REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEURE ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE IBN KHALDOUN - TIARET

MEMOIRE

Présenté à :

FACULTÉ MATHEMATIQUES ET INFORMATIQUE
DÉPARTEMENT D'INFORMATIQUE

Pour l'obtention du diplôme de :

MASTER

Spécialité : Réseaux et Télécommunication

Par :

BOUROUBA Hadjer
CHAOUCHE Ouidad

Sur le thème

Optimisation des IDS du Cloud Computing par les techniques de machines Learning

Soutenu publiquement le 07 / 09/ 2020 à Tiaret devant le jury composé de :

Mr DAOUD Mohamed Amine	Université Ibn Khaldoun	Encadreur
Mr ALEM Abdelkader	Université Ibn Khaldoun	Président
Mr Nassane Samir	Université Ibn Khaldoun	Examinateur

2019-2020

Remerciement…

En premier lieu, nous remercions ***DIEU*** *Pour le tout puissant , maître des cieux et de la terre,*
Qui nous a éclairé le chemin et permis de mener à bien ce travail.
Nous tenons également à exprimer toute notre reconnaissance à notre Promoteur Monsieur.
DAOUD Mohamed Amine, *qui s'est toujours montré disponible et à l'écoute durant toute la réalisation de ce présent mémoire et qui a su guider et structurer nos idées grâce à ses précieux conseils.*
Nos profonds remerciements s'adressent aux membres de jury qui nous font honneur en acceptant d'évaluer notre travail.
Un énorme merci à nos familles et amis pour leurs éternel soutient et la confiance qu'ils ont en nos capacité.
Enfin, nous remercions tous ceux qui ont contribué de près ou de loin à l'aboutissement de ce modeste travail.

Je dédie ce travail à …

A l'homme de ma vie, mon exemple éternel, mon soutien moral et source de joie et de bonheur, que dieu te garde dans son vaste paradis à toi mon père.

A la lumière de mes jours, la source de mes efforts, la flamme de mon cœur, Cela a toujours sacrifié ma vision la plus réussie, ma vie et mon bonheur maman que j'adore.

Aux personnes dont j'ai bien aimé la présence dans ce jour à tous mes frères et mes sœurs

A ceux qui m'ont toujours aidé et encouragé, qui étaient toujours à mes côtés, qui m'ont accompagnaient durant mon chemin d'études supérieures.

A tous ceux qui, d'une manière ou d'une autre, ont contribué à l'élaboration de ce mémoire ainsi qu'à la réussite de cette formidable année universitaire, mes aimables amis à vous Ouidad et Rahil & Abdelkader, sans oublier Hamid Qui m'a beaucoup aidé

A tous ceux qui compte pour moi et qui n'ont pas pu être cités ici.

Hadjer

Je dédie ce travail à …

Ma chère mère et mon père pour l'éducation qui l'on prodigue avec tous les moyens et au prix de tous les sacrifices

Qu'ils consentirent à mon égard, pour le sens du devoir qu'il mon enseigne depuis mon enfances,

À mon frère en leurs espérant le plein succès dans leur vie.

À mon binôme Hadjer pour les beaux moments que nous avons passé dans la réalisation de ce travail.

Et toutes mes amies qui m'ont toujours encouragée et soutenu.

Ouidad

Table des matières

Chapitre 01 : Cloud Computing

Chapitre 02 : les systèmes de détection d'intrusion

Chapitre 03: Machine Learning

Chapitre 04 : l'approche proposée

Liste des abréviations

IAAS : *Infrastructure as a Service*

PAAS: *Platform as a Service*

SAAS: *Software as a Service*

IPS : *Intrusion Prévention System*

IDS : *systèmes de détection d'intrusion*

TE : *Le taux d'exactitude*

TFA : *Le taux de fausse alerte*

DR : *Détection Rate*

IA : *intelligence artificielle*

ML: *Machine Learning*

DL: *Deep Learning*

RL: *Renforcement Learning*

Liste des figures

Liste des tableaux

Résumé

Avec le déploiement croissant du Cloud et l'utilisation intensive d'Internet, le cyber sécurité n'est pas seulement un besoin mais aussi une nécessité pour les organisations qui offrent leurs services via Internet et sont constamment la cible cyber-attaques.

Un système de détection d'intrusion (IDS-Intrusion Détection System) vise à identifier et à répondre aux cyber-attaques ciblant les services du Cloud. Le problème de la détection des intrusions revient à classer le flux d'activité des services du Cloud en deux catégories : flux d'activité normal ou flux d'attaques. Le problème formulé peut être considéré comme un problème de classification, dont l'objectif est d'avoir une bonne optimisation dans lequel la fonction objective est de maximiser le taux de détection. **N**otre objectif est de construire une IDS effective qui accompagne l'évolution des systèmes de détection afin d'améliorer le taux de détection et l'adaptabilité par des techniques de la machine Learning. Nous avons proposé un modèle IDS qui nous permet d'améliorer le taux de détection par rapport aux travaux précédents. On applique notre modèle sur un data set très récent CICIDS2017. Le modèle est une combinaison d'une technique de réduction de dimensions PCA et le classificateur SVM.

Mots-clés : Cloud, IDS, ML, PCA, SVM, CICIDS2017

Abstract

With the increasing deployment Our case study is the Cloud and the intensive use of the Internet, IT security is not only a need but also a necessity for the organizations which offer their services via Internet and are constantly the target Cyber-attacks.

An intrusion detection system (IDS-Intrusion Detection System) aims to identify and respond to Cyber-attacks targeting Cloud services The problem of intrusion detection amounts to classifying the activity flow of Cloud services in two categories: normal activity flow or attack flow. The problem thus formulated can be considered as a classification problem, which can itself be formulated as an optimization problem in which the objective function is to maximize the detection rate. We have proposed an IDS model which allows us to improve the detection rate compared to previous work. In addition, In addition, we have very large data set and with SVM managed several calculates them where we applied pca to reduce the data set and to eliminate redundant data and guarantee the survival of cloud computing in the event of a major disaster affecting the computer system . It is a question of restarting the activity as quickly as possible with the minimum of data loss. This plan is one of the essential points of the IT security policy of Cloud Computing. The required work consists in applying the analysis technique of the main PCA components which will be taken into account in the classification stage by the SVM classifier.

Keywords: Cloud, IDS, ML, PCA, SVM.CICIDS2017

INTRODUCTION GENERALE

Introduction générale

Les technologies de l'information et de la communication ont été évoluées et révolutionnée nos modes de vie et de travail. Le Cloud, est apparu, ces dernières années, comme un nouveau modèle de gestion et d'utilisation des systèmes informatiques.

Le concept consiste à déporter, sur des serveurs distants, les traitements et stockages habituellement effectués en local afin d'y accéder sous forme de service. Il Consiste à proposer des services informatiques sous forme de services à la demande, accessibles de n'importe quand n'importe où et par n'importe quel moyen qui en raison de une connexion internet. Si le Cloud Computing apporte de nombreux bénéfices « rêvés » par les entreprises, Beaucoup d'entreprises restent aporétique sur le Cloud Computing. La principale raison est la sécurité des données car les DSI (direction des systèmes d'informations) restent péteux de penser que leurs données critiques sont dans un endroit incontrôlé et souvent inconnu Et aussi à cause des cyber-attaques.

Le cyber sécurité dans le Cloud Computing est un service en forte croissance qui fournit un avantage pour la communauté du cloud. Cela comprend notamment la protection des informations critiques contre le vol, la fuite de données et la suppression. Dans ce contexte, il devient nécessaire de mettre en œuvre une solution pour éviter les attaques qui se produisent en utilisant ses ressources ou données, les systèmes de détection d'intrusion (IDS) sont en charge de détecter les attaques.

L'apprentissage automatique contient de nombreuses applications pour percevoir leur environnement (voir, reconnaître des choses comme des visages, des diagrammes, des langages naturels, des caractères manuscrits, des formes); Jouet ; ingénierie informatique ; Déplacez les robots; Parmi eux, la détection des fraudes, qui pourraient affecter de manière significative l'avenir des organisations.

L'apprentissage automatique permet aux gens d'accomplir des tâches plus rapidement et plus efficacement, et ce domaine comprend la construction d'un modèle de données grâce à l'utilisation d'un algorithme dans lequel nous avons présenté un modèle pour développer la sécurité par IDS pour détecter l'attaque dans Cloud

Notre travail consiste à proposer d'un IDS en utilisant les algorithmes des machines Learning (SVM et PCA) afin de régler certains problèmes tels que le nombre élevé de faux positifs et les faux négatifs.

Cet objectif permet d'assister et renforcer les analystes humains dans les domaines de la surveillance, de la détection et de la réaction aux attaques potentielles dans le Cloud, Pour cela, nous avons organisé notre travail en quatre chapitres :

— Dans le premier chapitre, on présente les notions et les concepts de base du Cloud Computing, ses services, ses types, et ses caractéristiques.

—Dans le deuxième chapitre, on présente les notions et concepts de base du système de détection d'intrusion, ses définitions, ses architecture, …

— Le troisième chapitre décrit les notions et concepts de base de la machine Learning ….

— Le quatrième chapitre est destiné à l'implémentation du notre approche proposée

PARTIE I

Recherche bibliographique (état de l'art)

Chapitre 1 :

Cloud Computing

Chapitre 01 : Cloud Computing

I. Introduction :

L'informatique a toujours évolué, au gré des nouvelles technologies, pour répondre à de nouvelles demandes. Le Cloud Computing se traduit en français par " Informatique dans les nuages", est un concept qui représente l'accès à la demande, à des informations et des services situés sur un serveur distant.

Le Cloud Computing est devenu ainsi, le sujet le plus débattu, aujourd'hui, dans le secteur des technologies de l'information. Avec le développement du le Cloud Computing, des problèmes de sécurité sont apparus mettant en péril, le cyber sécurité fait le point sur l'un des principaux challenges du Cloud.

II. Définition du Cloud Computing :

Le Cloud Computing, littéralement l'informatique dans les nuages est un concept qui consiste à déporter sur des serveurs distants, des stockages et des traitements informatiques traditionnellement localisés sur des serveurs locaux ou sur le poste de l'utilisateur. Il consiste à proposer des services informatiques sous forme de service à la demande, accessible de n'importe où, n'importe quand et par n'importe qui, grâce à un système d'identification, via un PC et une connexion à Internet [1]

Figure 1: Cloud Computing

Selon la définition du National Institute of Standards and Technologie (NIST), Le Cloud Computing est un modèle qui permet un accès réseau pratique sur demande à un pool partagé de ressources informatiques configurables (par exemple, des réseaux, des serveurs, du stockage, des applications et des services) [2]. Ce modèle de Cloud favorise la disponibilité, il est composé de cinq caractéristiques essentielles, de trois modèles de service et de quatre modèles de déploiement. [3]

III. Services du Cloud Computing :

III.I. Infrastructure as a Service (IaaS) :

Le NIST dit que l'IaaS donne au consommateur la capacité de provisionner le traitement, le stockage, les réseaux et d'autres ressources informatiques fondamentales où le consommateur déploie et exécute des logiciels arbitraires qui peuvent inclure des systèmes d'exploitation et des applications. L'IaaS fournit aux consommateurs des serveurs et des réseaux physiques ou virtuels loués ainsi qu'un stockage dans un environnement Cloud sur une base de paiement à l'utilisation. IaaS est le modèle de service le plus élémentaire que les entreprises technologiques utilisent pour accéder à la puissance de calcul brute sans les responsabilités d'installation ou de maintenance**.** [4]

III.II. Platform as a Service (PaaS)**:**

Le PaaS signifie « Platform as a Service » est une architecture composée de tous les éléments nécessaires pour soutenir la construction, la livraison, le déploiement et le cycle de vie complet des applications et des services exclusivement disponibles à partir d'internet.
Le PaaS offre des facilités à gérer le déroulement des opérations lors de la conception, du développement, du test, du déploiement et de l'hébergement d'applications web à travers des outils et des services tels que :

-Le travail collaboratif (« team collaboration »).
-L'intégration des services web et bases de données.

Ces services sont fournis au travers une solution complète destinée aux développeurs et disponible immédiatement via l'internet. [5]

III.III. Software as a Service (SaaS)**:**

Le NIST définit le SaaS comme un modèle de service dans lequel un consommateur ne gère ni ne contrôle l'infrastructure Cloud sous-jacente, y compris le réseau, les serveurs, les systèmes d'exploitation, le stockage ou même les capacités d'application individuelles Les applications SaaS offrent de nombreuses options de configuration et des environnements de développement qui permettent aux clients de coder leurs propres modifications et ajouts. Les utilisateurs accèdent au service via un navigateur Web ou une application achetant le service par siège ou par utilisateur. La beauté du SaaS réside dans sa simplicité, car l'installation locale du logiciel SaaS n'est pas nécessaire. [4]

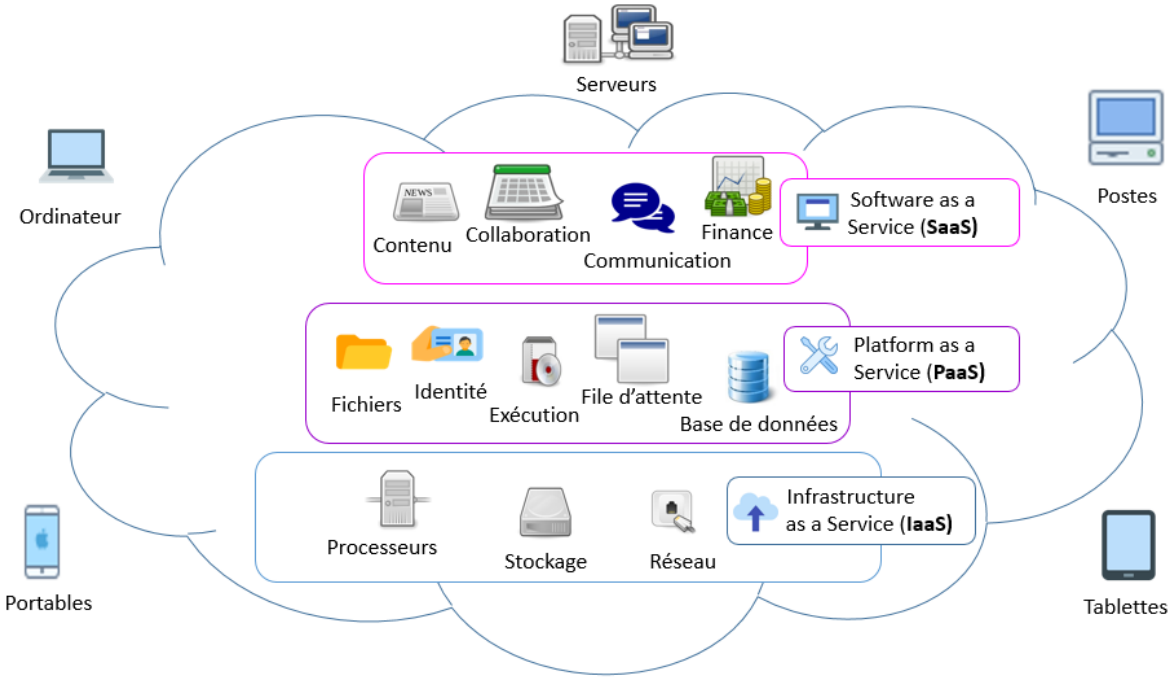

Figure 2: Les Service du Cloud Computing

IV. Modèles de déploiement :

1. Cloud privé :

Quand le Cloud est « privé », une seule organisation utilise la plateforme de Cloud. Ce Cloud s'appuie sur des ressources informatiques (infrastructure, serveurs, etc.) propres à l'organisation ou qui lui sont dédiées, et qui peuvent être gérées et administrées par l'organisation elle-même ou par un tiers. Dans tous les cas, l'organisation garde la maîtrise de son infrastructure et de ses données.

Le Cloud privé est le type de déploiement le plus simple à gérer pour les entreprises, que ce soit en termes de sécurité ou de respect de la législation, dans la mesure où il reste intégralement maîtrisé par l'organisation qui le met en œuvre. [6]

2. Cloud communautaire :

L'infrastructure Cloud est partagée par plusieurs organisations, réunies au sein d'une communauté et partageant des préoccupations spécifiques communes (par exemple, la mission, les exigences de sécurité, des politiques et des considérations de conformité). Elle peut être gérée par les entreprises elles-mêmes ou par un tiers et peut exister sur site ou hors site. [2]

3. Cloud public :

Le Cloud public, quant à lui, consiste à mutualiser les serveurs, les systèmes de stockage et les applications entre un grand nombre de clients. Le client final n'a généralement aucun moyen de savoir quels autres usagers sont présents sur le serveur sur lequel ses tâches

sont exécutées. La plupart des offres de Cloud public sont de plus des offres standards dont les fonctionnalités et les caractéristiques sont définies uniquement par le prestataire en fonction du marché qu'il vise. S'il a un souhait particulier que le fournisseur ne prend pas en charge, le client de Cloud public devra généralement changer de prestataire ou revenir à un service géré dans ses locaux.

Le Cloud public soulève de nombreuses questions en termes de sécurité et de respect de la vie privée car son utilisation conduit globalement à une perte de contrôle du client sur le traitement de ses données. [6]

4. Cloud hybride :

L'infrastructure Cloud est une composition de deux ou plusieurs Cloud (privés, communautaires ou publics). Ceux-ci demeurent des entités uniques mais sont connectés par une technologie normalisée ou exclusive qui permet le partage des données et des applications des nuages) [2].

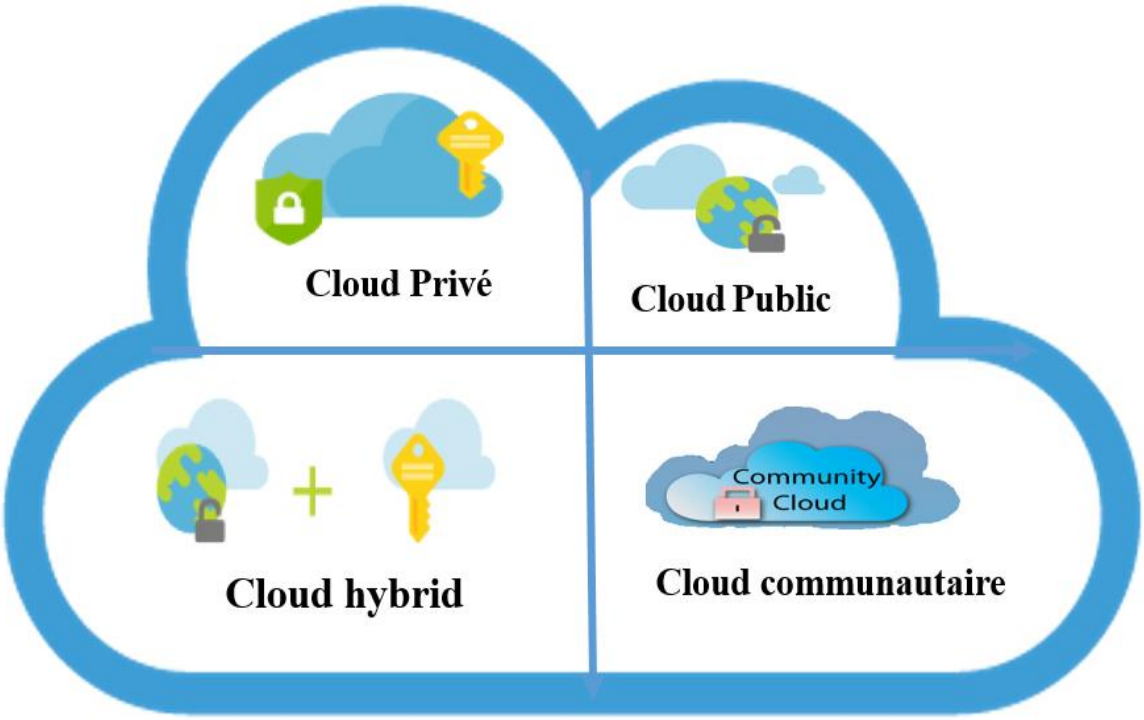

Figure 3 : Les Types du Cloud Computing.

V. Les caractéristiques essentielles du Cloud Computing :

a. Libre-service à la demande :

Un consommateur peut fournir unilatéralement des capacités informatiques telles que l'heure du serveur et le stockage en réseau, au besoin, sans nécessiter d'interaction humaine avec le fournisseur de chaque service.

b. Large accès au réseau :

Les capacités sont disponibles sur le réseau et sont accessibles via des mécanismes standards qui favorisent l'utilisation par des plates-formes clientes minces ou épaisses et hétérogènes (par exemple, les téléphones mobiles, les ordinateurs portables et les assistants personnels numériques).

c. Mise en commun des ressources :

Les ressources informatiques du fournisseur sont regroupées pour servir plusieurs consommateurs à l'aide d'un modèle multi-locataire, avec différentes ressources physiques et virtuelles attribuées dynamiquement et réattribuées en fonction de la demande du consommateur. Il y a un sentiment d'indépendance de localisation puisque le client n'a généralement aucun contrôle ou connaissance sur l'emplacement exact des ressources fournies, mais peut être capable de spécifier l'emplacement à un niveau d'abstraction plus élevé (par exemple, pays, état ou centre de données). Des exemples de ressources comprennent le stockage, le traitement, la mémoire, la bande passante du réseau et les machines virtuelles.

d. Une souplesse rapide :

Les capacités peuvent être rapidement et élastiquement provisionnées, parfois automatiquement, pour évoluer rapidement et être disponible dans un temps réduit. Pour le consommateur, les capacités disponibles pour l'approvisionnement semblent souvent illimitées et peuvent être achetées en n'importe quelle quantité et à tout moment.

e. Service mesuré :

Les systèmes de Cloud contrôlent et optimisent automatiquement l'utilisation des ressources en exploitant une capacité de comptage à un niveau d'abstraction approprié au type de service (par exemple stockage, traitement, bande passante et comptes d'utilisateurs actifs) [2]

VI. Avantages inconvénients du Cloud Computing

VI.1. Les avantage du Cloud Computing :

Le Cloud Computing offre de multiple avantage aux entreprises et aux utilisateurs finaux :

- La réduction des coûts.
- L'accessibilité.
- L'élasticité.
- Le déploiement rapide et la simplicité d'intégration.
- La disponibilité du service.
- L'adoption rapide par les utilisateurs finaux. [7]

VI.2. Les inconvénients du Cloud Computing :

•***Connexion Internet obligatoire :*** L'accès aux services du Cloud Computing se fait par le baillet de l'Internet. La rupture de la connexion Internet implique la perte d'accès aux applications et aux données [8]

• ***Sécurité des données :*** Dans le cas du Cloud Computing, l'entreprise devra connecter ses postes à Internet, et les exposer à un risque d'attaque et d'intrusion, et de vol de données par piratage [8]
• ***La bande passante peut faire exploser votre budget :*** La bande passante qui serait nécessaire pour stocker les données dans le Cloud est gigantesque, et les coûts seraient tellement importants qu'il est plus avantageux d'acheter le stockage plutôt que de payer quelqu'un d'autre pour s'en charger. [8]
• ***Stockage de données*** **:** Le stockage physique des données dans le Cloud est effectué par les fournisseurs des services ce qui limite la manipulation de ces derniers par les clients [9]
• ***Identification des clients*** **:** Avec l'utilisation croissante du Cloud et l'utilisation multi location de ces ressources, il devient de plus en plus difficile d'identifier par qui et de quel endroit les données ont était modifiées [9]
• ***Taille de l'entreprise :*** Si votre entreprise est grande alors vos ressources sont grandes, ce qui inclut une grande consommation du Cloud. Vous trouverez peut-être plus d'intérêt à mettre au point votre propre Cloud plutôt que d'en utiliser un externalisé. Les gains sont bien plus importants quand on passe d'une petite consommation de ressources à une consommation plus importante [10]

VII. La sécurité du Cloud Computing :

Une des plus grandes préoccupations des utilisateurs sur le Cloud Computing est sa sécurité. Dans les centres de données internet. Les fournisseurs de services offrent les grilles et les réseaux seulement, et les appareils restants doivent être prépares par les utilisateurs eux-mêmes, y compris les serveurs, le pare-feu, les logiciels, les périphériques de stockage, etc.

Certains utilisateurs utilisent l'isolement physique pour protéger leur serveur. Du point de vue de la technologie, la sécurité des données des utilisateurs peut être réfléchie dans les règles suivantes [11]*:*

- ***la confidentialité des données de stockage des utilisateurs :*** Le stockage des données d'utilisateurs ne peut pas être lu ou modifié par d'autres personnes (y compris l'opérateur).
- ***la confidentialité des données d'utilisateur lors de l'exécution :*** les données d'utilisateur ne peuvent pas être lu ou modifié par d'autres personnes lors de l'exécution (c.à.d. Charger dans la mémoire système).
- ***le secret des données privées d'utilisateur lors du transfert à travers le réseau :*** il comprend la sécurité de transfert des données vers le Cloud Computing. Il ne peut pas être affiché ou modifié par d'autres personnes.
- ***authentification et autorisation nécessaire pour les utilisateurs d'accéder à leurs données :*** Les utilisateurs peuvent accéder efficacement à leurs données et peuvent autoriser d'autres utilisateurs d'y accéder.

VIII. Les solutions de sécurité dans le Cloud :

Les infrastructures Cloud sont, comme tout système informatique réparti, exposées à des problématiques de sécurité. En effet, le nombre et la diversité des utilisateurs de ces infrastructures ainsi que la quantité importante de composants matériels et logiciels impliquent la présence de menaces de sécurité. Pour se prémunir

des attaques reposant sur l'utilisation des réseaux, des mécanismes de sécurité réseau sont déployés pour protéger les données hébergées dans les infrastructures virtuelles :

- les pare-feu sont responsables du filtrage de paquets afin de contrôler l'accès réseau.
- Les systèmes de détection d'intrusion sont en charge de détecter les attaques survenant sur les canaux de communication.
- L'objectif des administrateurs sécurité (des clients ou des fournisseurs) est de prévenir et de détecter les attaques tout en ne perturbant pas le bon fonctionnement du Cloud. [12]
- Arrêtez l'infiltration par les systèmes de prévention des intrusions adossés à des IDS [13]

IX. Conclusion :

Au cours de cette première partie, nous avons fourni une base théorique sur le Cloud Computing. Ce modèle Cloud se compose de cinq caractéristiques essentielles, trois modèles de services et quatre modèles de déploiement. Le modèle Cloud Computing offre la promesse d'économies massives combinées à une agilité informatique accrue. Cependant, Donc il est nécessaire de sécuriser les services et ressources Cloud en utilisant des mécanismes de sécurité tel que les systèmes de détection d'intrusion (IDS) qu'il sera l'intérêt du prochain chapitre.

Chapitre 2 :

Les systèmes de détection d'intrusion

Chapitre 02 : les systèmes de détection d'intrusion

I. Introduction :

Aujourd'hui, de nombreuses organisations déplacent vers le Cloud. Cela rend leurs traitements beaucoup plus facilement par un accès à distance. Cependant, il souffre également de nouvelles menaces des attaques. Les risques d'intrusion sont donc plus avec l'aptitude des nouvelles attaques à cause de sa nature distribuée. Diverses techniques d'analyses de sécurité existent dont les systèmes de détection d'intrusion sont essentiels pour le Cloud. En utilise IDS pour contrer les attaques malveillantes dans les réseaux virtuels du Cloud.

II. Définition d'un système de détection d'intrusions :

La détection d'intrusion est le processus de la surveillance des événements qui se produisent dans un ordinateur ou dans un réseau et de les analysent pour des signes d'intrusions, qui sont définis comme des tentatives qui compromettent la confidentialité, l'intégrité ou la disponibilité ou bien qui dépassent les conditions de la sécurité d'un ordinateur ou un réseau [14]

IDS signifie Intrusion Détection System. Il s'agit d'un équipement permettant de surveiller l'activité d'un réseau ou d'un hôte donné, afin de détecter toute tentative d'intrusion et éventuellement de réagir à cette tentative [15]

IDS est un appareil ou une application qui alerte l'administrateur en cas de faille de sécurité, de violation de règles ou d'autres problèmes susceptibles de compromettre son réseau informatique. [16]

IDS est un périphérique ou processus actif qui analyse l'activité du système et du réseau pour détecter toute entrée non autorisée et / ou toute activité malveillante. La manière dont un IDS détecte des anomalies peut beaucoup varier ; cependant, l'objectif principal de tout IDS est de prendre sur le fait les auteurs avant qu'ils ne puissent vraiment endommager vos ressources [17]

IDS analyse les configurations des systèmes et leurs vulnérabilités, ainsi, ils vérifient l'intégrité des fichiers. Ils peuvent reconnaitre des schémas d'attaque classiques. Pour ce faire, ils analysent les comportements anormaux et suivent les violations de règles par les utilisateurs. [17]

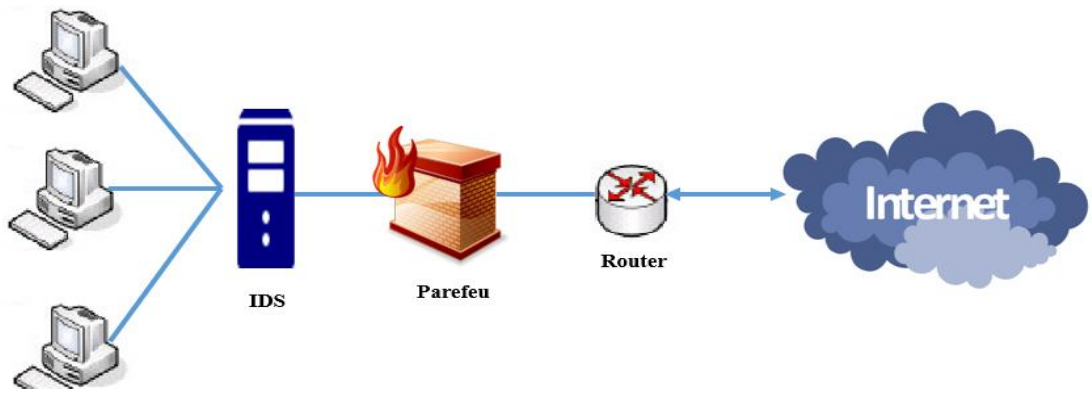

Figure 4: Système de détection d'intrusions

Ses fonctions principales peuvent être résumées dans les points suivants :

- Journaliser l'événement source d'information et vision des menaces courantes.
- Comparer les données collectées avec des données de référence qui correspondent à des opérations interdites ou autorisées.
- Avertir un système avec un message (Exemple : appel SNMP[1]).
- Avertir un humain avec un message (Courrier électronique, SMS, interface web,…).
- Amorcer certaines actions sur un réseau ou hôte (Exemple : mettre fin à une connexion réseau, ralentir le débit des connexions,…).
- Appliquer des mesures correctives en cas de détection d'une intrusion. [17]

III. Architecture Intrusion Détection System :

Plusieurs schémas ont été proposés pour décrire les composants d'un système de détection d'intrusions. Parmi eux, nous avons retenu celui issu des travaux d'Intrusions Détection exchange format Working Group (IDWG) de l'Internet Engineering Task Force (IETF) comme base de départ, car il résulte d'un large consensus parmi les intervenants du domaine. [18]

L'objectif des travaux du groupe IDWG est la définition d'un standard de communication entre certains composants d'un système de détection d'intrusions. La figure (5) illustre ce modèle et permet d'introduire un certain nombre de concepts :

[1] Simple Network Management Protocol (abrégé SNMP), en français « protocole simple de gestion de réseau », est un protocole de communication qui permet aux administrateurs réseau de gérer les équipements du réseau, de superviser et de diagnostiquer des problèmes réseaux et matériels à distance. SNMP est utilisé pour administrer les équipements et/ou surveiller le comportement des équipements.

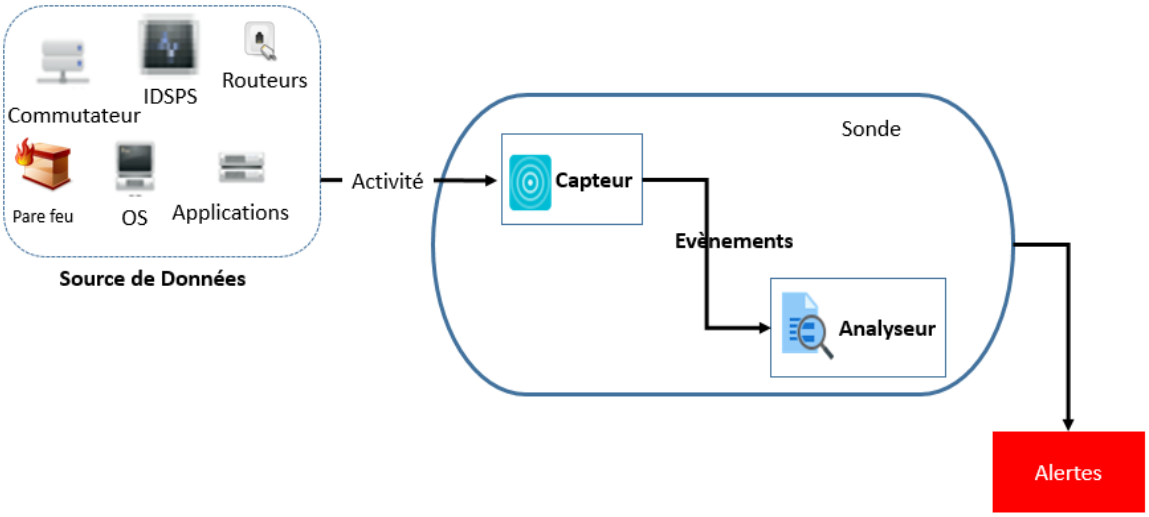

Figure 5: Modèle générique de la détection d'intrusions proposé par l'IDWG

L'architecture IDWG d'un système de détection d'intrusions contient des capteurs qui envoient des événements à un analyseur. Les capteurs couplés avec un analyseur forment une sonde, cette dernière envoie des alertes qui la notifient à un opérateur humain. Les différents éléments de cette architecture sont :

- ✓ **Source de données :** dispositif générant de l'information sur les activités des entités du système d'information.
- ✓ **Capteur :** génère des événements en filtrant et formatant les données brutes provenant d'une source de données.
- ✓ **Evénement :** message formaté et renvoyé par un capteur. C'est l'unité élémentaire utilisée pour représenter une étape d'un scénario d'attaques connu.
- ✓ **Analyseur :** c'est un outil logiciel qui met en œuvre l'approche choisie pour la détection (comportementale ou par scénarios), il génère des alertes lorsqu'il détecte une intrusion.
- ✓ **Sonde :** un ou des capteurs couplés avec un analyseur.
- ✓ **Alerte :** message formaté émis par un analyseur s'il trouve des activités intrusives dans une source de données.

Dans ce modèle qui représente le processus complet de la détection ainsi que l'acheminement des données au sein d'un IDS. L'administrateur configure les différents composants (capteur(s), analyseurs(s)) selon une politique de sécurité bien définie. Les capteurs accèdent aux données brutes, les filtrent et les formatent pour ne renvoyer que les événements intéressants à un analyseur. Les analyseurs utilisent ces événements pour décider de la présence ou non d'une intrusion et envoient dans le cas échéant une alerte, qui notifie

l'opérateur humain, une réaction éventuelle peut être menée automatiquement ou manuellement. [19]

IV. Classification des systèmes de détections d'intrusion :

La classification adoptée selon différents critères qui ne sont pas forcément mutuellement exclusifs n'est pas, elle présente tour à tour et au même niveau les catégories caractérisant chaque IDS, et utilise les critères suivants (figure 6). [19]

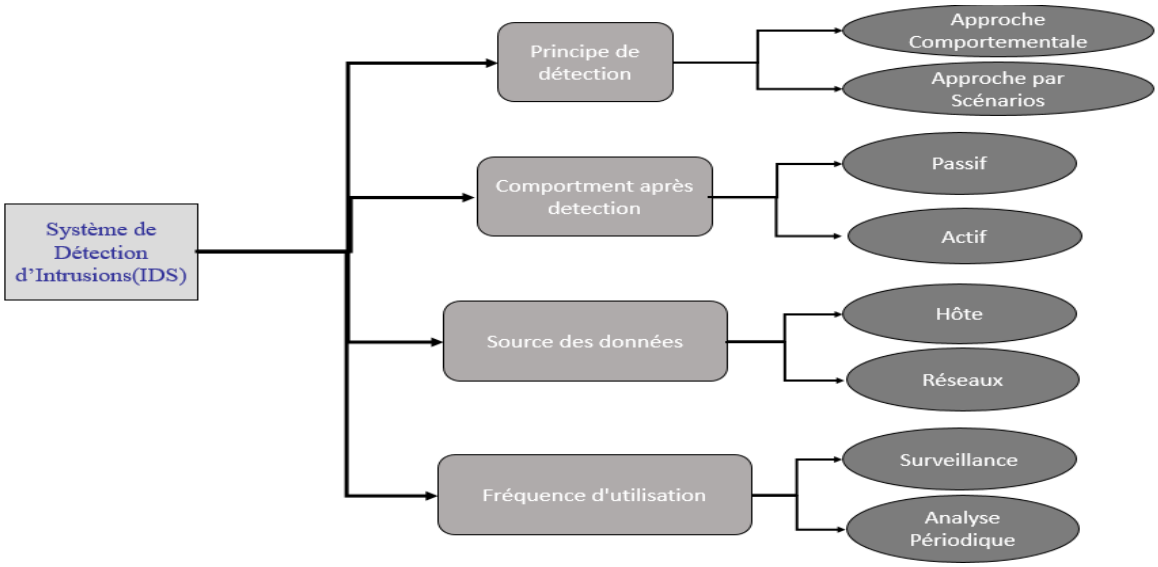

Figure 6: Taxonomie des systèmes de détection d'intrusions.

- La méthode de détection utilisée (principe).
- Le comportement après détection
- La source des données à analyser.
- La fréquence de l'analyse.

1. Les méthodes d'analyses des systèmes de détections d'intrusion (principe) :

Deux techniques de détection d'intrusion sont généralement mises en œuvre par les IDS courants. La Méthode de détection décrit les caractéristiques de l'analyseur, lorsque le système de détection d'intrusion utilise des informations sur le comportement normal des systèmes qu'il surveille, on qualifie de comportement (détection par comportement). Lorsque le système de détection d'intrusion utilise des informations sur les attaques, on qualifie (détection par signature).

a) Détection par signature (scénario)

La détection par signature considère comme normal tout ce qui n'est pas hostile, et elle adopte la politique suivante : «si ce n'est pas dangereux, alors c'est normal ». Donc, il est impératif de disposer d'une base de toutes les attaques connues.

Dans la détection par signature (aussi appelée détection de mauvaise utilisation), l'IDS analyse l'information recueillie et la compare avec une base de données de signatures (motifs définis, caractéristiques explicites) d'attaques connues (i.e., qui ont déjà été documentées), et toute activité correspondante est considérée comme une attaque (avec différents niveaux de sévérité). [20]

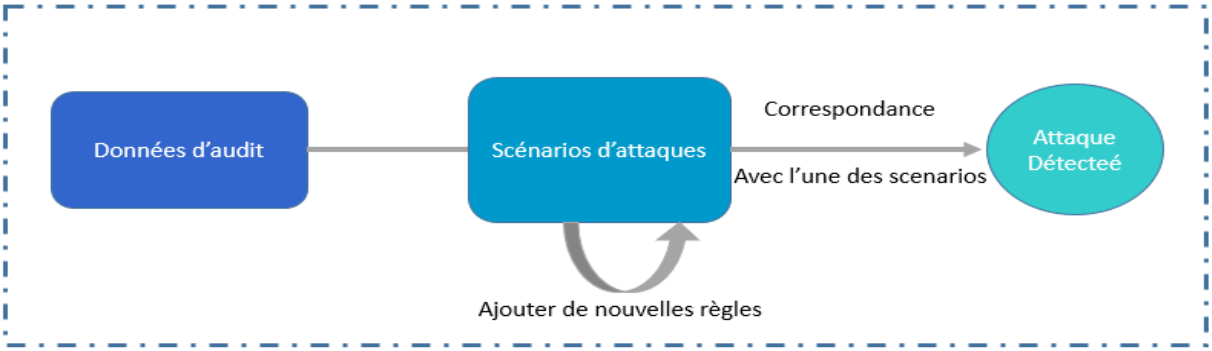

Figure 7: Fonctionnement d'un IDS par l'approche basée connaissance

Cette méthode est très efficace pour détecter des attaques sans produire un grand nombre de fausses alarmes. Elle peut rapidement et sûrement diagnostiquer l'utilisation d'un outil spécifique ou une technique d'attaque. Ceci peut aider les responsables de la sécurité à donner la priorité aux mesures correctives. Cependant, elle peut seulement détecter les attaques connues, dont les signatures sont introduites dans le système, donc le système de détection doit être constamment mis à jour avec les signatures des nouvelles attaques. De plus, beaucoup de systèmes adoptant cette approche sont conçus pour employer un nombre limité de signatures qui peuvent être définies, ce qui les empêchent de détecter les variantes de ces attaques.

b) Détection par comportement

La détection par comportement consiste à considérer comme hostile tout ce qui n'est pas normal, au sens où on cherchera plutôt à bien définir ce qui est un comportement normal sur le système pour pouvoir y opposer toute déviation, que l'on considérera comme étant une attaque : « si ce n'est pas normal, alors c'est dangereux ».

L'idée principale est de modéliser durant une période d'apprentissage le comportement "normal" d'un système/programme en définissant une ligne de conduite (dite profil), et de

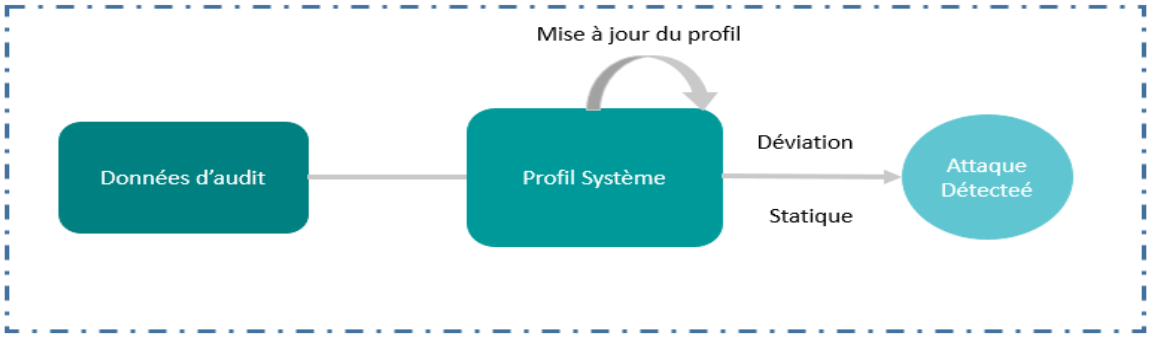

Figure 8 :Fonctionnement d'un IDS par l'approche comportementale.

considérer ensuite (en phase de détection) comme suspect tout comportement inhabituel (les déviations significatives par rapport au modèle de comportement "normal"). Les modèles de détection comportementale incluent fréquemment des modèles statistiques. La détection par comportement revient donc à repérer tout ce qui sort du cadre de la normalité. [21]

La détection comportementale est la capacité à détecter le comportement peu commun. Elle a ainsi la capacité de détecter les symptômes des attaques connues et inconnues sans la connaissance spécifique des détails. De plus, cette approche permet de produire l'information utile pour la définition des signatures pour les systèmes de détection d'intrusions à base de signatures. Cependant, cette approche produit un grand nombre de fausses alarmes dues aux comportements imprévisibles des utilisateurs du réseau. Elle exige souvent l'historique à long terme des événements enregistrés afin de caractériser les modèles normaux de comportement.

Les systèmes basés sur cette approche doivent être dotés d'une certaine intelligence pour raison d'apprentissage automatique de la normalité. Il existe plusieurs méthodes de détection d'intrusion utilisées pour implémenter cette approche. Voici quelques principales méthodes utilisées : la méthode statistique, le système expert, etc… pour plus d'information dirigez-vous dans [22]

2. Comportement après détection :

Nous pouvons également faire une distinction entre les IDS en se basant sur le type de réaction lorsqu'une attaque est détectée :

a) Passive :

Généralement, la plupart des systèmes de détection d'intrusions n'apportent qu'une réponse passive à l'intrusion ; c'est à dire lorsqu'une attaque est détectée, ils génèrent une alarme et notifient l'administrateur système par e-mail, message dans une console, voire même par beeper ou SMS. C'est alors l'opérateur qui devra prendre les mesures qui s'imposent.

b) Active:

Des systèmes de détection d'intrusions peuvent, en plus de la notification à l'opérateur, prendre automatiquement des mesures pour stopper l'attaque en cours. Par exemple, ils peuvent couper les connexions suspectes ou même, pour une attaque externe, reconfigurer le

pare-feu pour qu'il refuse tout ce qui vient du site incriminé. Toutefois, il apparait que ce type de fonctionnalité automatique est potentiellement dangereux car il peut mener à des dénis de service provoqués par l'IDS. Un attaquant déterminé peut, par exemple, tromper l'IDS en usurpant des adresses du réseau local qui seront alors considérées comme la source de l'attaque par l'IDS. Il est préférable de proposer une réaction facultative à un opérateur humain (qui prend la décision finale). [19]

3. Source des données à analyser

Parmi les caractéristiques essentielles des systèmes de détection d'intrusions, les sources de données à analyser constituent la matière première du processus de détection. Ces données proviennent soit de logs (journaux) générés par le système d'exploitation, soit de logs des applications, soit d'informations provenant du réseau, soit encore d'alertes générées par d'autres IDS. [19]

4. Fréquence de l'analyse.

Une autre caractéristique des systèmes de détection d'intrusions est leur fréquence d'utilisation, dans ce cas nous distinguons deux (2) types :

a) IDS online (continue) :

Ce sont des IDS qui font leur analyse des fichiers d'audit ou des paquets réseau de manière continue ou en permanence afin de détecter une attaque au moment de sa production, c'est une détection en temps réel. Ce type d'IDS consomme un taux élevé de ressources systèmes car il faut analyser à la volée tout ce qui se passe sur le système et ce qu'il le rend non préférable en cas de ressources précieuses telle que les serveurs de messagerie par exemple.

b) IDS offline (périodique) :

Ce type d'IDS fait l'analyse dans des durées périodiques afin de détecter des traces d'attaques au but de modéliser des signatures d'attaques pour la base du système, l'avantage de ce type est qu'il ne consomme pas beaucoup de ressources système. Cela peut être suffisant dans des contextes peu sensibles (nous ferons alors une analyse journalière, par exemple). L'inconvénient majeur de ce type est sa détection tardive des attaques ce qui risque de provoquer des dégâts dangereux. [19]

V. Types des IDS

Suivant l'emplacement de l'IDS dans l'architecture du réseau informatique à surveiller, nous distinguons trois types d'IDS :

1. IDS réseau

L'IDS réseau (Network IDS ou NIDS) est situé sur un réseau isolé et ne voit qu'une copie du trafic, c'est-à-dire des paquets qui circulent sur le réseau. En cas de détection d'une menace, le NIDS peut lever des alertes et ordonner les actions pour le blocage d'un flux. En termes d'architecture, le NIDS est situé sur un réseau isolé et analyse une copie du trafic du

réseau à surveiller, entre ses points d'entrées et les terminaux du réseau. A noter qu'il est entièrement passif et il n'est pas capable de dialoguer avec le réseau surveillé.

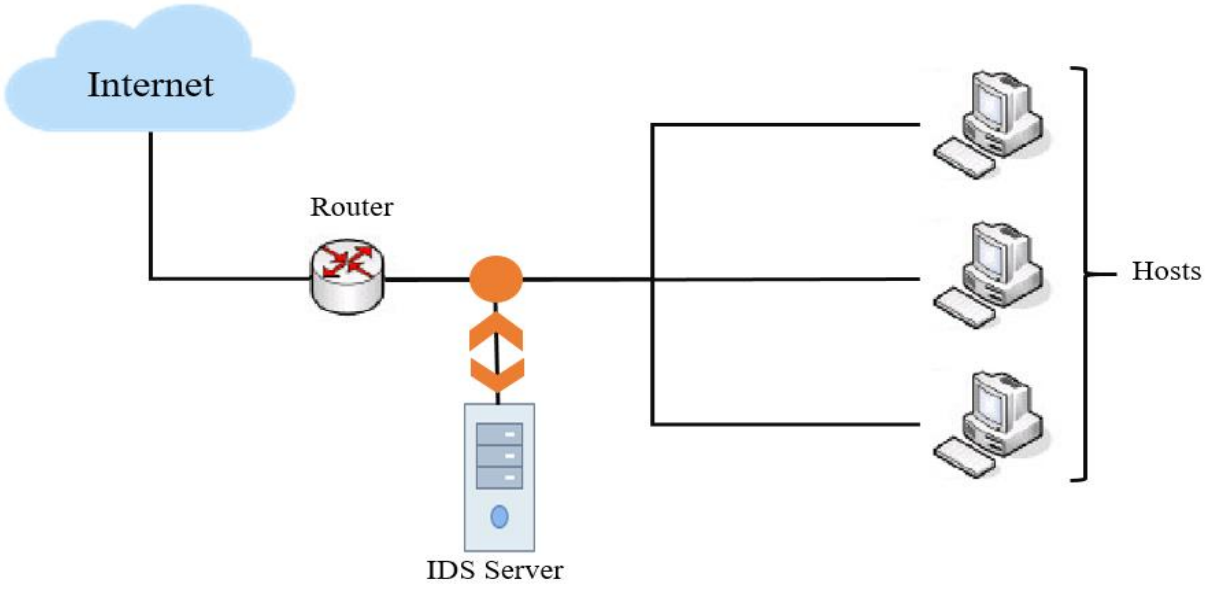

Figure 9 : Exemple d'une architecture d'un NIDS

Il est fréquent de trouver plusieurs IDS sur les différentes parties du réseau. Nous trouvons souvent une architecture composée d'une sonde placée à l'extérieur du réseau afin d'étudier les tentatives d'attaques et d'une sonde en interne pour analyser les requêtes ayant traversé le pare-feu [23]

1.1 avantages de NIDS :

- Les capteurs peuvent être bien
- sécurisés puisqu'ils se contentent d'observer le trafic.
- Détecter plus facilement les scans grâce aux signatures.
- Filtrage de trafic.
- Assurer la sécurité contre les attaques puisqu'il est invisible. [24]

1.2 Inconvénients de NIDS :

- La probabilité de faux négatifs (attaques non détectées) est élevée et il est difficile de contrôler le réseau entier.
- Ils doivent principalement fonctionner de manière cryptée d'où une complication de l'analyse des paquets.
- A l'opposé des IDS basés sur l'hôte, ils ne voient pas les impacts d'une attaque. [24]

2. IDS hôte

Il y a ensuite les IDS hôte (Host IDS ou HIDS) ou IDS système. Les HIDS (Host Intrusion Détection System), surveillent l'état de la sécurité des hôtes selon différents critères :

- Activité de la machine (comme par exemple le nombre et listes de processus, le nombre d'utilisateurs, ressources consommées, etc.).

- Le second critère de surveillance est l'activité de l'utilisateur sur la machine : horaires et durée des connexions, commandes utilisées, programmes activés, etc.

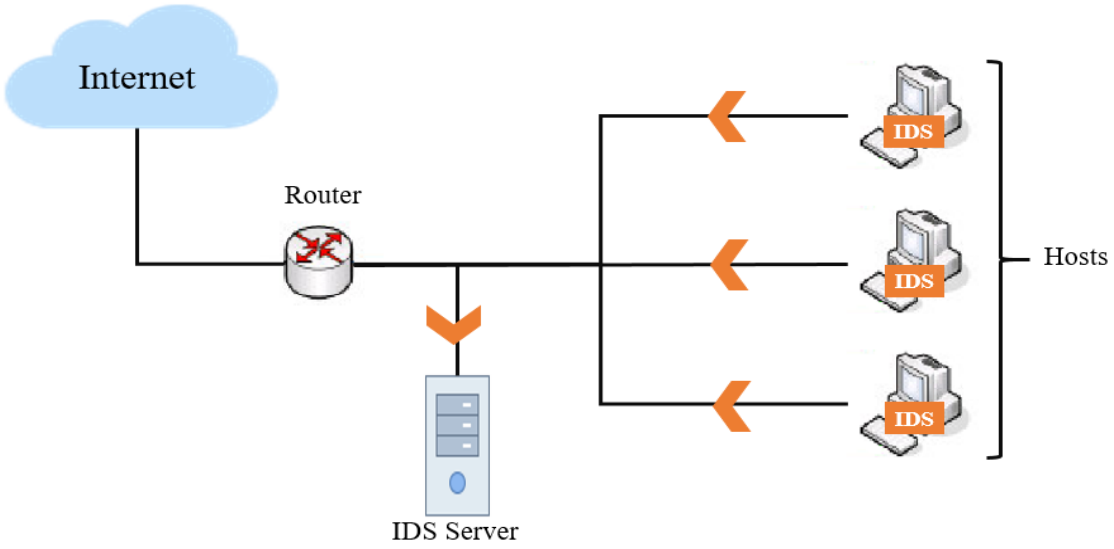

Figure 10 : Exemple d'une architecture d'un HIDS

Évidemment le HIDS analyse toute activité potentielle liée à l'activité d'un ver, d'un virus ou cheval de Troie. En termes d'architecture, les HIDS sont, généralement, placés sur des machines sensibles, susceptibles de subir des attaques et possédantes des données sensibles pour l'entreprise. Les serveurs web et applicatifs, peuvent notamment être protèges par un HIDS. Le HIDS récupère les informations remontées par une machine sur laquelle un client HIDS est installé. Ensuite, le HIDS va analyser ces informations sur le fonctionnement et l'état des machines afin de détecter les menaces. [23]

2.1 avantages de HIDS :

- Découvrir plus facilement un Cheval de Troie puisque les informations et les possibilités sont très étendues.
- Détecter des attaques qui inaccessibles à détecter avec des IDS réseau puisque le trafic est souvent crypté.
- Observer les activités sur l'hôte avec précision. [24].

Inconvénients de HIDS :

- Ils ont moins de facilité à détecter les scans.
- Ils sont plus vulnérables aux attaques de type Dos.
- Ils consomment beaucoup de ressources CPU. [24]

3. IDS hybride

Les IDS hybrides sont, généralement, utilisés dans un environnement décentralisé, ils permettent de réunir les informations de diverses sondes placées sur le réseau, et agissent comme NIDS et/ou HIDS suivant leurs emplacements. Leur appellation « hybride » provient du fait qu'ils sont capables de réunir aussi bien des informations provenant d'un système HIDS qu'un NIDS. Toutes ces sondes HIDS et NIDS remontent alors les alertes à une machine qui va centraliser le tout, et agréger/lier les informations d'origines multiples. Ainsi, nous comprenons que les IDS hybrides sont basés sur une architecture distribuée, où chaque composant unifie son format d'envoi (par exemple IDMEF2). Cela permet de communiquer et d'extraire des alertes plus pertinentes [23]

3.1 Avantages d'IDS hybride :

- moins de faux positifs.
- meilleure corrélation (la corrélation permet de générer de nouvelles alertes à partir de celles existantes).
- possibilité de réaction sur les analyseurs [24]

3.2 Inconvénients d'IDS hybride :

- taux élevé de faux positifs. [24]

VI. Mesures d'évaluations (performances) des systèmes de détection d'intrusions :

La matrice de confusion est utilisée pour visualiser, pour chaque classe de modèle, les vraies classifications et les classifications prédites.

		Classe Prévision	
		Classe négative (normal)	**Classe positive (attaque)**
Classe actuelle	**Classe négative (normal)**	**Vrai négative(VN)**	**Faux positif(FP)**
	Classe positive (attaque)	**Faux négative(FN)**	**Vrai positif(VP)**

Tableau 1: Matrice De Confusion

Vrai positif (VP) : est un résultat où le modèle prédit correctement la classe positive.
Vrai négatif (VN): est un résultat où le modèle prédit correctement la classe négative.
Faux positif (FP) : est un résultat où le modèle prédit incorrectement la classe positive.
Faux négatif (FN): est un résultat où le modèle prédit incorrectement la classe négative. [72]

Par Exemple :

- "Loup" est une **classe positive**.
- "Pas de loup" est une **classe négative**.

Nous pouvons résumer notre modèle de prédiction de loup avec une matrice de confusion 2x2 qui illustre les quatre résultats possibles :

Vrai positif (VP) : • Réalité : attaque de loup. • Le berger dit : "Loup". • Résultat : le berger est un héros.	**Faux positif (FP) :** • Réalité : pas d'attaque de loup. • Le berger dit : "Loup". • Résultat : les villageois en veulent au berger de les avoir réveillés.
Faux négatif (FN) : • Réalité : attaque de loup. • Le berger dit : "Pas de loup". • Résultat : le loup dévore tous les moutons.	**Vrai négatif (VN) :** • Réalité : pas d'attaque de loup. • Le berger dit : "Pas de loup". • Résultat : tout le monde est sain et sauf.

Tableau 2 : Exemple Matrice de Confusion

Les vrais négatifs ainsi que les vrais positifs correspondent à un fonctionnement correct de la technique de machine Learning ce qui signifie que la technique de data machine Learning a prédit avec succès respectivement le comportement normal et les attaques. Les faux négatifs sont des attaques incorrectement prédites comme des comportements normaux.

Les métriques traditionnelles de classification comprennent le taux d'exactitude, le taux de fausse alerte et le taux d'erreur de la classification, elles sont définies comme suit :

1. **Le taux d'exactitude (TE)** : montre à quel point le système est exact, c'est le nombre de type bien classé sur le nombre de type de tout le corpus.

$$\boldsymbol{Exactitude} = \frac{VP + VN}{VP + VN + FP + FN}$$

.

2. **Le taux de fausse alerte(TFA)** : ce critère mesure le taux de fausses alertes générées par un IDS dans un environnement donné et pendant une durée donnée. C'est le nombre des alertes généré comme attaque sur le nombre des types classés comme normal existants dans le corpus.

$$\boldsymbol{FAR} = \frac{FP}{VP + FP}$$

3. **Le taux de détection (DR : Détection Rate)** mesure le taux des attaques détectées par un IDS dans un environnement donné et pendant une durée donnée. C'est le nombre des attaques détecté sur le nombre des attaques existants dans le corpus.

$$\boldsymbol{DR} = \frac{VP}{VP + FN}$$

Dans [25], il est défini trois critères pour évaluer l'efficacité des systèmes de détection d'intrusion :

- **L'exactitude (*accuracy*)** : on parle de l'exactitude quand le système de détection d'intrusion déclare comme malicieux une activité légale. Ce critère correspond au faux positif.
- **La performance *(performance*)** : la performance du système de détection d'intrusion est le taux de traitement des évènements. Si ce taux est faible, la détection en temps réel est donc impossible
- **La complétude (*completeness*)** : on parle de la complétude quand le système de détection d'intrusion ne rate pas la détection d'une attaque. Ce critère est le plus difficile, parce qu'il est impossible d'avoir une connaissance globale sur les attaques. Ce critère correspond au faux négatif.

Debar dans [26] a rajouté également ces critères suivants :

- **La tolérance aux fautes (*Fault tolerance*)** : le système de détection d'intrusions doit lui-même résisté aux attaques, particulièrement au déni de service. Ceci est important, parce que plusieurs systèmes de détection d'intrusion s'exécutent sur des matériels ou logiciels connus comme vulnérables aux attaques.

- **La réaction à temps (*Timeliness*) :** le système de détection d'intrusion doit s'exécuter et propager les résultats de l'analyse le plus tôt possible, pour permettre à l'officier de sécurité de réagir avant que de graves dommages n'aient lieu
- **Rapidité :** Un système de détection d'intrusions doit exécuter et propager son analyse d'une manière prompte pour permettre une réaction rapide dans le cas d'existence d'une attaque pour permettre à l'agent de sécurité de réagir.

VII. Les limites d'un IDS :

Parmi les faiblesses des IDS on trouve : [27]

- Nombreux faux positifs.
- Configuration complexe et longue.
  a) Nombreux faux positifs après configuration.
- Pas de connaissance de la plate-forme.
  a) De ses vulnérabilités.
  b) Du contexte métier.
- Les attaques applicatives sont difficilement détectables.
  a) Injection SQL.

b) Exploitation de CGI mal conçus.

- Des évènements difficilement détectables.
  a) Scans lents / distribués
  b) Canaux cachés / tunnels.
- Pollution des IDS.
  a) Consommation des ressources d'IDS.
  b) Perte de paquets.
  c) Déni de service contre IDS / opérateur.
  d) Une attaque réelle peut passer inaperçue.
- Attaque contre IDS lui-même.
- Ils ne peuvent pas compenser les trous de sécurité dans les protocoles réseaux.
- Ils ne peuvent pas compenser des manques significatifs dans votre stratégie de sécurité, votre politique de sécurité ou votre architecture de sécurité.

VIII. Conclusion :

Ce chapitre nous a permis de constater que les IDS sont de plus en plus fiables, d'où le fait qu'ils soient souvent intégrés dans les solutions modernes de sécurité. Les avantages qu'ils présentent par rapport aux autres outils de sécurité les favorisent. Il nous a également permis de comprendre que ces derniers sont indispensables aux fournisseurs du Cloud afin d'assurer leur sécurité Cloud. Dans le chapitre qui suit nous présenterons les différentes techniques de l'apprentissage automatique.

Chapitre3:

Machine Learning

Chapitre 03: Machine Learning

I. Introduction

L'intelligence artificielle (IA) a reçu une attention croissante ces dernières années. Il contient l'apprentissage automatique qui a conclu de nombreuses discussions contemporaines que l'investissement dans plusieurs secteurs. Le Machine Learning, aussi appelé apprentissage automatique en français, est une sous forme d'intelligence artificielle permettant aux ordinateurs d'apprendre sans avoir été programmés explicitement à cet effet. Dans ce chapitre, nous avons présenté machine Learning et ses types et les grandes familles d'algorithmes pour chaque type.

II. Définition des concepts :

1. Intelligence artificielle

Intelligence artificielle, souvent abrégée en IA, est définie par l'un de ses créateurs, Marvin Lee Minsky, comme : " *la construction de programmes informatiques qui donnent des tâches de façon plus satisfaisante par des êtres humains car elles demandent des processus mentaux de haut niveau*" tels que : l'apprentissage perceptuel, l'organisation de la mémoire et le raisonnement critiquée [28]. Elle est utilisée dans diffèrent domaines d'application et des usages potentiels tels que : la compréhension du langage naturel, la reconnaissance visuelle, robotique, un système autonome. [29]

2. Machine Learning :

Le machine Learning ou « apprentissage automatique » en français est un concept qui fait de plus en plus parler de lui dans le monde de l'informatique, et qui se rapporte au domaine de l'intelligence artificielle. Encore appelé « apprentissage statistique », ce terme renvoie à un processus de développement, d'analyse et d'implémentation conduisant à la mise en place de procédés systématiques. Pour faire simple, il s'agit d'une sorte de programme permettant à un ordinateur ou à une machine d'un apprentissage automatisé, de façon à pouvoir réaliser un certain nombre d'opérations très complexes.

L'objectif visé est de rendre la machine ou l'ordinateur capable d'apporter des solutions à des problèmes compliqués, par le traitement d'une quantité abusif d'informations. Cela offre ainsi une possibilité d'analyser et de mettre en évidence les corrélations qui existent entre deux ou plusieurs situations données, et de prédire leurs différentes implications. [30]

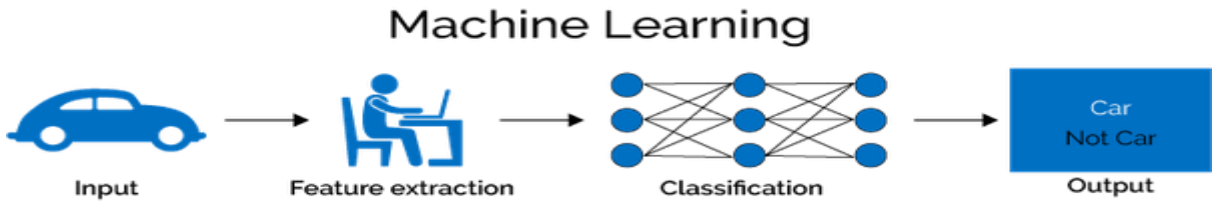

Figure 11 :Machine Learning.

3. Deep Learning :

Le Deep Learning ou apprentissage profond est un type d'intelligence artificielle dérivé du machine Learning (apprentissage automatique) où la machine est capable d'apprendre par elle-même, contrairement à la programmation où elle se contente d'exécuter à la lettre des règles prédéterminées**.**[31]

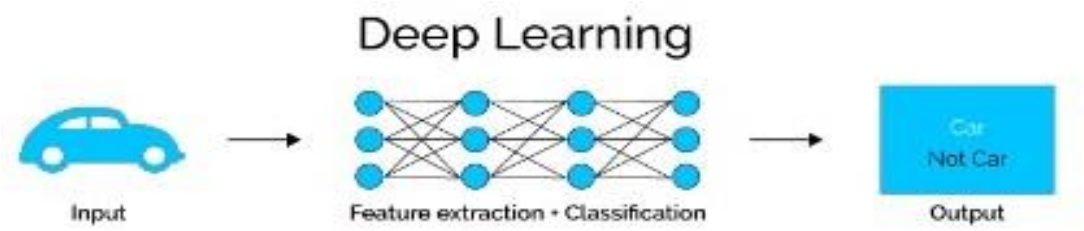

Figure 12: Deep Learning.

Le deep Learning s'appuie sur un réseau de neurones artificiels s'inspirant du cerveau humain. Ce réseau est composé de dizaines voire de centaines de « couches » de neurones, chacune recevant et interprétant les informations de la couche précédente. Le système apprendra par exemple à reconnaître les lettres avant de s'attaquer aux mots dans un texte, ou il détermine s'il y a un visage sur une photo avant de découvrir de quelle personne il s'agit. [31]

Si le Machine Learning (ML) et le Deep Learning (DL) sont des Intelligences Artificielles, l'inverse n'est pas vrai. Par exemple, les graphiques de connaissances ou les moteurs de règles sont des Intelligences Artificielles mais ne relèvent pas du ML ni du DL. Le Deep Learning est, quant à lui, une branche du Machine Learning. [29]

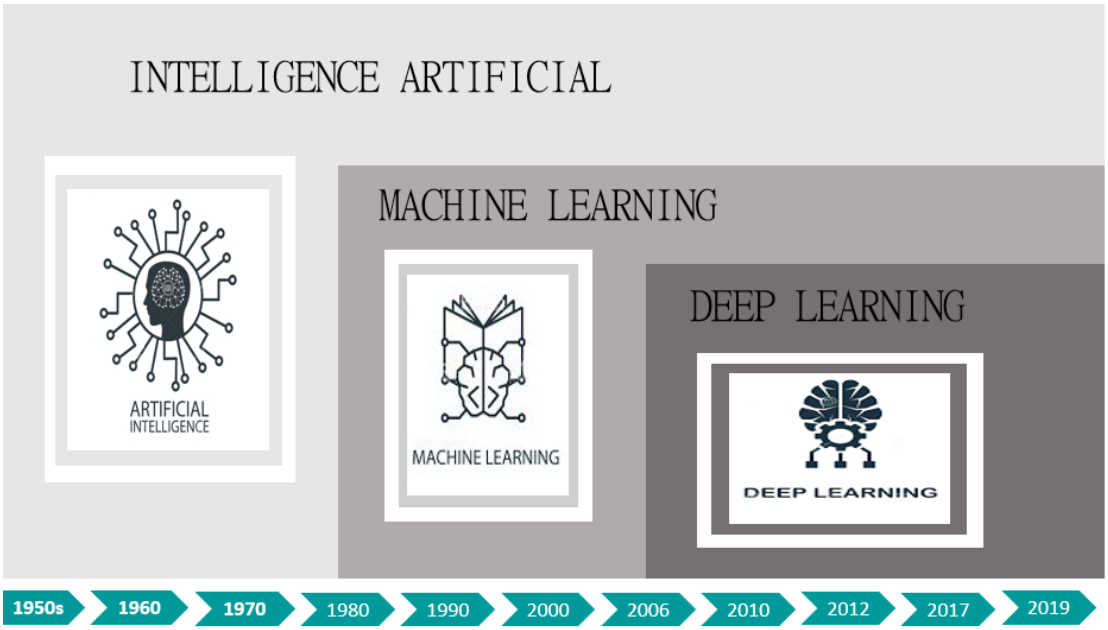

Figure 13 :Intelligence Artificielle

III. Machine Learning

L'apprentissage automatique est un sous-domaine de l'intelligence artificielle (IA). En général, l'objectif de l'apprentissage automatique est de comprendre la structure des données et de les intégrer dans des modèles qui peuvent être compris et utilisés par les tout le monde. [32]

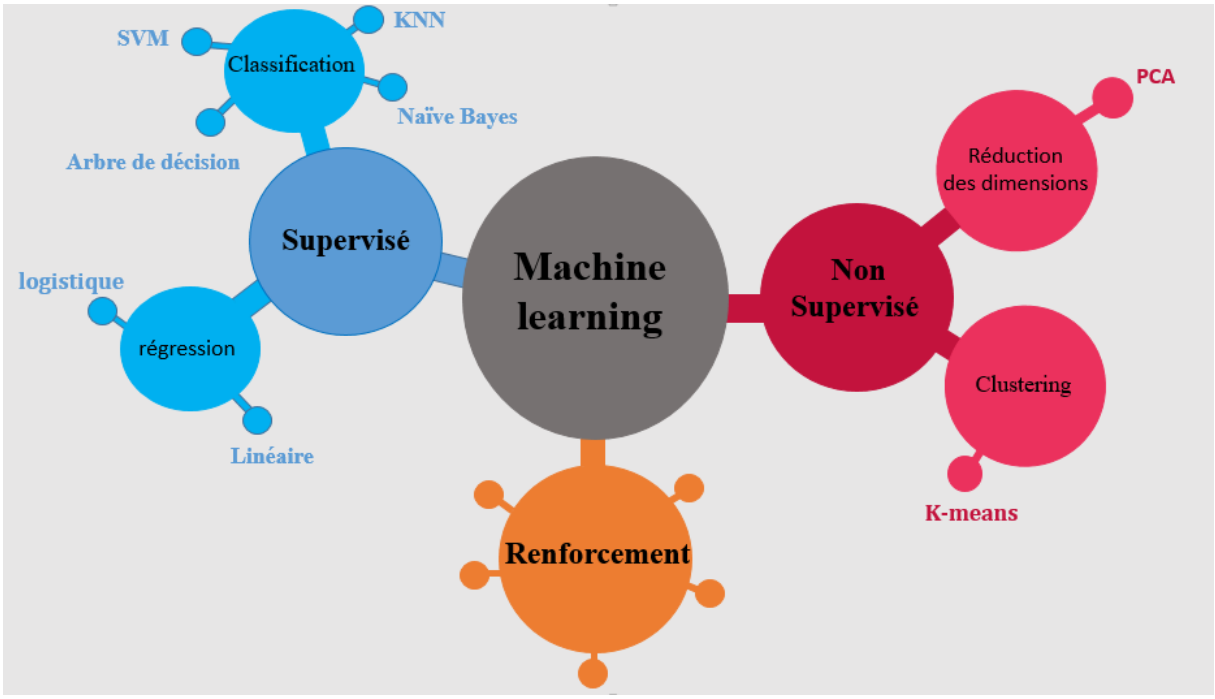

Figure 14 : Les types de Machine Learning.

1. Les types du ML :

L'apprentissage automatique vous permet d'entraîner les ordinateurs à agir de manière indépendante afin que nous n'ayons pas à rédiger des instructions détaillées pour l'exécution de certaines tâches. Pour cette raison, l'apprentissage automatique apporte une grande valeur pour n'importe quel domaine, mais tout d'abord, il fonctionnera bien là où il y a la science des données. L'apprentissage du Machine Learning est basé sur ces trois principaux types :[33]

a) L'apprentissage supervisé :

La majorité des apprentissages automatiques utilisent un apprentissage supervisé (supervised Learning). L'apprentissage supervisé consiste à des variables d'entrée (x) et une variable de sortie (Y). C'est un algorithme qui apprend une fonction de cartographie de l'entrée à la sortie Y = f (X).[34]

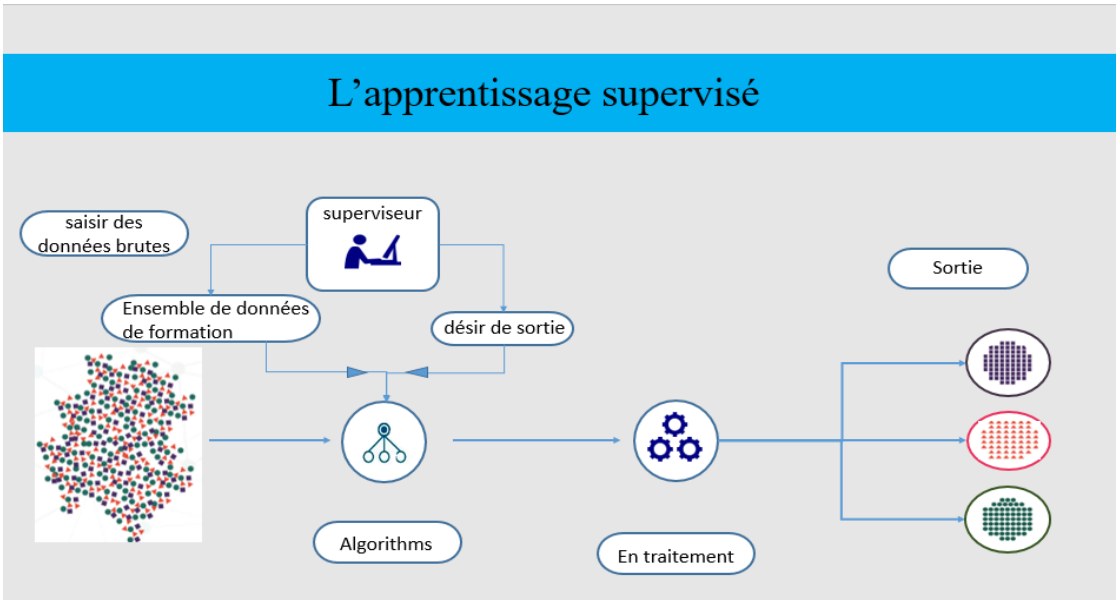

Figure 15: L'apprentissage Supervise

Dans l'apprentissage supervisé, l'ordinateur est fourni avec des exemples d'entrées qui sont étiquetés avec les sorties souhaitées. Le but de cette méthode est que l'algorithme puisse «apprendre» en comparant sa sortie réelle avec les sorties «enseignées» pour trouver des erreurs et modifier le modèle en conséquence. L'apprentissage supervisé utilise donc des modèles pour prédire les valeurs d'étiquettes sur des données non étiquetées supplémentaires.[35]

b) L'apprentissage non supervisé :

L'apprentissage non supervisé consiste à ne disposer que de données d'entrée (X) et pas de variables de sortie correspondantes. L'objectif de l'apprentissage non supervisé est de modéliser la structure ou la distribution sous-jacente des données afin d'en apprendre davantage sur les données.

On appelle apprentissage non supervisé car, contrairement à l'apprentissage supervisé ci-dessus, il n'y a pas de réponse correcte ni d'enseignant. Les algorithmes sont laissés à leurs propres mécanismes pour découvrir et présenter la structure intéressante des données.

L'apprentissage non supervisé comprend deux catégories d'algorithmes : Algorithmes de regroupement et d'association. [34]

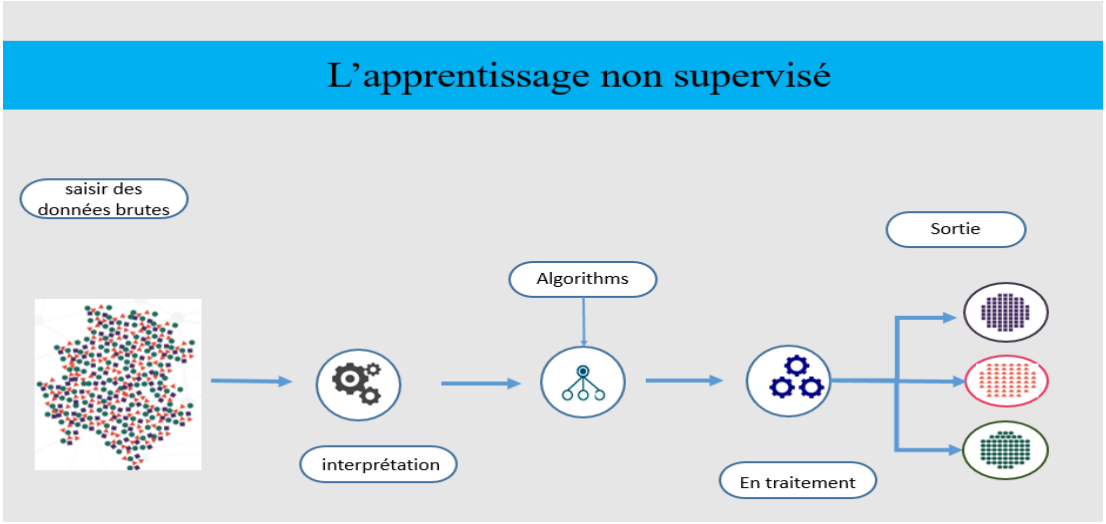

Figure 16 : L'apprentissage Non Supervise

Dans l'apprentissage non supervisé, les données sont non étiquetées, de sorte que l'algorithme d'apprentissage trouve tout seul des points communs parmi ses données d'entrée. Les données non étiquetées étant plus abondantes que les données étiquetées, les méthodes d'apprentissage automatique qui facilitent l'apprentissage non supervisé sont particulièrement utiles.

L'objectif de l'apprentissage non supervisé peut être aussi simple que de découvrir des modèles cachés dans un ensemble de données, mais il peut aussi avoir un objectif d'apprentissage des caractéristiques, qui permet à la machine intelligente de découvrir automatiquement les représentations nécessaires pour classer les données brutes. [35]

c) L'apprentissage de renforcement :

Il se produit lorsque vous présentez l'algorithme avec des exemples qui manquent d'étiquettes, comme dans l'apprentissage non supervisé. Cependant, vous pouvez accompagner un exemple de rétroaction positive ou négative selon la solution proposée par l'algorithme. (figure 17)

Figure 17 : L'apprentissage Renforcement

RL (Renforcement Learning) est connecté aux applications pour lesquelles l'algorithme doit prendre des décisions, et ces décisions ont des conséquences. Un exemple intéressant de RL se produit lorsque les ordinateurs apprennent à jouer à des jeux vidéo par eux-mêmes. [32]

2. Les algorithmes des machines Learning :

a) Algorithme de régression :

C'est un processus de recherche d'un modèle ou d'une fonction permettant de distinguer les données en valeurs réelles continues au lieu d'utiliser des classes ou des valeurs discrètes. Il peut également identifier le mouvement de distribution en fonction des données historiques. Parce qu'un modèle prédictif de régression prédit une quantité, la compétence du modèle doit donc être signalée comme une erreur dans ces prédictions.

Prenons un exemple similaire dans la régression également, où nous trouvons la possibilité de pluie dans certaines régions particulières à l'aide de certains paramètres enregistrés précédemment. Ensuite, il y a une probabilité associée à la pluie. [36], il existe 2 types de cet algorithme :

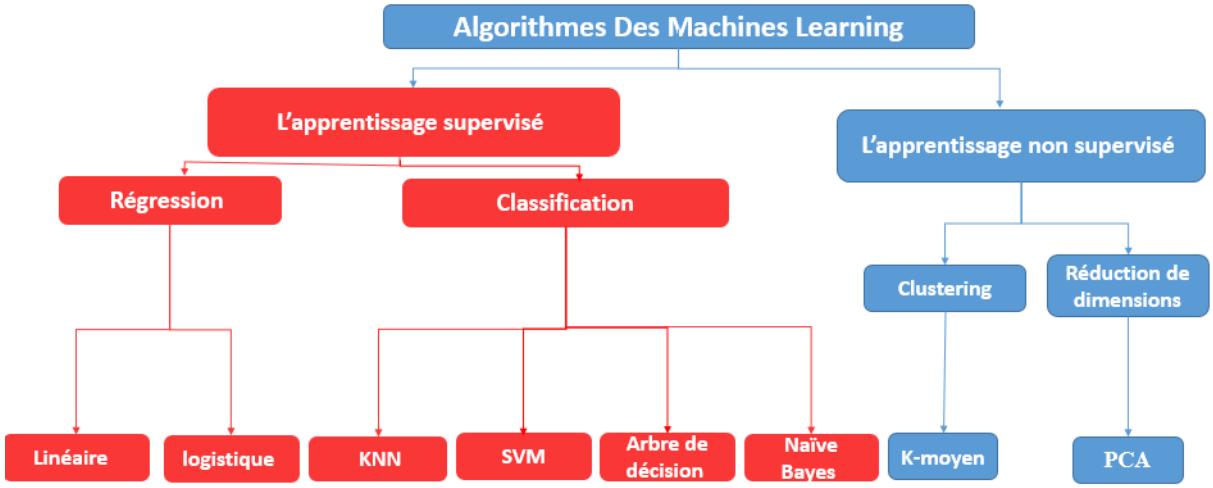

Figure 18: Algorithmes Des Machines Learning.

1. La régression Linéaire :

Les algorithmes de régression linéaire modélisent la relation entre des variables prédictives et une variable cible. La relation est modélisée par une fonction mathématique de prédiction. Le cas le plus simple est la régression linéaire uni variée. Elle va trouver une fonction sous forme de droite pour estimer la relation. La régression linéaire multi

variée intervient quand plusieurs variables explicatives interviennent dans la fonction de prédiction. Finalement, la régression polynomiale permet de modéliser des relations complexes qui ne sont pas forcément linéaires. [37]

2. La régression logistique :

La régression logistique est devenue un outil important dans la discipline de l'apprentissage automatique. Cette approche permet d'utiliser un algorithme dans l'application d'apprentissage automatique pour classer les données entrantes en fonction des données historiques. Plus il y a de données pertinentes en entrée, plus l'algorithme est en mesure de prédire des classifications au sein des jeux de données. [38]

b) Algorithme de Classification :

La classification automatique est le processus qui permet d'analyser et d'organiser un ensemble de données, selon leurs caractéristiques, dans des classes de similarité. Elle se base principalement sur des représentations classiques de données dont les limites de traitement sont connues et, qui dans la plupart du temps, demande un temps de calcul énorme. [39]

1. K plus proches voisins (KNN) :

L'algorithme de k-plus proche voisin est un modèle de reconnaissance de modèle qui peut être utilisé pour la classification et la régression. Souvent abrégé en k-NN, le k est un entier positif, typiquement petit. Dans la classification ou la régression, l'entrée consistera en les k exemples d'entraînement les plus proches dans un espace. [40]

L'algorithme KNN est l'un des plus simples de tous les algorithmes d'apprentissage automatique. Il est un type d'apprentissage basé sur l'apprentissage paresseux (lazy Learning). En d'autres termes, il n'y a pas de phase d'entraînement explicite ou très minime. Cela signifie que la phase d'entraînement est assez rapide.

La méthode KNN suppose que les données se trouvent dans un espace de caractéristiques. Cela signifie que les points de données sont dans un espace métrique. Les données peuvent être des scalaires ou même des vecteurs multidimensionnels. La méthode des k plus proches voisins est utilisée pour la classification et la régression. Dans les deux cas, l'entrée se compose des k données d'entraînement les plus proches dans l'espace de caractéristiques. [41]

L'algorithme kNN :

1. Charger les données
2. Initialiser **k** au nombre de plus proches voisins choisi
3. Pour chaque exemple dans les données :
   3.1 Calculer la distance entre notre requête et l'observation itérative actuelle de la boucle depuis les données.
   3.2 Ajouter la distance et l'indice de l'observation concernée à une collection ordonnée de données
4. Trier cette collection ordonnée contenant distances et indices de la plus petite distance à la plus grande (dans ordre croissant).

5. Sélectionner les **k** premières entrées de la collection de données triées (équivalent aux k plus proches voisins)
6. Obtenir les étiquettes des **k** entrées sélectionnées
7. Si **régression**, retourner la moyenne des **k** étiquettes
8. Si **classification**, retourner le mode (valeur la plus fréquente/commune) des **k** étiquettes. [42]

2. Algorithme les machines à support de vecteurs (SVM) :

Les machines à support de vecteurs sont parmi les techniques les plus connues. Il est développé par Vapnik en 1995, elles sont considérées comme une alternative récente pour la classification. Elles se basent principalement sur l'utilisation des fonctions appelées kernel, qui facilite la séparation des données.

En général, les SVM peuvent être utilisées pour résoudre plusieurs problèmes réels, tels que, la classification textuelle et la régression. Pour cela, on doit construire une fonction *f* qui accepte un vecteur d'entrée *x* et qui retourne un vecteur de sortie *y*, avec : y= *f(x)* [43]

L'algorithme des machines à vecteurs de support a été développé dans les années 90 par le russe Vladimir Vapnik. Initialement, les SVM ont été développé comme un algorithme de classification binaire supervisée. Il s'avère particulièrement efficace de par le fait qu'il peut traiter des problèmes mettant en jeu de grands nombres de descripteurs, qu'il assure une solution unique (pas de problèmes de minimum local comme pour les réseaux de neurones) et il a fourni de bons résultats sur des problèmes réels [73]. L'algorithme sous sa forme initiale revient à chercher une frontière de décision linéaire entre deux classes, mais ce modèle peut considérablement être enrichi en se projetant dans un autre espace permettant d'augmenter la séparabilité des données. On peut alors appliquer le même algorithme dans ce nouvel espace, ce qui se traduit par une frontière de décision non linéaire dans l'espace initial [73].

a. Les domaines d'applications SVM :

Est une méthode de classification qui montre de bonnes performances dans la résolution de problèmes variés. Cette méthode a montré son efficacité dans de nombreux domaines d'applications tels que le traitement d'image, la catégorisation de textes ou le diagnostiques médicales et ce même sur des ensembles de données de très grandes dimensions.

La réalisation d'un programme d'apprentissage par SVM se ramène à résoudre un problème d'optimisation impliquant un système de résolution dans un espace de dimension conséquente. L'utilisation de ces programmes revient surtout à sélectionner une bonne famille de fonctions noyau et à régler les paramètres de ces fonctions. Ces choix sont le plus souvent faits par une technique de validation croisée, dans laquelle on estime la performance du système en la mesurant sur des exemples n'ayant pas été utilisés en cours d'apprentissage.

L'idée est de chercher les paramètres permettant d'obtenir la performance maximale. Si la mise en oeuvre d'un algorithme de SVM est en général peu coûteuse en temps, il faut cependant compter que la recherche des meilleurs paramètres peut requérir des phases de test assez longues [74].

b. SVM principe de fonctionnement général :

b.1. Notions de base: Hyperplan, marge et support vecteur

Pour deux classes d'exemples donnés, le but de SVM est de trouver un classificateur qui va séparer les données et maximiser la distance entre ces deux classes. Avec SVM, ce classificateur est un classificateur linéaire appelé hyperplan. Dans le schéma qui suit, on détermine un hyperplan qui sépare les deux ensembles de points [74]

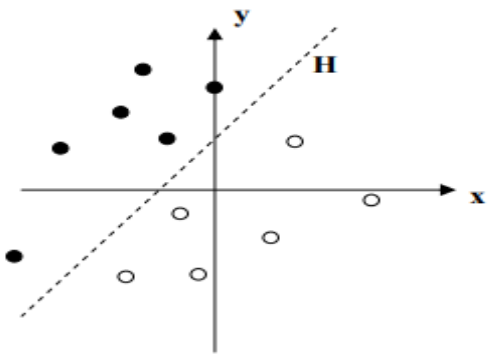

Figure 19 : Exemple d'un hyperplan séparateur .

Les points les plus proches, qui seuls sont utilisés pour la détermination de l'hyperplan, sont appelés vecteurs de support.

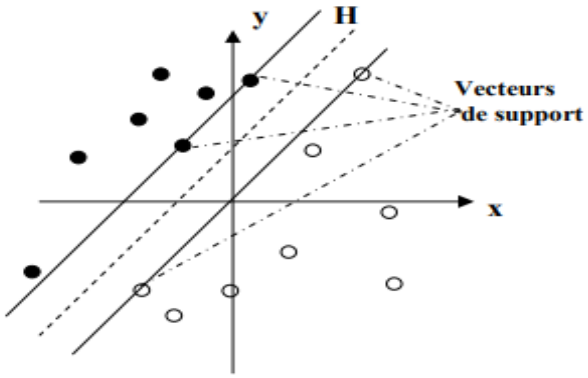

Figure 20 : Exemple de vecteurs de support

Il est évident qu'il existe une multitude d'hyperplan valide mais la propriété remarquable des SVM est que cet hyperplan doit être optimal. Nous allons donc en plus chercher parmi les hyperplans valides, celui qui passe « au milieu » des points des deux

classes d'exemples. Intuitivement, cela revient à chercher l'hyperplan le « plus sûr » [75]. En effet, supposons qu'un exemple n'ait pas été décrit parfaitement, une petite variation ne modifiera pas sa classification si sa distance à l'hyperplan est grande. Formellement, cela revient à chercher un hyperplan dont la distance minimale aux exemples d'apprentissage est maximale [75]. On appelle cette distance « marge » entre l'hyperplan et les exemples. L'hyperplan séparateur optimal est celui qui maximise la marge. Comme on cherche à maximiser cette marge, on parlera de séparateurs à vaste marge [75].

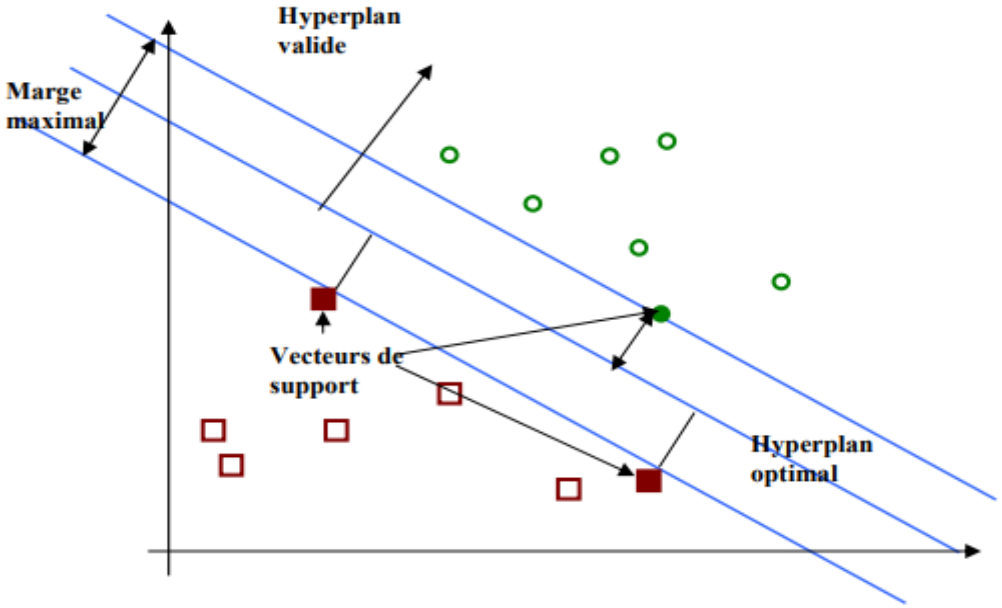

Figure 21 : Exemple de marge maximal (hyperplan valide).

b.2. Pourquoi maximiser la marge ?

Intuitivement, le fait d'avoir une marge plus large procure plus de sécurité lorsque l'on classe un nouvel exemple. De plus, si l'on trouve le classificateur qui se comporte le mieux vis-à-vis des données d'apprentissage, il est clair qu'il sera aussi celui qui permettra au mieux de classer les nouveaux exemples. Dans le schéma qui suit, la partie droite nous montre qu'avec un hyperplan optimal, un nouvel exemple reste bien classé alors qu'il tombe dans la marge. On constate sur la partie gauche qu'avec une plus petite marge, l'exemple se voit mal classé [74].

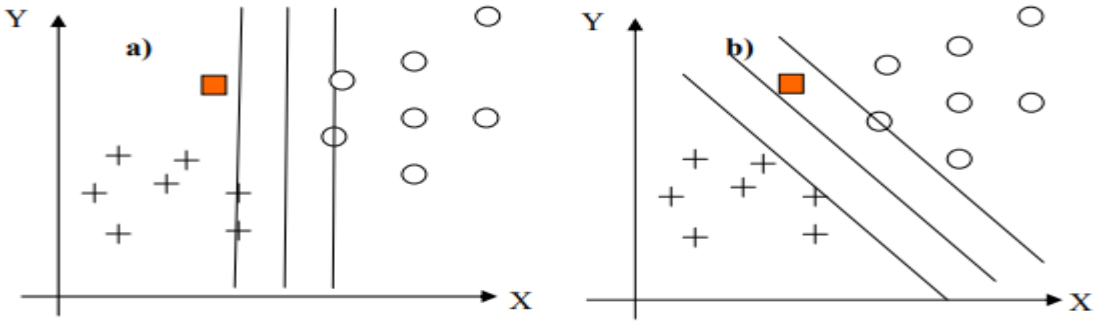

Figure 22 : a) Hyperplan avec faible marge, b) Meilleur hyperplan séparateur

En général, la classification d'un nouvel exemple inconnu est donnée par sa position par rapport à l'hyperplan optimal. Dans le schéma suivant, le nouvel élément sera classé dans la catégorie des « + ».

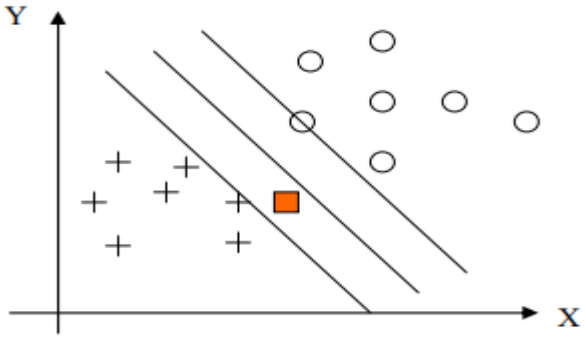

Figure 23 :Exemple de classification d'un nouvel élément.

b.3. Linéarité et non-linéarité :

Parmi les modèles des SVM, on constate les cas linéairement séparable et les cas non linéairement séparable. Les premiers sont les plus simples de SVM car ils permettent de trouver facilement le classificateur linéaire. Dans la plupart des problèmes réels il n'y a pas de séparation linéaire possible entre les données, le classificateur de marge maximale ne peut pas être utilisé car il fonctionne seulement si les classes de données d'apprentissage sont linéairement séparables [74].

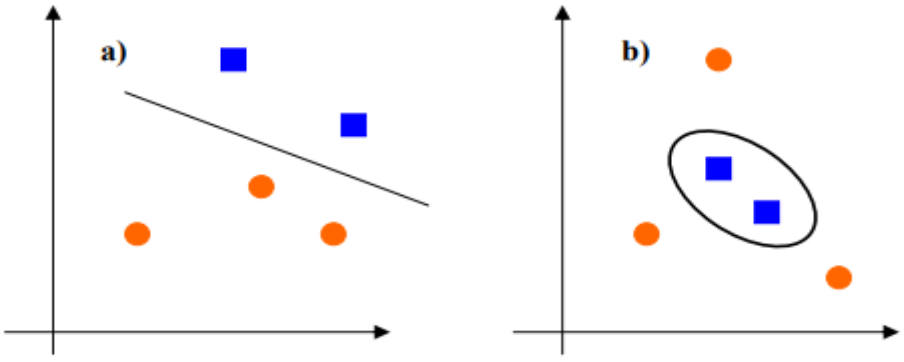

Figure 24 : a) Cas linéairement séparable, b) Cas non linéairement séparable .

b.4. Cas non linéaire :

Pour surmonter les inconvénients des cas non linéairement séparable, l'idée des SVM est de changer l'espace des données. La transformation non linéaire des données peut permettre une séparation linéaire des exemples dans un nouvel espace. On va donc avoir un changement de dimension. Cette nouvelle dimension est appelé « espace de re-description ».En effet, intuitivement, plus la dimension de l'espace de redescription est grande, plus la probabilité de pouvoir trouver un hyperplan séparateur entre les exemples est élevée. Ceci est illustré par le schéma suivant [74]:

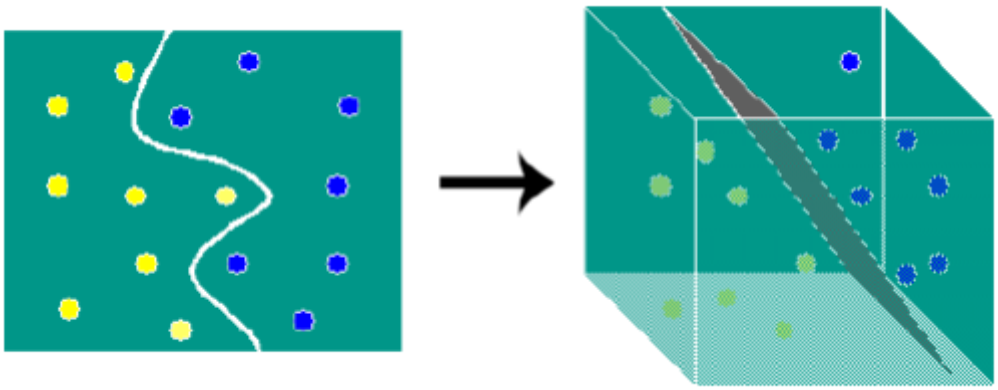

Figure 25 : Exemple de changement de l'espace de données.

On a donc une transformation d'un problème de séparation non linéaire dans l'espace de représentation en un problème de séparation linéaire dans un espace de redescription de plus grande dimension. Cette transformation non linéaire est réalisée via une fonction noyau [74].

En pratique, quelques familles de fonctions noyau paramétrables sont connues et il revient à l'utilisateur de SVM d'effectuer des tests pour déterminer celle qui convient le mieux pour son application. On peut citer les exemples de noyaux suivants : polynomiale, gaussien, sigmoïde et laplacien [74].

b.5. Illustration de transformation de cas non linéaire :

Le cas XOR Le cas de XOR n'est pas linéairement séparable, si on place les points dans un plan à deux dimensions, on obtient la figure suivante : Coordonnées des points : (0,0) ; (0,1) ; (1,0) ; (1,1)

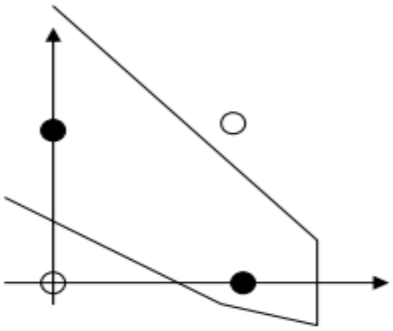

Figure 26 : Illustration de cas non linéairement séparable (le cas XOR)

Si on prend une fonction polynomiale (x,y) → (x,y,x.y) qui fait passer d'un espace de dimension 2 à un espace de dimension 3, on obtient un problème en trois dimensions linéairement séparable :

(0,0) → (0,0,0)
(0,1) → (0,1,0)
(1,0) → (1,0,0)
(1,1) → (1,1,1)

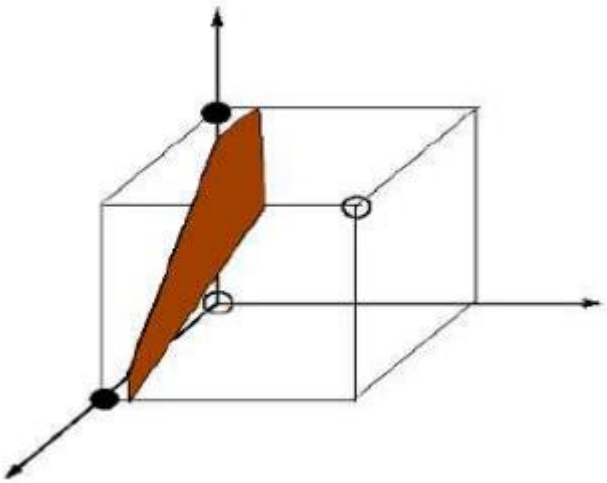

Figure 27 : Illustration de passage d'un espace 2D à un espace 3D

3. Algorithme arbre de décision :

Un arbre de décision est un schéma représentant les résultats possibles d'une série de choix interconnectés. Il permet à une personne ou une organisation d'évaluer différentes actions possibles en fonction de leur coût, leur probabilité et leurs bénéfices. Il peut être

utilisé pour alimenter une discussion informelle ou pour générer un algorithme qui détermine le meilleur choix de façon mathématique.[44]

Un arbre de décision commence généralement par un nœud d'où découlent plusieurs résultats possibles. Chacun de ces résultats mène à d'autres nœuds, d'où émanent d'autres possibilités. Le schéma ainsi obtenu rappelle la forme d'un arbre. [44]

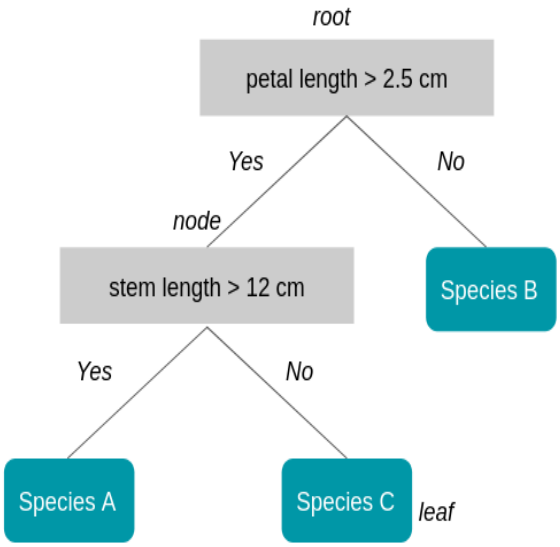

Figure 28 : Exemple D'Arbre de décision.

4. L'algorithme de Naïve Bayes :

Les méthodes Naïve Bayes sont considérées parmi les modèles probabilistes les plus connus. Elles se basent principalement sur le théorème de Bayes (Bayes, 1963).

Les algorithmes Naïves Bayes sont souvent utilisés dans la catégorisation et la classification de documents. Ils permettent d'estimer la probabilité de chaque classe parmi les exemples, étant donné un document, et affectent à ce dernier la classe la plus probable. On appelle ce procédé «Prior probabilités». [45]

Un exemple d'utilisation du naive bayes est celui du filtre anti-spam.

c) Clustering:

C'est un algorithme de Machine Learning très pratique pour identifier des groupes de comportements similaires. Cet algorithme permet de travailler sur les données et de les classifier. Avec un algorithme de clustering, il est possible de découvrir automatiquement des groupes (ou clusters) de clients qui ont des comportements similaires. Cela permet donc de grouper les audiences selon des comportements similaires, de repérer facilement les « électrons libres » qui n'appartiennent pas à un groupe et même de découvrir les comportements inconnus a priori. Il devient ainsi plus facile de personnaliser son service et d'offrir donc une expérience de qualité à ses clients. [46]

1. K-means (K-moyen) :

K-means est un algorithme non supervisé de clustering non hiérarchique. Il permet de regrouper en clusters distincts les observations du data set. Ainsi les données similaires se retrouveront dans un même cluster. Par ailleurs, une observation ne peut se retrouver que dans un cluster à la fois (exclusivité d'appartenance). Une même observation, ne pourra donc, appartenir à deux clusters différents. [47]

L'algorithme de K-moyen :

Le fonctionnement de K-moyen se résume dans les étapes suivantes :

1. On choisit k objets au hasard qu'on considère comme des centres pour les classes initiales.

2. On affecte chaque objet au centre le plus proche pour obtenir une partition de k classes.

3. On recalcule les centres de chaque classe.

4. La répétition des étapes 2 et 3 jusqu'à la stabilité des centres.

La complexité de l'algorithme du K-moyen est de (lkn).

Où : l : est le nombre d'itérations.

k : le nombre des classes telles que $k<n$. [48]

d) Réduction de dimensions :

1. PCA :

Est un algorithme d'apprentissage automatique non supervision qui tente de réduire la dimensionnalité (nombre de fonctions) au sein d'un ensemble de données tout en conservant autant d'informations que possible. Cette action s'effectue en recherchant un nouvel ensemble de variables appelées *composantes*, qui constituent les composés des caractéristiques originales décarrelées les unes les autres. Les composants sont également contraints de telle sorte que le premier composant représente la plus grande variabilité possible dans les données, le deuxième composant la deuxième variabilité la plus importante, et ainsi de suite.

Dans Amazon Sage Maker, l'algorithme PCA opère selon deux modes, en fonction du scénario :

a. **rgular :** pour les ensembles de données avec données fragmentées et un nombre modéré d'observations et de caractéristiques.

b. **randomized** : pour les ensembles de données avec un grand nombre d'observations et de caractéristiques. Ce mode utilise un algorithme d'approximation [49]

Mode 1 : Rgular

Les agents de travail calculent $\sum x_i^T x_i$ et $\sum x_i$.

Note :

Comme x_i sont 1 * d des vecteurs de ligne, $x_i^T x_i$ est une matrice (non un scalaire). L'utilisation des vecteurs de ligne au sein du code nous permet d'obtenir une mise en cache efficace.

La matrice de covariance est calculée comme $\sum x_i^T x_i - (1/n)(\sum x_i)^T \sum x_i$ et ses principaux vecteurs num_components forment le modèle.

Note :

Si subtract_mean a la valeur False, nous évitons de calculer et de soustraire $\sum x_i$.

Utilisez cet algorithme lorsque la dimension d des vecteurs est suffisamment petite pour que d^2 puisse contenir en mémoire. [49]

Mode 2 : Randomized

Lorsque le nombre de fonctions de l'ensemble de données en entrée est de grande taille, nous utilisons une méthode pour estimer approximativement la métrique de covariance. Pour chaque mini-lot X_t de dimension b * d, nous initialisons de façon aléatoire une matrice (num_components + extra_components) * b que nous multiplions par chaque mini-lot, afin de créer une matrice (num_components + extra_components) * d. La somme de ces matrices est calculée par les workers et les serveurs exécutent SVD sur la matrice (num_components + extra_components) * d finale. Les vecteurs num_components singuliers en haut à droite représentent l'approximation des vecteurs singuliers supérieurs de la matrice d'entrée.

ℓ = num_components + extra_components. Soit un mini-lot X_t de dimension b * d, le travail trace une matrice aléatoire H_t de dim ension $\ell * b$. Selon que l'environnement utilise un GPU ou une UC et la taille de la dimension, la matrice est une matrice de signe aléatoire où chaque entrée est +-1 ou une transformation *FJLT* (Fast Johnson Lindenstrauss Transform ; pour plus d'informations, consultez FJLT Transforms et les articles afférents). Le travail calcule ensuite $H_t X_t$ et maintient $B = \sum H_t X_t$. Le travail maintient aussi h^T , la somme des colonnes de $H_1, \ldots, H_T$ (T étant le nombre total de mini-lots), et s, la somme de toutes les lignes en entrée. Après le traitement de la totalité de la partition de données, le worker envoie au serveur B, h, s et n (nombre de lignes en entrée).

Indiquez les différentes entrées au serveur comme B^1, h^1, s^1, n^1 Le serveur calcule B, h, s, n les sommes des entrées respectives. Puis, il calcule $C = B - (1/n)h^T s$ et recherche sa décomposition en valeurs singulières. Les vecteurs singuliers en haut à droite et les valeurs singulières de C sont utilisés comme solution approximative au problème.[49]

APPRENTISSAGE	ALGORITHMES	AVANTAGES		INCONVENIENTS
Supervisé	**Classification**	KNN	1. facile à implémente. 2. efficace. [50] 3. L'algorithme est polyvalent [28]	1. Calculer chaque fois la similarité entre les k. [51] 2. grande capacité de stockage. 3. utilise de nombreuses données de références pour classifier les nouvelles entrées [50]
		SVM	1. Leur capacité à manipuler de grandes quantités de données 2. Le faible nombre d'hyper paramètres. 3. Elles sont bien fondées théoriquement. [52]	1. complexes pour la classification des corpus. 2. demande un temps énorme pendant les phases de test. [43]
		Arbre de décision	1. faciles à comprendre. 2. Ils permettent de sélectionner l'option la plus appropriée parmi plusieurs. 3. Il est facile de les associer à d'autres outils de prise de décision. [53]	1. instables. [53] 2. Certains concepts sont difficiles à exprimer à l'aide d'arbres de décision (comme XOR). [29]
		Naïve Bayes	1. La facilité et la simplicité de leur implémentation. 2. Leur rapidité. 3. Les méthodes Naïve Bayes donnent de bons résultats. [54]	1. faire le même travail de classification. [55] [56]
	REGRESSION	Linéaire	1. Simplicité d'interprétation. 2. facilité de calcul [30]	Elle ne traite pas les valeurs manquantes de variables continues sensible aux valeurs hors norme de variables continues [57]
Non Supervisé	**REDUCTION DES DI MENSIONS**	PCA	1. Simplicité mathématique 2. Simplicité des résultats 3. Puissance 4. Flexibilité [58]	1. l'ACP n'a pas réellement 2.s'applique simplement sur des cas précis 3.Perte d'information par l'emploi fréquent de la 1ère composante principale uniquement. [58]
	CLUSTERING	K-means	1. Simple 2. Flexible 3. Efficace 4. Complexité temporelle. [48]	1. Ensemble non optimal de clusters 2. Manque de cohérence 3. Limitation des calculs 4. Spécifiez les valeurs k [48]

Tableau 3: Avantages ET Inconvénients Les Algorithmes De ML

Conclusion :

Ce chapitre présente les concepts de base de l'apprentissage automatique. Premièrement, sa définition et ses types ont été abordés pour donner une image claire, et deuxièmement, les algorithmes ont été expliqués en détail, en particulier les algorithmes svm, et pca et donnez des exemples. Après cela, nous discuterons d'une application utilisant la technologie ML.

PARTIE II

Contribution

Chapitre 4 :

L'approche proposée

Chapitre 04 : l'approche proposé

I. Introduction :

Nous présentons dans ce dernier chapitre, notre approche pour le développement de notre problématique, nous décrivons et exposons toutes les étapes de réalisation de notre approche. On explique le principe du fonctionnement détaillé et tous les difficultés rencontrées de notre application.

La méthode proposée évalue par un ensemble de données CICIDS2017, les ensembles de données sont prétraités pour convenir à l'application des techniques d'apprentissage automatique. En général, IDS traite une énorme quantité de données, même pour un petit réseau, qui contient non pertinent et redondant fonctionnalités. Les fonctionnalités étrangères peuvent rendre plus difficile la détection de comportements suspects, entraînant un processus de formation et de test lent, une consommation de ressources plus élevée et un faible taux de détection. [73]

La sélection des fonctionnalités est l'un des sujets clés d'IDS, elle améliore les performances de classification en recherchant le sous-ensemble de fonctionnalités, qui classe le mieux les données d'entraînement [74]. En cas de problème d'espace de dimension élevée, certaines des fonctionnalités peuvent être redondantes ou non pertinentes. La suppression de ces fonctionnalités redondantes ou non pertinentes est très importante ; ils peuvent donc détériorer les performances des classificateurs.

L'analyse des composants (PCA) est une méthode efficace pour réduire la dimensionnalité en fournissant une carte linéaire de l'espace des caractéristiques à n dimensions à un espace des caractéristiques à m dimensions. PCA est appliqué pour la réduction de dimension de caractéristique. Pour CICIDS2017 ensemble de données, nous sélectionnons 35 fonctionnalités pour réduisent les dimensions des données d'entrée

Nous évaluons l'ensemble de données CICIDS2017 et nous proposons un système de détection d'intrusions d'anomalies basé sur SVM, voir la figure suivante qui représente l'enchaînement de la combinaison des PCA et SVM :

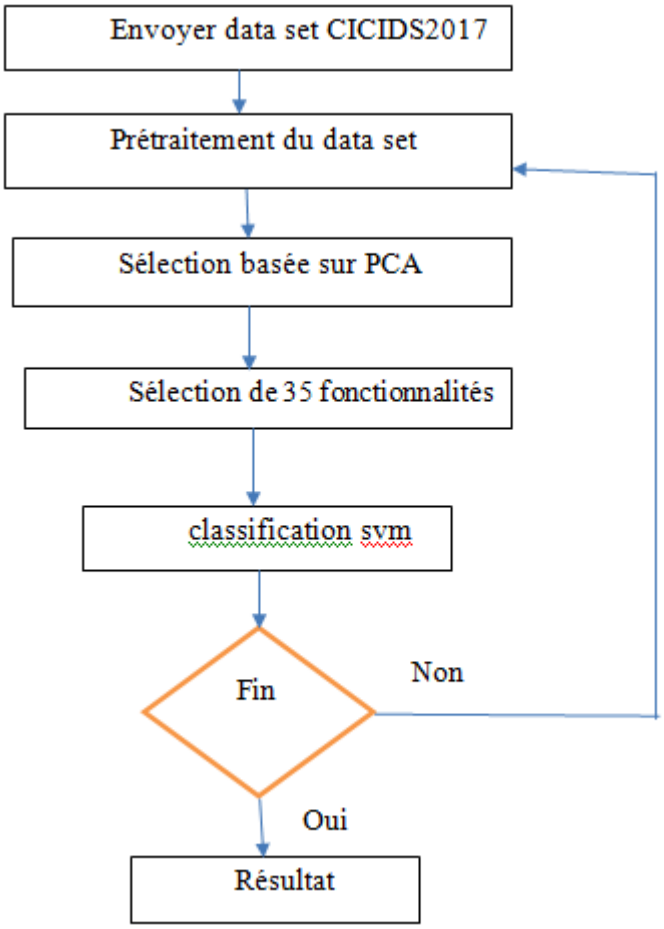

Figure 29:Diagramme de Combinaison PCA et SVM.

II. Les outils de développement :

1. Définition du langage Python en informatique :

Python est le langage de programmation open source le plus employé par les informaticiens. Ce langage s'est propulsé en tête de la gestion d'infrastructure, d'analyse de données ou dans le domaine du développement de logiciels. En effet, parmi ses qualités, Python permet notamment aux développeurs de se concentrer sur ce qu'ils font plutôt que sur la manière dont ils le font. Il a libéré les développeurs des contraintes de formes qui occupaient leur temps avec les langages plus anciens. Ainsi, développer du code avec Python est plus rapide qu'avec d'autres langages. [59]

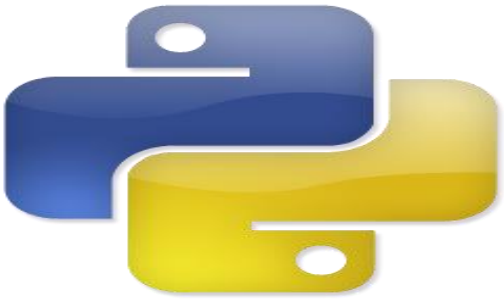

Figure 30 : Logo Python.

2. Définition de l'anaconda :

Anaconda est une plate-forme informatique scientifique et de traitement de donnée basée sur Python. Il a intégré de nombreuses bibliothèques tierces très utiles. [60]

Figure 31 : logo Anaconda.

Anaconda est donc un utilitaire dont on ne peut quasiment pas se passer lorsque l'on a un projet incluant du Python. [61]

3. Définition jupyter :

Jupyter se présente comme un outil extrêmement simple à mettre en œuvre qui vous permettra de transformer vos Jupyter Notebooks en applications web ou en Dashboard quasiment automatiquement.[62]

Figure 32 : Logo Jupytre.

Bibliothèques Supplémentaires :

Afin d'atteindre les objectifs de ce projet, nous avons utilisé d'autres bibliothèques externes pour effectuer certaines tâches spécifiques. En plus de celles fournies par la bibliothèque standard de Python.

- **Matplotlib :**

Est probablement l'un des packages Python les plus utilisés pour la représentation de graphiques en 2D. Il fournit aussi bien un moyen rapide de visualiser des données grâce au langage Python, que des illustrations de grande qualité dans divers formats [63]

- **Seaborn :**

Est une librairie qui vient s'ajouter à Matplotlib, remplace certains réglages par défaut et fonctions, et lui ajoute de nouvelles fonctionnalités. Seaborn vient corriger trois défauts de Matplotlib :

- Matplotlib, surtout dans les versions avant la 2.0, ne génère pas des graphiques d'une grande qualité esthétique.
- Matplotlib ne possède pas de fonctions permettant de créer facilement des analyses statistiques sophistiquées.
- Les fonctions de Matplotlib ne sont pas faites pour interagir avec les Data frames de Panda (que nous verrons au chapitre suivant).

Seaborn fournit une interface qui permet de pallier ces problèmes. Il utilise toujours Matplotlib "sous le capot", mais le fait en exposant des fonctions plus intuitives. Pour commencer à l'utiliser, rien de plus simple. [64]

- **Scikit-learn :**

Est une bibliothèque développée en Python, un langage de programmation de haut niveau. Elle est dédiée à l'apprentissage statistique (machine Learning) et peut être utilisée comme middleware, notamment pour des tâches de prédiction. [65]

- **NumPy :**

Est le package fondamental pour le calcul scientifique avec Python. Il contient entre autres :

- un puissant objet tableau N-dimensionnel
- fonctions sophistiquées (diffusion)
- outils d'intégration de code C / C ++ et Fortran
- algèbre linéaire utile, transformée de Fourier et capacités de nombres aléatoires

Outre ses utilisations scientifiques évidentes, NumPy peut également être utilisé comme un conteneur multidimensionnel efficace de données génériques. Des types de données arbitraires peuvent être définis. Cela permet à NumPy de s'intégrer de manière transparente et rapide à une grande variété de bases de données.

NumPy est sous licence BSD, permettant une réutilisation avec peu de restrictions. [66]

- **Pandas :**

Est une librairie Python qui a pour objectif de vous faciliter la vie en matière de manipulation de données. Les structures de données gérées par Pandas peuvent contenir tout type d'éléments à savoir (dans le jargon Pandas) des Séries et Data Frame et des Panel. Dans le cadre de nos expérimentations on utilisera plutôt les Data frame car ils offrent une vue bidimentionnelle des données (comme un tableau Excel), et c'est exactement ce que l'on va chercher à utiliser pour nos modèles. [67]

- **SciPy:**

SciPy est un logiciel open source pour les mathématiques, les sciences et l'ingénierie. La bibliothèque SciPy dépend de NumPy, qui fournit une manipulation de tableau N dimensionnelle pratique et rapide.

La bibliothèque SciPy est conçue pour fonctionner avec les baies NumPy et fournit de nombreuses routines numériques conviviales et efficaces telles que des routines pour l'intégration et l'optimisation numériques. [68]

- **glob :**

En Python, le module glob est utilisé pour récupérer des fichiers / noms de chemin correspondant à un modèle spécifié. Les règles de modèle de glob suivent la règle standard d'extension de chemin Unix. Il est également prédit que selon les repères, il est plus rapide que les autres méthodes de faire correspondre les noms de chemin dans les répertoires. [70]

III. Ensemble de données d'évaluation de détection d'intrusion (CICIDS2017) :

L'ensemble de données CICIDS2017 contient les attaques communes bénignes, qui ressemblent aux vraies données réelles .Il inclut également les résultats de l'analyse du trafic réseau à l'aide de CICFlowMeter avec des flux étiquetés basés sur l'horodatage, les IP source et de destination, les ports source et de destination, les protocoles et les attaques.

Le jeu de données CIDSID2017 contient l'attaque la plus courante basée sur le rapport McAfee 2016 (Dos, DDos, Web based, Brute force, Infiltration, Heart-bleed, Bot et Scan) avec plus de 80 fonctionnalités extraites du trafic réseau généré. [71]

CICFlowMeter génère des flux bidirectionnels (Biflow), où le premier paquet détermine les directions avant (source vers destination) et arrière (destination vers source), d'où les 83 caractéristiques statistiques telles que la durée, le nombre de paquets, le nombre d'octets, la longueur des paquets, etc. sont également calculés séparément dans le sens avant et arrière. La sortie de l'application est le format de fichier CSV avec six colonnes étiquetées pour chaque flux, à savoir Flow ID, Source IP, Destination IP, Source Port, Destination Port et Protocol avec plus de 80 fonctionnalités de trafic réseau, dans ce tableau nous présentons chaque fonction avec sa description.[72]

Nom de la fonction	Description
Feduration	Durée du flux en microseconde
Flux Feduration	Durée du flux en microsecondes
Total FWwd Packet	Total des paquets dans le sens aller
Total de paquets Bwd	Total de paquets dans le sens inverse
Longueur totale du paquet avant	Taille totale du paquet vers l'avant
Longueur totale du paquet Bwd	Taille totale du paquet vers l'arrière
Fwd Packet Length Min	Taille minimale du paquet vers l'avant
Fwd Packet Length Max	Taille maximale du paquet dans le sens direct
Fwd Packet Length Moyenne	Taille moyenne du paquet vers l'avant
Fwd Packet Length Std	Taille de l'écart type du paquet vers l'avant
Bwd Packet Length Min	Taille minimale du paquet vers l'arrière
Bwd Packet Length Max	Taille maximale du paquet vers l'arrière
Bwd Packet Length Moyenne	Taille moyenne du paquet vers l'arrière
Bwd Packet Length Std	Taille de l'écart type du paquet vers l'arrière
Octet de flux / s	Nombre d'octets de flux par seconde
Paquets / s de flux	Nombre de paquets de flux par seconde
Flow IAT Mean	Temps moyen entre deux paquets envoyés dans le flux
Flow IAT Std	Temps d'écart type entre deux paquets envoyés dans le flux
Flow IAT Max	Temps maximum entre deux paquets envoyés dans le flux
Flow IAT Min	Temps minimum entre deux paquets envoyés dans le flux
Fwd IAT Min	Temps minimum entre deux paquets envoyés dans le sens direct
Fwd IAT Max	Temps maximum entre deux paquets envoyés dans le sens direct
Fwd IAT Mean	Temps moyen entre deux paquets envoyés dans le sens direct
Fwd IAT Std	Temps d'écart type entre deux paquets envoyés dans le sens direct

Fwd IAT Total	Temps total entre deux paquets envoyés dans le sens direct
Bwd IAT Min	Temps minimum entre deux paquets envoyés vers l'arrière
Bwd IAT Max	Temps maximum entre deux paquets envoyés vers l'arrière
Bwd IAT Mean	Temps moyen entre deux paquets envoyés vers l'arrière
Bwd IAT Std	Temps d'écart type entre deux paquets envoyés vers l'arrière
Bwd IAT Total	Temps total entre deux paquets envoyés vers l'arrière
Fwd PSH flag	Nombre de fois que le drapeau PSH a été mis en paquets dans le sens aller (0 pour UDP)
Bwd PSH Flag	Nombre de fois que le drapeau PSH a été défini dans des paquets se déplaçant vers l'arrière (0 pour UDP)
Fwd URG Flag	Nombre de fois que le drapeau URG a été défini dans des paquets se déplaçant vers l'avant (0 pour UDP)
Bwd URG Flag	Nombre de fois où le drapeau URG a été défini dans des paquets se déplaçant vers l'arrière (0 pour UDP)
Fwd Header Length	Nombre total d'octets utilisés pour les en-têtes dans le sens direct
Longueur d'en-tête Bwd	Nombre total d'octets utilisés pour les en-têtes vers l'arrière
Paquets FWD / s	Nombre de paquets de transfert par seconde
Bwd Packets / s	Nombre de paquets en arrière par seconde
Min Packet Length	Longueur minimale d'un paquet
Max Packet Length	Longueur maximale d'un paquet
Packet Length Mean	Longueur moyenne d'un paquet
Packet Length Std	Longueur d'écart type d'un paquet
Packet Length Variance	Longueur de variance d'un paquet
FIN Flag Count	Nombre de paquets avec FIN
SYN Flag Count	Nombre de paquets avec SYN
RST Flag Count	Nombre de paquets avec RST
PSH Flag Count	Nombre de paquets avec PUSH

ACK Flag Count	Nombre de paquets avec ACK
URG Flag Count	Nombre de paquets avec URG
CWR Flag Count	Nombre de paquets avec CWE
ECE Flag Count	Nombre de paquets avec ECE
down/Up Ratio	Ratio de téléchargement et de téléchargement
Average Packet Size	Taille moyenne des paquets
Avg Fwd Segment Size	Taille moyenne observée vers l'avant
AVG Bwd Segment Size	Nombre moyen d'octets débit en masse dans le sens direct
Fwd Header Length	Longueur de l'en-tête pour le paquet en avant
Fwd Avg Bytes / Bulk	Nombre moyen d'octets en vrac dans le sens direct
Fwd AVG Packet / Bulk	Nombre moyen de paquets en vrac dans le sens aller
Fwd AVG Bulk Rate	Nombre moyen de taux en vrac vers l'avant
Bwd Avg Bytes / Bulk	Nombre moyen d'octets en vrac dans le sens retour
Bwd AVG Packet / Bulk	Nombre moyen de paquets en vrac dans le sens arrière
Bwd AVG Bulk Rate	Nombre moyen de taux en vrac vers l'arrière
Subflow Fwd Packets	Le nombre moyen de paquets dans un sous-flux dans le sens direct
Subflow Fwd Bytes	Le nombre moyen d'octets dans un sous-flux dans le sens direct
Subflow Bwd Packets	Le nombre moyen de paquets dans un sous-flux vers l'arrière
Subflow Bwd Bytes	Le nombre moyen d'octets dans un sous-flux vers l'arrière
Init_Win_bytes_forward	Le nombre total d'octets envoyés dans la fenêtre initiale dans le sens direct
Init_Win_bytes_backward	Le nombre total d'octets envoyés dans la fenêtre initiale vers l'arrière
Act_data_pkt_forward	Nombre de paquets avec au moins 1 octet de charge utile de données TCP dans le sens aller
min_seg_size_forward	Taille minimale de segment observée vers l'avant
Active Min	Durée minimale pendant laquelle un flux était actif

	avant de devenir inactif
Active Mean	Durée moyenne pendant laquelle un flux était actif avant de devenir inactif
Active Max	Durée maximale pendant laquelle un flux était actif avant de devenir inactif
Actif Std	Écart type de temps un flux a été
min_idle	Temps minimum pendant lequel un flux était inactif avant de devenir actif
mean_idle	Temps moyen pendant lequel un flux était inactif avant de devenir actif
max_idle	Durée maximale pendant laquelle un flux était inactif avant de devenir actif
std_idle	Temps d'écart type pendant lequel un flux était inactif avant de devenir actif
Init_Win_bytes_backward	Le nombre total d'octets envoyés dans la fenêtre initiale vers l'arrière
Init_Win_bytes_forward	Le nombre total d'octets envoyés dans la fenêtre initiale dans le sens direct
Act_data_pkt_forward	Nombre de paquets avec au moins 1 octet de charge utile de données TCP dans le sens aller

Tableau 4: fonctionnalités de trafic réseau avec la description

Ainsi, notre data set Contient 6 types d'attaques voici le tableau suivant :

	Flow Type
0	BENIGN
1	DoS Hulk
2	DoS slowloris
3	Dos slowhttptest
4	DoS GoldenEye
5	Heartbleed

Tableau 5 : Type d'attaque.

IV. Entraînement et Paramétrage des Modèles :

• Prétraitement des données (Preprocessing Data sets) :

Pour fournir des données plus appropriées pour le classificateur, l'ensemble de données est passé par un groupe d'opérations de prétraitement. Ces opérations sont résumées ci-dessous :

- La première étape consiste à diviser l'ensemble de données en un ensemble d'entités (voir Tableau 4) et des étiquettes (voir Tableau 5) correspondantes, stocke les ensembles d'entités dans la variable X et la série d'étiquettes correspondantes dans la variable y.
- Supprimez les espaces blancs, certaines des étiquettes multi-classes du jeu de données incluent des espaces blancs.
- Encodage des étiquettes multi-classes de l'ensemble de données sont fournies avec les noms de l'attaque, qui sont des valeurs de chaînes. Ainsi, il est important de coder ces valeurs en valeurs numériques, afin que le classificateur puisse apprendre le numéro de classe auquel appartient chaque tuple.

- La conversation numérique est fait avec pandas.to_numeric () est l'une des fonctions générales de Pandas qui est utilisée pour convertir l'argument en type numérique.
- Normalisation des données, les données numériques de l'ensemble de données sont de différentes plages, ce qui pose certains défis au classificateur pendant la formation pour compenser ces différences. Ainsi, il est important de normaliser les valeurs de chaque attribut, de sorte que la valeur minimale de chaque attribut soit nulle, tandis que le maximum est un. Cela fournit des valeurs plus homogènes au classificateur tout en maintenant la relativité entre les valeurs de chaque attribut. avec la technique StandardScaler () on a normalisé les entités ,l'idée derrière StandardScaler est qu'il transformera les données de telle sorte que leur distribution aura une valeur moyenne de 0 et un écart-type de 1.

• Application de PCA :

Pour exécuter PCA on a utilisé la bibliothèque Scikit-Learn de Python. La classe PCA est utilisée à cet effet. PCA dépend uniquement de l'ensemble de fonctionnalités et non des données d'étiquette.

La réalisation de PCA à l'aide de Scikit-Learn est un processus en deux étapes :

- Initialisez la classe PCA en passant le nombre de composants au constructeur.

- Appelez l'ajustement, puis transformez les méthodes en transmettant l'ensemble de fonctionnalités à ces méthodes. La méthode de transformation renvoie le nombre spécifié de composants principaux, voici le code suivant :

```
In [13]: model = PCA().fit(x)
         variances = model.explained_variance_ratio_
```

Figure 33 : Entrainement du model(PCA).

Dans le code ci-dessus, nous créons un objet PCA nommé model. Nous n'avons pas spécifié le nombre de composants dans le constructeur. Par conséquent, les 76 fonctionnalités de l'ensemble de fonctionnalités seront renvoyées pour les ensembles de formation et de test.

La classe PCA contient **explained_variance_ratio_** qui renvoie la variance causée par chacun des composants principaux, La variable **explained_variance_ratio_** est maintenant un tableau de type flottant qui contient des ratios de variance pour chaque composant principal, voici la représentation du tableau sur un graph :

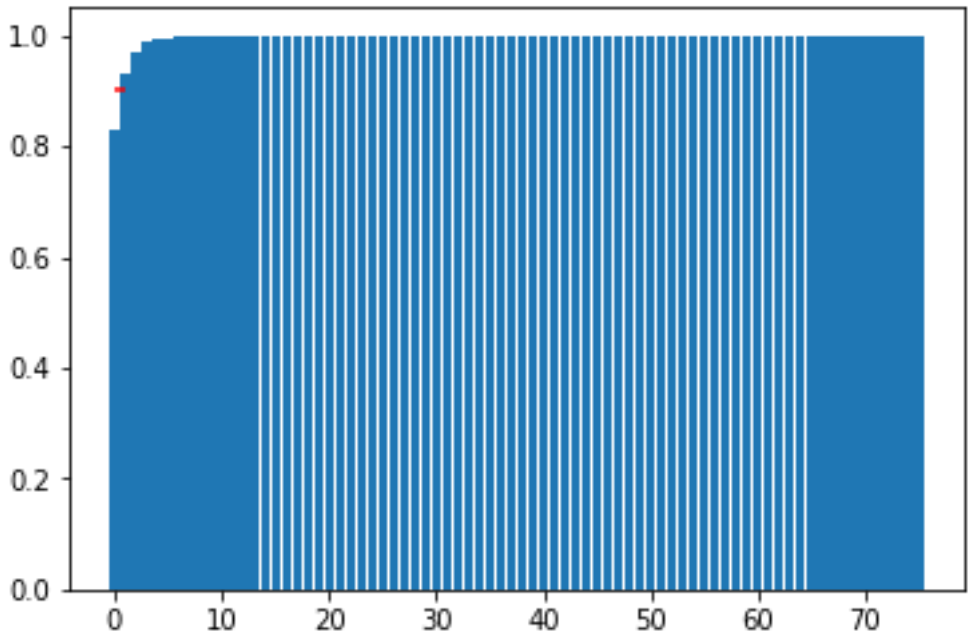

Figure 34 : graph repésente explained_variance_ratio_.

La ligne bleu montre le somme cumulée (cela vient de **model.explaines-variance-ratio-cumsum()).**, a partir du graphe, nous pouvons lire le pourcentage de la variance dans les données expliquées lorsque nous ajoutons les principales composantes. La première

composante principale explique donc 82% de la variance de l'ensemble de données. Les 2 premières composantes principales expliquent 90%, etc.

Et de la on peux décider que on souhaite conserver 99% de la variance dans nos modélisation. On conserverais donc les 35 premières composantes principales, expliquant 99.51% de la variance, et éliminerais les 41 autres composantes principales, voici le code suivant :

```
In [18]: modell = PCA(n_components=35)
         modell.fit(x)
         X_compress = modell.transform(x)
```

Figure 35: Application d'algorithme PCA.

La prochaine étape consiste à diviser les données ensembles d'apprentissage et de test, afin d'atteindre notre objectif, 80% des données de chacun des mélanges sont utilisés pour entrainer le classificateur sur la relation contrainte-déformation réelle, puis 20% des données ont été utilisés pour valider le model.

```
X_train, X_test, y_train, y_test =\
    train_test_split(X_compress, y, test_size=0.2, random_state=0)
```

Figure 36 : Séparation du data set à des les données d'apprentissage et de test.

On a utilisé la technique SVM pour faire les classifications. Dans cette tâche, on a suggéré d'utiliser l'approche One-vs-Rest, qui est implémentée dans la classe OneVsRestClassifier. En tant que classificateur de base, on a utilisé LinearSVC voici le code suivant :

```
# classifier
clf = OneVsRestClassifier(LinearSVC(C=1,random_state=0,dual=False))
y_score = clf.fit(X_train, y_train).decision_function(X_test)
```

Figure 37 : Application du techinque PCA.

```
In [30]: y_pred = clf.predict(X_test)
```

Figure 38 : Classification du données de test.

Après la prédiction on calculer le taux de précision avec la 'métrics accuracy' ,notre modèle a atteindre vers 98.60 % de taux de précision dans la classification de l'ensemble de données.

```
In [27]: y_pred = clfrr.predict(X_test)
         val_acc = metrics.accuracy_score(y_test,y_pred)*100
         print('accuracy %s' % val_acc)

         accuracy 98.60402335770638
```

Figure 39 : calcul de la précision.

- **La courbe ROC :**

Les courbe ROC sont créé, En traçant le vrai taux positif par rapport au taux de faux positifs à différents paramètres de seuil pour chaque classe, pour montre la capacité de diagnostic du notre classifieur par le calcule (TPR) et (FPR).

Et on a le résultat suivant qui montre que pour chaque class la courbe est au-dessus de la diagonale, et pour chaque classe on a :

- Seuil: En bas à gauche, point (0,0)

Taux de faux positifs (FPR): 0. Le classificateur n'a identifié aucun échantillon négatif réel comme positif
True Positive Rate (TPR): 0 . Le classificateur n'a pu attraper aucun des échantillons True Positive

- Seuil2: En haut à droite, point (1.0, 1.0) (la barre de maintien est en bas)

FPR: 1,0 .Le classificateur a identifié tous les échantillons négatifs réels comme positifs
TPR: 1.0 Classifier a montré une bonne performance sur la capture de tous les positifs réels
Ainsi, le point idéal est donc le coin supérieur gauche du graphique: les faux positifs sont proches de 0 et les vrais positifs sont proches de 1 .

Cependant, pour qu'il y ait discrimination, il est nécessaire que la courbe monte très vite. Il faut avoir simultanément forte sensibilité et forte spécificité. Une mesure du pouvoir discriminante est obtenue à l'aide de l'aire sous la courbe ROC [76]

Si ROC = 0.5	Pas de discrimination
Si 0.5 < ROC <0.7	Discrimination insuffisante
Si 0.7 = ROC <0.8	Discriminante acceptable
Si 0.8 = ROC <0.9	Discriminante excellente
Si 0.9 = ROC < 1	Discriminante exceptionnelle

Tableau 6 : Mesures de discrimination.

Pour notre modèle, sa valeur est éloignée vers 1. Elle traduit là une discrimination exceptionnelle.

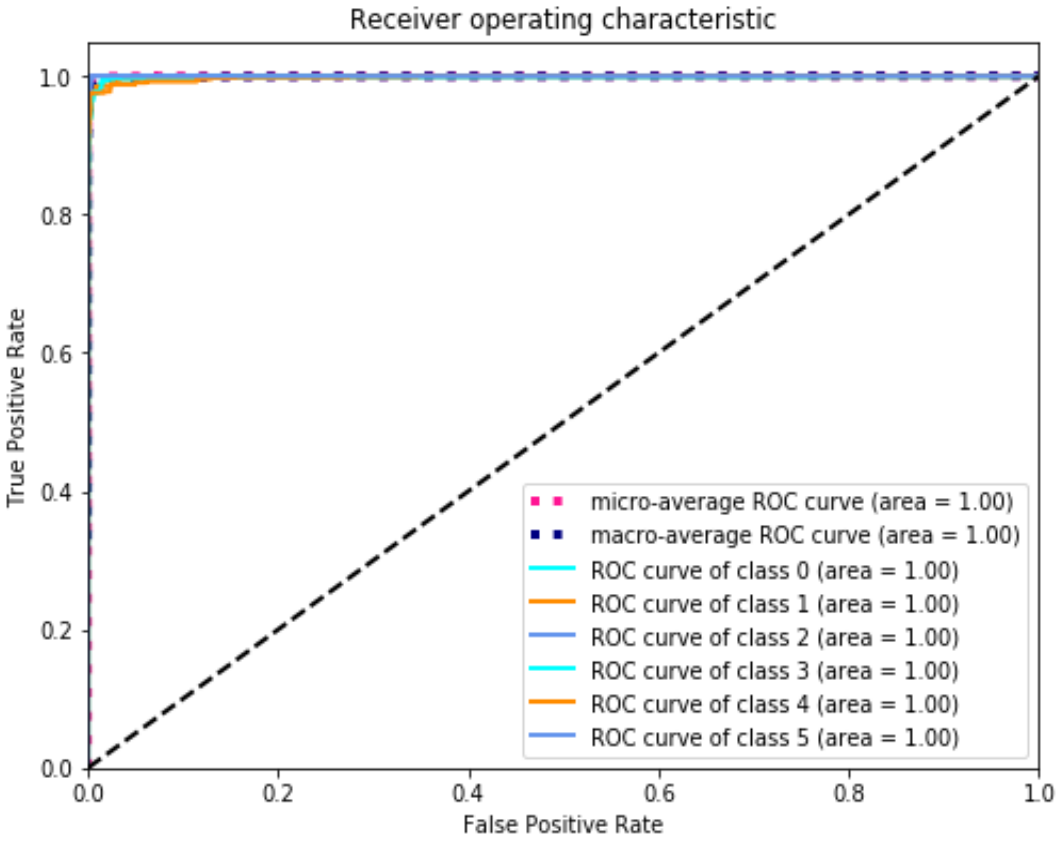

Figure 40 : Courbe ROC

Ce travail montre à quel point le jeu de données CICIDS2017 est très utile pour tester différents classificateurs, les travaux se concentrent sur la phase de prétraitement de CICIDS2017 afin de préparer des expériences fiables et des données de test indépendantes randomisées. Parmi les techniques de classification, le classifier SVM a atteint le taux de précision le plus élevé pour la détection et la classification de tous les types d'attaques de jeux de données CICIDS2017.

Chaque table de confusion va analyser une unique classe. Ainsi, dans notre cas, nous aurons besoins de 6 classes.comme on peut l'observer, une matrice de confusion a en colonne et en ligne les mêmes intitulés.

En ligne, on lit les labels des individus et en colonne les labels prédits par le modèle. Ainsi, on peut conclure par exemple que :

Dans 87710 situation où il appartient à la class 0, le modèle a bien prédit les classes, le modèle a prédit16 fois qu'il appartient à la classe 1 alors qu'en réalité il appartient à la classe 0,la classe 2 déduite par le modèle contient 45565 observations bien classées et 533 mauvaises

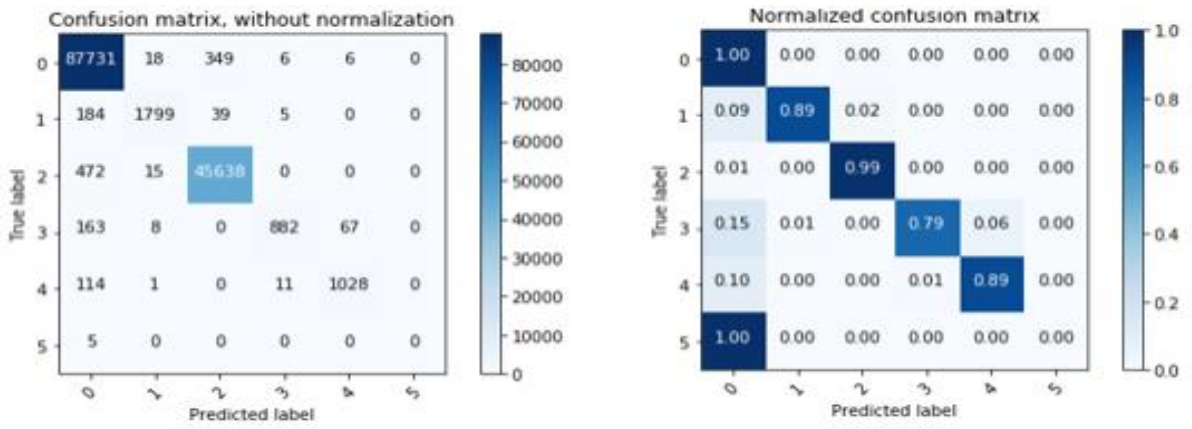

Figure 41: Matrices Confusion.

- ## Rapport de classification (Classification report) :

Le rapport de classification est utilisé pour mesurer la qualité des prévisions de notre classifieur. Combien de prédictions sont vraies et combien sont fausses. Plus précisément, les vrais positifs, les faux positifs, les vrais négatifs et les faux négatifs sont utilisés pour prédire les mesures d'un rapport de classification, comme indiqué ci-dessous :

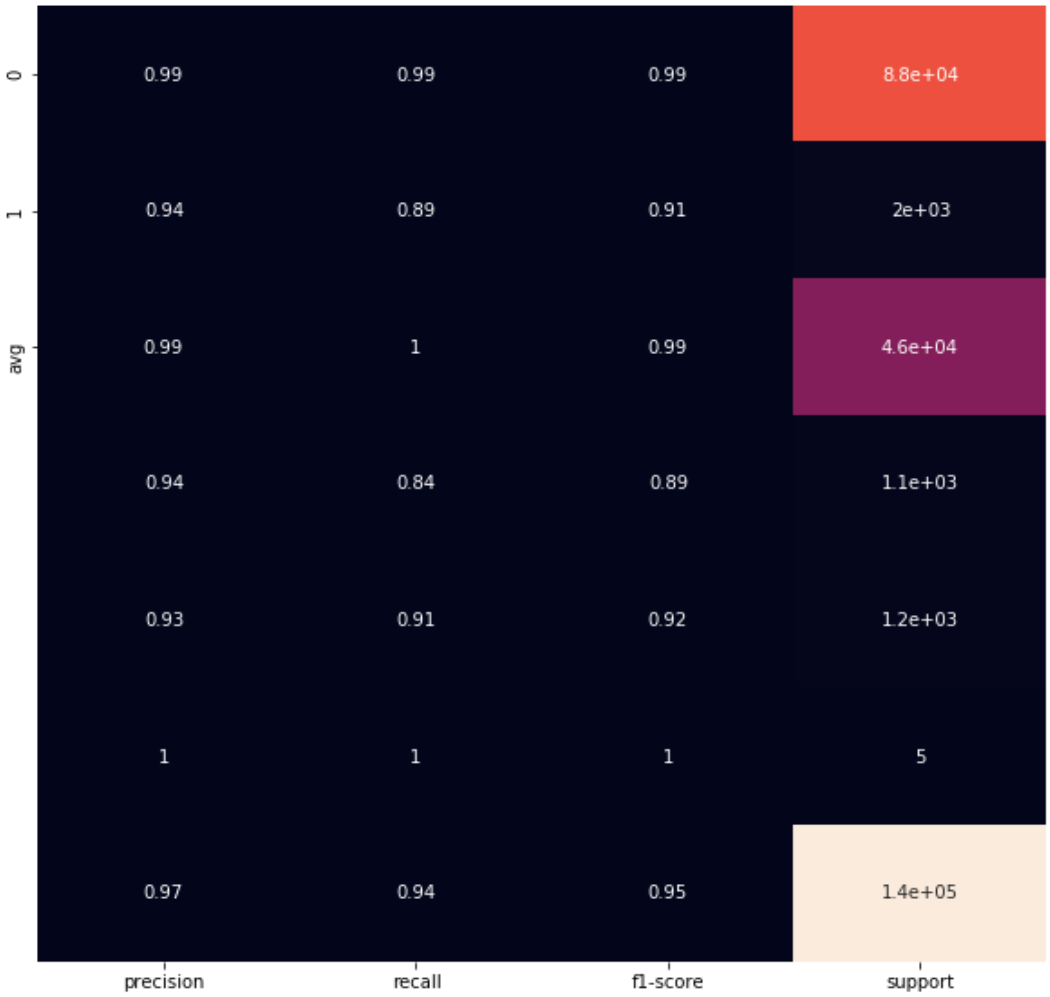

Figure 41 : rapport de classification.

Le rapport montre la précision des mesures de classification, le rappel et le score f1 principaux par classe. Les métriques sont calculées en utilisant des vrais et des faux positifs, des vrais et des faux négatifs.

On a la précision, le rappel, le score f1 et le support pour chaque classe que nous avons essayé de trouver.

- Le rappel signifie "combien de cette classe vous trouvez sur le nombre entier d'éléments de cette classe".
- La précision sera "combien sont correctement classés dans cette classe"
- Le score f1 est la moyenne harmonique entre précision et rappel
- Le support est le nombre d'occurrences de la classe donnée dans votre ensemble de données.

V. Conclusion :

Dans ce chapitre nous avons, en premier lieu, présenté les différents outils et langages que nous avons utilisés pour implémenter notre model. Ainsi une approche hybride de (PCA+SVM) a été utilisée.

Ce travail montre à quel point le jeu de données CICIDS2017 est très utile pour tester différents classificateurs, les travaux se concentrent sur la phase de prétraitement de CICIDS2017 afin de préparer des expériences fiables et des données de test indépendantes randomisées. Parmi les techniques de classification, le classifier SVM a atteint le taux de précision le plus élevé pour la détection et la classification de tous les types d'attaques de jeux de données CICIDS2017.

Conclusion

Conclusion générale

Le cyber sécurité est un domaine de recherche, et il existe de nombreuses solutions pour protéger les informations et rendre le système plus sécurisé. C'est l'un des sujets les plus impressionnants, compte tenu des problèmes rencontrés par les utilisateurs d'Internet et des services proposés comme le Cloud Computing. Pour sécuriser le Cloud, nous devons mettre en place des solutions de protection contre tout piratage ou attaque. IDS est la meilleure méthode d'analyse, de surveillance et de détection, car c'est l'outil de défense le plus important contre les attaques réseau sophistiquées et croissantes.

Parce qu'il existe des ensembles de données fiables pour les tests et la validation, il existe plusieurs méthodes de détection d'intrusion parmi lesquelles IDS 2017 a enregistré une occurrence d'enregistrement où les échantillons de données résultant de l'analyse des flux réseau sont stockés et traités dans des fichiers et c'est une tâche ardue car ces fichiers contiennent un grand nombre d'instances. Les données de chaque fichier prennent beaucoup de temps car il s'agit d'une fausse recherche

Cela a conduit à une réflexion sur l'amélioration des systèmes de détection d'intrusions existants basés sur des techniques d'apprentissage automatique, nous avons sélectionné les technologies les plus adaptées à notre choix technique, et avons principalement appliqué les techniques PCA et SVM pour scruter les informations en profondeur de ce processus.

Les techniques utilisées (PCA+SVM) présentent une grande capacité de modélisation pour IDS et une grande précision dans la classification utilisant les techniques d’apprentissage automatique. Dans le cadre de la tâche de classification multi classée sur le jeu de données CICIDS2017, le modèle peut effectivement améliorer à la fois la précision de la détection d'intrusion et la capacité de reconnaître le type d'intrusion.

Notre projet a été une opportunité pour approfondir nos connaissances dans le domaine de Machine Learning et d’apprendre ses différents modèles et leurs applications. Il est important pour nous de dire que l’un des avantages majeurs de ce travail est de familiariser avec la compréhension des articles et la maîtrise de plusieurs bibliothèques où nous avons vu et les exploiter pour la création des modèles.

Le système proposé utilise la combinaison d'un algorithme d'apprentissage automatique supervisé et non supervisé afin de détecter les attaques. On s'attend à ce que le système proposé qui utilise une combinaison du PCA et de SVM offre de meilleures

performances que les systèmes de détection d'intrusion existants, nous avons utilisé l'analyse des composants principaux (PCA) pour réduire la dimensionnalité de l'ensemble de données CICIDS2017 , et l'avons passé à un classificateur de vecteur d'état (SVC).Le classificateur de vecteur d'état a atteint une précision de 98.60% avec 35 composants principaux; l'ensemble de données se prête bien à la réduction de dimensionnalité.

Le modèle peut être développé avec différentes combinaisons d'algorithmes d'apprentissage automatique pour obtenir de meilleures performances et IDS avec deux ou plusieurs algorithmes d'apprentissage automatique peuvent être développé et testé sur différents environnements cloud.

Annex A

Administrateur : personne chargée de mettre en place la politique de sécurité, et par conséquent, de déployer et configurer les IDS.

Attaque : synonyme d'intrusion. Exploit : terme utilisé pour désigner un programme d'attaque.

Analyseur : outil logiciel qui met en œuvre l'approche choisie pour la détection (comportementale ou par scénarios). Il génère des alertes lorsqu'il détecte une intrusion à partir des événements remontés par les capteurs ou à partir d'alertes générées par d'autres analyseurs.

Opérateur : personne chargée de l'utilisation du manager associé à l'IDS. Elle propose ou décide de la réaction à apporter en cas d'alerte. C'est parfois la même personne que l'administrateur.

Intrusion : action (ou tentative d'action) qui a pour conséquence de compromettre l'intégrité, la confidentialité ou la disponibilité d'une ressource (violation de la politique de sécurité).

Signature : règle utilisée par certains analyseurs pour identifier parmi les activités surveillées celles qui sont caractéristiques d'une intrusion.

Détection d'intrusions : processus logiciel de recherche des intrusions qui s'appuie sur la surveillance des activités des entités dans les systèmes et les réseaux. Nous réservons le terme de détection d'intrusions à l'analyse logicielle et automatique par opposition à l'analyse manuelle de logs effectuées par un opérateur humain.

IDS : acronyme de "Intrusion Détection System". Voir système de détection d'intrusions.

Approche comportementale : ensemble des techniques utilisées par les IDS qui basent leur processus de détection sur l'hypothèse que toute déviation significative du comportement observé d'une entité par rapport à son modèle de comportement normal constitue une intrusion potentielle.

Système de détection d'intrusions : ensemble complet composé de capteur(s), d'analyseur(s) et de manager(s).

Politique de sécurité : spécification des règles à respecter dans le réseau d'une organisation, afin de garantir l'intégrité, la confidentialité et la disponibilité des ressources sensibles. Elle définit quelles activités sont autorisées et Les quelles sont interdites.

Faux positif : alerte émise en présence d'une action légitime rapportée à tort comme étant une intrusion par un système de détection d'intrusions (fausse alerte).

Alerte : message formaté qui décrit un événement relatif à une action qui compromet la sécurité d'un système ou d'un réseau. Les alertes sont produites par un analyseur. Nous faisons l'hypothèse que le format utilisé est l'IDMEF.

Approche par scénarios : ensemble des techniques utilisées par les IDS qui détectent les intrusions en recherchant dans les activités courantes celles qui sont caractéristiques de scénarios d'attaques connus (comparaison avec une base de signatures d'attaques) . Aussi appelée approche par signatures.

Capteur : logiciel qui génère les événements enfutant et formatant les données brutes intéressantes provenant d'une unique source d'information (paquets du réseau, logs du système ou logs applicatifs).

Manager : composant d'un IDS permettant à l'opérateur de gérer les autres composants. Ses fonctions comportent généralement la configuration des capteurs et analyseurs, la notification des alertes à l'opérateur et éventuellement la réaction.

Scénario : suite des étapes d'une intrusion. Réaction : mesures passives ou actives qui peuvent être prises en réponse à la détection d'une attaque, pour la stopper ou pour corriger ses effets.

Sonde : regroupement (logique ou fonctionnel) d'un capteur et d'un analyseur.

Faux négatif : absence d'alerte en présence d'une action qui constitue bien une intrusion mais qui n'a pas été détectée comme telle par un système de détection d'intrusions.

Annex B

1. Algorithme PCA :

1: **Input**: a D-dimensional training set $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}$ and the new (lower) dimensionality d (with $d \leq D$)
2: Compute the mean $\bar{\boldsymbol{x}} = \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{x}_i$
3: Compute the covariance matrix $\text{Cov}(\boldsymbol{x}) = \frac{1}{N} \sum_{i=1}^{N} (\boldsymbol{x}_i - \bar{\boldsymbol{x}}_i)(\boldsymbol{x}_i - \bar{\boldsymbol{x}}_i)^T$
4: Find the spectral decomposition of $\text{Cov}(\boldsymbol{x})$, obtaining the eigenvectors $\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \ldots, \boldsymbol{\xi}_D$ and their corresponding eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_D$. Note that the eigenvalues are sorted, such that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_D \geq 0$
5: For any $\boldsymbol{x} \in \mathbb{R}^D$, its new lower dimensional representation is:

$$\boldsymbol{y} = \left(\boldsymbol{\xi}_1^T(\boldsymbol{x} - \bar{\boldsymbol{x}}), \boldsymbol{\xi}_2^T(\boldsymbol{x} - \bar{\boldsymbol{x}}), \ldots, \boldsymbol{\xi}_d^T(\boldsymbol{x} - \bar{\boldsymbol{x}})\right)^T \in \mathbb{R}^d,$$

and the original $\boldsymbol{x}$ can be approximated as

$$\boldsymbol{x} \approx \bar{\boldsymbol{x}} + (\boldsymbol{\xi}_1^T(\boldsymbol{x} - \bar{\boldsymbol{x}}))\boldsymbol{\xi}_1 + (\boldsymbol{\xi}_2^T(\boldsymbol{x} - \bar{\boldsymbol{x}}))\boldsymbol{\xi}_2 + \cdots + (\boldsymbol{\xi}_d^T(\boldsymbol{x} - \bar{\boldsymbol{x}}))\boldsymbol{\xi}_d$$

2. Algorithme SVM :

AttributeSupportVector(*ASV*)
={Closest Attribute Pair from Opposite Classes}
1: while margin constraint violating points exist do
2: Find the violator
3: ASV = ASV∪ Violator
4: if any $\alpha_p < 0$ because of addition of c to S then
5: $ASV = \frac{ASV}{p}$
6: Repeat all the violating points are pruned
7: end if
8: end while

3. Algorithme de k-means

Entrée

- Un ensemble de données D, où chaque instance X_i est décrite par un vecteur de d dimensions et par une classe $Y_i \in \{1, \ldots, J\}$.
- Le nombre de clusters souhaité, noté K.

Début

1) Prétraitement des données.
2) Initialisation des centres.

Pour un nombre fixé de partitions, noté **R faire**

Répéter

3) *Affectation :* générer une nouvelle partition en assignant chaque instance X_i au groupe dont le centre est le plus proche.

$$X_i \in C_k \forall j \in 1, \ldots, K \qquad k = min_j \parallel X_i - \mu_j \parallel$$

avec μ_k est le centre de gravité du cluster C_k.

4) *Représentation :* calculer les centres associés à la nouvelle partition

$$\mu_k = \frac{1}{N_k} \sum_{X_i \in C_k} X_i$$

jusqu'à ce que (convergence de l'algorithme)

Fin Pour

5) Choix de la meilleure partition parmi les R partitions.
6) Attribution des classes aux clusters formés.
7) Prédiction de la classe des nouvelles instances.

Fin

Sortie

- Chaque cluster est représenté par un prototype qui possède la même prédiction de classe.
- Chaque cluster est associé à une description donnée par le biais de langage B.
- L'inertie intra-clusters est minimale (l'homogénéité des instances est maximale).
- L'inertie inter-clusters est maximale (la similarité entre les clusters est minimale).
- Le taux de bonnes classifications est maximal.

4. Algorithme arbre de décision

```
entrée : échantillon S
début
    Initialiser l'arbre courant à l'arbre vide ; la racine est le nœud courant
    répéter
        Décider si le nœud courant est terminal

    Si le nœud est terminal alors
        Lui affecter une classe
    sinon
        Sélectionner un test et créer autant de nouveaux nœuds fils
        qu'il y a de réponses possibles au test
    FinSi
    Passer au nœud suivant non exploré s'il en existe
Jusqu'à obtenir un arbre de décision
fin
```

Bibliographie

[1] «Fatma., Bensaha Fatima et Smail,Composition des services Cloud à base de séries temporelles. 2017. »

[2] «https://www.hebergeurcloud.com/definition-cloud-computing-selon-nist/Accède le 20/01/2020 a 22 :20,»

[3] «https: //www.supinfo.com/articles/single/2760-presentation-cloud-computing Accède le12/01/2020 a 12:26,»

[4] «https://blog.rsisecurity.com/nist-definition-of-cloud-computing/Accède le 10/01/2020 a 13:54,»

[5] «N. Grevet. Le cloud computing : évolution ou révolution ? Pourquoi, quand, comment et surtout faut-il prendre le risque ?, Août 2009.»

[6] «https://connect.ed-diamond.com/MISC/MISC-060/Introduction-au-Cloud-Computing-risques-et-enjeux-pour-la-vie-priveeAccède le 24/02/2020 a 12:00,»

[7] «https://support.cloudwatt.com/kb/faq/lecloud/quels-sont-les-avantages-du-cloud-computing.htmlAccède le 24/01/2020 a 18:09,»

[8] «http://www.renaudvenet.com/cloud-computing-avantages-et-inconvenients-2011-01-26.html Accède le 20/01/2020 a 22 :26,»

[9] «H. Saouli. Découverte de services web via le Cloud computing à base d'agents mobiles. Thèse de doctorat. Université Mohamed Khider de Biskra, 2015.»

[10] «K. Maioua, A. Mansouri. Approche basée Agents Mobiles intelligents dans un environnement de cloud Computing. Mémoire de master. Université KasdiMerbah Ouargla,2014. »

[11] «Vincent Kherbache, Mohamed Moussalih, Yannick Kuhn, Allan Lefort, Cloud Computing, Edition Eucalyptus, 2010. »

[12] «J. Zhu.Cloud Computing Technologies and Application. Springer Science+Business Media, LLC2010 »

[13] «https://www.techniques-ingenieur.fr/base-documentaire/technologies-de-l-information-th9/attaques-et-mesures-de-protection-des-si-42313210/pare-feu-te7550/ids-et-ips-pour-detecter-et-reagir-aux-intrusions-te7550v2niv10009.htmlAccède le 10/02/2020 a 23:09,»

[14] « Martin Arvidson Markus Carlbark ,Intrusion Detection Systems – Technologies, Weaknesses and Trends 2003. »

[15] «Nicolas Baudoin, MarionKarle :NT Réseaux IDS et IPS 2003/2004. »

[16] « LiranLERMAN, L, Les systèmes de détection d'intrusion basés sur du machine learning, Université LIBRE de BRUXELLES. »

[17] «E. Hodo, X. Bellekens, A. Hamilton, C. Tachtatzis, and R. Atkinson.―Shallow and deep networks intrusion detection system: A taxonomy and survey,‖arXiv preprint arXiv:1701.02145, 2017. »

[18] «Hervé Debar, Benjamin Morin, FrédéricCuppens, Fabien Autrel, LudovicMé, Bernard Vivinis Salem Benferhat, MireilleDucassé, RodolpheOrtalo, Détectiond'intrusions : corrélationd'alertes. Article de synthèse, Caen, France, 2004. »

[19] «Cédric Michel, Langage de description d'attaques pour la détection d'intrusions par corrélation d'événements ou d'alertes en environnement réseau hétérogène, thèse de doctorat de l'Université de Rennes1,16 Décembre 2003. »

[20] «Nathalie Dagorn, Détection et prévention d'intrusion : présentation et limites', Rapport de recherche, 6 july 2006. »

[21] «A. Ahmed, Système de détection d'intrusion adaptatif et distribué,2014, pp.66-67. »

[22] « Cédric Liorens, Laurent Levier, Cenis Valois. 2ièm édition" Editions Eyrolles, 61, bld Saint-Germain, Tableaux de bord de la sécurité réseau. 2003, 2006. »

[23] « M. Amand, M. Nsiri,―Etude d'un système de détection d'intrusion comportemental pour l'analyse du trafic aéroportuaire,‖ Rapport de projet tutoré, Jan 2011. »

[24] « Belkhatmi, Keltouma, and Ouarda Benamara. Mise en place d'un systéme de détection et de prévention d'intusion. Diss. Université de Bejaia, 2016.»

[25] « A. Phillip, Porras et A. Valdes,Live traffic analysis of tcp/ip gateways. Proc. ISOC Symposium on Network and Distributed System Security (NDSS98). San Diego, Mars 1998. »

[26] «Cédric Liorens, Laurent Levier, Cenis Valois. 2ièm édition" Editions Eyrolles, 61, bld Saint-Germain, Tableaux de bord de la sécurité réseau. 2003, 2006. »

[27] «Yann Berthier, Jean-Baptiste Marchand, Détection d'intrusions et analyse forensique.»

[28] «http://tpe-intelligence-artificielle-2013.e-monsite.com/pages/definition-de-l-intelligence-artificielle.html Accès le 16/01/2020 à 11 :52,»

[29] « https://www.oracle.com/fr/cloud/deep-learning-intelligence-artificielle.html Accès le 16/01/2020 à 12 :00,»

[30] «https://digitalinsiders.feelandclic.com/construire/definition-quest-machine-learning Accès le 30/03/2020 à 19 :23,»

[31] «https://www.futura-sciences.com/tech/definitions/intelligence-artificielle-deep-learning-17262/ Accès le 29/03/2020 à 21 :18,»

[32] «https://www.supinfo.com/articles/single/6041-machine-learning-introductionAccès le 16/01/2020 à

16:31,»

[33] «https://le-datascientist.fr/lapprentissage-du-machine-learning Accès le 17/01/2020 à 17:38,»

[34] «https://le-datascientist.fr/apprentissage-supervise-vs-non-supervise Accès le31/03/2020 à 17:05,»

[35] «https://www.supinfo.com/articles/single/6041-machine-learning-introduction-apprentissage-automatique Accès le 29/03/2020 à 22 :28»

[36] «https://www.geeksforgeeks.org/ml-classification-vs regression/?fbclid=IwAR0p3EgOEsEPevFPw7tL6i41ULybNoOCvZWpqtibLZ86NoFatzwAbtFkdU Accès le 16/01/2020 à 17 :15»

[37] «https://mrmint.fr/9-algorithmes-de-machine-learning-que-chaque-data-scientist-doit-connaitre Accès le 31/03/2020 à 17 :10»

[38] «https://whatis.techtarget.com/fr/definition/Regression-logistique Accès le 29/03/2020 à 22 :37»

[39] « Mathian, H. and L. Sanders (2006). Les méthodes de classification de données spatiales.»

[40] «https://www.supinfo.com/articles/single/6041-machine-learning-introduction Accès le 01/04/2020 à 18 :10»

[41] «Thirumuruganathan, s., a detailed introduction to k-nearest neighbor (knn) algorithm , <https://saravananthirumuruganathan.wordpress.com/2010/05/17/a-detailed-introduction-to-k-nearest-neighbor-knn-algorithm/>, may 2010.»

[42] «https://moncoachdata.com/blog/algorithme-k-plus-proches-voisins/ Accès le 04/04/2020 à 10 :38»

[43] «Mohamadally, H. and B. Fomani (2006). SVM : Machines à vecteurs de support ou séparateurs à vastes marges. Versailles St Quentin.»

[44] «https://www.lucidchart.com/pages/fr/quest-ce-quun-arbre-de-decision Accès le 05/04/2020 à 9:10»

[45] «Nakache, D. (2007). Extraction automatique des diagnostics à partir des comptes rendus médicaux textuels. Laboratoire CEDRIC - équipe ISID. Paris, Conservatoire National des Arts et Métiers: 219.»

[46] «https://invenis.co/3-algorithmes-de-machine-learning-bien-utiles-business/ Accès le 04/04/2020 à 10 :38»

[47] «https://lovelyanalytics.com/2016/09/06/k-means-comment-ca-marche/ Accès le 30/03/2020 à 19 :38»

[48] «Hilali, h., *application de la classification textuelle pour l'extraction des règles d'association maximales.* thèse de maitrise en informatique, université du québec à trois-rivières, trois-rivières, 2009.»

[49] «https://docs.aws.amazon.com/fr_fr/sagemaker/latest/dg/pca.html Accès le 24/05/2020 a 12:40»

[50] « Fertil, B. (2006). Reconnaissance des formes : Classement d'ensembles d'objets.»

[51] « Quang, C. T. (2005). Classification automatique des textes vietnamiens Hanoi, Institut de la Francophonie pour l'informatique.»

[52] « John, Shawe-Taylor (2000), Nello Cristianini, Support Vector Machines and other kemel-based leaming methods, Cambridge University Press.»

[53] «https://www.supinfo.com/articles/single/8381-reseaux-quand-ips-ids-s-mele?fbclid=IwAR1DL5ATCcrakiGIEFWswFh9AXMSqb0SRKVXXXkcjkKtj6dQQ_0lRhusbE. html Accès le 24/05/2020 a 12:40»

[54] «Graham-Cumming, J. (2006). "Interview de John Graham-Cumming, l'auteur du logiciel»

[55] «Denoue, L. (2003). Classification supervisée de documents.»

[56] «Zeitouni, K. (2006). Analyse et extraction de connaissances des bases de données»

[57] «TUFFERY Stéphane, Data Mining et statistique décisionnelle: L'intelligence des données .»

[58] «https://www.stat4decision.com/fr/voila-dashboards-a-partir-de-vos-jupyter-notebooks/Accède le 09/05/2020 a 22 :30.»

[59] «https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1445304-python-definition-et-utilisation-de-ce-langage-informatique/Accède le 09/05/2020 a 20 :26. »

[60] «https://www.it-swarm.dev/fr/python/quel-est-le-lien-entre-anaconda-et-python/829887380/Accède le 09/05/2020 a 20 :26. »

[61] «https://www.yubigeek.com/developper-en-python-avec-anaconda/Accède le 09/05/2020 a 22 :29.»

[62] «https://www.stat4decision.com/fr/voila-dashboards-a-partir-de-vos-jupyter-notebooks/Accède le 09/05/2020 a 22 :30. »

[63] «https://python.developpez.com/tutoriels/graphique2d/matplotlib/Accède le20/05/2020 a 22 :04.»

[64] «https://openclassrooms.com/fr/courses/4452741-decouvrez-les-librairies-python-pour-la-data-science/5559011-realisez-de-beaux-graphiques-avec-seaborn Accède le 20/05/2020 a 22:41. »

[65] «https://www.inria.fr/fr/lancement-de-linitiative-scikit-learn?fbclid=IwAR1r89W0NsQHyu7BN31qRQJq5YEUS0iORwj37i51Zj0ds35stAwHCL-8N8c Accède le 20/05/2020 a 22 :11. »

[66] «https://numpy.org/ Accède le 20/05/2020 a 22 :26. »

[67] «https://www.datacorner.fr/pandas_1/ Accède le 20/05/2020 a 22 :31. »

[68] «https://pypi.org/project/scipy// Accède le 20/05/2020 a 23 :18. »

[69] «https://pypi.org/project/requests/2.7.0/ Accède le 24/05/2020 a 10:39. »

[70] «https://www.geeksforgeeks.org/how-to-use-glob-function-to-find-files-recursively-in-python/ Accède le 24/05/2020 a 10 :51.»

[71] «https://www.unb.ca/cic/datasets/nsl.html IDS 2017 | Datasets | Research | Canadian Institute for Cybersecurity | UNB Accède le 17/03/2020 a 10 :51. »

[72] «https://developers.google.com/machine-learning/crash-course/classification/true-false-positive-negative?hl=fr Accède le 24/06/2020 a 19 :06»

[73] « P. Mahé :" Noyaux pour graphes et Support Vector Machines pour le criblage virtuel de molécules ". Rapport de stage, DEA MVA 2002/2003,Septembre 2003»

[74] « Mohamadally Hasan,Fomani Boris :" SVM machine à vecteurs de support ou séparateur à vaste marge ". BD Web, ISTY3,Versailles St Quentin, France, janvier 2006.»

[75] «A. Cornuéjols :" Une nouvelle méthode d'apprentissage : Les SVM. Séparateurs à vaste marge". Université de Paris-Sud, Orsay, France, Juin 2002.»

[76] «https://www.memoireonline.com/12/19/11370/m_Facteurs-explicatifs-de-linadequation-professionnelle10.html Accède le 25/06/2020 a 19 :38»