الجمهورية الجزائرية الديمقراطية الشعبية

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة التعليم العالي والبحث العلمي

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

جامعة 20 اوت -1955 سكيكدة

UNIVERSITE 20 AOUT 1955- SKIKDA


Faculté des Sciences Département d'informatique

Département d'informatique

Mémoire Présenté en Vue de l'Obtention du Diplôme de Master

Spécialité : Génie Logiciel Avancée et Application

Intitulé :

# Combinaison de la Visualisation et du Deep Learning pour la Détection d'Intrusions sur la Base NSL-KDD.

Realisé par :

Zemmali Fatima zahra

Encadré par :

Dr. Cheikh Mohamed

Année universitaire :2023/2024

# Remerciement

*Louange à Dieu qui m'a guidé à la réalisation de ce modeste projet de fin d'études.*

*Je souhaite exprimer ma profonde gratitude au Dr. Mohamed Cheikh pour m'avoir proposé un sujet si intéressant, pour son encadrement patient, ses précieux conseils et son soutien constant tout au long de ce travail.*

*Mes remerciements vont également aux membres du jury pour leur évaluation de mon travail.*

# Dédicace

*Je dédie mon travail de fin d'étude à ma très chère mère, Kafi Sabiha, pour son amour inconditionnel, son soutien sans faille et ses encouragements constants qui ont été ma source d'inspiration tout au long de ce voyage académique.*

*À mon père, Abderrahmane.*

*À ma sœur, Chaïma, pour son soutien indéfectible, et à mon frère, Mohamed.*

*À Monsieur Dennis, pour m'avoir encouragé et soutenu.*

# Résumé

Les attaques de déni de service (DoS) et distribuées (DDoS) ciblent la disponibilité des ressources réseau légitimes, perturbant gravement les services en ligne par la surcharge des serveurs ou l'exploitation de vulnérabilités dans les protocoles réseau. Les systèmes de détection d'intrusion (IDS) jouent un rôle crucial dans la prévention de ces attaques en surveillant le trafic réseau et en identifiant les comportements malveillants. Ce travail propose une méthode novatrice de détection basée sur l'analyse en temps réel et la représentation graphique des paramètres réseau. En utilisant l'ensemble de données NSL KDD, notre approche démontre une efficacité accrue dans la détection des attaques DoS en exploitant les techniques d'apprentissage profond, notamment les réseaux neuronaux convolutifs (CNN), et en utilisant des visualisations géométriques pour détecter les comportements suspects. Cette approche renforce la capacité des administrateurs à réagir promptement aux menaces de sécurité, améliorant ainsi la résilience des infrastructures face aux attaques informatiques.



**Les mots clés :** Système de Détection d'intrusion, Attaques de déni de service (DoS), Attaques distribuées (DDoS) ,Réseaux neuronaux convolutifs (CNN), , NSL KDD.

## ملخص

هجمات حجب الخدمة (DoS) والهجمات الموزعة لحجب الخدمة (DDoS) تستهدف موارد الشبكة الشرعية، مما يؤدي إلى إعاقة خدمات الإنترنت بشكل كبير من خلال تحميل الخوادم أو استغلال الثغرات في بروتوكولات الشبكة. تلعب أنظمة كشف الاختراق (IDS) دورًا حاسمًا في منع هذه الهجمات من خلال مراقبة حركة المرور عبر الشبكة وتحديد السلوكيات الخبيثة. يقترح هذا العمل طريقة اكتشاف مبتكرة تعتمد على تحليل في الوقت الحقيقي وتمثيل بيانات الشبكة بشكل رسومي. باستخدام مجموعة بيانات NSL KDD ، تظهر نهجنا كفاءة محسّنة في اكتشاف هجمات DoS من خلال استخدام تقنيات التعلم العميق، بما في ذلك الشبكات العصبية التكرارية (CNN)، واستخدام التصورات الهندسية لاكتشاف السلوكيات المشبوهة. يعزز هذا النهج قدرة المسؤولين على الاستجابة الفورية للتهديدات الأمنية، مما يعزز من مرونة البنية التحتية ضد الهجمات الإلكترونية.



**الكلمات الرئيسية**: نظام كشف الاختراق، هجمات حجب الخدمة(DoS) ، هجمات حجب الخدمة الموزعة(DDoS) ، شبكات العصب العصبي التكراري(CNN) ، مجموعة بيانات.NSL KDD

# Abstract

Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks target the availability of legitimate network resources, severely disrupting online services by overloading servers or exploiting vulnerabilities in network protocols. Intrusion Detection Systems (IDS) play a crucial role in preventing these attacks by monitoring network traffic and identifying malicious behaviors. This work proposes an innovative detection method based on real-time analysis and graphical representation of network parameters. Using the NSL KDD dataset, our approach demonstrates enhanced effectiveness in detecting DoS attacks by leveraging deep learning techniques, specifically Convolutional Neural Networks (CNNs), and utilizing geometric visualizations to detect suspicious behaviors. This approach enhances administrators' ability to promptly respond to security threats, thereby improving infrastructure resilience against cyber-attacks.



**Keywords** : Intrusion Detection System, Denial of Service (DoS) attacks, Distributed Denial of Service (DDoS) attacks, Convolutional Neural Networks (CNN), NSL KDD .

**Table des matières** :

## Chapitre 2 : Les systèmes de détection d'intrusions

## Chapitre03 : Conception et Implémentation

# Liste des figures

# Liste des tableaux

# Introduction générale :

Les progrès technologiques, le développement des moyens de communication et l'ouverture aux nouvelles technologies ont rendu les réseaux et systèmes informatiques indispensables au fonctionnement et au développement de la plupart des entreprises. Ces systèmes sont largement utilisés dans divers domaines tels que l'industrie, le marketing, l'assurance, la médecine et l'éducation. Cependant, cette interdépendance croissante des systèmes et des réseaux les rend vulnérables à un groupe diversifié et croissant d'utilisateurs potentiellement malveillants. Il n'est donc pas surprenant de constater une certaine méfiance à l'égard des réseaux, car ils peuvent être la cible d'attaques visant à accéder, lire, modifier ou détruire des informations sensibles, perturbant ainsi leur bon fonctionnement. La sécurisation des réseaux est donc devenue un enjeu crucial pour éviter toute intrusion non autorisée.

La sécurité des réseaux peut être mise en place de manière préventive ou réactive. Cependant, même avec une approche préventive, il est difficile d'assurer une sécurité totale en raison de la complexité des systèmes et des failles potentielles. Afin de protéger les systèmes contre toutes les éventuelles attaques, une approche interactive est souvent adoptée. Cette méthode consiste à détecter rapidement les attaques pour pouvoir y réagir promptement. Parmi les outils de sécurité, le système de détection d'intrusion (IDS) joue un rôle crucial. La détection d'intrusion consiste essentiellement à rechercher des signes d'attaque. Lorsqu'une intrusion est détectée, l'IDS peut prendre les mesures nécessaires selon son type et sa programmation par l'administrateur.

Dans ce travail, nous nous concentrons sur l'approche de classification grâce à une technique de détection basée sur une représentation visuelle du trafic réseau. Notre approche combine la visualisation des modèles de trafic réseau avec les capacités du deep learning, en particulier les réseaux neuronaux convolutifs (CNN). Cette représentation vise à identifier des modèles

visuels d'attaques DoS (Denial of Service) et à les distinguer du trafic normal. Ces modèles sont ensuite utilisés dans un modèle CNN pour la classification afin de détecter les intrusions . Ainsi, on peut considérer le problème de la détection d'intrusion comme un problème de reconnaissance de formes. La classification n'est donc pas faite sur la base de paramètres souvent complexes, mais plutôt sur la base de formes issues d'une transformation géométrique. L'objectif est d'obtenir de meilleurs résultats de détection en utilisant cette approche visuelle, qui pourrait simplifier et améliorer la précision du processus de classification des intrusions.

Ce travail est structuré comme suit :

•**Chapitre** 1 : aborde les définitions et généralités sur les notions de base de la sécurité informatique.

•**Chapitre2** : On détail sur les systèmes de détection d'intrusion, architecture, méthodes, leur différent types, emplacement et avantages.

• **Chapitre 3** : On parle sur la conception de modèle et après l'implémentation toute en passant sur la plateforme utilisée, les langages de programmation et les résultats obtenus.

# CHAPITRE I :
# La sécurité informatique.

# 1 Introduction :

Dans un monde où la technologie façonne de plus en plus nos interactions quotidiennes, la sécurité informatique émerge comme un pilier essentiel pour protéger nos données, nos activités et nos systèmes. Avec la prolifération des menaces en ligne, il est crucial de comprendre les principes fondamentaux de la sécurité informatique ainsi que les mesures nécessaires pour prévenir les cyberattaques et assurer la protection des informations sensibles.

# 2 La sécurité informatique

## 2.1 Définition de la sécurité :

La sécurité informatique est l'ensemble des moyens techniques qui visent à empêcher l'utilisation non autorisée des ressources matérielles ou logicielles. On peut dire aussi que la sécurité informatique est un ensemble des moyens mis en œuvre pour réduire la vulnérabilité d'un système contre les menaces accidentelles ou intentionnelles. [1]

## 2.2 Les types de sécurité informatique :

D'après les experts, il peut y avoir trois, six ou même plus de types de sécurité informatique. Chaque expert en sécurité a ses propres catégorisations. De plus, à mesure que les réseaux continuent de se développer avec le cloud et d'autres nouvelles technologies, de plus en plus de types de sécurité informatique émergeront.

Cependant, pour la plupart, il existe trois grands types de sécurité informatique : la sécurité réseau, la sécurité des terminaux et la sécurité Internet (la sous-catégorie de la cybersécurité). Les autres types de sécurité informatique peuvent généralement relever de ces trois types :

- **Sécurité du réseau :**

Dans sa forme la plus simple, la sécurité réseau fait référence à l'interaction entre les différents appareils d'un réseau. Cela inclut le matériel et le logiciel. La sécurité réseau, selon le SANS Institute, s'efforce de protéger l'infrastructure réseau sous-jacente contre les accès non autorisés, les abus, les dysfonctionnements, les modifications, les destructions ou les

divulgations inappropriées, créant ainsi une plate-forme sécurisée permettant aux ordinateurs, aux utilisateurs et aux programmes d'exécuter leurs fonctions critiques autorisées dans un environnement sécurisé.

- **Sécurité des terminaux :**

une mesure cruciale visant à garantir que seuls les appareils authentifiés ont accès au système ou aux données. Elle se concentre sur la protection contre les menaces de sécurité au niveau des appareils tels que les ordinateurs portables, les téléphones portables, les tablettes, etc. Chaque nouvelle connexion sur le réseau d'une entité étend le champ de l'intelligence sur les menaces.

- **Sécurité Internet :**

La sécurité Internet désigne la protection des activités et des transactions effectuées sur Internet. C'est un élément spécifique des concepts plus larges de cybersécurité et de sécurité informatique, englobant des sujets tels que la sécurité des navigateurs, le comportement en ligne et la sécurité des réseaux.[2]

## 2.3 Exigences de la sécurité :

Lorsque nous abordons le problème de sécurité, nous visons à atteindre certains objectifs. Les objectifs de sécurité sont classés comme principaux et secondaires.

Les principaux objectifs sont connus comme objectifs standards de sécurité tel que : la confidentialité, l'authentification, l'intégrité et la disponibilité. Les objectifs secondaires sont : la fraicheur de données, la non-répudiation, le contrôle d'accès, l'auto-organisation, la synchronisation et la localisation sécurisée.

### 2.3.1 Les principaux objectifs de la sécurité informatique :

Les systèmes d'information représentent l'ensemble des données de l'entreprise et les infrastructures matérielles et logicielles. La sécurité informatique d'une manière générale,

consiste à assurer que les ressources d'une organisation, soient uniquement utilisées dans le cadre prévu.

- La confidentialité : les données ne doivent être visibles que pour les personnes autorisées.
- L'authentification : consiste à assurer l'identité d'un utilisateur, c'est-à-dire garantir à chacun des correspondants, que son partenaire est bien celui qu'il croit être.
- L'intégrité : il faut pouvoir garantir que les données protégées n'ont pas été modifiées par une personne non autorisée. Le but étant de ne pas altérer les informations sensibles de l'entreprise.
- La non-répudiation : il s'agit de garantir qu'aucun des correspondants ne pourra nier la transaction effectuée.
- La disponibilité : les données doivent restées accessibles aux utilisateurs. C'est la capacité à délivrer un service permanent à l'entreprise.

## 2.4 Principaux défauts de la sécurité informatique :

Les défauts de sécurité d'un système d'information les plus souvent constatés sont :

- Installation des logiciels et matériels par défaut.
- Mises à jour non effectuées.
- Mots de passe inexistants ou par défaut.
- Services inutiles conservés (Netbios…).
- Traces inexploitées.
- Pas de séparation des flux opérationnels des flux d'administration des systèmes.
- Télémaintenance sans contrôle fort.
- Procédures de sécurité obsolètes (périmés).
- Authentification faible.

## 2.5 Les Causes pour sécuriser les réseaux

## 2.6 Les enjeux

- **Enjeux économique :**

Les organismes ou entreprises à but lucratif ont presque toujours la même finalité : c'est de réaliser des bénéfices sur l'ensemble de leurs activités. Cette réalisation est rendue possible grâce à son système d'information considéré comme moteur de développement de l'entreprise. D'où la nécessité de garantir la sécurité de ce dernier. La concurrence fait que des entreprises s'investissent de plus en plus dans la sécurisation de leurs systèmes d'information et dans la qualité de service fournit aux clients . [3]

- **Enjeux politiques :**

La plupart des entreprises ou organisations se réfèrent aux documents officiels de sécurité élaborés et recommandés par l'État. Ces documents contiennent généralement des directives qui doivent être appliquées par toute structure engagée dans un processus de sécurisation du réseau. Dans le cadre du chiffrement des données par exemple, chaque État définit des cadres et mesures d'utilisation des algorithmes de chiffrement et les recommande aux entreprises exerçant sur son territoire. Le non-respect de ces mesures et recommandations peut avoir des conséquences graves sur l'entreprise. A ce niveau, l'enjeu est plus politique parce que chaque État souhaite être capable de décrypter toutes les informations circulant dans son espace .[3]

- **Enjeux juridiques** :

Dans un réseau, on retrouve de l'information multiforme (numérique, papier, etc.). Le traitement de celle-ci doit se faire dans un cadre bien définit et dans le strict respect des lois en vigueur. En matière de juridiction, le non-respect des lois et exigences relatives à la manipulation des informations dans un système d'information peut avoir des conséquences graves sur l'entreprise .[3]

### 2.6.1 Les Vulnérabilités

Tous les systèmes informatiques sont vulnérables. Peu importe le niveau de vulnérabilité de ceux-ci. Une vulnérabilité est une faille ou une faiblesse pouvant être exploitée par une personne mal intentionnée pour nuire .[3]

Les vulnérabilités des systèmes peuvent être classées en catégorie (humaine, technologique, organisationnelle, mise en œuvre).

- **Vulnérabilités humaines** : L'être humain, de par sa nature, est vulnérable.

La plupart des vulnérabilités humaines proviennent des erreurs (négligence, manque de compétences, surexploitation, etc.), car ne dit-on pas souvent que l'erreur est humaine Un SI étant composé des humains, il convient d'assurer leur sécurité si l'on veut garantir un maximum de sécurité dans le SI .[3]

- **Vulnérabilités technologiques** : Avec la progression exponentielle des outils informatiques,

les vulnérabilités technologiques sont découvertes tous les jours. Ces vulnérabilités sont à la base dues à une négligence humaine lors de la conception et la réalisation. Pour être informé régulièrement des vulnérabilités technologiques découvertes, il suffit de s'inscrire sur une liste ou des listes de diffusion mises en place par les CERT (Computer Emergency Readiness ou Response Team) . [3]

- **Vulnérabilités organisationnelles** : Les vulnérabilités d'ordre organisationnel sont dues à l'absence des documents cadres et formels, des procédures (de travail, de validation) suffisamment détaillées pour faire face aux problèmes de sécurité du système. Quand bien même ces documents et procédures existent, leur vérification et mises à jour ne sont pas toujours bien assurées. [3]
- **Vulnérabilités mise en œuvre** : Les vulnérabilités au niveau mise en œuvre peuvent être dues au non prise en compte de certains aspects lors de la réalisation d'un projet. [3]

### 2.6.2 Les Menaces :

Une menace est un événement, d'origine accidentelle ou délibérée, capable s'il se réalise de causer un dommage au sujet étudié. Le réseau informatique comme tout autre réseau informatique est en proie à des menaces de toutes sortes qu'il convient de recenser. [3]

### 2.6.3 Les logiciels malveillants :

Ce sont des logiciels développés par des hackers dans le but de nuire à un système d'informations.

- **Les Virus** : un virus est un segment de programme qui, lorsqu'il s'exécute, se reproduit en s'adjoignant à un autre programme (du système ou d'une application), et qui devient ainsi un cheval de Troie. Puis le virus peut ensuite se propager à d'autres ordinateurs (via un réseau) à l'aide du programme légitime sur lequel il s'est greffé. Il peut également avoir comme effets de nuire en perturbant plus ou moins gravement le fonctionnement de l'ordinateur infecté. [4] Les virus peuvent être classés suivant leur mode de propagation et leurs cibles : [5]
- **Le virus de boot** : il est chargé en mémoire au démarrage et prend le contrôle de l'ordinateur.
- **Le virus d'application** : ils infectent les programmes exécutables, c'est-à-dire les programmes (.exe, .com ou .Sys) en remplaçant l'amorce du fichier, de manière à ce que le virus soit exécuté avant le programme infecté. Puis ces virus rendent la main au programme initial, camouflant ainsi leur exécution aux yeux de l'utilisateur .
- **La macro virus** : il infecte des logiciels de la suite Microsoft Office les documents bureautiques en utilisant leur langage de programmation, qui contaminera tous les documents basés sur lui, lors de leur ouverture.
- **Les Vers** : Un ver est un programme autonome qui se reproduit et se propage à l'insu des utilisateurs. Contrairement aux virus, un ver n'a pas besoin d'un logiciel hôte pour se dupliquer. Le ver a habituellement un objectif malicieux, par exemple :

- Espionner l'ordinateur dans lequel il réside.
- Offrir une porte dérobée à des pirates informatiques.
- Détruire des données sur l'ordinateur infecté.
- Envoyer de multiples requêtes vers un serveur internet dans le but de le saturer. [4]

• **Les chevaux de Troie** : Un cheval de Troie est une forme de logiciel malveillant déguisé en logiciel utile. Son but : se faire exécuter par l'utilisateur, ce qui lui permet de contrôler l'ordinateur et de s'en servir pour ses propres fins. Généralement d'autres logiciels malveillants seront installés sur votre ordinateur, tels que permettre la collecte frauduleuse, la falsification ou la destruction de données. [4]

• **Le logiciel espion** : (Espiogiciel ou logiciel espion) est un programme ou un sous-programme, conçu dans le but de collecter des données personnelles sur ses utilisateurs et de les envoyer à son concepteur, ou à un tiers via Internet ou tout autre réseau informatique, sans avoir obtenu au préalable une autorisation explicite et éclairée desdits utilisateurs. [4]

• **Le spam** : correspond à l'envoi intempestif de courriers électroniques, publicitaires ou non, vers une adresse mail. Le spam est une pollution du courrier légitime par une énorme masse de courrier indésirable non sollicité. [6]

### 2.6.4 Les intrusions :

Une intrusion est définie comme une faute malveillante d'origine interne ou externe résultant d'une attaque qui a réussi à exploiter une vulnérabilité. Elle est susceptible de produire des erreurs pouvant provoquer une défaillance vis-à-vis la sécurité, c'est-à-dire une violation de la politique de sécurité du système. Le terme d'intrusions sera employé dans le cas où l'attaque est menée avec succès et où l'attaquant a réussi à s'introduire et/ou compromettre le système. [7]

### 2.6.5 Les attaques :

Les motivations des attaques peuvent être liées à divers objectifs : [8]

- Obtenir un accès au système.
- Voler des informations, tels que des secrets industriels ou des propriétés intellectuelles.
- Collectionner des informations personnelles sur un utilisateur
- S'informer sur l'organisation.
- Récupérer des données bancaires.
- S'informer sur l'organisation (entreprise de l'utilisateur, etc.).
- Troubler le bon fonctionnement d'un service.
- Utiliser le système de l'utilisateur comme « rebond » pour une attaque.
- Utiliser les ressources du système de l'utilisateur, notamment lorsque le réseau sur lequel il est situé possède une bande passante élevée.
- Faire du chantage.
- Par simple jeu ou par défi.
- Pour terrorisme ou pour des fins politique.
- Pour apprendre.

## 2.7 Les attaques informatiques :

### 2.7.1 Définition :

Une attaque est définie comme faute d'interaction malveillante visant à violer une ou plusieurs propriétés de sécurité.

C'est une faute externe créée avec l'intention de nuire, y compris les attaques lancées par des outils automatiques : vers, virus, etc. La notion d'attaque ne doit pas être confondue avec la notion d'intrusions. [7]

### 2.7.2 Anatomie d'une attaque :

Fréquemment appelés « les 5 P » dans la littérature, ces cinq verbes anglophones constituent le squelette de toute attaque informatique : Probe, Penetrate, Persist, Propagate, Paralyze.

- Probe (Analyser) : Dans un premier temps, une personne mal intentionnée va chercher les failles pour pénétrer le réseau.
- Penetrate (Pénétrer) : Une fois une ou plusieurs failles identifiées, le pirate va chercher à les exploiter afin de pénétrer au sien du SI.
- Persiste (Persister) : une fois le réseau infiltré, le pirate cherchera à y revenir facilement. Pour cela, il installera par exemple des back doors. Cependant, en général, il corrigera la faille par laquelle il s'est introduit afin de s'assurer qu'aucun autre pirate n'exploitera sa cible
- Propagate (Propager) : Le réseau est infiltré, l'accès est facile. Le pirate pourra alors explorer le réseau et trouver de nouvelles cibles qui l'intéresseraient.
- Paralyse (Paralyser) : Les cibles identifiées, le pirate va agir et nuire au sein du SI. [9]

### 2.7.3 Les types d'attaque informatique :

Les systèmes informatiques utilisent différents composants, allant de l'électricité aux machines en fonctionnement, en passant par les programmes exécutés sur le système d'exploitation et utilisant le réseau. Des attaques peuvent se produire dans chaque lien vers cette chaîne, s'il existe une faille pouvant être exploitée [10].

Pour l'aspect technique, on définit que l'attaque c'est une exploitation d'une faille pour des fins illégales. Il existe cinq formes d'attaque que nous détaillons comme suit [11] :

- **L'attaque passive** : Les attaques passives sont toute action nous permettant d'analyser et de déchiffrer le trafic, de surveiller les communications et de capturer des informations d'authentification. L'attaquant utilise ce type d'attaque pour obtenir des informations ou de données facilement et à l'insu de la victime en interceptant les mots de passe, les numéros de carte de crédit et les emails.

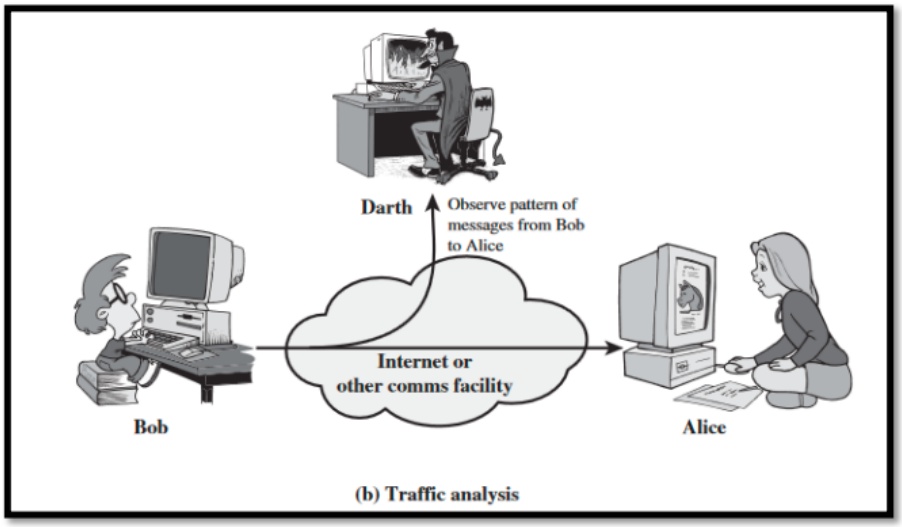


Figure 1:L'attaque passive (L'analyse du trafic réseau). [12]

- **L'attaque active :** Les attaques actives incluent les tentatives visant à contourner ou casser des

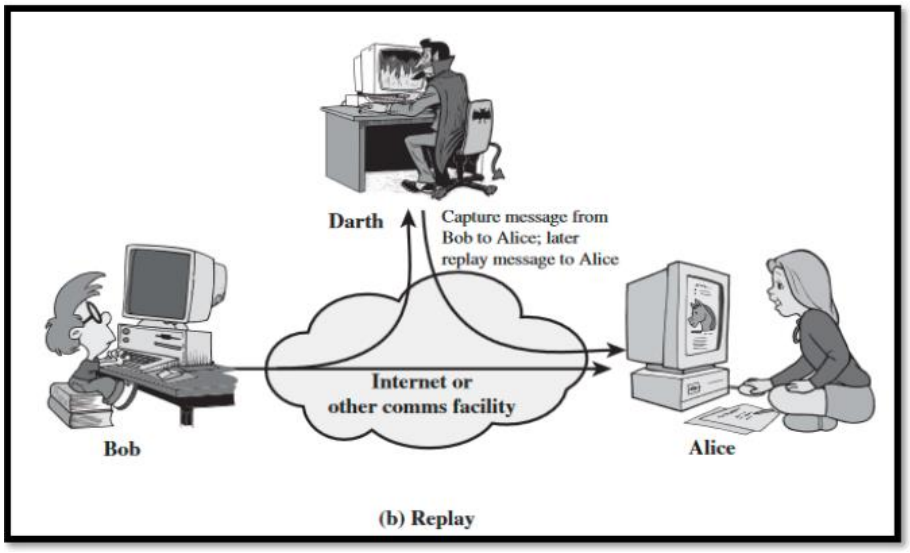


Figure 2:L'attaque active (rejeu). [12]

Fonctionnalités de sécurité afin de falsifier ou de voler des informations en insérant un code malveillant dans le système d'exploitation ou le réseau, ainsi que des menaces d'attaques actives visant à détecter ou à publier des fichiers de données, à refuser le service ou à modifier les données.

- **L'attaque externe :** L'attaquant doit utiliser la proximité physique pour pouvoir se connecter à des systèmes ou des réseaux via un accès secret ou ouvert afin de modifier, comparer et gérer l'accès aux informations.
- **L'attaque interne :** Dans ces cas d'attaques, l'attaquant peut faire partie de l'entreprise ou utiliser l'ingénierie sociale avec les personnes impliquées à la suite d'abus, de négligence ou de manque de connaissances. Dans les deux cas, ces attaques essayent d'espionner, de voler ou de détruire des informations, de les utiliser frauduleusement ou d'empêcher l'accès à d'autres utilisateurs autorisés.
- **L'attaque de distribution** : les attaques de distribution représentent toute modification malveillante du matériel ou du logiciel en usine ou lors de la distribution. Ces attaques consistent à introduire un code malveillant dans un produit comme un port dérobé pour obtenir un accès non autorisé à des informations ou une fonction système.

## 2.8 Les Moyens de la Sécurité Informatique :

### 2.8.1 Techniques de sécurité :

C'est l'ensemble de procédures ou dispositifs qui sont conçu pour détecter, prévenir ou récupérer les attaques qui menacent la sécurité informatique, il existe plusieurs outils de prévention contre-attaques informatiques, Nous avons cité ci-dessous quelques mécanismes : [13]

- **La protection physique** : avant de parler sur la sécurité des systèmes d'information premièrement il faut assurer la sécurité des matériels informatique et leurs emplacements.
- **Chiffrement** : Les algorithmes utilisent des clés pour convertir les données afin d'obtenir une sécurité robuste. Leur sécurité dépend du niveau de sécurité des clés.
- **Signature numérique** : Un mécanisme pour assurer l'intégrité des données et également pour authentifier l'auteur du document.
- **Bourrage de trafic** : Mécanisme assurant la confidentialité des données sur le volume de trafic en cas d'interception par des attaquants.
- **Contrôle d'accès** : Vérifier l'authentification des utilisateurs et leurs autorisations d'accéder aux données et vérifier leurs privilèges.
- **Antivirus** : Logiciel censé protéger l'ordinateur contre les logiciels (ou fichiers potentiellement exécutables) néfastes. Ne protège pas contre un intrus qui emploie un logiciel légitime, ou contre un utilisateur légitime qui accède à une ressource alors qu'il n'est pas autorisé à le faire.
- **Le pare-feu** : Un élément (logiciel ou matériel) du réseau informatique contrôlant les communications qui le travers. Il a pour fonction de faire respecter la politique de sécurité du réseau, celle-ci définissant quelles sont les communications autorisées ou interdites. N'empêche pas un attaquant d'utiliser une connexion autorisée pour attaquer

le système. Ne protège pas contre une attaque venant du réseau intérieur (qui ne le traverse pas).

- **Détection d'intrusion** : Identifiez une activité anormale ou suspecte sur le moniteur réseau. Ne pas détecter les accès incorrects mais autorisés par les utilisateurs légitimes. Le problème c'est comment minimiser les taux de faux positifs et de faux négatifs.
- **Journalisation ("logs")** : Enregistrement des activités de chaque acteur. Permet de constater que des attaques ont eu lieu, de les analyser et potentiellement de faire en sorte qu'elles ne se reproduisent pas.
- **Analyse des vulnérabilités ("Security audit")** : Identification des points de vulnérabilité du système. Ne détecte pas les attaques ayant déjà eu lieu, ou lorsqu'elles auront lieu. [13]

## 3 Conclusion :

La sécurité informatique est un domaine en constante évolution, crucial pour garantir la confidentialité, l'intégrité et la disponibilité de nos données et de nos systèmes. Dans ce chapitre, nous avons exploré les différents aspects de la sécurité informatique, des objectifs principaux et les types d'attaques en passant par les moyens de protection . En comprenant ces concepts et en mettant en œuvre des pratiques de sécurité efficaces, on protège mieux ses activités en ligne contre les menaces croissantes.

# CHAPITRE II :
# Les systèmes de détection d'intrusions (IDS).

# 1 Introduction :

Les cybermenaces sont en constante augmentation, la sécurisation des réseaux informatiques est devenue primordiale pour protéger les données sensibles et les infrastructures critiques. Les systèmes de détection d'intrusion (IDS) sont essentiels dans cette sécurisation. Ils surveillent les activités réseau et détectent les comportements anormaux pouvant indiquer une intrusion. En identifiant rapidement les menaces potentielles, les IDS permettent de réagir rapidement et de renforcer la défense contre les attaques malveillantes, assurant ainsi une protection efficace des environnements informatiques.

# 2 Généralité sur le système de détection d'intrusion :

## 2.1 Historique :

Les systèmes de détection ont été imposés en raison de la nécessité d'améliorer leur capacité à auditer et à surveiller la sécurité informatique. James Anderson [14] a été le premier à introduire le concept de systèmes de détection en 1980, mais le premier modèle a été créé par Denning Dorothy en 1987 et plusieurs prototypes ont été produits. Des budgets importants ont investi dans la recherche dans ce domaine jusqu'à ce jour. [16]

## 2.2 Définition :

La détection d'intrusions est un terme général qui désigne des méthodes automatiques qui, basées sur l'analyse de séquences d'événements temps réel et/ou enregistrés, peuvent alerter l'administrateur de sécurité de possibles violations de sécurité. [17]

La détection d'intrusions fait référence à la capacité d'un système informatique de déterminer automatiquement, à partir d'événements relevant de la sécurité, qu'une violation de sécurité se produit ou s'est produite dans le passé. [17]

Pour se faire, la détection d'intrusions nécessite qu'un grand nombre d'événements de sécurité soient collectés et enregistrés afin d'être analysés.

## 2.3 Les types des IDS :

Il existe plusieurs types d'IDS disponibles aujourd'hui, car ils sont joués un rôle important dans la capacité de survie du système d'information et préservent sa sécurité des attaques. Un IDS peut être classé comme :

### 2.3.1 Le IDS hybride :

Généralement utilisés dans un environnement décentralisé, ils permettent de réunir les informations de diverses sondes placées sur le réseau. Leur appellation « hybride » provient du fait qu'ils sont capables de réunir aussi bien des informations provenant d'un système HIDS « Host Intrusion Detection Systems » qu'un NIDS « Network Intrusion Detection Systems ».

### 2.3.2 Le IDS basé hôte (host- based IDS) :

HIDS est un système de détection d'intrusion spécifique à un ordinateur unique qui surveille la sécurité de ce système ou de cet ordinateur contre les attaques internes et externes. Les attaques internes font référence au cas où il détecte quel programme a accès à quelle ressource et qu'il y a une faille de sécurité. Dans la deuxième partie, il s'agit d'attaques externes, HIDS analyse les paquets en provenance et à destination de ce système (ordinateur) sur ses interfaces. HIDS répond en enregistrant l'activité et en informant l'autorité désignée. [17]

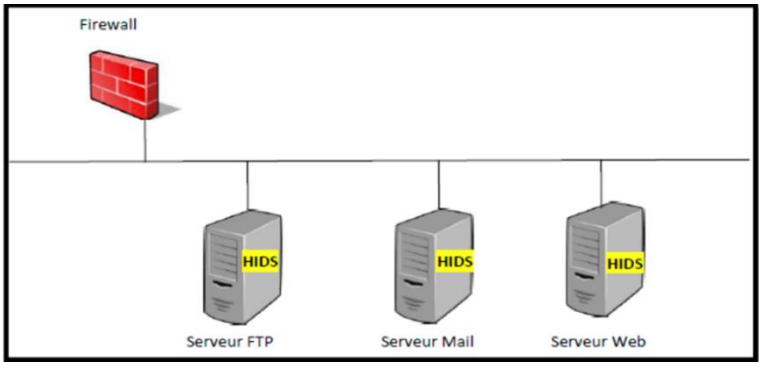


Figure 3:Exemple d'un HIDS(L'IDS-N iveau systéme).[18]

### 2.3.3 Le IDS basé réseau (Network- based IDS) :

Le système de détection d'intrusion basée réseau (NIDS) surveille le trafic réseau et analyse les paquets en transit pour détecter les attaques. Lors de l'identification d'une attaque ou lorsqu'un comportement anormal est détecté, une alerte peut être envoyée àl'administrateur [19]

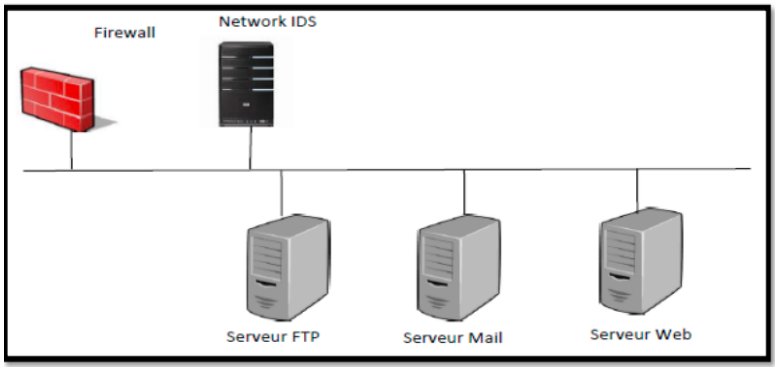


Figure 4: Exemple d'un IDS dans un réseau (NIDS) [19]

## 2.4 Emplacement d'un système de détection d'intrusions :

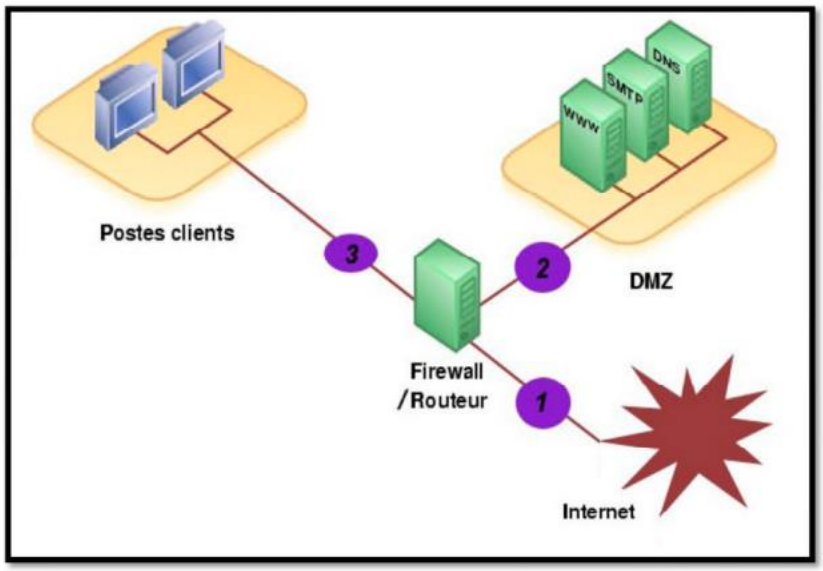


Figure 5: Endroits typique pour un IDS.

Il existe plusieurs endroits stratégiques où il convient de placer un IDS. Le schéma suivant illustre un réseau local ainsi que les positions que peut y prendre un IDS :

• **Position (1)** : Sur cette position, le IDS va pouvoir détecter l'ensemble des attaques frontales, provenant de l'extérieur, en amont du firewall. Ainsi, beaucoup d'alertes seront remontées ce qui rendra les logs difficilement consultables.

• **Position (2)** : Si le IDS est placé sur la DMZ(demilitarized zone), il détectera les attaques qui n'ont pas été filtrées par le firewall et qui relèvent d'un certain niveau de compétence. Les logs seront ici plus clairs à consulter puisque les attaques bénignes ne seront pas recensées.

• **Position (3)** : Le IDS peut ici rendre compte des attaques internes, provenant du réseau local de l'entreprise. Il peut être judicieux d'en placer un à cet endroit étant donné le fait que 80% des attaques proviennent de l'intérieur. De plus, si des trojans ont contaminé le parc informatique (navigation peu méfiante sur internet) ils pourront être ici facilement identifiés pour être ensuite éradiqués.

## 2.5 Classification des systèmes de détection d'intrusions :

Nous pouvons classifier les systèmes de détection d'intrusions selon cinq critères : [20]

- La source des données à analyser.
- Le lieu de l'analyse des données.
- La fréquence de l'analyse.
- Le comportement en cas d'attaque détectée
- La méthode de détection utilisée.

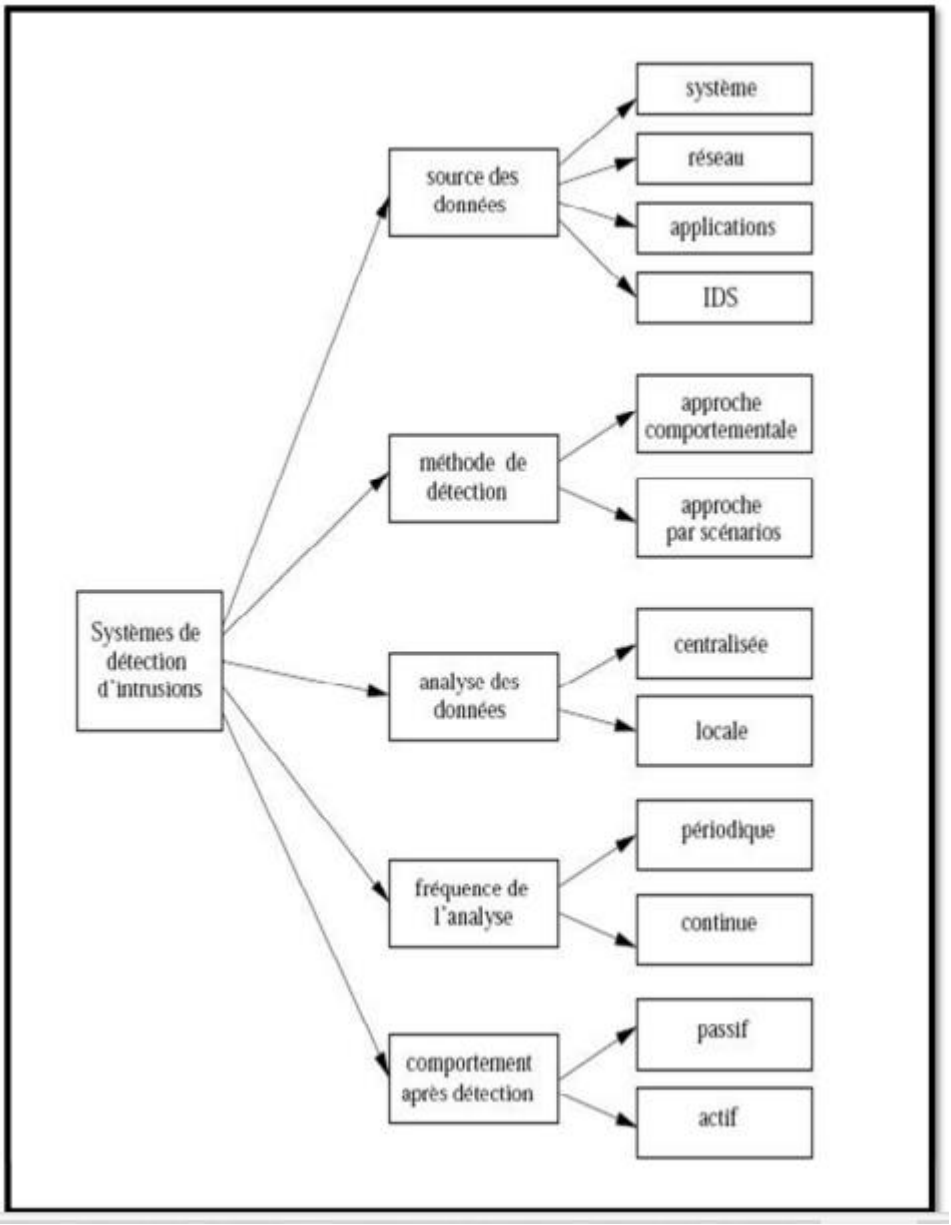


Figure 6: Classification des sustémes de détection d'intrusion .[26]

### 2.5.1 Source des données à analyser :

Les sources possibles de données à analyser sont une caractéristique essentielle des systèmes de détection d'intrusions puisque ces données constituent la matière première du processus de détection. Les données proviennent soit de logs générés par le système d'exploitation, soit de logs applicatifs, soit d'informations provenant du réseau, soit encore d'alertes générées par d'autres IDS. [20]

### 2.5.2 Localisation de l'analyse des données :

On peut également faire une distinction entre les IDS en se basant sur la localisation réelle de l'analyse des données : [20]

- **Analyse centralisée :** certains IDS ont une architecture multi-capteurs (ou multisondes). Ils centralisent les événements (ou alertes) pour analyse au sein d'une seule machine. L'intérêt principal de cette architecture est de faciliter la corrélation entre événements puisqu'on dispose alors d'une vision globale. Par contre, la charge des calculs (effectués sur le système central) ainsi que la charge réseau (due à la collecte des événements ou des alertes) peuvent être lourdes et risquent de constituer un goulet d'étranglement.
- **Analyse locale :** si l'analyse du flot d'événements est effectuée au plus près de la source de données (généralement en local sur chaque machine disposant d'un capteur), on minimise le trafic réseau et chaque analyseur séparé dispose de la même puissance de calcul. En contrepartie, il est impossible de croiser des événements qui sont traités séparément et l'on risque de passer à côté de certaines attaques distribuées.

### 2.5.3 Fréquence de l'analyse :

Une autre caractéristique des systèmes de détection d'intrusions est leur fréquence d'utilisation : [19]

- **Périodique** : certains systèmes de détection d'intrusions analysent périodiquement les fichiers d'audit à la recherche d'une éventuelle intrusion ou anomalie passée. Cela peut être suffisant dans des contextes peu sensibles (on fera alors une analyse journalière, par exemple).
- **Continue** : la plupart des systèmes de détection d'intrusions récents effectue leur analyse des fichiers d'audit ou des paquets réseau de manière continue afin de proposer une détection en quasi temps-réel. Cela est nécessaire dans des contextes sensibles (confidentialité) et/ou commerciaux (confidentialité, disponibilité). C'est toutefois un

processus coûteux en temps de calcul car il faut analyser à la volée tout ce qui se passe sur le système.

### 2.5.4 Comportement après détection :

Une autre façon de classer les systèmes de détection d'intrusions consiste à les classer par type de réaction lorsqu'une attaque est détectée : [20]

- **Passive :** la plupart des systèmes de détection d'intrusions n'apportent qu'une réponse passive à l'intrusion. 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. C'est alors lui qui devra prendre les mesures qui s'imposent
- **Active :** d'autres 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. Des outils tels que RealSecure ou NetProwler proposent ce type de

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

### 2.5.5 Méthode de détection :

Deux approches de détection ont été proposées : [20]

- L'approche comportementale : cette approche se base sur l'hypothèse selon laquelle nous pouvons définir un comportement normal de l'utilisateur et que toute déviation par rapport à celui-ci est potentiellement suspecte.

- L'approche par signature : elle s'appuie sur un modèle constitué des sections interdites dans le système d'informatique, ce modèle s'appuie sur la connaissance des techniques employées par les attaquants : on tire des scénarios d'attaque et on recherche dans les traces d'audit leur éventuelle survenue.

### 2.5.5.1 L'approche comportementale :

Les détecteurs d'intrusions comportementaux reposent sur la création d'un modèle de référence représentant le comportement de l'entité surveillée en situation de fonctionnement normal. Ce modèle est ensuite utilisé durant la phase de détection afin de pouvoir mettre en évidence d'éventuelles déviations comportementales. Pour cela, le comportement de l'entité surveillée est comparé à son modèle de référence. Une alerte est levée lorsqu'une déviation trop importante (notion de seuil) vis-à-vis de ce modèle de comportement normal est détectée.

Le principe de cette approche est de considérer tout comportement n'appartenant pas au modèle de comportement normal comme une anomalie symptomatique d'une intrusion ou d'une tentative d'intrusion. [21] On peut distinguer deux catégories de profils :

### A. Profils construits par apprentissage :

Parmi les méthodes proposées pour construire les profils par apprentissage, les plus marquantes sont les suivantes : [20]

- **Méthode statistique :** le profil est calculé à partir de variables considérées comme aléatoires et échantillonnées à intervalles réguliers. Ces variables peuvent être le temps processeur utilisé, la durée et l'heure des connexions, etc. Un modèle statistique est alors utilisé pour construire la distribution de chaque variable et pour mesurer, au travers d'une grandeur synthétique, le taux de déviation entre un comportement courant et le comportement passé.
- **Système expert :** ici, c'est une base de règles qui décrit statistiquement le profil de l'utilisateur au vu de ses précédentes activités. Son comportement courant est comparé aux règles, à la recherche d'une anomalie. La base de règles est rafraîchie régulièrement.

- **Réseaux de neurones :** la technique consiste à apprendre à un réseau de neurones le comportement de l'entité à surveiller. Par la suite, lorsqu'on lui fournira en entrée les actions courantes effectuées par l'entité, il devra décider de leur normalité.
- **Analyse de signatures :** Il s'agit de construire un modèle de comportement normal des services réseaux. Le modèle consiste en un ensemble de courtes séquences d'appels système représentatifs de l'exécution normale du service considéré. Des séquences d'appels étrangères à cet ensemble sont alors considérées comme l'exploitation potentielle d'une faille du service. Pour toutes ces méthodes, le comportement de référence utilisé pour l'apprentissage étant rarement exhaustif, on s'expose à des risques de fausses alarmes (faux positifs). De plus, si des attaques ont été commises durant cette phase, elles seront considérées comme normales (risque de faux négatifs).

**B. Profils spécifiant une politique de sécurité (policy-based) :**

Pour les IDS dits policy-based, il n'y a pas de phase d'apprentissage. Leur comportement de référence est spécifié par une politique de sécurité : la détection d'une intrusion intervient chaque fois que la politique est violée. Le profil est ici une politique de sécurité qui décrit la suite des appels systèmes licites d'une application. [20]

L'approche comportementale possède un certain nombre d'avantages et d'inconvénients :

- **Les avantages :**

➢ L'analyse comportementale n'exige pas des connaissances préalables sur les attaques.

➢ Elle permet la détection de la mauvaise utilisation des privilèges.

➢ Elle permet de produire des informations qui peuvent être employées pour définir des signatures pour l'analyse basée connaissance.

- **Les inconvénients :**

➢ Les approches comportementales produisent un taux élevé des alarmes type faux positif en raison des comportements imprévisibles des utilisateurs et des réseaux.

- Ces approches nécessitent des phases d'apprentissage pour caractériser les profils de comportement normaux.
- Les alarmes génériques par cette approche ne sont pas significatives.

#### 2.5.5.2 L'approche par scénarios :

On construit des scénarios d'attaque en spécifiant ce qui est caractéristique de l'attaque et qui doit être observé dans les traces d'audit. L'analyse des traces d'audit se fait à la recherche de ces scénarios. Les méthodes proposées à ce jour sont les suivantes : [20]

- **Système expert :** le système expert comporte une base de règles qui décrit les attaques. Les événements d'audit sont traduits en des faits qui sont interprétables par le système expert. Son moteur d'inférence décide alors si une attaque répertoriée s'est ou non produite.
- **Analyse de signatures :** il s'agit là de la méthode la plus en vue actuellement. Des signatures d'attaques sont fournies à des niveaux sémantiques divers selon les outils (de la suite d'appels système aux commandes passées par l'utilisateur en passant par les paquets réseau). Divers algorithmes sont utilisés pour localiser ces signatures connues dans les traces d'audit. Ces signatures sont toujours exprimées sous une forme proche des traces d'audit. Si l'on prend l'exemple des NIDS, les algorithmes de recherche de motifs utilisés permettent d'obtenir de bonnes performances en vitesse de traitement mais génèrent de nombreuses fausses alertes.
- **Automates à états finis :** plusieurs IDS utilisent des automates à états finis pour coder le scénario de reconnaissance de l'attaque. Cela permet d'exprimer des signatures complexes et comportant plusieurs étapes. On passe d'un état initial sûr à un état final attaqué via des états intermédiaires. Chaque transition entre états est déclenchée par des conditions sur les événements remontés par les capteurs.

L'approche par scénario possède un certain nombre d'avantages et d'inconvénients :

- **Les avantages :**

➢ L'analyse basée connaissance est très efficace pour la détection d'attaque avec un taux très bas des alarmes de type faux positif.

➢ Les alarmes générées sont significatives.

- **Les inconvénients :**

➢ Cette analyse basée connaissance permet seulement la détection des attaques qui sont connues au préalable. Donc, la base de connaissances doit être constamment mise à jour avec les signatures des nouvelles attaques.

➢ Le risque que l'attaquant peut influencer sur la détection après la reconnaissance des signatures.

## 2.6 Les principales tâches d'un IDS :

Un IDS permet de repérer des anomalies dans le trafic réseau comme suit :

- Détecter les tentatives de découvertes du réseau.
- Détecter dans certains cas, si l'attaque a réussi ou non.
- Détecter le Dénis de Service.
- Détecter le niveau d'infection du système informatique et les zones réseaux touchées.
- Repérer les machines infectées.
- Alerter de façon centrale pour toutes les attaques.
- Réagir aux attaques et corriger les problèmes éventuels.

Si on compare les HIDS et NIDS, les HIDS présente un avantage considérable par rapport à un NIDS dans le cas où le trafic est crypté. En effet, un NIDS n'a pas connaissance des clés de cryptage et ne peut appliquer ses algorithmes de détection au niveau des données chiffrées. La détection est effectuée à l'extrémité de la chaine de communication, une fois le flux est décrypté. Ceci est réalisé en mettant en œuvre un agent HIDS directement sur le serveur cible.

Les flux chiffrés sont ainsi décodés par la cible et transmis ensuite au monteur d'analyse du NIDS.

## 2.7 Les limites d'un IDS :

Parmi les faiblesses des IDS on trouve : [22]

- Configuration complexe et longue.
- Nombreux faux positifs après configuration.
- Pas de connaissance de la plate-forme.
- De ses vulnérabilités.
- Du contexte métier.
- Les attaques applicatives sont difficilement détectables.
- Injection SQL.
- Exploitation de CGI mal conçus.
- Des évènements difficilement détectables.
- Scans lents / distribués. – Canaux cachés / tunnels.
- Pollution des IDS.
- Consommation des ressources de le IDS.
- Perte de paquets.
- Déni de service contre le IDS.
- Une attaque réelle peut passer inaperçue.
- Attaque contre le SDI 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 desécurité, votre politique de sécurité ou votre architecture de sécurité.

## 3 Conclusion :

Dans ce chapitre, nous avons examiné en détail les systèmes de détection d'intrusion (IDS),des outils essentiels dans la défense des réseaux informatiques contre les menaces croissantes. Grâce à leur capacité à surveiller en temps réel les activités du réseau, à analyser les événements et à détecter les comportements suspects, les IDS permettent de répondre rapidement aux tentatives d'intrusion. Cependant, malgré leur efficacité, les IDS présentent également des limites et des défis, notamment en termes de configuration complexe et de gestion des faux positifs. Dans un paysage cybernétique en constante évolution, investir dans des IDS modernes et les intégrer de manière stratégique dans les infrastructures de sécurité globales est crucial pour garantir une protection optimale contre les cybermenaces.

# Chapitre III :
# Conception et Implémentation.

# 1 Introduction :

Au cours des dernières années, les technologies de l'information ont connu une expansion rapide, engendrant une prolifération des attaques informatiques qui, de surcroît, sont devenues plus nombreuses, variées et sophistiquées. Face à cette montée en puissance des attaques exploitant les failles des systèmes logiciels et matériels, plusieurs solutions ont été envisagées, notamment les systèmes de détection d'intrusion (IDS).

Ce chapitre est divisé en deux parties principales : conception et implémentation. Dans la partie conception, nous détaillons les principes théoriques qui sous-tendent notre approche. Nous expliquons comment des modèles géométriques peuvent être appliqués pour identifier les motifs caractéristiques des attaques DoS.

Dans la partie implémentation, nous décrivons en détail notre travail, y compris la représentation graphique des données, la méthode de classification, et les résultats obtenus. Cette section couvre également la méthodologie employée pour tester et évaluer l'efficacité de notre système de détection d'intrusion, mettant en évidence les performances en termes de taux de détection et de précision. En présentant nos résultats, nous discutons des défis rencontrés et des perspectives d'amélioration future.

# 2 Conception

## 2.1 Objectif :

Cette approche repose sur des formes géométriques simples et vise deux objectifs clés :

1. Identifier des modèles d'attaques par déni de service (DoS), en mettant particulièrement l'accent sur la capacité à les distinguer du trafic normal. Ces modèles sont ensuite utilisés dans un modèle CNN pour la classification afin de détecter efficacement les attaques DoS et les flux de trafic régulier. Des modèles spécifiques ont été élaborés pour discriminer entre les attaques DoS et le trafic normal.

2. Améliorer le taux de détection, ce qui constitue un défi majeur pour les systèmes de détection d'intrusion (IDS). Cette amélioration vise à renforcer la capacité du IDS à repérer les activités suspectes et les attaques potentielles, tout en minimisant les faux positifs et en optimisant les performances globales du système.

Dans notre mémoire, nous utilisons le travail préalablement réalisé par M. Cheikh, Salima Hacini, Zizette Boufaida dans leurs article intitulés " Classification of DOS Attacks Using Visualization Technique en 2014 ».[22] Ce travail repose sur la technique de détection basée sur une représentation visuelle du trafic réseau après normalisation de certains paramètres dans le KDD. Cette représentation vise à trouver des modèles visuels d'attaques DoS et à pouvoir les distinguer du trafic normal ,ces modèles sont ensuite intégrés dans des classifieurs, [22] .

Dans notre travail, nous avons étudié l'ajout d'autres paramètres appliqués à la base de test NSLKDD pour ensuite les intégrer dans un réseau de neurones convolutionnel (CNN) pour la phase de classification.

On peut donc considérer le problème de la détection d'intrusion comme un problème de reconnaissance de formes. Ainsi, la classification n'est pas faite sur la base de paramètres souvent complexes, mais plutôt sur la base de formes issues d'une transformation géométrique.

## 2.2 Acquisition de Données :

La première étape de notre processus consiste à acquérir les données nécessaires pour entraîner et tester notre système. Nous avons utilisé le dataset NSL-KDD, une version améliorée du célèbre dataset KDD'99, qui est spécifiquement conçu pour les recherches en détection d'intrusions. Le dataset NSL-KDD présente plusieurs avantages, notamment l'absence d'entrées redondantes et une meilleure représentativité des instances, ce qui permet une évaluation plus juste des systèmes de détection.

Le dataset NSL-KDD contient 41 caractéristiques (features) décrivant divers aspects des connexions réseau, telles que la durée de la connexion, le nombre de paquets envoyés, les types

de service, et bien d'autres. Chaque enregistrement est étiqueté soit comme "normal" ou "attaque".

## 2.3 Classification :

### 2.3.1 Définition :

La classification est le processus de reconnaissance, de compréhension et de regroupement des idées et des objets en catégories prédéfinies ou en « sous-populations ». Utilisation d'ensembles de données de formation pré-catégorisées, de machines les programmes d'apprentissage utilisent une variété d'algorithmes pour classer les futurs ensembles de données en catégories.

Les algorithmes de classification dans l'apprentissage automatique utilisent les données d'entraînement d'entrée pour prédire la probabilité que les données ultérieures tombent dans l'une des catégories prédéterminées. L'une des plus utilisations courantes de la classification consistent à filtrer les emails en « spam » ou « non-spam ».

En bref, la classification est une forme de « reconnaissance de formes », avec des algorithmes de classification appliqué aux données d'entraînement pour trouver le même motif (mots, nombre séquences, etc.) En utilisant des algorithmes de classification pour classer le trafic réseau en deux classes : « normal » et « intrusion », Un processus en deux étapes :

- Training : description d'un ensemble de classes prédéterminées. Chaque échantillon est supposé appartenir à une classe prédéfinie, déterminée par l'attribut d'étiquette de classe. L'ensemble de tuples utilisé pour la construction du modèle est appelé "training set". Le modèle est représenté sous forme de règles de classification, d'arbres de décision ou de formules mathématiques.
- Classification : des objets et estimation de l'exactitude du modèle. La classe connue de l'échantillon de test est comparée au résultat classifié du modèle. Le taux de précision est le pourcentage d'échantillons de l'ensemble de test correctement classés par le modèle.

L'ensemble de test est indépendant de l'ensemble d'apprentissage pour éviter le surajustement.

# 3 Implémentation

## 3.1 Environnement de programmation :

Nous allons parler de l'implémentation en détaillant l'aspect matériel, l'environnement de développement, et les différents outils utilisés pour réaliser l'application.

- **Aspect matériel :**

Notre projet a été développé sur un pc :

• Type : Système d'exploitation 64 bits.

• Processeur : Intel(R) Core(TM) i5-6200U CPU @ 2.30GHz 2.40 GHz.

• RAM: 8,00 Go.

## 3.2 Environnement de développement :

Dans cette section, nous présentons les outils et les logiciels que nous avons utilisés : Pycharm IDE , Python 3.12.2 et Weka

### 3.2.1 Pycharm :

PyCharm est un environnement de développement intégré (IDE) pour Python, créé par JetBrains. Il offre des fonctionnalités avancées telles que l'édition de code intelligent, le débogage intégré, le support des tests unitaires, et l'intégration avec les frameworks web (comme Django et Flask).

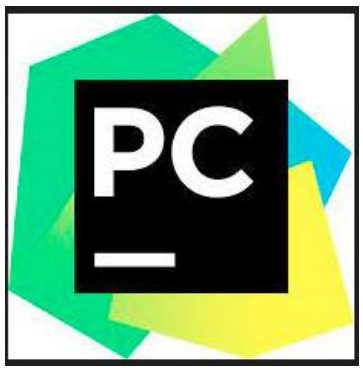

Figure 7: Le logo de PyCharm.[24]

PyCharm facilite également le travail avec les bases de données et les outils de contrôle de version. Il est extensible grâce à de nombreux plugins et adapté aux besoins des développeurs scientifiques avec le support de bibliothèques comme NumPy et Pandas .[24]

### 3.2.2 Python 3.12.2 :

une version spécifique du langage de programmation Python, publiée par la Python Software Foundation. Python est un langage de programmation interprété, orienté objet et de haut niveau avec une syntaxe claire et une lisibilité accrue, ce qui le rend facile à apprendre et à utiliser. Python 3.12.2 apporte des améliorations et des corrections de bugs par rapport aux versions précédentes. Les principales caractéristiques de Python incluent une gestion automatique de la mémoire, un support étendu pour l'intégration avec d'autres langages et outils, ainsi qu'une vaste bibliothèque standard et un écosystème riche de modules tiers.

### 3.2.3 Weka:

Weka est une collection d'algorithmes d'apprentissage automatique pour les tâches d'exploration de données. Il contient des outils pour la préparation des données, la classification, la régression, le clustering, l'exploration des règles d'association et la visualisation.[25]

Figure 8: Le logo de Weka .*[25]*

A. **Interface utilisateur graphique de WEKA** :

- Explorer : Un environnement pour explorer les données avec WEKA (la suite de cette documentation traitée de cette application plus en détail).
- Expérimentateur : un environnement pour effectuer des expériences et effectuer des tests entre les schémas d'apprentissage.
- Knowledge Flow : cet environnement prend essentiellement en charge les mêmes fonctions que Explorer mais avec une interface glisser-déposer. L'un des avantages est qu'il prend en charge apprentissage.
- Simple CLI : Fournit une interface de ligne de commande simple qui permet l'exécution directe de Commandes WEKA pour les systèmes d'exploitation qui ne fournissent pas leur propre ligne de commande interface.


Figure 9: Interface graphique Weka .

Nous ne décrirons que le premier composant "Explorer".

• Explorateur : Les fenêtres de l'Explorateur WEKA affichent différents onglets en commençant par le prétraitement.

• Initialement, l'onglet de prétraitement est actif, car l'ensemble de données est d'abord prétraité avantd'être appliqué algorithmes et exploré le jeu de données. Les onglets sont les suivants :

• Preprocess : Choisissez et modifiez les données chargées.

• Classifier : appliquer des algorithmes d'entraînement et de test aux données qui seront classifiées et régressées les données.

• Cluster : Formez des clusters à partir des données.

• Associer : extrayez la règle d'association pour les données.

• Sélectionner les attributs : les mesures de sélection des attributs sont appliquées.

• Visualiser : la représentation 2D des données est visible.

• Barre d'état : la section la plus basse de la fenêtre affiche la barre d'état. Cette section montre ce qui se passe actuellement sous la forme d'un message, tel qu'un fichier est en cours chargé. Faites un clic droit dessus, les informations sur la mémoire peuvent être vues, et également Run ramasse-miettes pour libérer de l'espace peut être exécuté.

• Bouton Log : il stocke un journal de toutes les actions dans Weka avec l'horodatage. Les journaux sont affichés dans une fenêtre séparée lorsque le bouton Log est cliqué.

• Icône d'oiseau WEKA : présente dans le coin inférieur droit montre l'oiseau WEKA avec représente le nombre de processus exécutés simultanément (par x.). Lorsque le processus est courir l'oiseau se déplacera.

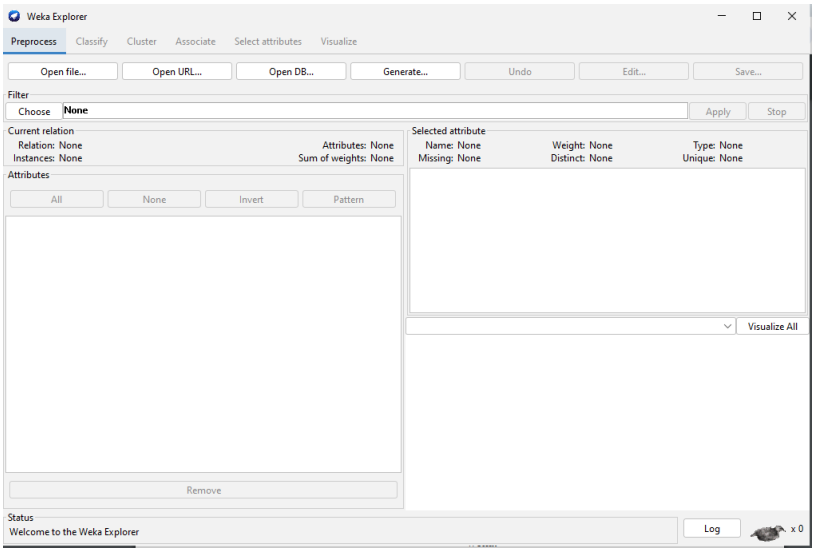


Figure 10: Interface de l'explorateur Weka.

## 3.3 Dataset NSL-KDD :

### 3.3.1 Définition :

Le NSL-KDD dataset est une version améliorée du KDD Cup 1999 dataset, largement utilisé pour la recherche en détection d'intrusion et cybersécurité. Il a été conçu pour corriger les problèmes du KDD'99, tels que la présence de nombreuses données redondantes, ce qui rendait l'évaluation des systèmes de détection d'intrusion biaisée. Le NSL-KDD élimine ces redondances, réduisant ainsi la taille des données et améliorant la qualité de l'ensemble de données. Il contient des enregistrements représentant des activités normales et diverses attaques classées en plusieurs catégories : DOS (Denial of Service), U2R (User to Root), R2L (Remote to Local), et Probe. Les enregistrements sont composés de 41 caractéristiques et une étiquette indiquant l'activité. Le dataset est utilisé pour l'évaluation et le développement de modèles de

machine learning pour la détection d'intrusions, et pour l'analyse des techniques de sécurité réseau.

Le NSL-KDD dataset est subdivisé en plusieurs ensembles pour faciliter l'entraînement et l'évaluation des modèles de détection d'intrusion. On cite de ces ensembles :

1. **NSL-KDDTrain** :

Ensemble d'entraînement complet, utilisé pour entraîner les modèles de détection d'intrusion. Il contient un large échantillon de données étiquetées couvrant à la fois les activités normales et les diverses attaques.

2. **NSL-KDDTest** :

Ensemble de test complet, utilisé pour évaluer les performances des modèles sur des données non vues pendant l'entraînement. Il permet de tester la capacité du modèle à détecter des intrusions et à généraliser sur de nouvelles données.

### 3.3.2 Types d'Attaques dans le Dataset :

Le NSL-KDD dataset classifie les attaques en plusieurs catégories, permettant une évaluation détaillée de la performance des systèmes de détection d'intrusion :

- **DOS (Denial of Service)** : Attaques visant à rendre une machine ou un réseau indisponible pour ses utilisateurs.
- **U2R (User to Root)** : Attaques où l'attaquant obtient des privilèges d'administrateur à partir d'un compte utilisateur normal.
- **R2L (Remote to Local)** : Attaques où l'attaquant obtient un accès local à partir d'un accès distant.
- **Probe** : Attaques de surveillance et d'exploration du réseau.

### 3.3.3 Structure des Enregistrements :

Chaque enregistrement dans ces fichiers est composé de 41 caractéristiques (features) décrivant les connexions réseau, telles que la durée, le protocole, le nombre de paquets, etc., et d'une étiquette (label) indiquant si l'enregistrement représente une activité normale ou un type d'attaque spécifique.

| Type | Number of Records | |
|---|---|---|
| | **Traning dataset** | **Test dataset** |
| Normal | 67343 | 9711 |
| Denial of Service | 45927 | 7456 |
| Probe | 11656 | 2421 |
| Remote to Local | 955 | 2756 |
| User to Root | 52 | 200 |
| **Total** | 125973 | 22544 |

Tableau 1 : Instances de répartition NSL-KDD.

### 3.3.4 Explication des caractéristiques du NSL-KDD Dataset :

Le NSL-KDD dataset comprend 41 caractéristiques distinctes qui décrivent chaque connexion réseau. Ces caractéristiques sont divisées en trois catégories principales :

1. **Caractéristiques Intrinsèques :**

- Ces caractéristiques sont liées aux propriétés individuelles de chaque connexion, telles que :

- **Duration :** La durée de la connexion.
- **Protocol_type** : Le type de protocole utilisé (TCP, UDP, etc.).
- **Service :** Le type de service réseau (HTTP, FTP, SMTP, etc.).
- **Flag** : L'état de la connexion.

- **Src_bytes et Dst_bytes** : Le nombre de bytes envoyés depuis/sur la connexion.

2. **Caractéristiques Contenant des Informations sur les Erreurs :**

   - Ces caractéristiques donnent des indications sur les erreurs et les échecs potentiels de la connexion :
   - **Wrong_fragment :** Le nombre de fragments erronés.
   - **Urgent :** Le nombre de paquets urgents.

3. **Caractéristiques Dérivées de l'Historique :**

   - Ces caractéristiques sont calculées sur la base des données historiques pour évaluer le comportement des connexions :
   - **Count :** Le nombre de connexions vers la même destination au cours des deux dernières secondes.
   - **Srv_count :** Le nombre de connexions vers le même service au cours des deux dernières secondes.
   - **Dst_host_count et Dst_host_srv_count :** Le nombre de connexions vers le même hôte/destinataire et vers le même service.

Exemples de Caractéristiques :

- **Same_srv_rate** : La proportion de connexions ayant le même service.
- **Dst_host_same_src_port_rate** : La proportion de connexions au même hôte utilisant le même port source.
- **Logged_in** : Indicateur de connexion réussie (1 si réussi, 0 sinon).

➢ **Utilité des Caractéristiques** : Ces caractéristiques sont cruciales pour identifier des motifs d'activité réseau normale et anormale. En utilisant ces 41 caractéristiques, les

algorithmes de machine learning peuvent apprendre à distinguer entre letrafic normal et les intrusions potentiellement malveillantes.

| Number | Data features | Number | Data features | Number | Data features | Number | Data features |
|---|---|---|---|---|---|---|---|
| 1 | Duration | 12 | Logged_in | 23 | Count | 34 | Dst_host_same_srv_rate |
| 2 | Protocol_type | 13 | Num_compromised | 24 | Srv_count | 35 | Dst_host_diff_srv_rate |
| 3 | Service | 14 | Root_shell | 25 | Serror_rate | 36 | Dst_host_same_src_port_rate |
| 4 | Flag | 15 | Su_attempted | 26 | Srv_serror_rate | 37 | Dst_host_srv_diff_host_rate |
| 5 | Src_bytes | 16 | Num_root | 27 | Rerror_rate | 38 | Dst_host_serror_rate |
| 6 | Dst_bytes | 17 | Num_file_creations | 28 | Srv_rerror_rate | 39 | Dst_host_srv_serror_rate |
| 7 | Land | 18 | Num_shells | 29 | Same_srv_rate | 40 | Dst_host_rerror_rate |
| 8 | Wrong_fragment | 19 | Num_access_files | 30 | Diff_srv_rate | 41 | Dst_host_srv_rerror_rate |
| 9 | Urgent | 20 | Num_outbound_cmds | 31 | Srv_diff_host_rate | | |
| 10 | Hot | 21 | Is_host_login | 32 | Dst_host_count | | |
| 11 | Num_failed_logins | 22 | Is_guest_login | 33 | Dst_host_srv_count | | |

Figure 11: Les 41 paramétres du NSL-KDD .[26]

### 3.3.5 Convertir les données arrf en csv :

- Ouvrez Weka. Si vous travaillez dans Weka, vous disposez d'un outil intégré qui convertira vos fichiers. CSV au format. ARFF. Vous trouverez généralement Weka dans le dossier Applications.
- Cliquez sur le menu « tools ». Il se trouve dans la barre de menus en haut de la fenêtre Weka
- Cliquez sur « ArffViewer ». Cela ouvre une fenêtre vide appelée ARFF-Viewer.
- Cliquez sur le menu « File ». Il se trouve en haut de la fenêtre ARFF-Viewer.
- Cliquez sur « Open ». Une fenêtre de navigateur de fichiers apparaîtra.
- Accédez au dossier qui contient le fichier. CSV.
- Sélectionnez « CSV data files (*.csv) » dans le menu Fichiers de type. Vous devriez maintenant voir le fichier. CSV que vous devez convertir dans la fenêtre.
- Sélectionnez-le. CSV et cliquez sur « Open ». Cela ouvre le fichier dans le visualiseur.
- Cliquez sur le menu « file ».
- Cliquez sur « Save As ».

- Nommez le fichier. Le nom du fichier doit se terminer par. ARFF (p. ex., mydata.ARFF).
- Cliquez sur « save ». Le fichier. CSV est maintenant converti au format. SLIA.

### 3.3.6 Prétraitement des Données :

Le prétraitement des données est une étape critique pour s'assurer que les données utilisées sont propres, cohérentes et prêtes pour l'analyse. Les étapes de prétraitement incluent :

- **Normalisation** : Pour éviter que certaines caractéristiques ne dominent les autres en raison de leurs différentes échelles, nous avons normalisé les valeurs des caractéristiques. La normalisation a été réalisée en échelonnant les valeurs entre 0 et 1, en utilisant la méthode de min-max scaling.
- **Sélection des Caractéristiques** : Parmi les 41 caractéristiques disponibles, nous avons sélectionné les plus pertinentes pour la détection des attaques DoS. Cette sélection s'est basée sur une analyse de corrélation et des tests préliminaires, visant à identifier les caractéristiques les plus discriminantes.

### 3.3.7 Le choix des paramètres de visualisation :

Nous nous concentrons dans ce travail sur l'application de notre technique aux attaques DoS. Le choix des paramètres basés sur NSL-KDD découle du travail réalisé par M. Cheikh, Salima Hacini, Zizette Boufaida dans leur article intitulé 'Classification of DOS Attacks Using Visualization Technique en 2014' [22], auxquels nous avons ajouté d'autres paramètres pertinents. Pour cela, les 41 paramètres NSL-KDD sont pris et une représentation des paramètres sous forme géométrique est appliquée. Les paramètres utilisés sont ceux qui donnent une meilleure discrimination des formes (voir Tableau 2)

| | Paramètres | Description |
|---|---|---|
| 1 | Pr (1) | Duration |
| 2 | Pr (23) | Count |
| 3 | Pr (24) | Srv count |
| 4 | Pr (13) | Compromised |
| 5 | Pr (25) | Serror rate |
| 6 | Pr (26) | Srv serror rate |
| 7 | Pr (29) | Same serror rate |
| 8 | Pr (34) | Dst host same srv rate |
| 9 | Pr (38) | Dst host serror rate |
| 10 | Pr (39) | Dst host srv serror rate |
| 11 | Pr(6) | Dst bytes |

Tableau 2: Les paramètres de détection

- Dans notre méthode de détection, nous avons intégré l'ensemble complet des instances normales, l'ensemble complet des instances de DoS, et l'ensemble complet des instances utilisées pour le test.

| TYPE | | Number of Records | |
|---|---|---|---|
| | | **Training dataset** | **Test dataset** |
| Normal | | **67343** | **9711** |
| Denial of Service | neptune | 41214 | 4657 |
| | smurf | 2646 | 665 |
| | back | 956 | 359 |
| | land | 18 | 7 |
| | teardrop | 892 | 12 |
| | pod | 201 | 41 |
| | **Total** | **45927** | **5741** |

*Tableau 3: Nombre total d'instances integrées .*

## 3.4 Transformation Géométrique et Visualisation :

Une des innovations de cette approche est la transformation des données en représentations géométriques, facilitant la visualisation des modèles de trafic réseau.

- **Transformation Géométrique** : Les valeurs des caractéristiques sélectionnées sont transformées en formes géométriques simples en utilisant un système de coordonnées polaires. Par exemple, chaque caractéristique peut être représentée par un angle spécifique et sa valeur par un rayon. Cette transformation permet de convertir des enregistrements de données en polygones.[22]

- **Visualisation** : Les polygones générés offrent une manière intuitive de visualiser les données réseau. Les formes et tailles distinctes des polygones permettent de distinguer visuellement les types de trafic normal et les différentes attaques DoS. Les visualisations sont générées en temps réel, facilitant la détection rapide des anomalies.[22]

### 3.4.1 La représentation graphique :

Dans notre projet, nous avons développé un programme Python pour effectuer une transformation des valeurs des paramètres en utilisant une représentation graphique similaire à un graphique radar. Chaque valeur de paramètre est représentée par des descripteurs di (i = 1, ... 10) correspondant aux dix paramètres de détection.

Le résultat de cette transformation est similaire à un graphique radar. La figure 1 montre la forme géométrique d'un paquet normal.

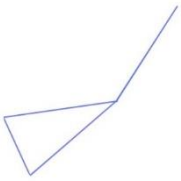

Figure 12: Représentation d'un paquet normal.

Pratiquement, si nous prenons tous les paquets normaux, le graphique garde le même rythme avec quelques changements insignifiants (la forme reste invariante dans l'espace).

## 3.5 Classification du Trafic Réseau :

Pour la classification des images de trafic réseau en anomalies ou en trafic normal, nous avons opté pour l'utilisation d'un modèle de réseau de neurones convolutionnels (CNN), implémenté en Python. Cette approche nous a permis de bénéficier d'une méthode robuste et efficace pour analyser et classifier le trafic réseau. En intégrant à notre ensemble de données à la fois des exemples de trafic normal et des exemples d'anomalies, nous avons enrichi

l'apprentissage du modèle, favorisant ainsi sa capacité à généraliser et à détecter diverses formes d'irrégularités dans le trafic réseau.

### 3.5.1 Réseaux de Neurones Convolutionnels (CNN) :

#### 3.5.1.1 Définition :

L'appellation convolutional neural network signifie « réseau neuronal convolutif » en Français. L'abréviation est CNN. Il s'agit d'une structure particulière d'un réseau de neurones artificiels spécialement conçu pour l'apprentissage automatique et le traitement d'images ou de données audio. Les CNN font partie de la famille des techniques d'apprentissage profond (deep learning) qui utilisent des réseaux de neurones profonds pour extraire des caractéristiques complexes et des motifs à partir des données brutes .[27]

Dans une certaine mesure, son fonctionnement est calqué sur les processus biologiques derrières les réflexions du cerveau humain. La structure est similaire à celle du cortex visuel d'un cerveau. Le convolutional neural network se compose de plusieurs couches. La formation d'un réseau de neurones convolutifs se déroule généralement de manière supervisée. L'un des fondateurs du réseau de neurones convolutifs est Yann Le Cun.[26]

#### 3.5.1.2 Mise en place d'un CNN :

Des neurones selon une structure entièrement ou partiellement maillés à plusieurs niveaux composent les réseaux de neurones conventionnels. Ces structures atteignent leurs limites lors du traitement d'images, car il faudrait disposer d'un nombre d'entrées correspondant au nombre de pixels. Le nombre de couches et les connexions entre elles seraient énormes et ne seraient gérables que par des ordinateurs très puissants. Différentes couches composent un réseau neuronal convolutif. Son principe de base est un réseau neuronal à propagation avant ou feedforward neural network partiellement maillé.

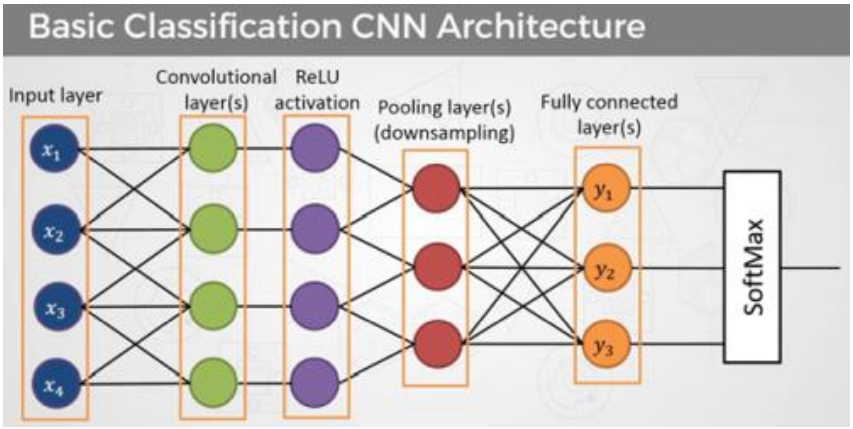


Figure 13 : Architecture de CNN .[27]

Les couches individuelles de CNN sont :

- **Convolutional layers** ou couches de convolution (CONV)
- **Pooling layers** ou couches de Pooling (POOL)
- **ReLU layers** ou couches d'activation ReLU (Rectified Linear Units)
- **Fully Connected layers** ou couches Fully Connected (FC)

  La couche de Pooling suit la couche de convolution et cette combinaison peut être présente plusieurs fois l'une derrière l'autre. La couche de Pooling et la couche de convolution étant des sous-réseaux maillés localement, le nombre de connexions dans ces couches reste limité et dans un cadre gérable, même avec de grandes quantités d'entrées. Une couche Fully Connected forme la fin de la structure.[27]

### 3.5.1.3 Motivations pour l'Utilisation de CNN :

Le choix des CNN comme classificateurs repose sur plusieurs avantages clés :

- **Capacité à Capturer des Motifs Complexes** : Les CNN sont capables de capturer des motifs complexes et des dépendances locales dans les données, ce qui est crucial pour identifier les caractéristiques subtiles des attaques DoS.

- **Efficacité pour les Données Visuelles** : En transformant les données de trafic en images, les CNN peuvent exploiter pleinement leur capacité à analyser des données visuelles et à extraire des caractéristiques discriminantes.
- **Robustesse et Flexibilité** : Les CNN sont robustes et peuvent être adaptés à diverses configurations et tailles de données, ce qui les rend flexibles pour différents types d'attaques et de scénarios de trafic réseau.

#### 3.5.1.4 Entraînement du Modèle CNN :

Le processus d'entraînement du modèle CNN implique plusieurs étapes clés :

- **Architecture du CNN** : Nous avons conçu une architecture CNN composée de plusieurs couches de convolution, suivies de couches de pooling et de couches entièrement connectées. Les couches de convolution permettent au modèle d'extraire des motifs locaux, tandis que les couches de pooling réduisent la dimensionnalité, rendant l'apprentissage plus efficace.
- **Fonction de Perte et Optimisation** : La fonction de perte d'entropie croisée a été utilisée pour évaluer la performance du modèle pendant l'entraînement. Cette fonction mesure la différence entre les prédictions du modèle et les étiquettes réelles des données. Pour optimiser le modèle, nous avons utilisé l'algorithme Adam, connu pour sa rapidité et son efficacité.
- **Entraînement** : Les données ont été utilisées pour l'entraînement. L'ensemble d'entraînement est utilisé pour ajuster les poids du modèle lors du processus d'apprentissage. Des techniques de régularisation telles que le dropout ont été appliquées pour améliorer la généralisation du modèle et éviter le surapprentissage.
- **Évaluation de la performance** : Pendant le processus d'entraînement, nous évaluons régulièrement la performance du modèle en utilisant des métriques telles que l'accuracy (précision) et la loss (perte). L'accuracy mesure la proportion d'images correctement

classées par rapport au nombre total d'images dans l'ensemble d'entraînement. Une accuracy élevée indique une meilleure performance du modèle. La loss, quant à elle, évalue à quel point les prédictions du modèle correspondent aux étiquettes réelles.

En ajustant les poids du modèle à chaque itération d'entraînement à l'aide de l'algorithme Adam, nous cherchons à minimiser la loss et à améliorer l'accuracy du modèle. Ces métriques nous permettent de surveiller et d'optimiser les performances du modèle pendant le processus d'entraînement, garantissant ainsi sa capacité à généraliser correctement aux données de test non vues.

#### 3.5.1.5 Phase de Classification avec le Modèle CNN :

Après avoir entraîné le modèle CNN, l'étape suivante implique son utilisation pour effectuer des prédictions sur un ensemble d'images de test. Chaque prédiction est interprétée pour déterminer si l'image est considérée comme "normale" ou "anormale".

En évaluant la performance du modèle sur ces données de test, nous calculons des métriques telles que le pourcentage de prédictions correctes. Cette évaluation nous permet d'apprécier la précision du modèle dans sa capacité à classifier les images correctement.

## 3.6 Les mesures de performances :

Nous commencerons par exposer quelques mesures permettant d'évaluer les performances des IDS. L'efficacité d'une technique de détection d'intrusion est évaluée en fonction de sa capacité à effectuer des détections correctes. Selon la concordance entre la réalité d'un événement donné et la prédiction de la technique utilisée, il existe quatre possibilités (voir Tableau 3) [28].

| | | Prédiction de la valeur | |
|---|---|---|---|
| | Type | Attaque | Normal |
| Valeur actuelle | Attaque | TP | FN |
| | Normal | FP | TN |

Tableau 4: La matrice de confusion.

Il existe quatre mesures pour évaluer les performances des IDS. L'évaluation de l'efficacité d'une technique de détection d'intrusion repose sur sa capacité à effectuer des détections correctes

[28] :

- Les vrais-positifs (TP) correspondent aux situations où l'IDS génère une alerte pour un événement qui est effectivement une attaque légitime.
- Les faux-positifs (FP) se produisent lorsque l'IDS génère une alerte pour un événement qui

n'est pas une véritable attaque.

- Les vrais-négatifs (TN) se produisent lorsque l'IDS ne génère pas d'alerte pour un événement qui n'est pas une attaque.
- Les faux-négatifs (FN) correspondent aux situations où l'IDS ne génère pas d'alerte pour un événement qui est pourtant une attaque.
  - **Taux de détection** : Le taux de détection représente la proportion d'attaques réellement identifiées par l'IDS par rapport au nombre total d'attaques réelles présentes dans les données.

    Taux de détection= $\frac{TP}{TP+FN}$

- **Exactitude (Accuracy)** : L'exactitude mesure la proportion totale de prédictions correctes (TP + TN) par rapport à l'ensemble des prédictions réalisées par l'IDS.

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

## 3.7 Résultats :

Avant de présenter les résultats de notre technique de détection d'anomalies, nous avons d'abord tenté de classifier les vecteurs de données sans utiliser de visualisation, en utilisant le même modèle CNN. Cette étape avait pour but de comparer les performances de notre approche basée sur la visualisation des données avec une méthode plus traditionnelle de classification vectorielle. En appliquant le modèle CNN directement sur les vecteurs de données.

| Type | Taux de classification correcte |
|---|---|
| Normal | 96.45% |
| DoS attaques | 82.02% |

Tableau 5 : Résultats relatifs aux Taux de Classification vectorielle correcte .

Le tableau suivant représente les résultats obtenus à partir de taux de classification correcte atteint pour détecter les intrusions avec notre technique :

| Type | Taux de classification correcte |
|---|---|
| Normal | 99.07% |
| DoS attaques | 82.71% |

Tableau 6: Résultats relatifs aux Taux de Classification Correcte .

- Les résultats obtenus confirment que la technique de visualisation par modèles images permet au modèle CNN d'exploiter efficacement les caractéristiques spatiales et structurelles des données, difficilement capturables par une représentation vectorielle classique. En convertissant les données en images, notre approche facilite la

reconnaissance de motifs complexes et la détection de comportements anormaux dans le trafic réseau. Ainsi, cette méthode améliore significativement la précision de la détection des attaques DoS et renforce la capacité des systèmes de détection d'intrusion à réagir de manière proactive aux menaces de sécurité.

### 3.7.1 Travaux Connexes :

La détection d'intrusion est un sujet largement étudié, et les travaux antérieurs offrent l'opportunité de comparer nos résultats avec des études similaires. Les résultats obtenus par notre technique sont présentés dans le tableau 7 avec le taux de précision de notre modèle et des méthodes proposées par des chercheurs précédents. Les résultats montrent six techniques et leur taux de précision prévu. Comme nous pouvons le voir dans le tableau 7, le taux de précision de notre modèle est supérieur à celui des autres méthodes de deep learning.

| Classifieurs | Accuracy (%) |
|---|---|
| DNN [29] | 75.75 |
| CNN [30] | 79.48 |
| 5-layer Autoencoder (AE) [31] | 90.61 |
| DLS -IDS (Deep Learning Spark Intrusion Detection System) [32] | 83.57 |
| DT -PCADNN [33] | 88.64 |
| Notre modèle (CNN) | 92.98 |

Tableau 7: Comparaison des performances du modèle proposé.

Les résultats montrent que notre modèle, basé sur CNN, obtient un taux de précision supérieur, avec une précision de 92,98%, comparé aux autres méthodes de deep learning comme CNN qui atteint 79,48%. En comparaison avec les méthodes de deep learning telles que DNN, Autoencoder (AE), DLS-IDS et DT-PCADNN, notre modèle démontre une nette amélioration de la précision. Cela souligne l'efficacité de notre approche basée sur CNN pour la détection d'intrusion.

En résumé, notre approche basée sur CNN démontre une amélioration significative dans la précision par rapport aux autres méthodes de deep learning.

## 4 Conclusion :

Ce chapitre présente notre travail visant à renforcer la détection des attaques par déni de service (DoS). Pour améliorer le taux de détection, nous avons développé une approche utilisant le jeu de données NSL-KDD pour identifier les modèles d'attaques DoS en nous basant sur des formes géométriques simples. Pour y parvenir, nous avons intégré un modèle CNN capable de discerner efficacement les attaques DoS du trafic normal. Les résultats obtenus, comparés aux techniques de détection classiques, démontrent l'efficacité de notre méthode de classification .

# Conclusion générale :

Le travail présenté s'inscrit dans le domaine de la sécurité informatique, et plus précisément dans les systèmes de détection d'intrusion. Nous nous sommes concentrés sur une technique déjà proposée dans l'article intitulé "Classification of DOS Attacks Using Visualization Technique" en 2014.[22]

Dans ce travail, nous avons développé une technique de détection d'intrusion capable de détecter, classifier et visualiser les attaques en temps réel. Les paquets relatifs au trafic sont représentés sous forme d'images avec des formes géométriques simples, ce qui permet de trouver des modèles d'attaques DoS visuelles et de les distinguer du trafic normal. Ces modèles sont ensuite utilisés pour la classification dans la détection d'intrusion.

Ainsi, nous considérons le problème de la détection d'intrusion comme un problème de reconnaissance de motifs, où la classification est effectuée non pas sur la base de paramètres arithmétiques souvent complexes, mais sur la base de formes dérivées d'une transformation géométrique. Pour cela, nous avons utilisé un réseau de neurones convolutionnels (CNN) pour entraîner un modèle capable de classifier et détecter si une attaque est présente. La visualisation se fait à partir de la base de données NSL-KDD, qui contient des signatures d'attaques pour deux catégories : « Attaque » et « Normal ».

Enfin, les résultats obtenus illustrent l'efficacité de cette technique avec un taux élevé de classification correcte. Pour améliorer le taux de détection, des recherches futures pourront explorer des techniques complémentaires et affiner davantage notre approche.

En conclusion, notre recherche a permis de renforcer les fondements des systèmes de détection d'intrusion et de proposer un système performant. Ces résultats ouvrent de nouvelles perspectives pour la sécurisation des réseaux et constituent une avancée significative dans le domaine de la sécurité informatique.

# Bibliographie

[1] Hidjeb, Ali. "Implémentation d'un protocole d'élection d'un serveur d'authentification dans l'internet des objets." Mémoire de fin de Cycle Master 2 Informatique Professionnel Option : ASR Administration et Sécurité des Réseaux, Université Abderrahmane Mira de Bejaïa.

[2] "Quels sont les différents types de sécurité informatique?" Sécurité RSI. (rsisecurity.com)

[3] Arnould, Gerard. "Étude et Conception d'Architectures Haut-Débit pour la Modulation et la Démodulation Numériques." Thèse de Doctorat, École Doctorale IAEM – Lorraine Département de Formation Doctorale Électronique – Électrotechnique, Université Paul Verlaine – Metz, Décembre 2006.

[4] Poinsot, Laurent. "Introduction à la sécurité informatique." Support de cours, Université Paris 13.

[5] "Les virus informatique." Clusif, 2005, page 10

[6] "Les virus et les spam." Page 37. (https://www.sophos.com/fr-fr/medialibrary/PDFs/case%20studies/fr/comviru_vrius_bfr.pdf?la=fr-FR)

[7] Biondi, Philippe. "Architecture expérimentale pour la détection d'intrusions dans un système informatique." Article de recherche, Avril-Septembre 2001.

[8] Bloch, Laurent, et Christophe Wolfhugel. "Sécurité informatique." EYROLLES, 2ème édition, 2005.

[9] Burgermeister, David, et Jonathan Krier. "Les systèmes de détection d'intrusions." 2006.

[10] ISO/IEC 27000. "Information technology — Security techniques — Information security management systems — Overview and vocabulary." 2009. (http://standards.iso.org/ittf/licence.html)

[11] Cole, E., Krutz, R., et Conley, J. "Network Security Bible." Wiley Publishing, Inc, 2005. ISBN13:978-0-7645-7397-2.

[12] Stallings, William. "Network Security Essentials: Applications and Standards." Fourth Edition, 2011.

[13] Aissaoui, Sihem. "Apprentissage automatique et sécurité des systèmes d'information : Application un système de détection d'intrusion basé sur les SVM." Université d'Oran, 2008.

[14] Anderson, J. "Computer security threat monitoring and surveillance." 1980.

[15] Denning, D. E. "An intrusion detection model." In IEEE Transactions on software engineering, SE-13:222-232, 1987.

[16] Biondi, Philippe. "Architecture expérimentale pour la détection d'intrusions dans un système informatique." 2001.

[17] Rahmani, Amine Boumedien Hassan. "La détection d'intrusion (Optimisation par classification)."

[18] Pharate, A., Bhat, H., Shilimkar, V., et Mhetre, N. "Classification of Intrusion Detection System." International Journal of Computer Applications, Volume 118 – No. 7, May 2015.

[19] Farhaoui, Yousef. "Évaluation des systèmes de détection et de prévention des intrusions et la conception d'un BIDS." Thèse de doctorat, Université Ibn Zohr, 2012.

[20] Michel, Cédric. "Langage de description d'attaques pour la détection d'intrusions par corrélation d'événements ou d'alertes en environnement réseau hétérogène." Thèse de doctorat, Université de Rennes 1, 16 Décembre 2003.

[21] Demay, Jonathan-Christofer. "Génération et évaluation de mécanisme de détection d'intrusions au niveau applicatif." Thèse de doctorat, école doctorale Matisse, Université de Rennes 1, Juillet 2011.

[22] Berthier, Yann, et Jean-Baptiste Marchand. "Détection d'intrusions et analyse forensique."

[23] https://www.jetbrains.com/fr-fr/pycharm/features/

[24] https://www.jetbrains.com/fr-fr/pycharm/features/

[25] https://www.cs.waikato.ac.nz/ml/weka/

[26] https://www.researchgate.net/figure/NSL-KDD-dataset-features_tbl1_302594395

[27] https://www.jeveuxetredatascientist.fr/convolutional-neural-network/

[28] Lamport, L. "Lower Bounds for Asynchronous Consensus." Distributed Computing 19, 2 (2006), 79-103, 2006.

[29] Tang T. A., Mhamdi L., McLernon D., Zaidi S. A. R., and Ghogho M. "Deep learning approach for Network Intrusion Detection in Software Defined Networking." International Conference on Wireless Networks and Mobile Communications (WINCOM), pp. 258–263, 2016. doi: 10.1109/WINCOM.2016.7777224.

[30] Wu K., Chen Z., and Li W. "A Novel Intrusion Detection Model for a Massive Network Using Convolutional Neural Networks." IEEE Access, vol. 6, pp. 50850–50859, 2018, doi: 10.1109/ACCESS.2018.2868993.

[31] Xu W., Jang-Jaccard J., Singh A., Wei Y., and Sabrina F. "Improving Performance of Autoencoder-Based Network Anomaly Detection on NSL-KDD Dataset." IEEE Access, vol. 9, pp. 140136–140146, 2021, doi: 10.1109/ACCESS.2021.3116612.

[32] Haggag M., Tantawy M.M., and El-Soudani M.M.S. "Implementing a Deep Learning Model for Intrusion Detection on Apache Spark Platform." IEEE Access, vol. 8, pp. 361669–163672, 2020, doi: 10.1109/ACCESS.2020.3019931.

[33] Alotaibi S.D. et al. "Deep Neural Network-Based Intrusion Detection System through PCA." Math. Probl. Eng., vol. 2922, p. 6488571, 2022, doi: 10.1155/2022/6488571