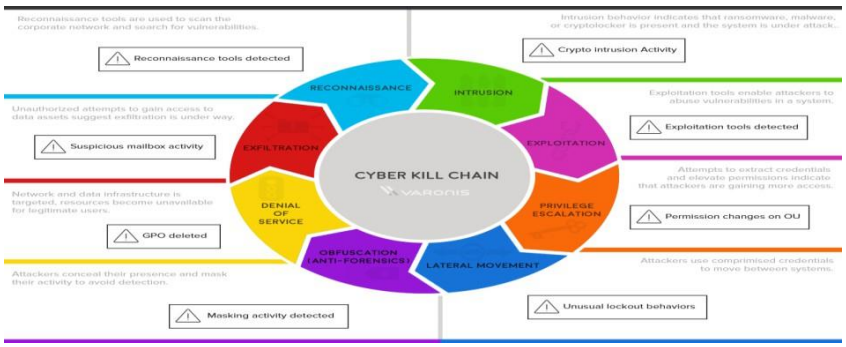


Figure1. 1 : Schéma d'une attaque [8]

I.4.2.Les types d'attaque :

Il existe principalement cinq catégories d'attaques informatiques :

I.4.2.1.Les attaques sur les réseaux :

Ce genre d'attaque exploite essentiellement les vulnérabilités présentes dans les protocoles de communication ou dans la manière dont ils sont implémentés. Le schéma suivant illustre de façon simplifiée les différents niveaux du réseau où des menaces potentielles en matière de sécurité peuvent survenir.

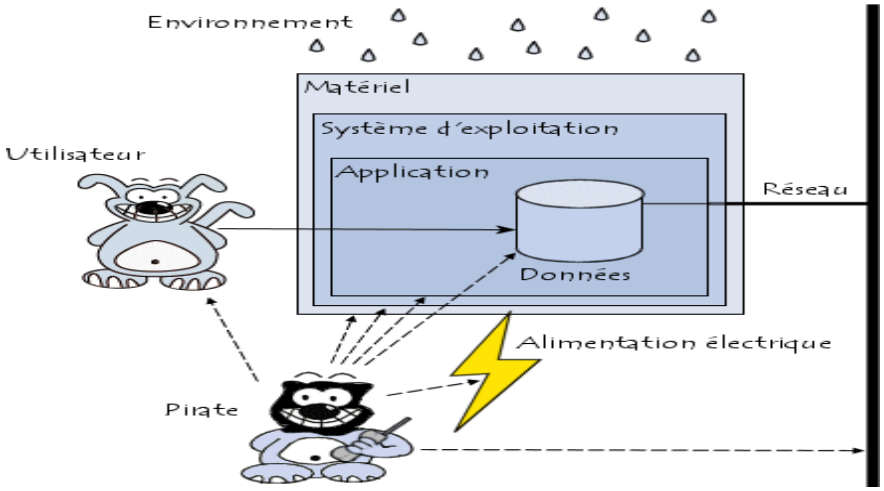


Figure 1.2 : schéma d'un attaque [9]

Nous allons décrire ci-après certaines attaques informatiques couramment rencontrées :

- **Techniques de scan :** Le scan de ports permet d'identifier les possibilités d'attaque sur une machine cible. Cette approche consiste à recueillir des données sur les systèmes analysés, notamment le système d'exploitation et les services en cours d'exécution, afin de détecter les vulnérabilités exploitables.

- **Usurpation d'adresse IP (IP Spoofing) :** Cette attaque consiste à manipuler l'adresse IP d'origine d'une requête pour simuler qu'elle provient d'un hôte autorisé. Une configuration correcte du routeur principal permet de limiter ce type d'intrusion.
- **DNS Spoofing :** Elle vise à duper un serveur DNS pour qu'il accepte des requêtes frauduleuses. Une méthode efficace pour s'en prémunir est de dissocier le serveur DNS interne de celui exposé à l'extérieur.
- **Flooding :** Le flooding utilise un programme appelé *flooder* pour bombarder une cible de requêtes (souvent des pings), dans le but de saturer sa connexion et provoquer sa

déconnexion . Chaque ping génère une réponse, ce qui peut rapidement submerger le système visé.

- **Smurf :** Cette attaque dite par réflexion repose sur l'exploitation d'adresses de diffusion (broadcast) pour submerger un réseau et le rendre inutilisable.
- **Web bug :** Il s'agit d'un message HTML contenant une image invisible. Lorsqu'il est ouvert, une requête est envoyée pour charger cette image, confirmant ainsi que le mail a été lu et validant l'adresse du destinataire.
- **Hoax (canular) :** Le hoax est une fausse alerte, souvent transmise par courrier électronique, et portant généralement sur des menaces de sécurité prétendument identifiées par des organismes reconnus. Ces messages peuvent perturber les réseaux d'entreprise. Il est donc essentiel d'en vérifier la véracité avant de les relayer.
- **Hacker :** Lors des débuts du réseau ARPA-net, une communauté de développeurs chevronnés et d'experts en réseaux partageait une même culture technique. C'est dans ce cadre qu'est né le terme "Hacker". Ces professionnels de l'informatique sont connus pour leur discrétion, leur opposition à l'autorité, ainsi que leur soif de découverte et de compréhension.
- **Cracker :** Les crackers sont des individus qui accèdent illégalement à des systèmes informatiques à distance. Leur but principal est souvent d'utiliser des logiciels payants sans en avoir acquis la licence, en utilisant des outils, souvent développés par d'autres, qu'ils trouvent en ligne.

I.4.2.2.Les attaques applicatives :

Les attaques ciblant les applications exploitent généralement des vulnérabilités présentes dans les logiciels utilisés ou résultant de mauvaises configurations. Comme pour d'autres types d'attaques, il est possible de les catégoriser selon leur origine.

- **Les erreurs de configuration :** Il est fréquent que les administrateurs réseau laissent les programmes avec leurs paramètres par défaut, sans les adapter aux exigences de sécurité. Ces paramètres sont souvent peu sécurisés pour simplifier l'usage initial du logiciel. Par ailleurs, une configuration incorrecte peut engendrer des failles sérieuses. Par exemple, une mauvaise configuration d'un serveur peut permettre l'accès à des fichiers sensibles ou compromettre la stabilité et la sécurité du système d'exploitation.
- **Les attaques par injection SQL :** Ce type d'attaque consiste à introduire du code SQL — le langage utilisé pour manipuler les bases de données — dans un champ de saisie prévu pour du texte. Le code malveillant est ensuite interprété et exécuté par l'application comme une

commande valide. Cela peut permettre à l'attaquant de contourner un écran de connexion, de consulter des données confidentielles, de modifier ou supprimer des informations dans la base de données, voire de lancer des commandes administratives directement sur celle-ci [5].

- **Man in the Middle (MITM) :** L'attaque de type « homme du milieu » désigne une situation où un individu malveillant s'intercale dans une communication entre un utilisateur et une application, dans le but soit d'écouter les échanges, soit de se faire passer pour l'un des interlocuteurs, tout en maintenant l'illusion d'une interaction normale.

L'objectif principal de ce type d'attaque est le vol d'informations sensibles comme les identifiants, les numéros de comptes ou de cartes bancaires. Les victimes visées sont généralement les utilisateurs de services financiers, les plateformes SaaS, les sites d'e-commerce ou tout site nécessitant une authentification.

Les données interceptées peuvent ensuite être exploitées à diverses fins, telles que l'usurpation d'identité, des virements frauduleux, ou encore la modification illégitime de mots de passe.

Par ailleurs, cette technique peut également servir à établir un point d'entrée dans un environnement sécurisé dans le cadre d'une attaque de type APT (Advanced Persistent Threat).

De manière imagée, une attaque MITM revient à ce qu'un facteur ouvre votre courrier bancaire, note vos informations confidentielles, referme l'enveloppe, puis la dépose comme si de rien n'était devant votre porte. [6]

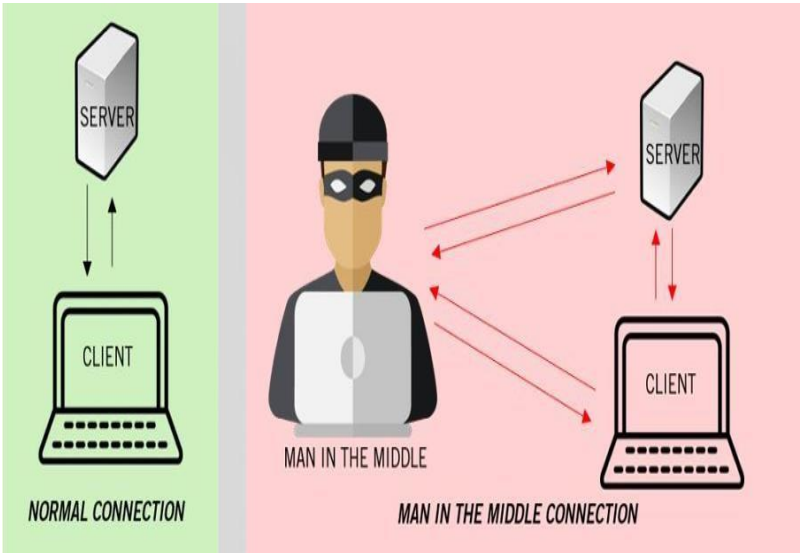


Figure 1.3 : man in the middle attack[24]

I.4.2.3.Le Déni de Service :

Comme mentionné plus haut, une attaque par déni de service (DoS) vise à rendre un service inaccessible. Cette attaque peut prendre différentes formes, comme une surcharge du réseau qui rend la machine cible injoignable, ou une attaque ciblant l'application elle-même, entraînant son plantage à distance.

Un exemple d'attaque consiste à exploiter un débordement de tampon (buffer overflow) pour faire planter une application. En injectant des instructions malveillantes dans le code d'une application vulnérable, un attaquant peut provoquer l'indisponibilité d'un service, que ce soit un serveur web, de messagerie ou un système entier.

Voici quelques exemples d'attaques réseau courantes qui rendent les services inaccessibles [4] :

- **SYN Flooding :** Cette attaque exploite le processus de connexion en trois étapes du protocole TCP (Three-Way Handshake : SYN/SYN-ACK/ACK). L'attaquant envoie un grand nombre de requêtes de connexion (SYN) sans répondre à l'accusé de réception (SYN-ACK). Ces connexions en attente occupent des ressources mémoire, ce qui entraîne une saturation du système et son éventuel effondrement.
- **UDP Flooding :** L'UDP étant plus prioritaire que le TCP, cette attaque consiste à envoyer un grand nombre de paquets UDP. Cette saturation de la bande passante empêche les connexions TCP d'être établies. Par exemple, l'attaquant envoie une requête sur le port 19 (service de génération de caractères) à une machine en usurpant l'adresse source, dirigeant ainsi les paquets vers un autre service comme le service echo, qui répète simplement les chaînes reçues.
- **Fragmentation de paquets :** Cette attaque repose sur une mauvaise gestion de la défragmentation des paquets IP au niveau du protocole ICMP. Un exemple connu de ce type d'attaque est le *ping of death*, où la taille des paquets envoyés dépasse la capacité maximale d'un paquet IP. Il est à noter que cette méthode peut aussi être utilisée pour contourner des filtres IP.
- **Smurfing :** L'attaquant envoie des requêtes ICMP ECHO à des adresses de diffusion (broadcast) en usurpant l'adresse source de la machine cible. Ainsi, la machine cible reçoit une quantité énorme de réponses provenant de toutes les machines du réseau, ce qui épuisera rapidement la bande passante de la cible.

I.4.2.4.Attaque par Déni de Service Distribué (DDoS) :

Dans ce type d'attaque, l'objectif est d'amplifier une attaque classique à une échelle bien plus vaste. Pour cela, le pirate va chercher à compromettre un grand nombre de machines. En exploitant des vulnérabilités comme les débordements de mémoire (buffer overflow) ou des failles dans les services à distance (comme RPC), il parvient à prendre le contrôle de ces machines à distance.

Une fois ce réseau de machines compromis en place, l'attaquant peut synchroniser une offensive en lançant simultanément les attaques depuis toutes ces machines. Cette action coordonnée transforme une attaque simple, telle qu'un **SYN Flooding**, en un assaut massif capable de paralyser complètement un système ou un réseau.

I.4.2.5.Attaques visant les données (contenu) :

Les informations véhiculées par les protocoles applicatifs peuvent elles-mêmes constituer une menace. En effet, certains contenus peuvent affecter l'intégrité du système destinataire. Les attaques de ce type incluent notamment les virus, vers, chevaux de Troie ou encore les applets Java, regroupés sous le terme général de **maliciels** ou **malwares**.

- **Virus :** Un virus informatique est un petit programme malveillant conçu pour se multiplier automatiquement. Il peut infecter un ordinateur à l'insu de l'utilisateur, sans son consentement. En pratique, un virus classique s'attache à un fichier exécutable. Chaque fois que ce fichier est lancé, le virus s'active et tente de contaminer d'autres fichiers exécutables.

Contrairement à une idée reçue, les virus ne naissent pas spontanément ; ils sont développés avec une intention bien précise. En plus de leur capacité à se propager, certains virus peuvent provoquer des effets variés : du simple affichage d'un message à la suppression complète de données.

- **Vers :** Les vers informatiques sont des programmes malveillants proches des virus, mais avec des spécificités propres. Bien qu'ils ne soient pas forcément capables de se propager d'eux-mêmes, leur objectif est souvent d'user les ressources du système : processeur, mémoire, espace disque ou bande passante.

Ce type de programme dépend du système d'exploitation ou d'un logiciel particulier. Il peut se transmettre via divers supports comme les disques amovibles, les CD-ROM ou les réseaux locaux (LAN) ou étendus (WAN). Bien que leur apparition ait diminué avec la montée en puissance des générateurs de virus, les vers existent toujours. Pour s'en protéger, on applique en général les mêmes mesures que pour les virus : logiciels antivirus, bonnes pratiques de sécurité, etc.

- **Le Cheval de Troie :** Le cheval de Troie est un programme qui se présente comme utile ou inoffensif, mais qui dissimule en réalité une activité nuisible. Le nom fait référence à la ruse mythologique utilisée pendant la guerre de Troie, où un grand cheval en bois offert en cadeau contenait des soldats ennemis cachés. Sur un ordinateur, ce type de programme peut sembler légitime, mais une fois exécuté, il peut entraîner des actions graves telles que l'effacement du disque dur, la récupération de mots de passe ou encore la transmission d'informations sensibles au pirate via Internet [7].
- **Bombes logiques :** Les bombes logiques sont des morceaux de code intégrés dans un logiciel ou un système, mais qui restent inactifs jusqu'à ce qu'un événement spécifique survienne. Ce déclencheur peut être une date, une action de l'utilisateur ou une condition particulière. Une fois activée, la bombe logique peut produire des effets destructeurs, affectant le fonctionnement normal du système [7].
- **Porte dérobée (Backdoor) :** Une porte dérobée est un mécanisme permettant à un tiers d'accéder à un système informatique sans passer par les procédures de sécurité classiques. Elle est souvent installée discrètement par des malwares comme les virus ou chevaux de Troie. Grâce à cette porte cachée, un attaquant peut se connecter à la machine ciblée à distance et y exécuter diverses commandes à son insu [8].
- **Logiciel espion :** Un **spyware** ou logiciel espion est un programme malveillant conçu pour surveiller l'activité d'un utilisateur à son insu. Il peut être installé sur un ordinateur ou un appareil mobile, et collecter des données personnelles telles que l'historique de navigation, les frappes clavier ou les identifiants de connexion. Ces informations sont ensuite transmises à des tiers, souvent par le biais d'Internet. La popularité croissante de ces logiciels est étroitement liée à l'expansion d'Internet [9].
- **Spam :** Le **spam**, également connu sous les termes de **courrier non désiré** ou **pourriel**, désigne toute forme de communication électronique envoyée sans le consentement du

destinataire, principalement par courrier électronique. Il s'agit le plus souvent de messages envoyés en masse, à des fins promotionnelles ou publicitaires[10].

I.5.Outils de sécurité :

Ce qui suit présente une sélection d'outils de sécurité, sans prétendre à l'exhaustivité :

I.5.1.Antivirus :

Un antivirus est un logiciel conçu pour prévenir, détecter, analyser et éliminer les virus informatiques. Une fois installé, il fonctionne généralement en arrière-plan afin d'assurer une protection continue contre les menaces virales.

Les suites antivirus complètes permettent de sécuriser aussi bien les fichiers que le matériel contre divers types de logiciels malveillants, tels que les vers, les chevaux de Troie ou les logiciels espions. En outre, elles peuvent intégrer des fonctionnalités supplémentaires comme des pare-feu personnalisables ou des outils de blocage de sites web malveillants.[11]

I.5.2.Pare-feu (Firewall) :

Un pare-feu, ou firewall, est un dispositif informatique (matériel, logiciel ou les deux) conçu pour sécuriser les données d'un réseau. Il peut s'agir, par exemple, de la protection d'un ordinateur personnel connecté à Internet ou de celle d'un réseau d'entreprise.

Ce dispositif garantit la sécurité des informations en filtrant le trafic entrant et en contrôlant le trafic sortant, selon des règles établies par l'administrateur.

Le pare-feu agit comme une passerelle filtrante, protégeant un ordinateur ou un réseau contre les intrusions provenant d'Internet. Il analyse les paquets de données échangés pour en autoriser ou bloquer le passage.

Il est parfois désigné par d'autres termes tels que coupe-feu, barrière de sécurité ou garde- barrière.

Un pare-feu dispose au minimum de deux interfaces : l'une reliée au réseau interne, l'autre au réseau externe.

Pour qu'un pare-feu soit efficacement intégré à un système, il est essentiel que :

- le système informatique soit sécurisé,
- le mécanisme de filtrage des paquets soit unique,
- la machine dispose de bonnes performances.[12]

Voici une figure qui montrer tout ça :

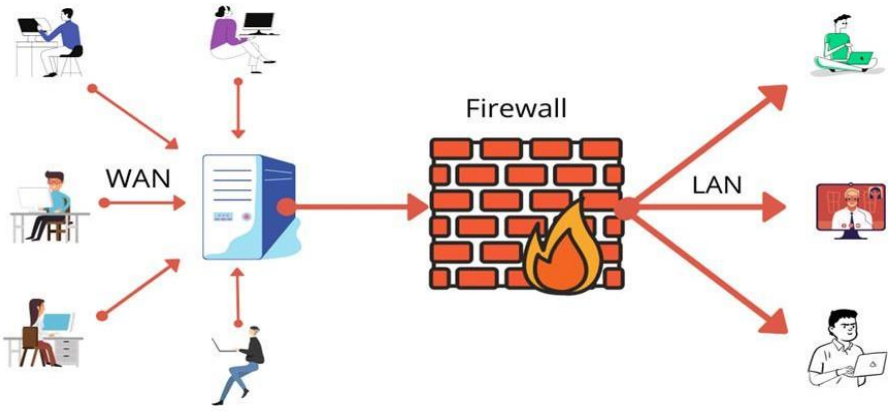


Figure 1.4 : Pare-feu (Firewall).[13]

I.5.3.Cryptographie :

La cryptographie est la discipline qui étudie les techniques permettant d'assurer la transmission sécurisée des données. Pour protéger un message, une transformation est appliquée afin de le rendre illisible ; cela s'appelle le chiffrement, qui convertit un texte lisible en un texte codé. À l'inverse, le déchiffrement consiste à récupérer le texte original à partir du texte codé, en utilisant un clé spécifique et un algorithme de décryptag.

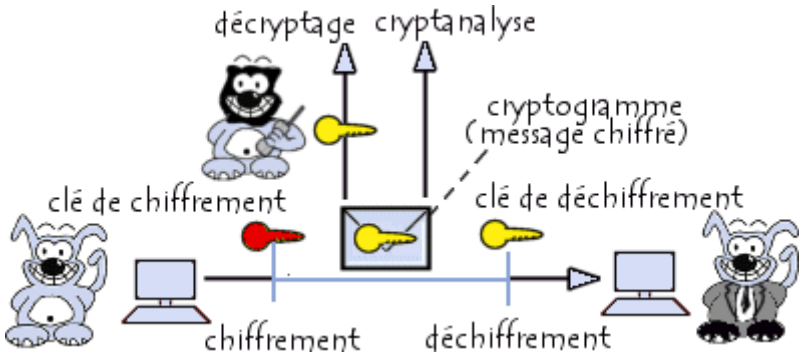


Figure1.5: la cryptographie [15]

I.5.4.VPN (Virtual Privat Network) :

Les réseaux privés virtuels, ou VPN (Virtual Private Network), offrent à l'utilisateur la possibilité d'établir une liaison sécurisée et virtuelle entre un point de départ et une destination.

Avec l'essor d'Internet, il devient essentiel de disposer de moyens fiables pour assurer un transfert de données sécurisé.

Le fonctionnement des VPN repose sur la technique dite du **tunneling**, qui consiste à créer un tunnel virtuel entre deux points clairement identifiés. Les données circulent à travers ce tunnel après avoir été chiffrées, garantissant ainsi leur confidentialité.

L'un des principaux avantages des VPN est la possibilité de mettre en place des réseaux privés à un coût réduit. En effet, le chiffrement des échanges donne l'illusion d'une communication en dehors d'Internet, tout en utilisant son infrastructure. Cependant, comme cette solution repose sur Internet, il est important de noter que la qualité de service (QoS) ne peut y être garantie.

Le principe fondamental du VPN repose donc sur le **tunneling**, qui consiste d'abord à identifier les deux extrémités de la communication (source et destination), puis à chiffrer les données avant de les envoyer via ce tunnel virtuel. Les informations transmises ne sont pas nécessairement de nature IP ; dans ce cas, le protocole de tunneling encapsule les données en y ajoutant un en-tête supplémentaire, permettant ainsi leur routage à travers le tunnel. Le **tunneling** regroupe l'ensemble des étapes d'encapsulation, de transmission, puis de désencapsulation des données.[16]

I.5.5.Système de Détection D'intrusion(IDS) :

Un système de détection d'intrusion (IDS : Intrusion Detection System) est un dispositif conçu pour identifier des comportements inhabituels ou suspects au niveau de la cible observée (réseau ou machine). Il permet ainsi de détecter aussi bien les tentatives d'intrusion abouties que celles ayant échoué.[17]

II. Détection d'intrusion:

II.1.Définition :

Un système de détection d'intrusion (IDS) est un dispositif destiné à identifier les comportements suspects ou anormaux affectant une cible spécifique (qu'il s'agisse d'un réseau ou d'un hôte), dans le but de traiter les incidents dans les plus brefs délais. Il permet de recueillir des informations concernant les tentatives d'intrusion, qu'elles soient couronnées de succès ou non. Grâce à leur importance opérationnelle, les IDS ont fait l'objet d'importantes recherches au cours des dernières années, visant à renforcer leur

performance. Ces travaux ont conduit à l'émergence de diverses catégories d'IDS, reposant sur plusieurs approches de détection, chacune étant optimisée pour un contexte particulier. Parmi ces catégories, on distingue les HIDS, qui prennent leurs décisions à partir des données collectées directement sur les hôtes, et les NIDS, qui fondent leurs analyses uniquement sur les flux d'informations circulant au sein des réseaux.[18]

II.2.Les types des IDS :

Les IDS représentent des logiciels ou des dispositifs élaborés pour surveiller les activités suspectes au sein d'un réseau ou d'un système informatique. Leur objectif est d'alerter les administrateurs en cas d'intrusions potentielles afin qu'ils puissent réagir rapidement pour les neutraliser. [19]

II.2.1.Systèmes de détection d'intrusions " réseaux " (NIDS):

Un système de détection d'intrusion réseau, ou NIDS (Network-based Intrusion Detection System), a pour mission principale d'examiner et d'interpréter les paquets circulant sur un réseau. L'installation d'un NIDS distant est généralement rigoureuse : des capteurs sont positionnés à des emplacements stratégiques du réseau, et génèrent des alertes lorsqu'ils détectent des paquets jugés dangereux. Ces alertes sont ensuite transmises à une console sécurisée pour analyse et éventuellement pour traitement.

Il est courant de trouver une architecture incluant une sonde située à l'extérieur du réseau pour observer les tentatives d'intrusion, ainsi qu'une interface dédiée à l'analyse des requêtes ayant franchi le pare-feu. Parmi les exemples notables de NIDS, on retrouve : NetRanger, NFR, Snort, DTK, ISS et RealSecure.

II.2.2.Systèmes de détection d'intrusions de type hôte (HIDS):

Les systèmes de détection d'intrusion basés sur l'hôte, connus sous le nom de HIDS (Host- based IDS), se concentrent sur l'analyse du fonctionnement interne des machines hôtes pour repérer les attaques dirigées contre des processus ou services actifs (démons). Contrairement aux NIDS, ils n'inspectent pas directement le trafic réseau mais uniquement les activités internes de l'hôte, ce qui leur confère une meilleure précision dans la détection de différents types d'attaques.

Ces systèmes s'appuient sur deux sources principales pour surveiller les activités : les fichiers journaux et les pistes d'audit du système d'exploitation. Parmi les HIDS les plus

connus, on peut citer :

- Tripwire, Tiger,
- WATCHER, un gestionnaire de la sécurité,
- Dragon Squire.

II.2.3.Systèmes de détection d'intrusions " hybrides ":

Les systèmes de détection d'intrusion hybrides (associant NIDS et HIDS) intègrent les fonctionnalités de plusieurs types d'IDS. En pratique, ce sont essentiellement les combinaisons de NIDS et de HIDS qui sont mises en œuvre. Ces systèmes permettent une surveillance simultanée du réseau et des hôtes finaux à l'aide d'une solution unique. Des sondes sont installées à des emplacements clés et agissent comme des NIDS et/ou des HIDS selon leur positionnement. Toutes ces sondes transmettent ensuite leurs alertes vers une plateforme centrale, qui se charge de regrouper et de corréler les informations issues des différentes sources.

II.3.Architecture fonctionnelle des(IDSs):

Cette partie présente les trois éléments fondamentaux qui composent traditionnellement un système de détection d'intrusions. Les interactions entre ces composants sont représentées dans la Figure 1.6 [20].

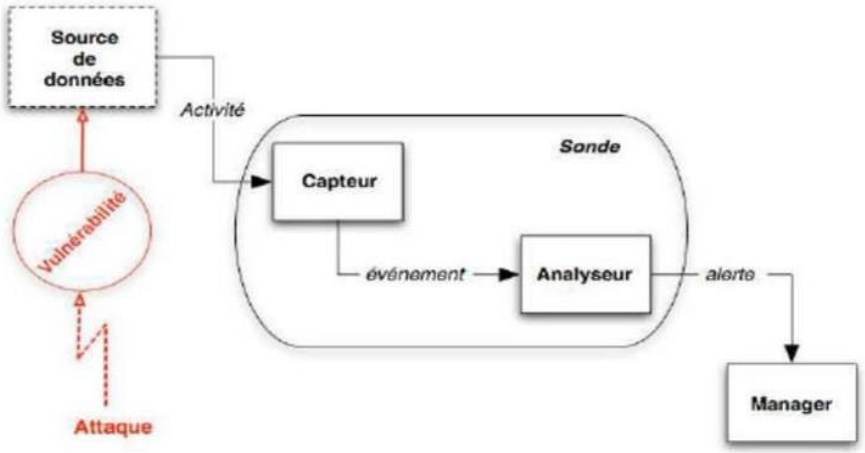


Figure1. 6: L'architecture d'un IDS [21]

II.3.1.Capteur : Le capteur est chargé de surveiller l'activité du système en s'appuyant sur une source de données. Il transmet à l'analyseur une suite d'événements décrivant l'évolution de l'état du système. Dans certains cas, ces données sont envoyées telles quelles, mais généralement, un prétraitement est réalisé avant leur transfert. Selon la nature des sources observées, on distingue principalement trois catégories de capteurs : les capteurs système, les capteurs réseau et les capteurs applicatifs.

II.3.2.Analyseur : La mission principale de l'analyseur est d'examiner le flux d'événements transmis par le capteur afin de détecter d'éventuels signes d'activités malveillantes ou anormales.

II.3.3.Manager : Le manager centralise les alertes générées par l'analyseur, les organise et les présente à l'opérateur. De plus, il peut être en charge de déclencher des réponses appropriées face aux incidents détectés, telles que :

- **Le confinement**, visant à limiter les impacts de l'attaque ;
- **L'éradication**, consistant à neutraliser la menace ;
- **Le recouvrement**, qui permet de restaurer le système dans un état fonctionnel ;
- **Le diagnostic**, ayant pour but d'identifier précisément la nature du problème.

II.4.Caractéristiques d'un système de détection d'intrusion :

Un système de détection d'intrusion efficace doit présenter plusieurs qualités essentielles, parmi lesquelles : [20]

- Être capable de résister aux tentatives de compromission, en détectant toute modification non autorisée qui pourrait l'affecter.
- Consommer le moins de ressources possible du système surveillé afin de ne pas perturber son fonctionnement.
- S'ajuster progressivement aux évolutions du système supervisé ainsi qu'aux modifications dans les comportements des utilisateurs.
- Offrir une grande flexibilité de configuration pour permettre l'application de politiques de sécurité spécifiques à chaque réseau.

II.5.Classification des systèmes de détection d'intrusions :

Dans cette section, nous présentons la technologie de détection d'intrusion selon une approche taxonomique. Il existe aujourd'hui plusieurs types de systèmes de détection d'intrusion (IDS), chacun se distinguant par sa méthode de surveillance et d'analyse. Chaque approche présente ses propres avantages et inconvénients. Ces différentes

approches peuvent être regroupées dans un modèle général d'IDS (voir la figure 1.8), basé sur plusieurs critères :[22]

- L'emplacement du système IDS
- Les méthodes de détection utilisées
- es possibles
- La fréquence d'utilisation

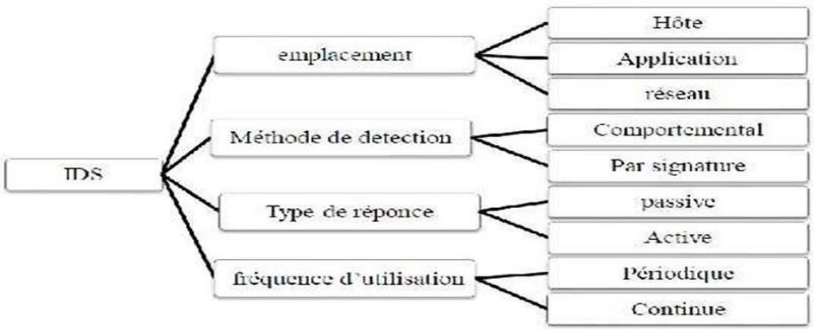


Figure1.7: Classification des IDS.

II.5.1.Emplacement d'un IDS :

Les systèmes de détection d'intrusions peuvent être répartis en trois grandes catégories selon leur emplacement :

- **HIDS (Host-based Intrusion Detection Systems)** : Ces systèmes sont installés directement sur une machine spécifique. Ils agissent comme des agents qui examinent les activités internes ou externes susceptibles de contourner les règles de sécurité définies du système.
- **NIDS (Network-based Intrusion Detection Systems)** : Ces dispositifs surveillent le trafic réseau en utilisant des sondes placées sur le segment à analyser. Ils détectent les anomalies en comparant les flux à un modèle de comportement normal.
- **Systèmes hybrides (HIDS + NIDS)** : Ces solutions associent les fonctions des IDS hôtes et réseau. Elles permettent une surveillance simultanée des postes et du réseau.

Les sondes, déployées à des emplacements stratégiques, peuvent agir comme HIDS, NIDS, ou les deux. Les alertes générées sont ensuite centralisées pour une corrélation et une analyse approfondies sur un système dédié.

II.5.2.Les Méthodes de détection :

Il existe deux principales méthodes utilisées pour la détection des intrusions :

- **L'approche comportementale** consiste à établir un modèle du comportement normal des utilisateurs, du système et du trafic réseau. Toute anomalie ou déviation par rapport à ce comportement de référence est alors considérée comme suspecte.
- **L'approche par signature** repose, quant à elle, sur un modèle basé sur les actions interdites dans le système informatique. Elle se fonde sur une connaissance éalable s techniques d'attaque utilisées. Des scénarios d'attaques sont définis et comparés aux traces d'audit pour identifier d'éventuelles correspondances.

II.5.2.1.L'approche par scénario (ou signature) :

Dans cette méthode, les détecteurs utilisent une base de données contenant des motifs d'attaques connus. Cette base sert à analyser en temps réel les informations recueillies par les sondes. Il s'agit d'un système de reconnaissance de motifs permettant d'identifier les intrusions en se basant sur ces signatures. L'efficacité dépend en grande partie de la qualité et de la précision de cette base.

Des signatures plus générales peuvent être élaborées afin de repérer des variantes d'une même attaque. Cela exige cependant une bonne compréhension du réseau et des attaques, pour éviter les faux positifs qui pourraient affecter le trafic légitime. Une signature peut décrire une attaque soit au niveau des paquets (jusqu'au niveau TCP ou UDP), soit au niveau des protocoles comme HTTP ou FTP.

Au niveau des paquets, l'IDS inspecte les paramètres de chaque paquet en transit et les compare aux signatures connues. Pour les protocoles, l'IDS vérifie la validité des commandes échangées afin de détecter d'éventuelles attaques, en particulier dans le cas du protocole HTTP. Toutefois, plus le nombre de signatures à analyser est élevé, plus le traitement est lent. L'usage de signatures optimisées permet donc de réduire le temps d'analyse.[23]

II.5.2.2.L'approche comportementale :

Les systèmes de détection d'intrusions basés sur le comportement reposent sur l'élaboration d'un modèle de référence représentant le comportement normal de l'entité surveillée.

Ce modèle est ensuite utilisé lors de la phase de détection pour comparer le comportement actuel à celui considéré comme normal. Lorsqu'un écart significatif, dépassant un certain seuil, est observé, une alerte est déclenchée. Cette approche part du principe que toute déviation par rapport au comportement habituel est considérée comme une anomalie, susceptible d'indiquer une intrusion ou une tentative d'intrusion.[24]

II.5.3.Types de réponses :

Une autre manière de catégoriser les systèmes de détection d'intrusions repose sur le type de réaction adoptée lors de l'identification d'une attaque. On distingue principalement deux approches :

- **Réaction passive :** Un système IDS fonctionnant de manière passive se limite à observer et analyser le trafic réseau, sans intervenir directement. Lorsqu'il détecte une activité suspecte ou une attaque potentielle, il envoie une alerte à un administrateur – que ce soit par e- mail, via la console système ou un dispositif d'alerte sonore.

La prise de décision et les mesures à appliquer sont alors laissées à l'appréciation de l'opérateur humain. Ce type de système présente l'avantage d'être simple à mettre en place et rapide à déployer, tout en n'imposant aucune modification majeure à l'infrastructure réseau.

- **Réaction active :** Contrairement au mode passif, une réponse active consiste à intervenir immédiatement dès qu'une attaque est identifiée, dans le but de l'arrêter ou d'en limiter les effets. Deux méthodes principales sont généralement utilisées à cet effet :

- **Reconfiguration du pare-feu :** Cette méthode bloque le trafic malveillant en fermant le port utilisé par l'attaquant ou en interdisant son adresse IP via le pare-feu. Toutefois, cette capacité dépend du modèle de pare-feu utilisé, car tous ne permettent pas une reconfiguration dynamique contrôlée par un IDS.

- **Interruption de la connexion TCP :** Dans ce cas, le système envoie un paquet TCP comportant le drapeau RST aux deux parties impliquées dans la connexion, simulant ainsi une déconnexion soudaine. Cela met fin à la session en cours et interrompt l'activité malveillante.

Lorsqu'une réaction active est envisagée, il est essentiel de vérifier que le trafic en question constitue effectivement une menace, afin d'éviter de perturber des utilisateurs légitimes.

Généralement, ces actions automatiques ne sont déclenchées que si l'alerte est validée comme étant une véritable attaque. L'analyse des journaux d'alerte générés par le système est donc cruciale pour comprendre la nature des menaces détectées.

Cependant, une automatisation excessive peut s'avérer risquée. Par exemple, un attaquant pourrait tromper le système en usurpant une adresse IP appartenant au réseau interne, ce qui pourrait conduire le système à bloquer un utilisateur légitime. Pour cette raison, il est recommandé de laisser la décision finale à un opérateur humain, notamment dans les contextes sensibles.

Enfin, un système de détection d'intrusion peut être exploité en mode continu (en ligne, temps réel) ou de manière périodique (hors ligne), selon les besoins et la configuration du réseau.[25]

II.5.4.Fréquence d'utilisation :

Il existe deux approches principales pour mesurer et évaluer la fréquence d'utilisation des systèmes de détection d'intrusions : **la surveillance en continu** et **l'analyse à intervalles réguliers**. [26]

- **Surveillance périodique** : Cette méthode consiste à effectuer une observation du système à des moments précis, planifiés à l'avance. L'administrateur procède à des analyses régulières pour identifier d'éventuelles intrusions ou anomalies ayant pu se produire dans le passé. Il s'agit donc d'un contrôle rétrospectif, effectué à intervalles définis.
- **Surveillance continue** : Contrairement à l'approche périodique, cette stratégie repose sur une observation permanente et en temps réel du système. Elle permet de détecter immédiatement toute activité suspecte, offrant ainsi une réactivité optimale face aux menaces en cours.

Conclusion :

En conclusion, la sécurité informatique est essentielle pour protéger les données à l'ère numérique. Grâce aux systèmes de détection d'intrusion (IDS) et à l'utilisation de l'intelligence artificielle pour améliorer l'efficacité et la précision, il est possible d'assurer une protection efficace contre les cyberattaques. Cette évolution reflète le besoin constant de mettre à jour les stratégies de sécurité et de relever les défis modernes pour garantir l'intégrité des informations.

Chapitre 2
Apprentissage automatique

Introduction :

Le domaine de la cybersécurité a connu une évolution remarquable ces dernières années, ce qui a suscité un intérêt croissant pour le développement de méthodes efficaces de détection des attaques informatiques. Parmi ces méthodes, les techniques d'apprentissage automatique (Machine Learning) se sont imposées comme des outils puissants capables d'analyser de vastes volumes de données afin d'identifier des schémas ou des comportements inhabituels pouvant indiquer la présence d'activités malveillantes au sein des systèmes ou des réseaux.

Ce chapitre explore les concepts et approches liés à l'intelligence artificielle, en mettant particulièrement l'accent sur les techniques d'apprentissage automatique et leur application dans le domaine de la détection d'intrusions. Il met en lumière leur efficacité et leur importance dans le renforcement de la sécurité des systèmes d'information face aux menaces croissantes.

I. L'intelligence artificielle (IA) :

I.1.Définition de l'intelligence artificielle (IA) :

Selon le dictionnaire Larousse, l'intelligence artificielle se définit comme « un ensemble de théories et de techniques visant à créer des machines capables d'imiter l'intelligence humaine ». Autrement dit, il s'agit d'ordinateurs ou d'appareils munis de programmes capables d'accomplir des tâches similaires à celles de l'esprit humain, voire de les surpasser grâce aux avancées technologiques.

Ces machines sont capables de :

- raisonner logiquement,
- traiter d'importants volumes de données,
- détecter des schémas imperceptibles à l'œil humain,
- comprendre et analyser ces schémas,
- interagir avec les êtres humains,
- apprendre de manière progressive,
- et améliorer constamment leurs performances [27].

Depuis sa création en 1950, l'intelligence artificielle n'a cessé d'évoluer. Elle aurait même, en janvier 2018, atteint un stade où elle pourrait dépasser l'intelligence humaine.

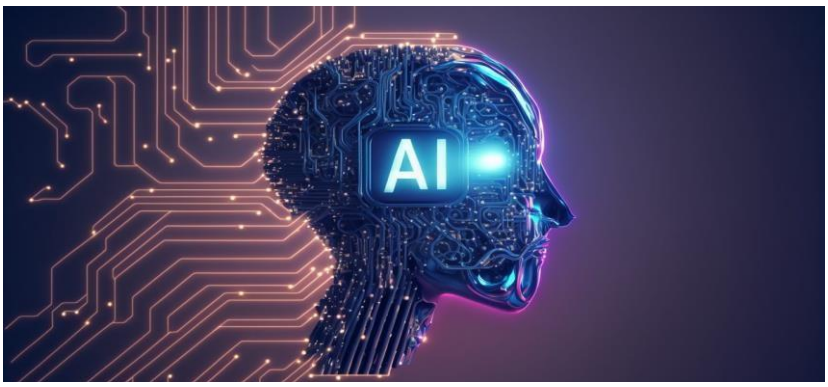


Figure 2.1: Intelligence Artificiel AI [28]

I.2.Comment fonctionne l'Intelligence Artificielle ?

Les machines équipées d'intelligence artificielle enregistrent des comportements. Cette capacité de mémorisation leur permet par la suite de résoudre des problèmes et de réagir de manière adéquate selon les situations rencontrées. Cet apprentissage s'appuie sur des bases de données et des algorithmes. Grâce à ce processus complexe, la machine peut évaluer l'importance d'un problème, examiner en détail les solutions envisageables ainsi que des cas similaires rencontrés auparavant afin d'agir de façon appropriée.

En réalité, il s'agit d'un système statistique avancé et très performant qui guide la machine dans la prise de décision ou l'adoption du comportement attendu. Pour évaluer son niveau d'intelligence, la machine est soumise au test de Turing. Ce test porte le nom de son créateur, Alan Turing, mathématicien britannique qui, dès 1950, s'est interrogé sur la capacité d'une machine à penser. Le test consiste à dialoguer avec la machine et à lui demander de produire quelque chose en respectant des critères bien définis [29].

I.3.Approches de l'intelligence artificielle :

L'intelligence artificielle forte représente la perspective future de l'IA, visant à concevoir des machines autonomes capables de conscience. À l'inverse, l'IA faible est dédiée à l'exécution de tâches spécifiques. Elle est capable d'effectuer des calculs, d'analyser de grandes quantités de données, de résoudre des problèmes ciblés et d'apprendre automatiquement. Cependant, elle ne possède pas de conscience propre et agit selon les instructions préétablies par les humains. De nos jours, l'intelligence artificielle que l'on retrouve dans les applications et dispositifs technologiques relève du domaine de l'IA faible. En revanche, l'IA forte demeure un véritable

défi, et sans doute l'ambition la plus grande dans le développement de l'intelligence artificielle [30].

I.4.Les sous-domaines de l'intelligence artificielle :

L'intelligence artificielle constitue un domaine d'étude étendu, regroupant un large éventail de théories, de méthodes et de technologies. Elle se divise en plusieurs branches, parmi lesquelles : [31]

I.4.1.L'apprentissage profond (Deep Learning) : Il s'agit d'une spécialisation du domaine de l'apprentissage automatique (Machine Learning), lui-même sous-ensemble de l'intelligence artificielle. Le Deep Learning repose sur l'utilisation de réseaux neuronaux artificiels composés de plusieurs couches – d'entrée, cachées et de sortie. Ces couches sont constituées d'unités qui transforment les données reçues en informations exploitables par la couche suivante, dans le cadre d'une tâche prédictive donnée. Cette architecture permet à la machine d'apprendre de manière autonome à partir des données qu'elle traite elle-même.

I.4.2.L'apprentissage automatique (Machine Learning) : Ce sous- domaine de l'intelligence artificielle regroupe un ensemble de techniques, comme le Deep Learning, permettant aux machines d'apprendre à partir de leurs expériences passées, afin d'optimiser l'exécution de leurs tâches.

I.4.3.L'informatique cognitive : Il s'agit d'un champ de l'intelligence artificielle qui cherche à établir une interaction plus naturelle, proche de celle des humains, entre les machines et les utilisateurs. L'objectif est de fusionner les capacités de l'IA avec celles de l'informatique cognitive pour reproduire les mécanismes cognitifs humains, notamment par l'interprétation des images et de la parole, tout en produisant une réponse verbale cohérente.

I.4.4.Le traitement du langage naturel : Cette discipline vise à doter les ordinateurs de la capacité à analyser, comprendre et générer le langage humain, en particulier sous sa forme orale. À un stade avancé, elle permet une interaction fluide en langage naturel, rendant possible la communication avec les machines dans une langue courante, telle qu'elle est utilisée au quotidien, pour accomplir diverses tâches.

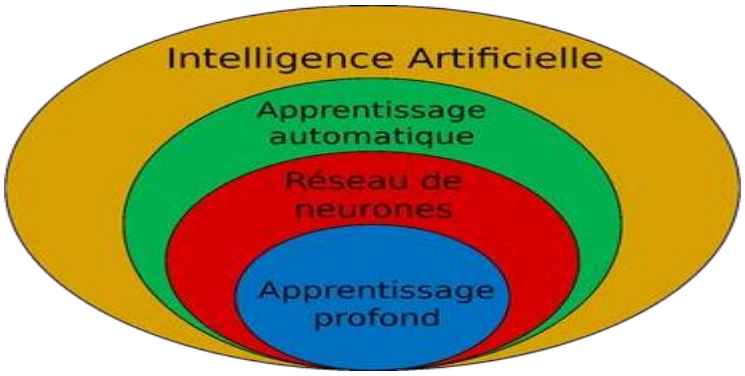


Figure 2. 2: L'intelligence artificielle et ses sous-domaines [32]

I.5.Les enjeux de l'IA dans la cybersécurité :

Les avancées en cybersécurité ont connu une évolution remarquable grâce à l'intégration croissante de l'intelligence artificielle (IA), notamment à travers l'utilisation des techniques de machine learning, de deep learning, ainsi que l'analyse de données et les approches statistiques. Ces méthodes, issues de disciplines variées, permettent de relever les défis complexes de la cybersécurité en tirant parti des volumes massifs de données générées par les réseaux, les systèmes d'exploitation et les infrastructures informationnelles.

La machine learning et le deep learning sont particulièrement efficaces lorsqu'ils sont appliqués dans les réseaux SDN pour identifier les anomalies. Ces technologies permettent de classer, détecter et anticiper les cyberattaques en reconnaissant des schémas de comportement typiques liés à divers types de menaces. Grâce à leur capacité à détecter les attaques en temps réel et à anticiper celles à venir, elles apportent une réponse proactive aux menaces informatiques.

Les techniques basées sur l'apprentissage profond apparaissent comme particulièrement prometteuses pour optimiser les systèmes de détection d'intrusion (IDS), en relevant les défis liés à leur conception et à leur évolution. Elles sont aussi capables de traiter efficacement les méga données issues du trafic réseau et des opérations de collecte d'informations, tout en améliorant les capacités de détection des anomalies et en réduisant le taux de fausses alertes.

En somme, le recours aux technologies de machine learning et de deep learning dans le domaine de la cybersécurité constitue une avancée majeure, en permettant une protection enforcée des systèmes informatiques et une détection anticipée et intelligente des cyber menaces .[33]

II.Apprentissage automatique :

II.1.Définition de Apprentissage automatique :

Le machine learning, ou apprentissage automatique, est une branche de l'intelligence artificielle qui cherche à doter les machines de la capacité d'apprendre à partir de données, en s'appuyant sur des modèles mathématiques. Concrètement, cela consiste à extraire des informations significatives à partir d'un ensemble de données utilisé pour l'entraînement.

L'objectif de cette étape est d'ajuster les paramètres d'un modèle de manière à optimiser ses performances, en particulier lors de l'exécution de la tâche pour laquelle il a été conçu.

Une fois la phase d'apprentissage achevée, le modèle est prêt à être utilisé en environnement réel, dit de production.[34]

II.2.Les types de l'apprentissage automatique :

Dans le cadre de notre étude, nous porterons une attention particulière aux principales catégories d'apprentissage, à savoir l'apprentissage supervisé et l'apprentissage non supervisé.

II.2.1.Apprentissage supervisé :

Bien que les deux types d'apprentissage s'inscrivent dans le domaine de l'intelligence artificielle, l'apprentissage supervisé se distingue par l'intervention d'un expert humain qui oriente l'algorithme dans sa phase d'apprentissage. Cela se fait par la fourniture d'exemples annotés, c'est-à-dire accompagnés des résultats attendus. L'intelligence artificielle apprend à partir de ces exemples en ajustant progressivement ses paramètres internes (notamment les poids des neurones) afin de réduire l'écart entre les prédictions générées et les valeurs réelles. À mesure que l'entraînement progresse, l'erreur diminue, avec pour objectif final la capacité à généraliser les connaissances acquises à de nouvelles données. C'est d'ailleurs la méthode la plus couramment utilisée par les spécialistes du machine learning.

L'apprentissage supervisé repose sur l'association entre des variables d'entrée (X) et une variable de sortie (Y). À l'aide d'un algorithme, on cherche à modéliser une fonction qui permet de faire correspondre une entrée à une sortie, autrement dit :**Y = f(X)**.

L'objectif est de modéliser cette fonction de façon suffisamment précise pour pouvoir prédire la sortie (Y) correspondant à toute nouvelle entrée (X). Ce processus est dit « supervisé » car il repose sur des données étiquetées et peut être comparé à un enseignement encadré, où l'algorithme apprend grâce à des corrections successives en fonction des erreurs qu'il commet. L'apprentissage se poursuit jusqu'à ce que l'algorithme atteigne un niveau de performance jugé satisfaisant.

Les problèmes d'apprentissage supervisé peuvent être classés en deux grandes catégories : la **classification** et la **régression**.

- **Classification** : Ce type de problème apparaît lorsque la variable de sortie correspond à une catégorie ou un label, comme par exemple « rouge » ou « bleu ».
- **Régression** : Ce cas se présente lorsque la sortie attendue est une valeur continue, comme un prix en dollars ou un poids.

Il convient également de noter que certains problèmes liés à la prévision, notamment ceux portant sur les séries temporelles, peuvent être abordés sous l'angle de la classification ou de la régression.

Voici quelques exemples courants d'algorithmes utilisés dans le cadre de l'apprentissage supervisé :

- La **régression linéaire**, souvent utilisée pour les tâches de régression.
- La **forêt aléatoire** (Random Forest), adaptée aussi bien à la classification qu'à la régression.
- Les **machines à vecteurs de support** (SVM), principalement utilisées pour les problèmes de classification.[35]

II.2.2.Apprentissage non supervisé :

Dans le cadre de l'apprentissage non supervisé, la machine apprend de manière entièrement autonome, sans qu'aucune indication explicite ne lui soit donnée sur les résultats attendus. Autrement dit, les données sont fournies à l'algorithme sans étiquettes ni sorties de référence.

Bien que ce type d'apprentissage puisse sembler avantageux en théorie — notamment parce qu'il ne nécessite pas de grands ensembles de données annotées — il convient de

noter que l'apprentissage non supervisé et l'apprentissage supervisé répondent à des objectifs distincts et s'appliquent à des contextes différents.

Dans un apprentissage non supervisé, seules les variables d'entrée (X) sont disponibles, sans correspondance avec des valeurs de sortie (Y). L'objectif principal est alors d'explorer la structure cachée ou la distribution sous-jacente des données, afin d'en extraire des connaissances utiles.

On parle d'« apprentissage non supervisé » car, contrairement au modèle supervisé, il n'existe ici ni solution correcte prédéfinie, ni supervision humaine. L'algorithme agit de manière indépendante pour identifier les motifs récurrents, les relations ou les groupements naturels présents dans les données.

Les tâches relevant de l'apprentissage non supervisé se divisent généralement en deux grandes catégories : **le regroupement (clustering)** et **l'association**.

- **Regroupement (clustering)** : Il s'agit d'identifier des structures de groupe naturelles au sein des données. Par exemple, segmenter des clients selon leurs comportements d'achat.
- **Association** : Ce type de problème vise à découvrir des règles qui décrivent les relations fréquentes entre les éléments d'un jeu de données, comme « les clients qui achètent X achètent souvent aussi Y ».

Parmi les algorithmes les plus utilisés en apprentissage non supervisé, on peut citer :

- **L'algorithme des k-moyennes (k-means)**, utilisé pour les tâches de regroupement.
- **L'algorithme A-priori**, appliqué à l'extraction de règles d'association.[35]

- **Table de comparaison :**

	Apprentissage supervisé	**Apprentissage non Supervisé**
Données d'entrée	Utilise les données connues et étiquetées comme entrées.	Données inconnues en entrée.
Complexité informatique	Très complexe.	Moins de complexité informatique.
Temps réél	Utilise l'analyse hors ligne.	Utilise l'analyse en temps réel des données.
Sous-domaines	Classification et régression.	Exploitation de règles de clustering et d'association.
Précision	Produit des résultats précis.	Génère des résultats modérés.
Nombre de classes	Nombre de classes connues.	Le nombre de classes n'est pas connu.

Tableau 2.1 : comparaison entre Apprentissage supervisé et non supervisé.[35]

II.2.3.Apprentissage par renforcement :

Couramment employé dans des contextes où un agent intelligent, tel qu'un véhicule autonome, doit évoluer dans un environnement donné, tout en recevant un retour différé sur la qualité de ses actions, qu'elles soient bonnes ou mauvaises. Ce type d'apprentissage est également utilisé dans les jeux où l'issue ne peut être déterminée qu'à la fin de la partie.[36]

II. IDS parles méthodes d'Apprentissage:

Voici une liste, non exhaustive, des algorithmes d'apprentissage susceptibles d'être utilisés dans la conception de systèmes de détection d'intrusion (IDS).

III.1.Arbre de décision :

Cette méthode aborde le problème de classification en le décomposant en plusieurs sous- problèmes. Elle repose sur la construction d'un arbre de décision, lequel sert ensuite à établir un modèle utilisé pour la tâche de classification. Le processus de classification s'effectue de
manière relativement intuitive : il suffit de suivre les réponses aux différentes questions en parcourant les branches de l'arbre.

Les arbres de décision sont des modèles non paramétriques capables d'extraire des

règles généralement puissantes. Ils sont adaptés au traitement de grands ensembles de données et peuvent manipuler des variables de nature mixte, qu'elles soient catégorielles ou numériques. Les variables redondantes sont automatiquement écartées, car l'arbre ne les sélectionne tout simplement pas.

À chaque nœud de l'arbre, deux décisions sont à prendre : choisir la variable (feature) à utiliser et déterminer le point optimal de séparation entre les classes, en minimisant soit l'erreur quadratique (en cas de régression), soit l'impureté (en cas de classification). L'arbre est ainsi développé jusqu'à sa taille maximale.

Les forêts aléatoires consistent en un ensemble d'arbres de décision construits séparément, chacun étant légèrement différent des autres. Pour prédire une nouvelle donnée, chaque arbre de la forêt fournit une classification, et la prédiction finale correspond à la valeur ayant obtenu le plus de votes parmi l'ensemble des arbres.

En résumé, les forêts aléatoires tirent leur nom du caractère aléatoire introduit lors de la génération des arbres. Cela se fait d'abord par la création d'échantillons aléatoires (bootstrap), puis par la sélection aléatoire des variables candidates à chaque division.[37]

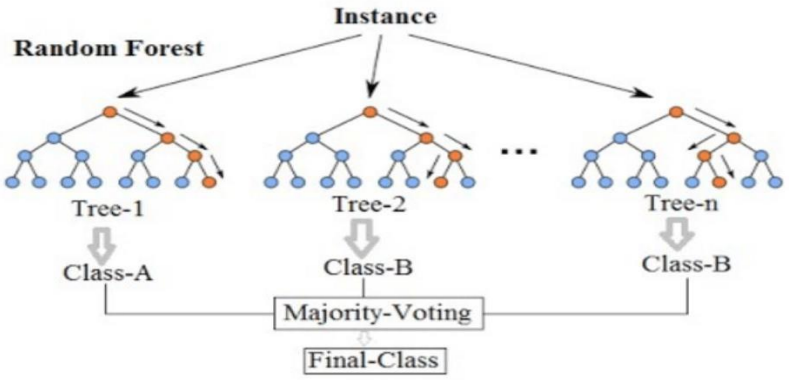


Figure 2.3: algorithme des arbres de décision [38].

III.2.L'algorithme des K plus proche voisin (KNN):

La méthode des k plus proches voisins, aussi appelée k-NN (pour *k-Nearest Neighbors*) ou k-PPV en français, repose sur un principe simple : pour classer un nouvel individu, on recherche les *k* individus les plus similaires parmi ceux déjà étiquetés, en fonction d'une mesure de distance

— la distance euclidienne étant la plus couramment utilisée. L'individu à classer est ensuite attribué à la classe la plus représentée parmi ces *k* voisins.

Cette approche, supervisée et non paramétrique, est réputée pour son efficacité dans de nombreux cas. Son apprentissage est trivial, car il se base sur la mémorisation directe des données. En revanche, le processus de prédiction peut s'avérer coûteux en temps, car il requiert de comparer le nouvel individu à l'ensemble des exemples d'apprentissage. Néanmoins, des techniques existent pour limiter ce nombre de comparaisons et améliorer l'efficacité.[39]

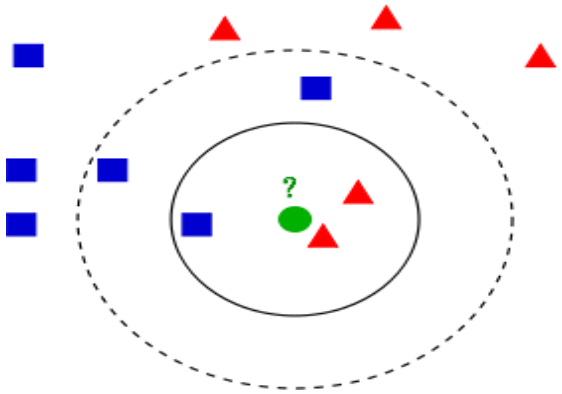


Figure 2.4: Algorithme KNN [40].

III.3.Naïve bayes :

Le classificateur Naïve Bayes repose sur le théorème de Bayes et fait l'hypothèse que les prédicteurs sont indépendants les uns des autres. Autrement dit, il considère que la présence d'une caractéristique donnée dans une classe est indépendante de la présence d'une autre. Par exemple, on peut identifier un fruit comme étant une pomme s'il est rouge, de forme ronde et mesure environ trois pouces de diamètre. Bien que ces attributs puissent être corrélés ou influencés par d'autres facteurs, le modèle les traite comme s'ils contribuaient séparément à la probabilité que le fruit soit une pomme. C'est cette supposition d'indépendance qui lui vaut l'appellation de « naïf ».

Le modèle Naïve Bayes présente l'avantage d'être simple à mettre en œuvre et se montre particulièrement efficace lorsqu'il s'agit de traiter de vastes ensembles de données. En plus de sa simplicité, l'un de ses atouts majeurs réside dans le fait qu'il requiert peu de données pour l'apprentissage. Il est d'ailleurs reconnu pour obtenir de très bons résultats, surpassant parfois même des méthodes de classification plus complexes.

Bayes theorem:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Equation 1 : Theorem de Naive Bayes.

Le terme P (A \ B) se lit : la probabilité que l'événement A se réalise sachant que l'événement B s'est déjà réalisé.[41]

$$P(A) = \frac{\text{cardinal de } A}{\text{cardinal total}}$$

$$P(B \cap A) = \frac{\text{cardinal de } (B \cap A)}{\text{cardinal total}}$$

Note:

- le cardinal d'un ensemble est le nombre d'éléments dans ce dernier
- cardinal total représente l'ensemble total des

instances Cas ou on a plusieurs paramètres /

caractéristiques :

$$P(y|x_1, ..., x_n) = \frac{P(x_1|y)P(x_2|y)...P(x_n|y)P(y)}{P(x_1)P(x_2)...P(x_n)}$$

Équation 2 : Naive bayes a plusieurs caractéristiques.

- **Avantage :**

Le Naïve Bayes Classifier est très rapide pour la classification : en effet les calculs de probabilités ne sont pas très coûteux.

La classification est possible même avec un petit jeu de données

- **Inconvénients :**

Contre intuitivement, malgré la violation de la contrainte d'indépendance des variables,

Naïve Bayes donne de bons résultats de classification.

III.4.SVM (support a vecteurs machine):

Le SVM (Support Vector Machine) est un algorithme de classification binaire. Lorsqu'on dispose d'un ensemble de points appartenant à deux classes distinctes dans un espace de dimension NNN, le SVM construit un hyperplan de dimension N−1N-1N−1 permettant de séparer ces deux groupes. Si les points des deux classes sont linéairement séparables, l'algorithme déterminera une droite (ou un hyperplan) qui les divise tout en maximisant la distance entre cette frontière et les points les plus proches de chaque classe.

À plus grande échelle, le SVM, avec des adaptations appropriées, a permis de résoudre plusieurs problèmes complexes. Parmi ces applications figurent la publicité contextuelle sur écran, la reconnaissance des sites de jonction chez l'humain, la détection du genre à partir d'images, ainsi que la classification d'images à grande échelle.[41]

III.5.Régression logistique :

La régression logistique constitue essentiellement une méthode de classification supervisée. Dans un contexte de classification, la variable cible y (ou variable de sortie) ne peut prendre que des valeurs discrètes, en fonction d'un ensemble donné de caractéristiques d'entrée X. Ce type de modèle cherche à estimer la probabilité qu'un point de données appartienne à la classe identifiée par le chiffre "1".

À l'instar de la régression linéaire, qui repose sur l'hypothèse d'une relation linéaire entre les variables, la régression logistique s'appuie sur la fonction sigmoïde pour modéliser les données. Elle n'est véritablement utilisée comme méthode de classification qu'à partir du moment où un seuil de décision est défini.

Le choix de ce seuil revêt une importance capitale, et dépend étroitement de la nature spécifique du problème de classification à résoudre. Cette décision est fortement influencée par deux indicateurs clés de performance : la précision (précision) et le rappel (rappel). Dans l'idéal, ces deux mesures devraient atteindre une valeur de 1, ce qui traduirait une performance parfaite du modèle, bien que cela soit rarement observé en pratique.

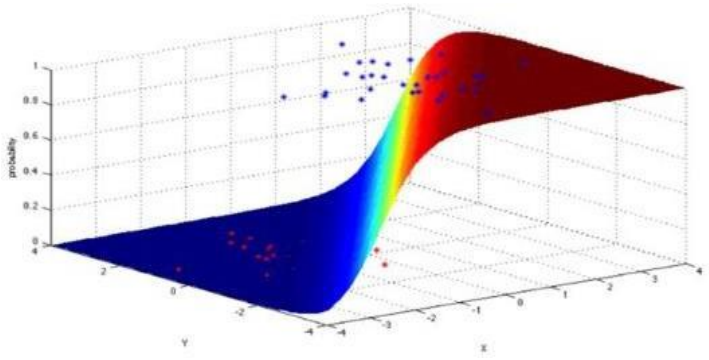


Figure 2.5 : Regression Logistique[42].

III.6.Régression linéaire:

La régression linéaire est une technique d'apprentissage supervisé couramment employée pour résoudre des problèmes de régression. Elle consiste à modéliser une valeur cible prédictive à partir de variables indépendantes, dans le but principal d'analyser les relations entre ces variables et de formuler des prévisions.

Les modèles de régression se distinguent en fonction du type de relation qu'ils établissent entre la variable dépendante et les variables explicatives, ainsi que du nombre de variables indépendantes prises en compte. L'objectif de la régression linéaire est d'estimer la valeur d'une variable dépendante y à partir d'une variable indépendante xxx, en définissant une relation linéaire entre x (entrée) et y (sortie).

À titre d'exemple, si l'on considère xxx comme représentant le nombre d'années d'expérience professionnelle et y comme le niveau de salaire, la ligne de régression correspond alors à la droite d'ajustement optimal qui décrit au mieux cette relation.

Lors de la phase d'apprentissage du modèle, on utilise :

- x : les données d'entrée, souvent univariées (c'est-à-dire une seule variable explicative).
- y : les étiquettes associées aux données, dans le cadre d'un apprentissage supervisé.

Le modèle ajuste alors une droite de régression optimale afin de prédire y à partir de x, en déterminant les valeurs des paramètres qui minimisent l'erreur. Une fois ces paramètres fixés, cette droite d'ajustement sert à estimer y pour toute nouvelle valeur de x.

Ainsi, la régression linéaire permet d'anticiper la valeur de la variable dépendante en fonction de la variable indépendante, grâce à la meilleure droite d'ajustement calculée au cours de l'apprentissage. [43]

III.7.Analyse discriminante linéaire:

L'analyse discriminante linéaire, également appelée analyse discriminante normale ou encore analyse par fonction discriminante, est une méthode de réduction de la dimensionnalité largement utilisée dans le cadre des problèmes de classification supervisée. Son objectif principal est de modéliser les distinctions entre différents groupes, autrement dit, de séparer efficacement deux classes ou plus.

Cette technique consiste à projeter les données issues d'un espace de dimension élevée vers un espace de dimension inférieure, tout en conservant au mieux l'information discriminante. Prenons l'exemple de deux classes que nous souhaitons distinguer de manière optimale. Chaque classe peut être décrite par plusieurs caractéristiques. Or, l'utilisation d'une seule de ces caractéristiques pour les différencier pourrait entraîner un chevauchement entre les classes, comme l'illustre la figure ci-dessous. Par conséquent, il devient nécessaire d'augmenter progressivement le nombre de caractéristiques prises en compte afin d'améliorer la qualité de la classification.[43]

III.8.K-Means :

L'algorithme **K-Means** est une méthode de **clustering** (regroupement) destinée à partitionner un ensemble de données non étiquetées en plusieurs clusters distincts. Le paramètre **K** détermine le nombre de clusters à former : par exemple, si K=2, deux groupes seront créés ; si K=3 , trois groupes, et ainsi de suite.

Cet algorithme permet de structurer les données en groupes homogènes, offrant ainsi une manière efficace d'identifier des catégories au sein d'un jeu de données sans nécessiter de phase d'apprentissage préalable. Il s'agit d'une technique fondée sur les centroïdes, chaque cluster étant représenté par un centroïde, soit un point central qui en résume les caractéristiques.

L'objectif principal de K-Means est de minimiser la somme des distances entre chaque point de données et le centroïde du cluster auquel il est rattaché. L'algorithme prend en entrée des données non étiquetées, les divise en **K** clusters, puis répète le processus

d'ajustement jusqu'à la convergence vers une répartition optimale. Il est important de noter que la valeur de **K** doit être définie à l'avance avant d'exécuter l'algorithme.

Conclusion :

Dans ce chapitre, nous avons abordé le domaine de l'intelligence artificielle dans ses différentes dimensions, en mettant particulièrement l'accent sur l'apprentissage automatique, considéré comme l'un de ses piliers fondamentaux. Nous avons mis en lumière les principes de base de cette discipline, en explorant les principales techniques de classification ainsi qu'un ensemble d'algorithmes et de méthodes permettant d'extraire des schémas significatifs à partir de vastes volumes de données.

Ces connaissances avancées constituent une base essentielle pour la conception d'un système performant de détection d'intrusions, capable de renforcer la sécurité et d'optimiser les capacités d'analyse et de réponse dans un environnement numérique en constante menace.

Chapitre 3
Description du projet

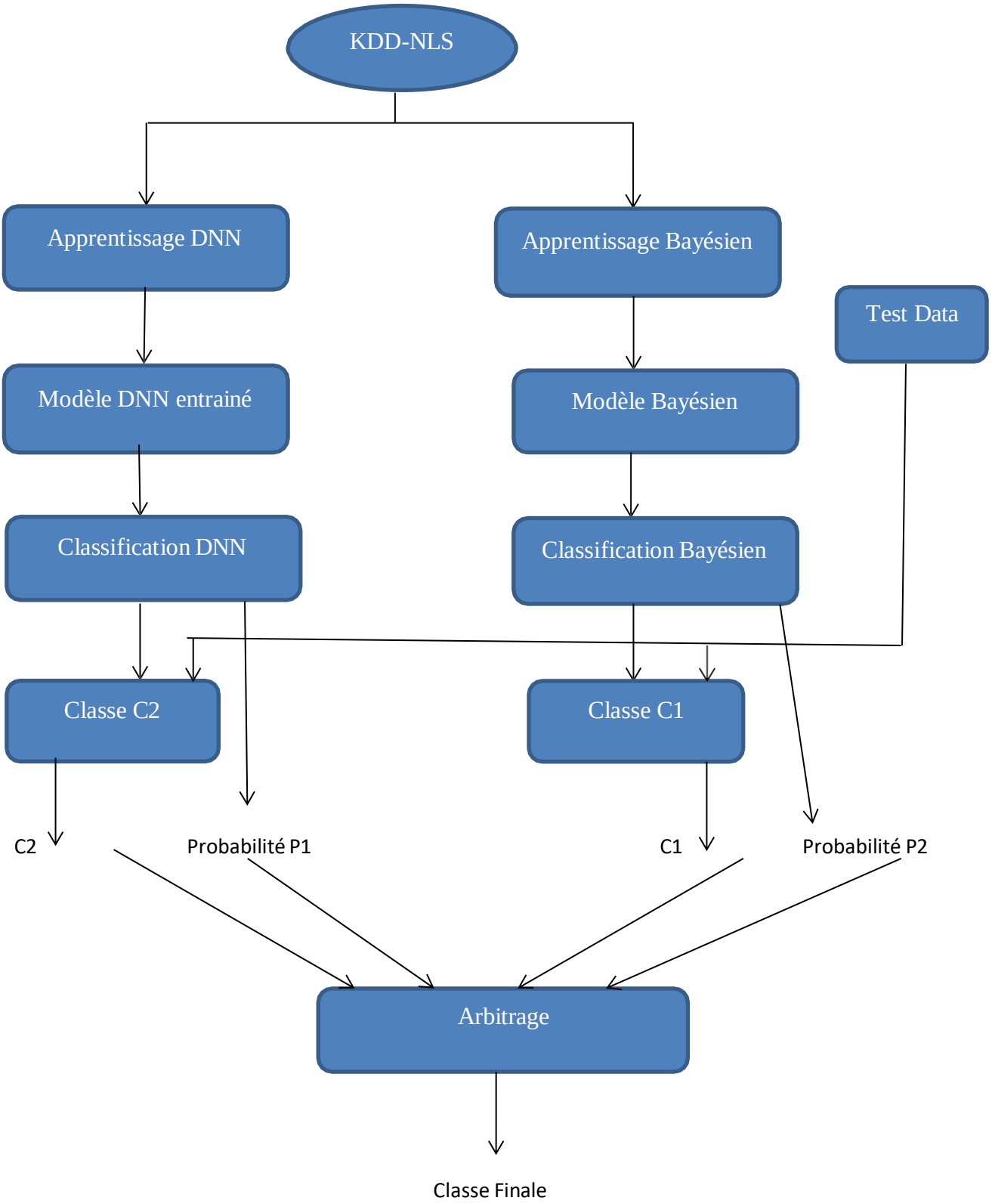
Introduction :

Le domaine de la détection d'intrusions connaît une évolution remarquable grâce aux techniques d'intelligence artificielle, ce qui nécessite l'adoption de modèles plus précis et performants. Dans ce chapitre, nous présentons une approche hybride basée sur la combinaison de deux modèles d'apprentissage automatique : un réseau de neurones profond et un modèle bayésien, dans le but d'améliorer les performances du système de détection.

Nous commençons par exposer le principe de l'intégration entre les deux modèles, puis nous détaillons les caractéristiques de chacun, y compris l'architecture des réseaux neuronaux, les mécanismes d'apprentissage, ainsi que les algorithmes de classification bayésienne. Nous abordons également les fonctions essentielles comme l'activation et les probabilités (a priori et a posteriori), avant de conclure par un plan expérimental destiné à évaluer ce système à l'aide du jeu de données KDD_NLS.

I. Principe d'intégration :

Dans ce volet, nous proposons une approche fondée sur l'intégration de plusieurs modèles d'intelligence artificielle en vue d'optimiser les performances du système de détection d'intrusions. Cette méthode repose sur l'exploitation du jeu de données KDD_NLS et combine deux techniques d'apprentissage supervisé : un réseau de neurones profond (DNN) et un modèle bayésien. Chaque modèle est entraîné séparément afin de générer des prédictions accompagnées de probabilités associées. Ces résultats sont ensuite fusionnés à travers une étape d'arbitrage, permettant de produire une décision finale plus fiable. Le schéma ci-dessous illustre les différentes étapes de ce processus d'intégration.


KDD-NLS
Apprentissage DNN
Apprentissage Bayésien
Test Data
Modèle DNN entrainé
Modèle Bayésien
Classification DNN
Classification Bayésien
Classe C2
Classe C1
C2
Probabilité P1
C1
Probabilité P2
Arbitrage
Classe Finale

I.1.Réseaux de neurones profonds (Deep Neural Network) :

Les réseaux de neurones profonds, souvent désignés par l'acronyme DNN (Deep Neural Networks), constituent une percée significative dans le champ de l'intelligence artificielle. Inspirés par la structure et le fonctionnement du cerveau humain, ces systèmes informatiques sophistiqués sont conçus pour simuler la manière dont les humains traitent et interprètent l'information. Grâce à leur architecture multicouche, ils sont capables d'apprendre automatiquement à partir de grandes quantités de données, en extrayant des caractéristiques complexes et en améliorant continuellement leurs performances sans intervention humaine directe. Cette capacité d'apprentissage profond leur permet de résoudre des problèmes complexes dans divers domaines, tels que la reconnaissance vocale, la vision par ordinateur, et l'analyse prédictive.[44]

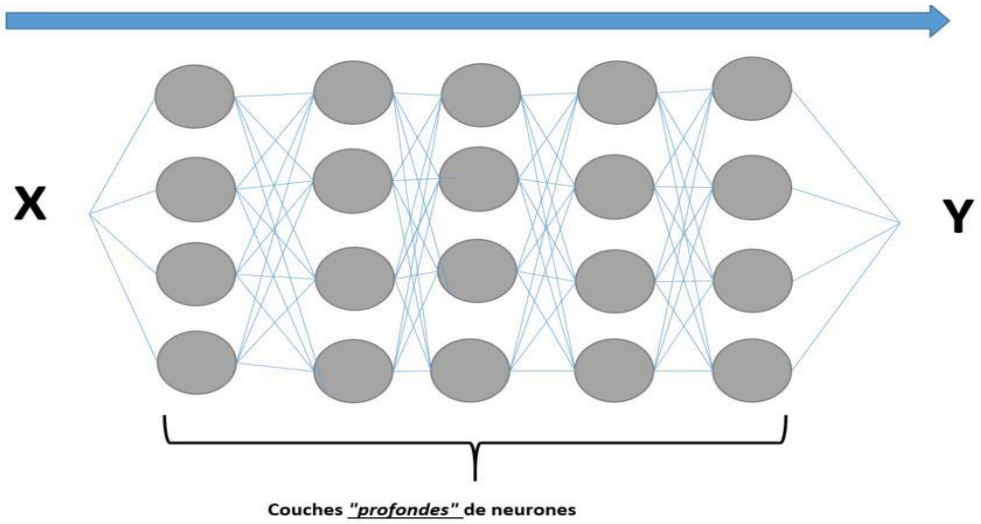


Figure3.1: Réseaux de neurones profonds [45]

I.1.1.Définition :

Un réseau neuronal profond (RNP) est un modèle avancé d'apprentissage automatique qui s'inspire du fonctionnement du cerveau humain pour traiter les informations. Contrairement aux algorithmes classiques, qui appliquent des règles fixes, ces réseaux ont la capacité d'apprendre directement à partir des données brutes. Ils détectent des schémas complexes et effectuent des prédictions en s'appuyant sur leur expérience acquise, de manière similaire à la manière dont un être humain apprend.

Les réseaux neuronaux profonds constituent la pierre angulaire de l'apprentissage profond (deep learning). Ils sont au cœur de nombreuses technologies modernes, telles que les assistants virtuels intelligents, la reconnaissance d'images, la compréhension du langage naturel et les chatbots. Grâce à cette capacité d'apprentissage, ces systèmes peuvent résoudre des problèmes complexes et offrir des interactions plus naturelles et efficaces entre l'homme et la machine. [46]

I.1.2.Architecture des réseaux de neurones profonds :

Les réseaux de neurones profonds se caractérisent par une structure composée de multiples couches interconnectées, coopérant pour analyser des données complexes. Chacune de ces couches joue un rôle fondamental dans le processus de traitement et de transformation de l'information, permettant au réseau d'atteindre une compréhension plus fine des entrées.[44]

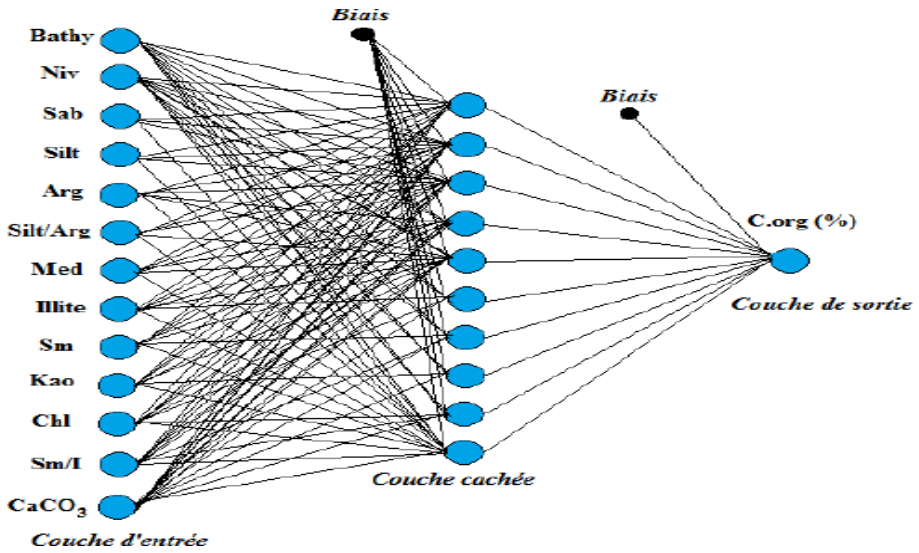


Figure3.2: Architecture des réseaux de neurones profonds **[**48]

Couche d'entrée et traitement initial : Le traitement commence au niveau de la couche d'entrée, chargée de recevoir les données à l'état brut. Dans un réseau neuronal convolutionnel, cette couche peut par exemple capter les pixels d'une image. Un prétraitement est souvent appliqué à ce stade, notamment la normalisation, afin de standardiser les données et améliorer l'efficacité de l'apprentissage.

Couches cachées et leur rôle : Les couches cachées constituent le noyau opérationnel du réseau. C'est à ce niveau que s'effectuent les calculs les plus sophistiqués, permettant d'extraire des caractéristiques de plus en plus abstraites. Par exemple, lors de la reconnaissance d'images,

les premières couches identifient des éléments simples comme les contours, les couches intermédiaires perçoivent des formes, et les couches finales reconnaissent des objets complets.

Couche de sortie et production des résultats : La couche de sortie représente l'étape finale du réseau, où le résultat est généré en fonction de la tâche assignée. En classification, elle peut contenir un neurone par catégorie, alors qu'en régression, un seul neurone peut suffire à estimer une valeur continue. La forme de cette couche dépend donc directement du type de problème à résoudre.

Type de réseau	Application	Durée de formation
Convolutionnel	Analyse d'images	14 heures
Récurrent	Traitement du langage	21 heures
Feedforward	Prévision boursière	7 heures

Tableau 3.1: Réseaux de neurones et leurs applications

Cette organisation hiérarchique des couches confère aux réseaux de neurones profonds une capacité d'adaptation remarquable à une grande variété de domaines, allant de la vision artificielle à la traduction automatique. Ce qui rend ces réseaux si puissants, c'est leur aptitude à construire, couche après couche, des représentations riches et adaptées à la complexité des données du monde réel.

I.1.3.La fonction d'activation:

Sur le plan mathématique, la **fonction d'activation** joue un rôle essentiel dans le fonctionnement des neurones artificiels. Elle permet à chaque neurone de **transformer les données d'entrée** qu'il reçoit en appliquant un mécanisme de lissage ou de normalisation, avant de décider s'il doit transmettre ou non l'information à la couche suivante. Ce processus s'enclenche uniquement lorsque **la valeur d'activation dépasse un certain seuil prédéfini**. Dans le cas contraire, le neurone reste inactif, à l'image du comportement observé dans les neurones biologiques.

Ces fonctions sont **fondamentales dans le processus d'apprentissage** des réseaux neuronaux, car elles introduisent la non-linéarité nécessaire pour modéliser des relations complexes entre les données.

Parmi les fonctions d'activation les plus couramment utilisées, on retrouve :

- la **fonction sigmoïde**, adaptée aux problèmes de classification binaire,
- la **fonction ReLU** (Rectified Linear Unit), fréquemment employée dans les couches cachées pour sa simplicité et son efficacité computationnelle,
- la **fonction tanh** (tangente hyperbolique), qui offre une sortie centrée autour de zéro,
- la **fonction Softmax**, utilisée dans la couche de sortie pour les tâches de classification multi-classes, car elle produit des probabilités d'appartenance à chaque classe.[47]

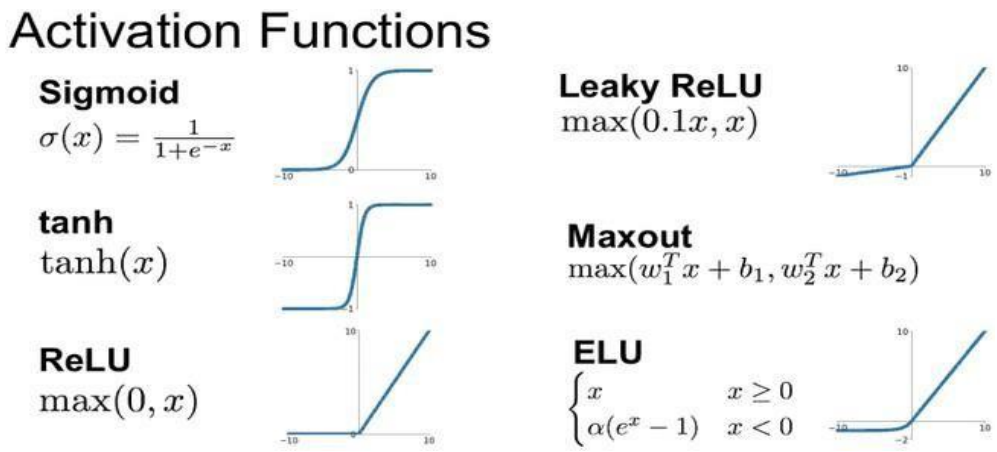


Figure3.3 : Exemples des fonctions d'activations.

I.1.4.La probabilité de classification :

La probabilité de classification nous fournit un indicateur sur la qualité de classification d'un élément donné selon le modèle utilisé. Dans la plupart des modèles d'apprentissage automatique, l'outil d'implémentation (Python par exemple) fournit en plus du vecteur de classes obtenues, les probabilités de ces classes. En d'autre terme, et pour un élément donné, il nous est fourni pour cet élément la probabilité que sa classe réelle est celle obtenue par la prédiction du modèle entrainé.

I.1.5.L'Apprentissage d'un DNN :

L'apprentissage automatique (Machine Learning – ML) est une technologie de l'intelligence artificielle qui permet aux ordinateurs d'apprendre à partir des données sans être explicitement programmés. Il constitue une base essentielle pour exploiter le potentiel du Big Data. Cependant,

les méthodes traditionnelles sont limitées dans le traitement des données brutes. C'est dans ce contexte que l'apprentissage profond s'impose comme une extension avancée du ML, s'inspirant du fonctionnement du cerveau humain. Il est utilisé dans divers domaines tels que la détection de la fraude, la reconnaissance vocale, la traduction automatique et la prise de décision. Sa particularité réside dans sa capacité à apprendre à partir de données non structurées et non étiquetées, sans supervision humaine directe.

I.1.6.L'algorithme d'apprentissage :

L'algorithme d'apprentissage constitue une méthode mathématique essentielle permettant l'ajustement progressif des poids de connexion dans un réseau neuronal, en vue de converger vers une solution optimale qui permet l'accomplissement de la tâche souhaitée. Ce processus s'apparente à une approche d'identification paramétrique visant à optimiser les valeurs des poids du réseau.

Divers algorithmes itératifs peuvent être utilisés pour ce processus, parmi lesquels figurent : l'algorithme de rétropropagation, la méthode Quasi-Newton, l'algorithme BFGS, entre autres.

Parmi ces approches, l'algorithme de rétropropagation est sans doute le plus répandu. Connu également sous le nom de **backpropagation**, il s'agit d'un exemple classique d'apprentissage supervisé, dont la notoriété s'est largement accrue grâce à certaines applications spectaculaires. On peut citer, à titre d'exemple, les travaux de Sejnowski et Rosenberg (1987) qui ont démontré l'efficacité de cet algorithme dans un système capable d'apprendre à lire un texte. D'autres succès notables incluent la prédiction des fluctuations boursières, ainsi que la détection de fraudes dans les transactions par carte bancaire.

La rétropropagation du gradient est une technique permettant de calculer, de manière rétroactive, le gradient de l'erreur pour chaque neurone, en partant de la couche de sortie jusqu'à la couche d'entrée. Les recherches antérieures montrent que cette méthode a été découverte indépendamment par plusieurs chercheurs, bien qu'elle ait porté des noms différents selon les publications.

Le fonctionnement de la rétropropagation repose sur trois étapes clés : le passage de l'information dans le réseau, le calcul rétroactif des sensibilités et des gradients, puis l'ajustement des paramètres à l'aide de la règle de descente du gradient. Toutefois, cette méthode présente certaines limitations, notamment le risque d'être piégée dans des minima locaux, ce qui

peut survenir lorsque les gradients ou leurs dérivées deviennent nuls. De plus, la convergence peut être particulièrement lente, surtout dans le cas des réseaux complexes possédant un grand nombre de paramètres à optimiser.

Pour pallier ces inconvénients et améliorer l'efficacité de l'optimisation, il est possible de recourir à des méthodes d'ordre supérieur, telles que les méthodes de type Quasi-Newton ou Newton modifiée.

I.2.La classification bayésienne :

Le classificateur de Bayes repose sur le théorème de Bayes pour estimer la probabilité qu'un message appartienne à une catégorie donnée, en se basant sur les probabilités conjointes des termes et des classes. Il cherche à choisir l'hypothèse la plus probable pour classer un message, en utilisant les probabilités a priori (avant observation) et a posteriori (après observation). La classe retenue est celle qui maximise la probabilité a posteriori, le dénominateur étant constant et donc ignoré. En raison de la complexité d'estimation des probabilités a posteriori à cause du grand nombre de paramètres, on adopte l'hypothèse d'indépendance conditionnelle des attributs (caractéristiques), connue sous le nom de modèle Naïve Bayes. Cela permet d'estimer les probabilités des termes individuellement. Cette hypothèse simple rend le modèle efficace, largement utilisé notamment pour la classification de documents textuels et le filtrage des spams.[49]

I.2.1. La règle de Bayes :

- **Probabilité conditionnelle et théorème de Bayes :**

La probabilité conditionnelle entre deux événements A et B est définie par :

$$Pr[A \cap B]=Pr[A|B].Pr[B]=Pr[B|A].P[A]$$

- **Théorème de Bayes (avec contexte) :**

Soient A, B, et C trois événements. Le théorème de Bayes généralisé s'écrit :

$$Pr[A|B,C]=\frac{Pr[B|A,C]Pr[A|C]}{Pr[B|C]}$$

Avec :

- Pr[B|A,C] : la **vraisemblance**, c'est-à-dire la probabilité de B sachant que A et C sont vrais,
- Pr[A|C]: la **probabilité a priori** de A sachant C,
- Pr[B|C]: la **probabilité marginale** de B sachant C,
- Pr[A|B,C]: la **probabilité a posteriori** de A sachant B et C.
- **Application à la classification :**

Dans le cadre de la classification, on cherche à estimer :

$$Pr[y|x,X]$$

C'est-à-dire : la probabilité que l'**étiquette** (ou classe) soit y, sachant l'observation d'une nouvelle donnée x et un ensemble d'exemples d'apprentissage X.

Cela représente donc la **probabilité que la donnée x appartienne à la classe y**, étant donné les exemples d'apprentissage disponibles.

En appliquant la règle de Bayes, on obtient :

$$Pr[y|x,X] = \frac{Pr[x|y,X].Pr[y|X]}{Pr[x|X]}$$

Ce qui permet d'évaluer la classe la plus probable pour une donnée nouvelle .[50]

I.2.2.La probabilité a priori :

Cette probabilité représente une estimation préalable de la probabilité qu'un élément appartienne à la classe y, en se basant sur l'ensemble d'exemples d'apprentissage X. Elle est généralement calculée comme la proportion d'exemples dans X qui appartiennent à la classe y.

Lorsqu'une information préalable sur la distribution des classes dans les données est disponible, il est possible d'en tirer parti pour estimer directement cette probabilité a priori, en utilisant la proportion de la classe y dans l'ensemble global des données. Cette estimation joue un rôle fondamental dans les modèles statistiques, notamment dans les algorithmes de classification probabilistes tels que Naïve Bayes, où elle constitue un élément clé dans le calcul de la probabilité a posteriori.[50]

I.2.3.La probabilité a Posteriori :

- **Classe MAP (Maximum A Posteriori) :**

Une fois avoir calculé Pr[y |x,X],∀y∈Y, on peut prédire sa classe comme étant celle qui maximise la probabilité a posteriori : c'est la classe **MAP** (Maximum A Posteriori), soit :

$$y_{MAP} = \underset{y \in Y}{argmax}\ Pr[y|x, \mathcal{X}]$$

Cela peut aussi s'écrire en appliquant l'équation (2) :

$$y_{MAP} = \underset{y \in Y}{argmax}\ Pr[x \mid y, \mathcal{X}] Pr[y \mid \mathcal{X}]$$

- **Classe ML (Maximum de Vraisemblance) :**

Si l'on ne tient pas compte de Pr[y|X] et qu'on ne considère que la vraisemblance Pr[x|y] on obtient la classe **ML** (Maximum Likelihood), soit :

$$y_{ML} = \underset{y \in Y}{argmax}\ Pr[x \mid y, \mathcal{X}]$$

Clairement, si les exemples sont uniformément répartis entre toutes les classes, soit :

$$\text{t } Pr[y|\mathcal{X}] = \frac{1}{|Y|}$$

alors les classes ML et MAP sont équivalentes. [50]

I.2.4.La vraisemblance :

- **Estimation de la vraisemblance Pr[x|y,X] et hypothèse de Bayes naïve :**

La probabilité conditionnelle Pr[x|y,X] représente la vraisemblance d'observer la donnée x sachant qu'elle appartient à la classe y, en tenant compte de l'ensemble d'exemples X. Cette probabilité est généralement difficile à estimer directement, en particulier lorsque les données sont complexes ou de grande dimension.

Pour simplifier cette estimation, on adopte fréquemment **l'hypothèse de Bayes naïve (HBN)**. Cette hypothèse repose sur l'idée que la donnée xxx est une conjonction de valeurs d'attributs, et suppose que ces attributs sont des variables aléatoires **indépendantes** entre elles (i.e., non corrélées). Bien que cette hypothèse d'indépendance soit rarement vérifiée en pratique, elle permet néanmoins de simplifier considérablement les calculs, tout en fournissant des résultats souvent pertinents. Lorsque des informations concernant d'éventuelles corrélations entre attributs sont disponibles, il est naturellement possible de les exploiter pour améliorer les estimations.

En appliquant l'HBN, et en supposant que la donnée x est décrite par P attributs notés aj prenant respectivement les valeurs vj , la vraisemblance s'écrit approximativement comme suit :

$$\begin{aligned} Pr[x|y,\mathcal{X}] &\approx Pr[a_1 = v_1|y,\mathcal{X}] \times \ldots \times Pr[a_P = v_P|y,\mathcal{X}] \\ &= \prod_{i=1}^{i=P} Pr[a_i = v_i|y,\mathcal{X}] \end{aligned}$$

Chaque terme Pr[aj = vj |y,X] est estimé à partir de l'ensemble d'exemples disponibles. La méthode d'estimation de ces probabilités dépend de la nature des attributs concernés, qu'ils soient **qualitatifs** (catégoriels) ou **quantitatifs** (numériques).[50]

Conclusion :

Dans ce chapitre, nous avons exploré une approche fondée sur l'intégration de deux modèles complémentaires : les réseaux de neurones profonds, reconnus pour leur capacité à apprendre à partir de données non structurées, et le modèle bayésien, caractérisé par sa structure probabiliste et ses inférences basées sur des connaissances préalables. La combinaison de ces deux paradigmes permet la construction d'un système de détection plus précis et adaptable, capable de faire face à la complexité croissante des menaces cybernétiques. Cette base théorique ouvrira la voie à la phase expérimentale, où l'efficacité du système sera évaluée à l'aide de données réelles.

Chapitre 4
Implémentation

Introduction :

Ce chapitre présente l'aspect pratique de ce travail, à travers la mise en œuvre d'un ensemble d'expériences visant à évaluer les performances des algorithmes utilisés pour la détection d'intrusions. En s'appuyant sur le jeu de données NSL-KDD, nous analysons la précision des modèles sélectionnés, en nous basant sur des métriques rigoureuses permettant de mesurer leur capacité à distinguer les attaques des connexions normales au sein du réseau.

I. Outils et application :

I.1.Présentation de Python :

Python est un langage de programmation de haut niveau, interprété et très polyvalent, dont la popularité n'a cessé de croître au fil du temps grâce à sa simplicité, sa clarté et sa fiabilité. Développé par Guido van Rossum et apparu pour la première fois en 1991, il a été conçu pour être à la fois facile à lire et à écrire. Cela en fait un excellent choix pour les débutants, tout en offrant des capacités avancées appréciées par les développeurs expérimentés.

Ce langage se distingue par une syntaxe épurée et directe, qui permet d'écrire moins de lignes de code pour accomplir une tâche donnée. Cette caractéristique favorise non seulement la productivité, mais aussi la facilité de maintenance des programmes.

Python est également compatible avec plusieurs systèmes d'exploitation, tels que Windows, macOS et Linux, ce qui permet d'exécuter le même code sur différentes plateformes sans nécessiter de modifications importantes.

Enfin, Python bénéficie d'une communauté dynamique et en constante expansion. Cette communauté joue un rôle clé dans l'évolution du langage et de ses bibliothèques, tout en offrant un soutien précieux à travers une abondante documentation, des forums d'échange, ainsi que de nombreux tutoriels.[51]

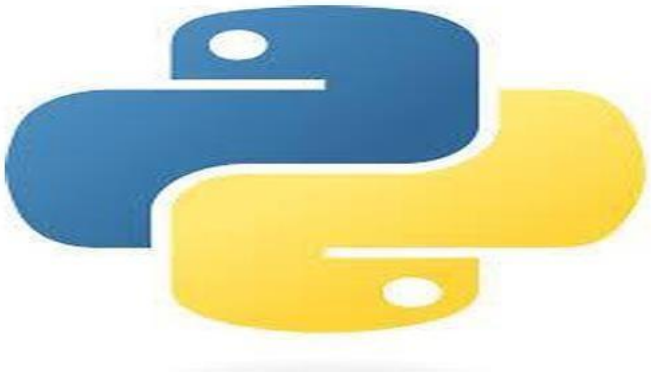


Figure4.1: Logo de langage python [51]

I.1.1.Applications de Python:

Python est utilisé dans une multitude de domaines, notamment :

- **Développement Web :** Frameworks tels que Django et Flask permettent de créer des applications web robustes et scalables. [52].

- **Science des Données :** Python est l'outil de prédilection pour les data scientists, grâce à ses bibliothèques puissantes comme Pandas, NumPy et Matplotlib.[53].

- **Apprentissage Automatique et Intelligence Artificielle :** Avec des bibliothèques comme Scikit-learn, TensorFlow et Keras, Python est largement utilisé pour développer et déployer des modèles de machine learning et deep learning.[53].

- **Automatisation et Scripting :** La simplicité de Python le rend idéal pour écrire des scripts qui automatisent des tâches répétitives.

- **Développement de Jeux :** Bibliothèques comme Pygame permettent le développement de jeux vidéo simples.[53]

I.1.2.Bibliothèque python :

Une **bibliothèque Python** désigne un regroupement de modules ou de packages déjà codés, proposant des fonctionnalités spécifiques et prêtes à l'emploi. Conçues pour être réutilisables, ces bibliothèques permettent aux développeurs d'économiser du temps et des efforts en évitant de redévelopper des fonctions standards à chaque projet.

L'écosystème Python offre une collection extrêmement riche de bibliothèques couvrant des domaines variés, allant du développement web à la science des données, en passant par l'intelligence artificielle et bien d'autres champs d'application.

Parmi les composants essentiels, la **bibliothèque standard de Python** fournit un ensemble de fonctionnalités de base telles que la gestion des fichiers, la manipulation de chaînes de caractères ou encore les opérations réseau. En complément, de nombreuses **bibliothèques tierces** viennent enrichir les possibilités offertes par Python : par exemple, **NumPy** pour le calcul scientifique, **Pandas** pour le traitement de données, ou encore **Django** pour le développement d'applications web.

Ces bibliothèques sont généralement accessibles via le **Python Package Index (PyPI)**, une plateforme qui facilite leur installation ainsi que la gestion des dépendances au sein des projets.

Voici quelques bibliothèques Python parmi les plus utilisées, accompagnées de leurs domaines d'application :

- **Pandas** : Il s'agit d'une bibliothèque très prisée pour charger et manipuler des fichiers de type Excel (ou d'autres formats) dans Python. Elle permet d'extraire des statistiques et d'alimenter des modèles via Scikit-learn.
- **Seaborn** : Cette bibliothèque est spécialisée dans la création de graphiques statistiques. Elle repose sur Matplotlib et s'intègre parfaitement avec les structures de données fournies par Pandas.
- **NumPy (Numerical Python)** : C'est une bibliothèque open source qui propose la gestion de tableaux et de matrices multidimensionnels, ainsi qu'un large éventail de fonctions mathématiques performantes. Elle est largement utilisée pour les calculs numériques et le traitement de données.
- **Matplotlib** : Outil de référence pour la visualisation, Matplotlib permet de tracer des graphiques en 2D et 3D, facilitant l'exploration et la présentation des données.

- **Beautiful Soup** : Spécialisée dans l'analyse de documents HTML et XML, cette bibliothèque est souvent utilisée pour extraire des informations à partir de sites web (scraping).
- **Requests** : Bibliothèque HTTP simple et élégante, conçue pour faciliter l'envoi de requêtes web dans les applications Python.
- **Scikit-learn** : Une bibliothèque d'apprentissage automatique très populaire, offrant un large choix d'algorithmes pour la classification, la régression, le regroupement, etc.
- **NLTK (Natural Language Toolkit)** : Dédiée au traitement automatique du langage naturel, elle propose divers outils comme la tokenisation, la lemmatisation et la reconnaissance des entités nommées.
- **Pygame** : Bibliothèque dédiée à la création de jeux 2D en Python, elle offre des fonctionnalités adaptées au développement ludique et interactif.
- **PyTorch** : Framework d'apprentissage profond développé par Facebook, apprécié pour sa flexibilité et ses performances élevées dans la conception de modèles de deep learning.

Ces bibliothèques, parmi tant d'autres, constituent des outils incontournables dans la communauté Python, permettant d'accélérer le développement de solutions logicielles performantes dans une grande variété de domaines.[54]

I.2.Présentation de Google colab :

Google Colab, également connu sous le nom de **Google Colaboratory**, est une plateforme gratuite proposée par Google qui permet d'écrire et d'exécuter du code Python directement depuis un navigateur web. Mis en ligne en 2017, ce service est largement adopté par les développeurs et les chercheurs grâce à son interface conviviale et aux ressources de calcul performantes qu'il met à disposition.

Figure4.2: Logo de google colab [56]

I.2.1.Les caractéristiques de Google colab :

Voici quelques caractéristiques importantes de Google Colab :

- **Environnement de Développement Basé sur Jupyter** : Google Colab repose sur l'utilisation des notebooks Jupyter, qui permettent de combiner dans un même document du code exécutable, du texte explicatif, des images et des graphiques. Cette approche interactive rend la présentation et le partage des travaux bien plus fluides, renforçant ainsi l'efficacité du travail collaboratif. [55]
- **Accès Gratuit aux Ressources de Calcul** : L'un des grands avantages de Colab est la mise à disposition gratuite de ressources matérielles comme les CPU, GPU et TPU.

  Cela donne la possibilité d'exécuter des tâches complexes, telles que des modèles de machine learning ou des calculs intensifs, sans avoir besoin d'un équipement coûteux. [55]
- **Intégration avec Google Drive** : Les notebooks créés dans Colab peuvent être directement enregistrés dans Google Drive, ce qui simplifie leur gestion et leur accessibilité. Il est également possible d'importer ou d'exporter les fichiers sous différents formats, tels que
.ipynb et .py. [57]
- **Partage et Collaboration en Temps Réel** : Grâce à Colab, les utilisateurs peuvent partager leurs notebooks avec d'autres personnes, qui peuvent alors commenter ou modifier le contenu instantanément. Ce mode de travail collaboratif est idéal pour les projets collectifs ou les recherches menées à plusieurs.
- **Installation de packages supplémentaires** : Colab offre la possibilité d'installer des bibliothèques Python supplémentaires via pip directement depuis l'interface du notebook. Cela permet aux utilisateurs d'élargir facilement les fonctionnalités de leurs projets en y intégrant des outils tiers. [58]

I.2.2.Applications de Google Colab :

- **Apprentissage Automatique et Deep Learning :** L'accès gratuit aux GPU et TPU fourni par Colab en fait une solution parfaitement adaptée pour entraîner des modèles complexes de machine learning et de deep learning, même à grande échelle.
- **Analyse de Données** : Grâce aux notebooks interactifs, il devient plus simple d'examiner, de manipuler et de visualiser les données, ce qui représente un atout majeur dans le cadre des projets en science des données.

- **Recherche et Développement** : Les chercheurs bénéficient de Colab pour concevoir rapidement des prototypes d'algorithmes et partager leurs découvertes avec leurs pairs ou la communauté scientifique de manière fluide.
- **Enseignement et Formation** : Colab se révèle particulièrement utile dans un contexte éducatif, car il permet aux enseignants de concevoir des cours interactifs et des exercices pratiques que les apprenants peuvent exécuter directement en ligne, sans installation préalable.

En résumé, Google Colab constitue une plateforme performante et facile d'accès, réunissant plusieurs atouts. Entre les ressources de calcul gratuites, une interface intuitive, et des fonctionnalités de collaboration en temps réel, il s'impose comme un outil incontournable pour les projets liés à la recherche, à l'analyse de données, ou à l'apprentissage machine.

II. Protocole expérimentale :

II.1.Le choix du Data Set :

Les systèmes intelligents de détection d'intrusion reposent sur la disponibilité d'un ensemble de données pertinent pour pouvoir être développés efficacement. Il est essentiel que cet ensemble de données soit riche en informations et qu'il reflète fidèlement le trafic en temps réel, afin de permettre un apprentissage et une évaluation fiables des performances desdits systèmes. Le jeu de données **NSL-KDD**, qui constitue une version améliorée du jeu **KDD 99** original, est utilisé dans le cadre de ce projet pour analyser l'efficacité de deux algorithmes de classification dans la détection d'anomalies au sein du trafic réseau.

Ce jeu de données comprend 42 attributs, ce qui représente une nette amélioration par rapport à **KDD 99**, notamment grâce à la suppression des doublons qui pouvaient engendrer des biais dans les résultats de classification. Bien que le nombre d'instances varie selon les configurations, les 42 attributs restent constants. L'attribut numéro 42, intitulé « class », revêt une importance particulière puisqu'il permet de déterminer si une instance correspond à une connexion normale ou à une attaque [59].

L'exploitation de ce jeu de données permet de former et d'évaluer différents algorithmes de classification, dans le but de mesurer leur aptitude à détecter les intrusions et les comportements anormaux au niveau du réseau. L'objectif principal est de repérer les méthodes les plus performantes afin de renforcer la sécurité des systèmes informatiques en assurant une détection rapide et une réponse efficace face aux menaces.

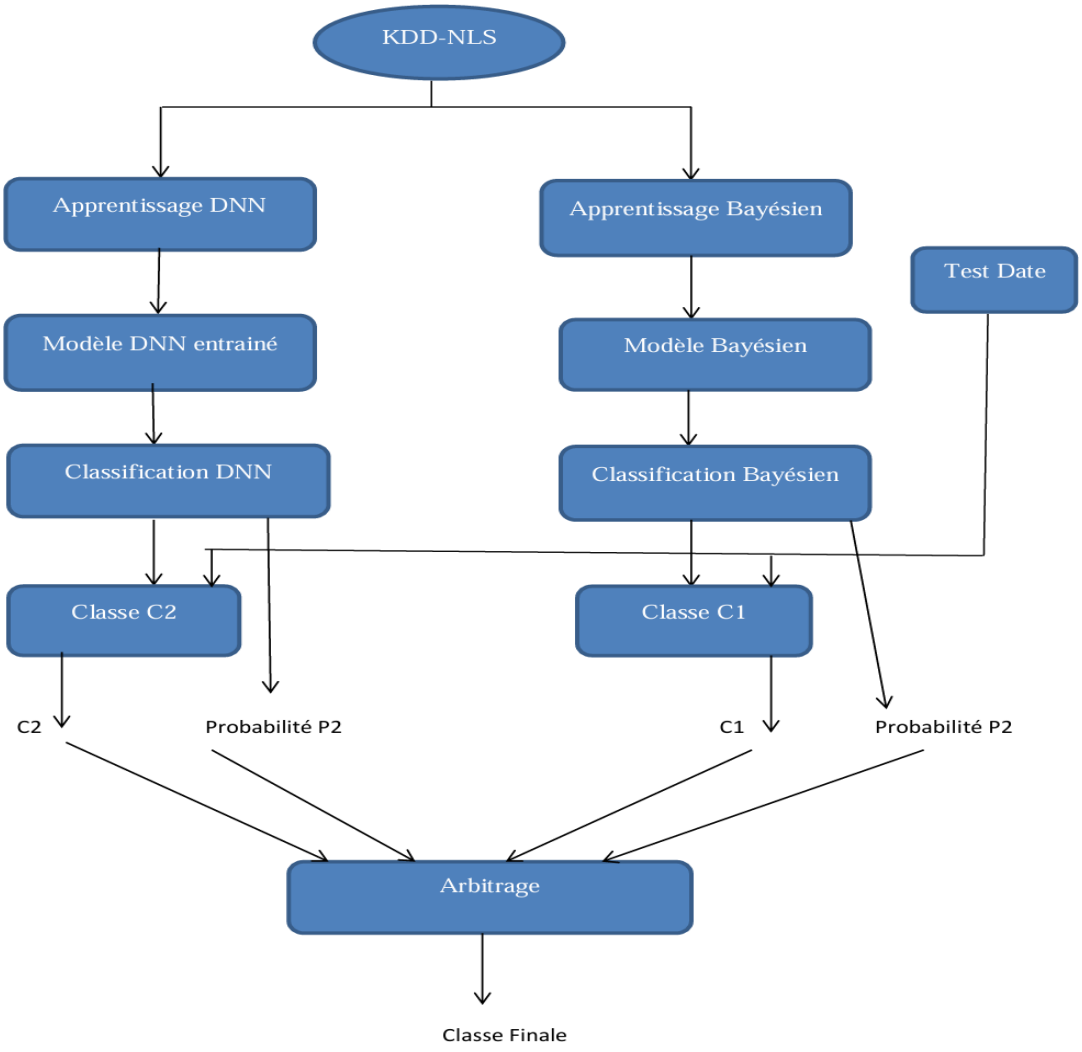


Figure 4.3 : Diagramme du protocole expérimental.

II.2.Métrique de Performance :

Afin d'évaluer l'efficacité et la robustesse d'un modèle de classification, notamment dans le domaine de la détection d'intrusions, il est essentiel de s'appuyer sur un ensemble de métriques de performance. Ces métriques permettent de quantifier la capacité du modèle à prédire correctement les différentes classes (attaque ou connexion normale, par exemple) et d'identifier ses éventuelles faiblesses. Parmi les mesures les plus couramment utilisées, on retrouve l'exactitude, la précision et le score F1, qui sont décrites ci-dessous :

Exactitude (Accuracy) : L'exactitude correspond à la proportion de prédictions correctes réalisées par le modèle parmi l'ensemble des prédictions effectuées (Shung, 2018). Autrement dit, elle mesure la capacité du modèle à classer correctement chaque échantillon. Elle se calcule ainsi :

$$\textbf{Exactitude} = \frac{\textbf{Total nombres de prédictions correctes}}{\textbf{Total nombres de prédictions}} = \frac{TP+TN}{TP+TN+FP+FN}$$

Précision (Precision) : La précision indique la part des prédictions positives qui sont effectivement justes. Elle mesure donc la fiabilité des prédictions positives du modèle, en tenant compte du nombre total des échantillons prédits comme positifs (vrais positifs et faux positifs). La formule est la suivante :

$$\textbf{Précision} = \frac{\textbf{Nombre total de prévisions correctes}}{\textbf{Nombre total de prévisions positives}} = \frac{TP}{TP+FP}$$

Score F1 (F1-Score) : Le score F1 est une mesure qui combine à la fois la précision et le rappel, afin de fournir un équilibre entre ces deux métriques. Cette mesure est particulièrement utile lorsque les classes sont déséquilibrées. Le score F1 se calcule comme suit (Shung, 2018) :

$$\textbf{Score F1} = 2 \times \textbf{Rappel} \times \textbf{Précision} + \textbf{Rappel}$$

III. Expérimentations :

- **Chargement et fusion des données :**

```
df_train =
pd.read_csv('/content/drive/MyDrive/KDD/NSL_KDD_Train.csv')
df_test =
pd.read_csv('/content/drive/MyDrive/KDD/NSL_KDD_Test.csv') df =
pd.concat([df_train, df_test], axis=0)
```

- **Encodage des variables nominales :**

```
encoder = preprocessing.OneHotEncoder(sparse_output=False)
df_nominal = encoder.fit_transform(df[['protocol_type', 'service',
'flag']])
```

- **Normalisation des variables numériques** :

```
scaler = preprocessing.MinMaxScaler()
df_numeric =
scaler.fit_transform(df_numeric)
```

- **Regroupement des étiquettes :**

```
def change_label(df):
    df['attack'] = df['attack'].replace([...], '1')
    df['attack'] = df['attack'].replace(['normal'], '0')
```

- **Séparation des données :**

```
X_train =
X[:120000] X_test
= X[120000:]
y_train = y[:120000]
y_test = y[120000:]
```

III.1.Métriques de performance :

Pour évaluer les modèles, nous avons utilisé les métriques suivantes :

- Accuracy (Exactitude) : proportion des prédictions correctes.
- Recall (Rappel) : capacité à identifier les vraies attaques.
- Precision (Précision positive) : proportion de vraies attaques parmi les détections.
- F1-score : moyenne harmonique entre précision et rappel.
- Matrice de confusion : représentation visuelle des résultats de classification.

III.2.Résultats expérimentaux :

➢ **Modèle KNN:**

```
knn = KNeighborsClassifier(n_neighbors=1)
knn.fit(X_train, y_train)
y_pred_knn = knn.predict(X_test)
```

- Accuracy : 82.44%
- F1-score (normal) : 82.98%
- F1-score (attaque) : 81.87%

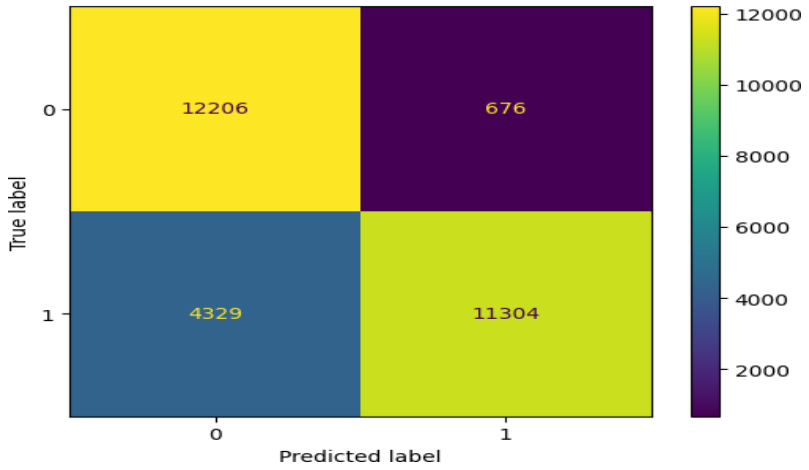


Figure 4.4 : Matrice de confusion du modèle KNN.

➢ **Modèle Naive Bayes :**

```
gnb = GaussianNB()
gnb.fit(X_train, y_train)
y_pred_bayes =
gnb.predict(X_test)
```

- Accuracy : 62.66%
- F1-score (normal) : 70.65%
- F1-score (attaque) : 48.71%

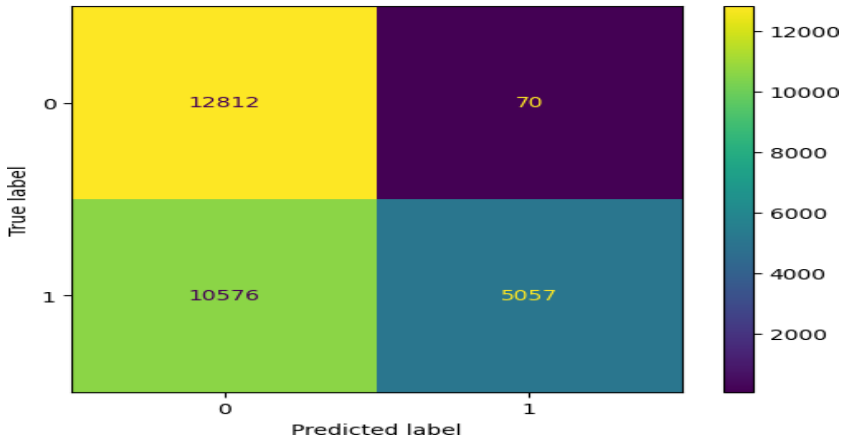


Figure 4.5 : Matrice de confusion du modèle Naive Bayes.

➢ **Réseau de neurones MLP :**

```
mlp = MLPClassifier(hidden_layer_sizes=(100, 50), max_iter=300)
```

```
mlp.fit(X_train, y_train)
y_pred_mlp =
mlp.predict(X_test)
```

- Accuracy : 85.35%
- F1-score (normal) : 85.32%
- F1-score (attaque) : 85.38%

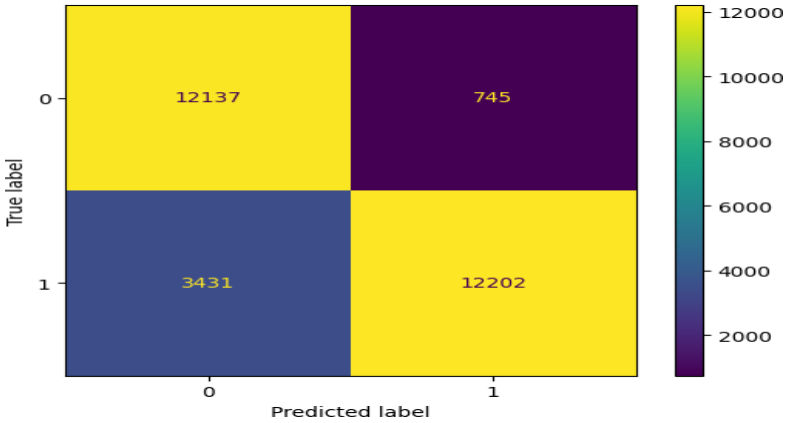


Figure 4.6 : Matrice de confusion du modèle MLP.

➢ **Fusion des modèles (Naive Bayes + MLP) :**

```
classe_finale = []
for i in range(len(y_pred_mlp)):
    if max(y_pred_prob_bayes[i]) > max(y_pred_prob_mlp[i]):
        classe_finale.append(y_pred_bayes[i])
    else:
        classe_finale.append(y_pred_mlp[i])
```

- Accuracy : 73.88%
- F1-score (normal) : 77.46%
- F1-score (attaque) : 68.94%

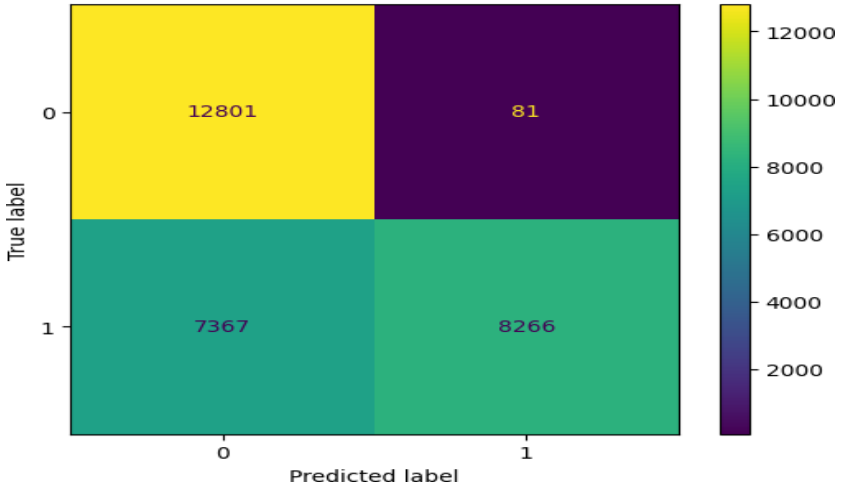


Figure 4.7 : Matrice de confusion du modèle fusionné.

➢ **Tableau récapitulatif des performances :**

Modèle	Accuracy (%)	F1-score (normal)	F1-score (attaque)
KNN	82.44	82.98	81.87
Naive Bayes	62.66	70.65	48.71
MLP	85.35	85.32	85.38
Fusion MLP + Bayes	73.88	77.46	68.94

Tableau 4.1 : Comparaison des performances des modèles.

III.3.Analyse des résultats :

- Le modèle MLP montre la meilleure précision et équilibre entre rappel et précision.
- Le modèle Naive Bayes est rapide mais limité par son hypothèse simplificatrice.
- Le modèle KNN est influencé par la dimensionnalité et le bruit.
- La fusion propose une alternative intéressante mais reste inférieure au MLP seul.

  Nous continuions a explorer les vecteurs de probabilités issus des prédictions des deux modèles, notamment ceux issus du MLP pour ajuster une fonction permettant d'améliorer les résultats de prédiction globale.

Conclusion :

Dans ce chapitre, nous avons évalué l'efficacité de plusieurs modèles d'apprentissage automatique pour la détection d'intrusions en utilisant la base de données NSL-KDD. L'étude a porté sur différents modèles tels que KNN, Naive Bayes et MLP, ainsi que sur une tentative de combinaison entre MLP et Naive Bayes. Les résultats ont montré que le réseau de neurones MLP offrait les meilleures performances en termes de précision et d'équilibre entre les différentes métriques, tandis que les autres modèles ont présenté des performances variables. Ce chapitre met en évidence l'importance de l'évaluation expérimentale dans le choix du modèle le plus approprié.

Conclusion générale

Conclusion générale :

En conclusion de ce travail, il apparaît clairement que la coopération entre les modèles d'intelligence artificielle constitue une approche prometteuse et efficace pour améliorer la précision et la performance des systèmes de détection d'intrusions dans les réseaux informatiques. À travers l'étude et l'application de modèles variés, tels que les réseaux de neurones profonds et les modèles bayésiens, nous avons démontré que l'intégration de ces techniques permet de compenser les faiblesses propres à chaque modèle, conduisant ainsi à une meilleure capacité de détection et à une réduction des taux d'erreur.

Cette recherche s'est appuyée sur une analyse approfondie des concepts de l'intelligence artificielle et de ses applications dans le domaine de la cybersécurité, en détaillant les algorithmes utilisés et les mécanismes d'apprentissage qui les sous-tendent. Une expérimentation concrète basée sur le jeu de données NSL-KDD, reconnu comme un standard pour l'évaluation des performances des systèmes de détection d'intrusions, a été réalisée.

Les résultats obtenus confirment l'efficacité de la coopération entre modèles pour classifier avec précision le trafic réseau entre connexions normales et attaques, renforçant ainsi la fiabilité des systèmes de protection et réduisant les risques sécuritaires auxquels font face les organisations.

Enfin, ce travail ouvre de nouvelles perspectives pour le développement de systèmes intelligents plus intégrés, capables de faire face aux menaces croissantes et aux nouvelles techniques d'attaques dans le domaine de la cybersécurité. Il constitue une base solide pour élargir les recherches vers l'intégration d'autres techniques de deep learning et l'amélioration des modèles d'adaptation automatique, contribuant ainsi efficacement à la sécurisation des systèmes d'information à l'avenir.

Bibliographie

Bibliographie

[1]- RIAHLA, Introduction à la sécurité informatique. PhD thesis, Département de phy sique/Infotronique IT/S6 de l'université de Boumerdes, 2008-2009.

[2]- https://www.commentcamarche.net/contents/1033-introduction-a-la-securiteinformatique

[3]- C. Asma, "Sécurité d'une application web à l'aide d'un systéme de détection d'intrusions comportementale," 2011-2012.

[4]- C. Llorens, L. Levier, D. Valois, and B. Morin, Tableaux de bord de la sécurité réseau. Editions Eyrolles, 2011.

[5]- F. Vinet, J.-C. Gaillard, J.-C. Denain, E. Clavé, F. Leone, S. Giyarsih, and S. Bachri, "En jeux et modalités spatiales de la reconstruction post-tsunami à banda aceh," Tsunarisque : le tsunami du, vol. 26, pp. 233–270, 2004.

[6]- https://www.cyberjobs.fr/actualites-articles/zoom-sur-les-5-objectifs-de-la-securite-informatique

[7]- Liran LERMAN, Les systèmes de détection d'intrusion basés sur du machine Learning, UNIVERSITÉ LIBRE DEBRUXELLES.

[8]https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.varonis.com%2Ffr%2Fbl og%2Fla-chaine-cybercriminelle-en-8-etapes&psig=AOvVaw1GyskcnI9hI48P7eVdp1aJ&ust=1745513159117000&source=images &cd=vfe&opi=89978449&ved=0CBUQjRxqFwoTCIic_6DN7owDFQAAAAAdAAAAABA E

[8]-

https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.varonis.com%2Ffr%2Fblog%2Fla-chaine-cybercriminelle-en-8-etapes&psig=AOvVaw3lQX-rBTVSfq_qXKVC_0Hx&ust=1748512981747000&source=images&cd=vfe&opi=89978449&ved=0CBUQjRxqFwoTCJiDjsb0xY0DFQAAAAAdAAAAABAE

[9]- https://web.maths.unsw.edu.au/~lafaye/CCM/attaques/attaques.htm

[10]- https://fr.wikipedia.org/wiki/Spam.html.

[11]- https://www.verizon.com/info/definitions/antivirus/

[12]- https://www.futura-sciences.com/tech/definitions/internet-firewall-474/

[13]-
https://www.google.com/url?sa=i&url=https%3A%2F%2Fgeekflare.com%2Ffr%2Ffirewall-introduction%2F&psig=AOvVaw2SlSO3ythI85meFRmk6mDB&ust=1745536883781000&source=images&cd=vfe&opi=89978449&ved=0CBUQjRxqFwoTCOi90NKl74wDFQAAAA AdAAAAABAE

[14]- Mme. BOUKHLOUF Djemaa, Une approche à base d'agents mobiles pour la sécurité des systèmes d'informations sur le web, Thèse de Doctorat, UNIVERSITE MOHAMED KHIDERBISKRA,2016.

[15]-
https://www.google.com/url?sa=i&url=https%3A%2F%2Fweb.maths.unsw.edu.au%2F~lafaye%2FCCM%2Fcrypto%2Fcrypto.htm&psig=AOvVaw3QHg8fJMW0epsfxe1flTTm&ust=1745537656976000&source=images&cd=vfe&opi=89978449&ved=0CBUQjRxqFwoTCPC2tri p74wDFQAAAAAdAAAAABAK

[16]- https://www.guill.net

[17]- https://fr.wikipedia.org/wiki/ Système_de_détection_d'intrusion.html

[18]- Liu, X., et al (2021). A Deep Learning-Based Method for Intrusion Detection in Smart Grid. IEEE Transactions on Smart Grid, 12(2), 1562-1570.

[19]- Alnajjar, K., et al (2021). Smart grid intrusion detection using deep learning : A survey. Sustainable Energy, Grids and Networks, 26, 100456.

[20]- https://dspace.ummto.dz/server/api/core/bitstreams/b038e9b4-4bac-46fd-bfec- 0c62abb53795/content

[21]-
https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.researchgate.net%2Ffigure% 2FArchitecture-classique-dun-IDS_fig2_30514332&psig=AOvVaw16-jf0b2hwQzJmSdVBENoj&ust=1745229107354000&source=images&cd=vfe&opi=89978449 &ved=0CBUQjRxqFwoTCIi6rJar5owDFQAAAAAdAAAAABAE

[22]- Debar, H., Dacier, M., & Wespi, A. (1999). "Towards a taxonomy of intrusion-detection systems." Computer Networks, 31(8), 805–822.

[23]- https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-94.pdf

[24]-_Denning, D. E. (1987). An Intrusion-Detection Model. IEEE Transactions on Software Engineering, 13(2), 222–232.

[25]- Northcutt, S. (2006). *An Introduction to Intrusion Detection Systems*. GIAC. Retrieved from https://www.giac.org/paper/gsec/4227/introduction-intrusion-detection-systems/106775

[26]- Interdata. (2024). *Fonctionnement des systèmes de détection d'intrusion (IDS)*. Récupéré de https://blog.interdata.fr/article-fonctionnement-ids-systeme-detection-intrusion

[27]- Microsoft Expériences, Tout savoir sur l'Intelligence Artificielle, (consulté le 09/02/2019),
disponiblesur:https://experiences.microsoft.fr/business/intelligenceartificielle-iabusiness/comprendreutiliser-intelligence-artificielle/

[28]- https://www.google.com/url?sa=i&url=https%3A%2F%2Felwatan-dz.com%2Fconsequences-du-developpement-de-lintelligence-artificielle-sur-leconomie-mondiale&psig=AOvVaw15qQVNAzZWTa-pB_Sr5pvM&ust=1748108979714000&source=images&cd=vfe&opi=89978449&ved=0CBU QjRxqFwoTCIDtysSTuo0DFQAAAAAdAAAAABAE

[29]- https://www.futura-sciences.com/tech/definitions/informatique-intelligence-artificielle- 555/

[30]- https://www.journaldunet.fr/web-tech/guide-de-l-intelligence-artificielle/1501845- intelligence-artificielle-forte-definition/

[31]- https://docs.microsoft.com/fr-fr/azure/machine-learning/concept-deep-learning-vs- machine-learning

[32]-
https://www.google.com/url?sa=i&url=https%3A%2F%2Fsti.eduscol.education.fr%2Fsi-ens- paris-saclay%2Fressources_pedagogiques%2Fdossier-intelligence-artificielle&psig=AOvVaw0C4Aay2NKjOKIqjq61YWSN&ust=1748511450110000&source
=images&cd=vfe&opi=89978449&ved=0CBUQjRxqFwoTCPi9yI7vxY0DFQAAAAAdAAAAABAE

[33]- S. Dua and X. Du, Data mining and machine learning in cybersecurity. CRC press, 2016.

[34]- https://www.cnil.fr/fr/definition/apprentissage-automatique

[35]- https://waytolearnx.com/2018/11/difference-entre-apprentissage-supervise-et-non-supervise.html

[36]- https://course.elementsofai.com/fr/4/1

[37]- Géron, Aurélien. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow.* 2nd Edition, O'Reilly Media, 2019.

[38]-
https://www.google.com/url?sa=i&url=https%3A%2F%2Flarevueia.fr%2Frandom-forest%2F&psig=AOvVaw26AvcNhlp7oBuzM1RisxYv&ust=1748186278746000&source=i
mages&cd=vfe&opi=89978449&ved=0CBUQjRxqFwoTCPi1zN6zvI0DFQAAAAAdAAAA ABAE

[39]- Université Saad Dahlab De BLIDA1 Faculté Des Sciences Département D'informatique. Mémoire réalisé par :Douid Rania: Système de détection d'intrusion réseau basé sur L'algorithme de Classification KNN: 2018-2019

[40]-
https://www.google.com/url?sa=i&url=https%3A%2F%2Fdataanalyticspost.com%2FLexique%2Fk-nearest-
neighbours%2F&psig=AOvVaw23V_HiAKxebVoSc32ZUvl7&ust=1748197318811000
&sou rce=images&cd=vfe&opi=89978449&ved=0CBUQjRxqFwoTCPjk-a3dvI0DFQAAAAAdAAAAABAE

[41]- https://mrmint.fr/naive-bayes-classifier

[42]- BAH DIDI EL MOKHTAR SALEM.Système de détection d'intrusion avec une

approche D'apprentissage automatique Université Saad Dahlab De BLIDA1

[43]- BOUKERTOUTA Mohammed Mémoire de Fin d'études MasterDétection des intrusions basée sur l'apprentissage automatique dans les systèmes IdO(Internet des Objets)Juin 2022

[44]- https://www.callmenewton.fr/guide-ia/reseau-neurones-profond/#Introduction_aux_reseaux_de_neurones_profonds

[45]- https://www.google.com/url?sa=i&url=https%3A%2F%2Ffr.m.wikiversity.org%2Fwiki%2FFichier%3AReseau_de_neurone_profond.png&psig=AOvVaw31n7Za4yNYRxy2YqkDrm5U&ust=1748536241793000&source=images&cd=vfe&opi=89978449&ved=0CBUQjRxqFwoTCPi5sLTLxo0DFQAAAAAdAAAAABAL

[46]- https://botpress.com/fr/blog/reseau-neuronal-profond#:~:text=Un%20r%C3%A9seau%20neuronal%20profond%20(RNP,cerveau%20humain%20traite%20les%20informations.

[47]- https://www.jedha.co/formation-ia/reseau-neurones-deep-learning

[48]- https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.researchgate.net%2Ffigure% 2FArchitecture-du-reseau-de-neurone-a-trois-couches-de-configuration-13-10-1-utilise_fig7_259972248&psig=AOvVaw3vbslgXRTVOzW7LsWeL0pL&ust=1748536765367000&source=images&cd=vfe&opi=89978449&ved=0CBUQjRxqFwoTCLiFp6LNxo0DFQ AAAAAdAAAAABAE

[49]- Ekpao Anani, PASSIGUE , Analyse et détection de pourriels textuels dans les réseaux sociaux par apprentissage, UNIVERSITÉ DU QUÉBEC EN OUTAOU 18 Août 2015

[50]- file:///C:/Users/hp/Downloads/Bayesien.pdf

[51]- https://python.sdv.univ-paris-diderot.fr/01_introduction

[52]- https://www.python.org/about/apps/

[53]- https://www.geeksforgeeks.org/python-applications-in-real-world

[54]- https://aws.amazon.com/fr/what-is/python/

[55]- https://datascientest.com/google-colab-tout-savoir

[56]- https://x.com/GoogleColab/header_photo

[57]- https://www.analyticsvidhya.com/blog/2020/03/google-colab-machine-learning-deep learning/

[58]- https://academy.hsoub.com/apps/productivity/

[59]- « Zahaf Mohamed El-Bachir » RAPPORT DE MINI-PROJET: Système de Détection 'Inrusion Réseau Basé sur l'ApprentissageAutomatique(2022-2023).