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

UNIVERSITE IBN KHALDOUN - TIARET

MEMOIRE

Présenté à :

FACULTÉ DES MATHEMATIQUES ET DE l'INFORMATIQUE
DÉPARTEMENT D'INFORMATIQUE

Pour l'obtention du diplôme de :

MASTER

Spécialité: Réseaux et Télécommunication
Par :

Saidi Wissam Halima
Sarir Fatima Zohra

Sur le thème

La mise en œuvre d'un système de détection d'injection SQL pour Les Applications web

---

Soutenu publiquement le 28/ 06 /2022 à Tiaret devant le jury composé de :

Mr KHERICI Cheikh	Grade Université MCB	Président
Mr DAOUD Mohamed Amine	Grade Université MAA	Encadrant
Mr ALEM Abdelkader	Grade Université MAA	Examinateur

2021-2022

Dédicaces

Nous dédions ce modeste travail à nos très chers parents que mille dédicaces ne puissent exprimer nos sincères sentiments pour leurs patiences, encouragements en témoignage de notre profond amour et respect.

À nos frères.

À nos amies.

A tous ce qui nous en chers.

REMERCIEMENT

Avant toute personne, nous remercions le bon Dieu de nous avoir prêté vie, santé et volonté pour achever ce travail.

Nous tenons à remercier notre encadrant M. DAOUD Mohamed Amine pour tout le temps qu'il nous a consacré, pour ses conseils précieux, pour toute son aide et son appui durant la réalisation de ce travail. Nous tenons à remercier, nos chers parents pour leur encouragement et soutien.

Nous tenons à remercier chacun des membres du jury pour nous avoir fait l'honneur d'examiner et d'évaluer notre travail.

Résumé :

Avec l'évolution augmentée d'Internet, les applications web sont devenues de plus en plus vulnérable et sensible aux attaques qui peuvent abime la confidentialité, l'intégrité ou la disponibilité des systèmes informatique, l'injection SQL est une cyber-attaque la plus répandue. Pour affronter à ces cyber-attaques, il est inévitable d'utiliser les systèmes de détection d'intrusion.

Une méthode de détection par injection SQL est proposée qui utilise les techniques de Deep Learning. Un type du Deep Learning, CNN (Convolutional Neural Network) est le meilleur chemin pour classifier les cyber-attaques et d'éviter le sur-ajustement et les sou-ajustement. Le modèle proposé a été efficace pour deux dataset différentes, en donnant des résultats très élevés concernant le taux de détection avec 97% pour le dataset SQLI, et de 99.99% pour le SQLIV2.

Abstract:

With the increased evolution of the Internet, web applications have become more and more vulnerable and sensitive to attacks that can damage the confidentiality, integrity or availability of computer systems, SQL injection is a cyber attack the most suitable. To face these cyber-attacks, it is inevitable to use intrusion detection systems.

A detection method by SQL injection is proposed which uses Deep Learning techniques. A type of Deep Learning, CNN (Convolutional Neural Network) is the best way to classify cyber attacks and avoid overfitting and underfitting. The proposed model was effective for two different datasets, giving very high results regarding the detection rate with 97% for the SQLI dataset, and 99.99% for the SQLIV2.

ملخص :

مع التطور المتزايد للإنترنت ، أصبحت تطبيقات الويب أكثر عرضة للخطر وحساسية للهجمات التي يمكن أن تضر بالسرية أو السلامة أو توفر أنظمة الكمبيوتر ، فإن حقن SQL هو الأكثر ملائمة للهجوم السيبراني. لمواجهة هذه الهجمات الإلكترونية ، لا مفر من استخدام أنظمة كشف التسلل.

تم اقتراح طريقة الكشف عن طريق حقن SQL والتي تستخدم تقنيات التعلم العميق. يعد CNN (الشبكة العصبية التلافيفية) أحد أنواع التعلم العميق ، وهو أفضل طريقة لتصنيف الهجمات الإلكترونية وتجنب الإفراط في التجهيز والتركيب. كان النموذج المقترح فعالًا لمجموعتين مختلفتين من البيانات، مما أعطى نتائج عالية جدًا فيما يتعلق بمعدل الكشف بنسبة 97٪ لمجموعة بيانات SQLI و 99.99٪ لـ SQLIV2.

Table des matières :

Liste des figures :

Table des figures :

I. Introduction générale :

La croissance des internautes chaque année augmente fortement. En janvier 2020, le nombre total d'internautes dans le monde a atteint plus de 4 milliards d'utilisateurs. Cela a été suivi par une augmentation des services d'applications Web et l'échange d'une quantité énorme d'informations, dont ces données sont stockées dans des bases de données relationnelles, ce qui les rend accessible via le Web ou / et des API (Application Program Interface). La même année, le nombre de sites Web et d'applications Web a atteint 1,74 milliard de sites Web. Beaucoup de ces sites Web sont vulnérables, ce qui attire également l'attention des attaquants pour effectuer divers types de Cyberattaques notamment les DDOS (Distributed Deni of Service) en rendant les services indispensables, car les portes dérobées sont laissées ouvertes par de mauvaises pratiques de codage qui ne respectent pas les bonnes normes de sécurité.

L'injection SQL est l'une des 10 principales vulnérabilités rencontrées par les applications Web selon une communauté en ligne OWASP (Open Web Application Security Project).L'attaque par injection SQL est une attaque par usurpation du serveur pour exécuter du code malveillant. Cette attaque peut divulguer des informations confidentielles sur les utilisateurs et les propriétaires d'applications. Cela se fait en soumettant la commande SQL à l'application Web via URL, formulaires ou autres pour obtenir des informations sensibles, afin de reprendre les droits d'accès et le contrôle de l'hôte. Cette attaque peut également être utilisée pour modifier ou endommager la base de données.

Il est fondamental d'avoir une méthode efficace pour sécuriser contre ce type d'attaque. Une méthode pour sécuriser les applications Web consiste à détecter les attaques d'injection SQL en utilisant une approche basée sur les anomalies. La méthode basée sur les anomalies classe activités dans la circulation en activités normales et activités suspectes. Cette méthode nécessite une analyse et une expertise dans la détection de modèles pour déterminer si une activité d'accès est normale ou constitue à une menace.

Les méthodes d'apprentissage automatique peuvent détecter l'injection SQL plus efficacement. Des travaux de recherche ont été menés dans ce contexte. Certains modèles d'apprentissage automatique rencontrent des problèmes de surajustement et de sous-ajustement. Ces problèmes diminuent la précision de la détection, le taux de détection et augmentent le taux de faux positifs et de faux négatifs. Le surajustement se produit lorsqu'un classificateur est formé avec beaucoup d'ensembles de données d'apprentissage. Il génère un modèle flexible qui classe de manière inexacte les attaques de l'ensemble de données de test. D'un autre côté, le sous-ajustement se produit lorsque moins d'ensembles de données d'entraînement et de caractéristiques extraites entraînent un modèle.

L'apprentissage en profondeur (Deep Learning) permet aux machines d'apprendre des modèles à partir des données, puis de les classer automatiquement dans des classes spécifiques. Une question de recherche a été pose spécifique a ce travail :

QR : À quoi sert la méthode Deep Learning pour détecter l'injection SQL ?

L'objet principal de ce mémoire est de proposer une approche basée sur l'apprentissage en profondeur pour la classification des attaques d'injection SQL afin de limiter les problèmes de

surajustement et sous-ajustements, puis nous donnons les résultats expérimentaux du système développé. Ce travail est une solution appropriée pour détecter efficacement l'injection SQL,

Notre mémoire est organisé comme suite :

- Dans le premier chapitre, on présente une description sur les cyberattaques ainsi que ses différents types d'attaques.

- Le deuxième chapitre décrit le système de détection d'intrusion (IDS), leurs différents types, ainsi leurs techniques.

- Le troisième chapitre présente les applications d'apprentissage automatique et ses types et spécialement l'apprentissage profond avec les types des algorithmes.

- Le dernier chapitre explique notre méthodologie de travail, qui consiste à détailler la méthode d'apprentissage élaborée, le chapitre contiendra l'implémentation du modèle adéquat a l'approche proposée.
- Et on conclut par une conclusion générale du notre travail.

Chapitre 1: Les Cyberattaques

Les Cyberattaques

I. Introduction :

Les cyberattaques sont généralement motivées par des objectifs criminels ou politiques. Les adversaires peuvent être une personne privée, un acteur étatique ou une organisation criminelle. Mais la principale réponse à la question concernant la raison pour laquelle ces attaques se produisent est de regarder les objectifs se cachant derrière chacune d'entre elles. Les cybercriminels ne veulent pas toujours la même chose, c'est pourquoi il n'existe pas une réponse simple à cette question.

Certains cybercriminels veulent de l'argent ou des informations, tandis que d'autres cherchent simplement à causer des problèmes. Ensuite, il y a ceux qui attaquent les systèmes dans le but de les détruire pour des raisons personnelles, comme c'est le cas parfois d'anciens employés mécontents**. [Logpoint]**

Figure 1 une cyberattaque

[Cyber]

II. Définition :

Une cyberattaque désigne toute action entreprise par des cybercriminels avec à l'esprit des objectifs malveillants. Les cybercriminels lancent leurs attaques en utilisant un ou plusieurs ordinateurs afin de frapper d'autres ordinateurs, réseaux ou systèmes d'information.

Diverses méthodes peuvent être utilisées pour lancer une cyberattaque, mais les objectifs sont généralement de :

- Voler des données.
- Détruire des informations ou des données.
- Modifier des données.
- Désactiver des ordinateurs.
- Obtenir un gain financier.
- Espionner. **[Logpoint]**

III. Types d'attaque :

Il existe de nombreux types de cyber attaques, mais certaines actions malveillantes sont plus courantes que d'autres. Les actions malveillantes les plus courantes incluent divers types de malwares, de déni de service et de phishing. **[Logpoint]**

1. Programmes malveillants :

Un programme malveillant est une expression utilisée pour décrire les logiciels malveillants, y compris les logiciels espions, les rançongiciels, les virus et les vers. Un programme malveillant s'introduit dans un réseau par le biais d'une vulnérabilité, généralement lorsqu'un utilisateur clique sur un lien dangereux ou sur une pièce jointe qui installe ensuite un logiciel à risque.

Une fois dans le système, le logiciel malveillant peut effectuer les tâches suivantes :

- Bloquer l'accès aux composants clés du réseau (rançongiciels)
- Installer d'autres logiciels malveillants ou des logiciels nuisibles supplémentaires
- Obtenir discrètement des informations en transmettant des données depuis le disque dur (logiciel espion)
- Perturber certains composants et rendre le système inutilisable. **[Cisco]**

2. Malware :

Malware vient de l'anglais « malicious software » qui signifie logiciel malveillant. Un malware est développé dans le but de nuire ou de récolter des informations d'un système. Il existe différentes familles de malware qui ont chacune leur manière de fonctionner et des objectifs divers**. [Cacciapaglia ,2018]**

a. **Virus :**

Un virus est un automate autoréplicatif, c'est-à-dire qu'il peut fabriquer autonomement une copie de lui-même en utilisant les ressources de son environnement.

Un virus est conçu pour rester indéfiniment caché, on dit alors qu'il s'exécute en arrière plan, il peut faire un certain nombre d'actions dans l'appareil hôte :

- Supprimer des données, soit pour effacer ses traces, soit pour nuire au système
- Endommager le système ou le ralentir.
- Installation de malware.
- Chiffrer des données pour demander une rançon.

Figure 2 un virus « antivirus »

[**Cacciapaglia**]

Exemple de virus :

Un exemple de virus assez courant est le faux antivirus. Il s'agit d'un virus qui se fait passer pour un antivirus et propose de supprimer les virus détecter contre une certaine somme. Le nombre de virus trouvés est toujours le même pour tous les appareils. **[Cacciapaglia ,2018]**

b. **Ver (Worm) :**

Un ver est un malware, il est similaire au virus car il a comme objectif de se propager et est auto réplicatif. Cependant, tandis que le virus a besoin d'un programme hôte pour se reproduire, le ver utilise les ressources de son environnement pour se multiplier et se propager.

Exemple d'attaque :

Le 4 mai 2000, un ver informatique nommé « I love You » a infecté, dans le monde, 10% des ordinateurs connectés à internet utilisant le système Windows. Le ver était exécuté par une pièce

jointe envoyée par mail : Love-letter-for-you.txt.vbs.
Le ver contenait un script VBS, il faut savoir qu'à l'époque Windows n'affichait pas l'extension « vbs » par défaut, c'est avec cette tactique que la personne malveillante a pu tromper ses victimes. De plus, le ver avait besoin que le système permettait l'exécution des fichiers de langage de Scripting (fichiers .vbs). **[Cacciapaglia ,2018]**

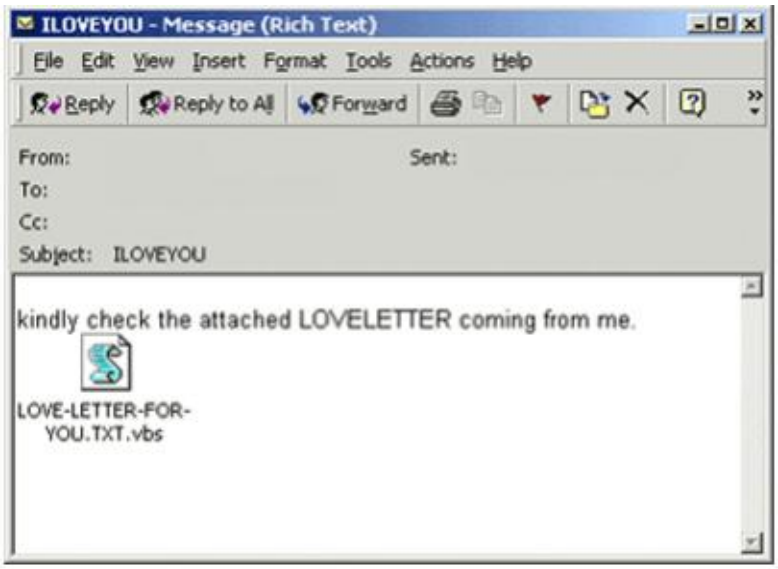

Figure 3 le ver « I love you »

[**Cacciapaglia**]

3. Attaque par dénie de service (Dos):

Le principe de cette attaque est de rendre indisponible un service en lui envoyant un nombre important de requêtes afin de saturer le serveur. Il faut savoir qu'un serveur a une capacité limitée de communication, si une personne malveillante parvient donc à envoyer des paquets/requêtes à répétition, le serveur peut alors commencer par ralentir et finalement crasher. **[Cacciapaglia ,2018]**

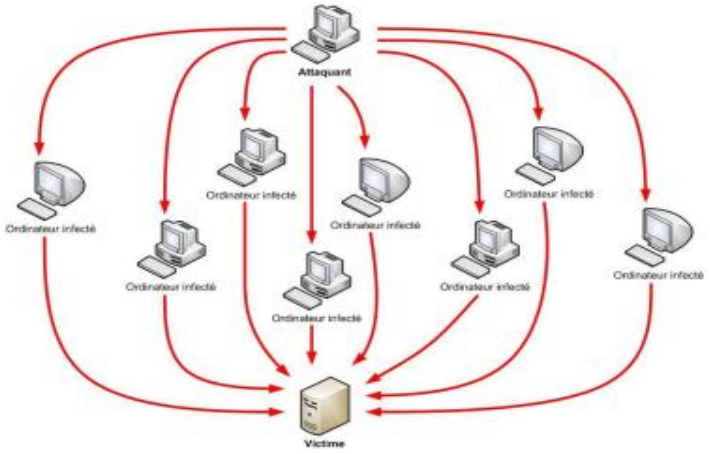

Figure 4 Structure d'une attaque par déni de service.

[**Cacciapaglia**]

a. **L'inondation SYN (SYN flooding):**

Comme la montre l'image ci-dessous, on voit que la personne malveillante (l'initiateur) ne va pas renvoyer d'ACK au receveur mais un SYN, le receveur va donc à nouveau envoyer un ACK et ainsi de suite. On voit dans la deuxième partie de l'image qu'une personne qui souhaite alors accéder au service ne reçoit aucune réponse de ce dernier car ce dernier est indisponible dû à l'attaque.**[Cacciapaglia,2018]**

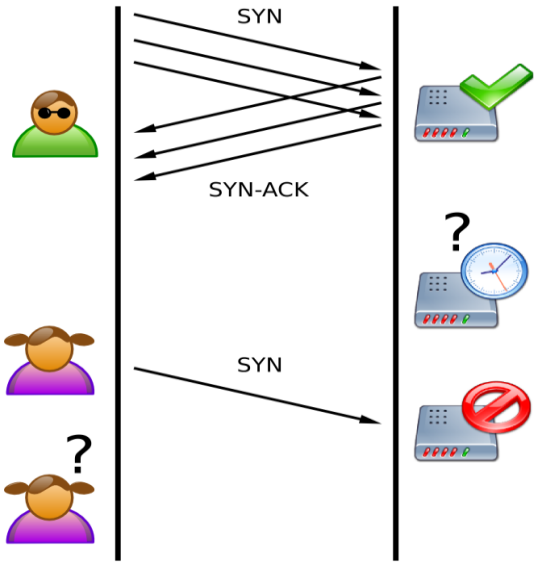

Figure 5 L'inondation syn

[**Wekimedia**]

b. **L'inondation UDP (UDP Flooding):**

Cette attaque utilise le protocole UDP qui permet la transmission de données entre deux machines. L'attaque consiste à créer une grande quantité de paquets UDP et de les envoyer au serveur ciblé. Attention, le protocole UPD est prioritaire sur le protocole TCP, ce qui fait qu'au bout d'un moment, le trafic UDP monopolisera toute la bande passante et ne laissera qu'une infime partie au trafic TCP.**[Cacciapaglia ,2018]**

Figure 6 L'inondation UDP

[**Open**]

c. **Attaque par rebond (Smurfing) :**

Cette attaque utilise le protocole ICMP qui est un protocole de message de contrôle internet donc des Echo. Le principe est d'envoyer un ICMP Echo à toutes les machines d'un réseau en mettant comme adresse IP source (destinataire) celle de la cible soit : 172.18.173.109, cela s'appelle de l'usurpation d'adresse IP. Comme le montre l'image ci-dessous, les ordinateurs du réseau répondront donc à l'écho à la cible de l'attaque. **[Cacciapaglia ,2018]**

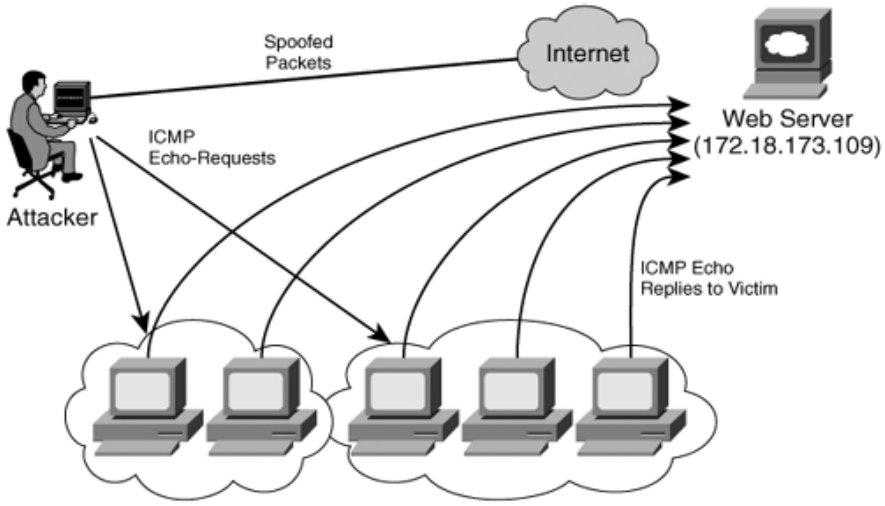

Figure 7 Attaque par rebond

[**Network**]

4. Injection SQL :

Le langage SQL (Structured Query Language ou langage de requête structurée) est un langage informatique permettant l'exploitation des bases de données relationnelles, il permet de faire des manipulations sur les données par des requêtes: la lecture, la modification, l'ajout et la suppression de ces dernières. SQL Injection est donc une attaque qui exploite la syntaxe SQL, il s'agit d'injecter dans une requête SQL du code supplémentaire pour provoquer une manipulation des données de la base de données. **[Cacciapaglia,2018]**

Exemple d'injection SQL:

SELECT * FROM Users WHERE Username='$username' AND Password='$password'

Cette requête est vulnérable à SQLi car elle utilise directement les entrées fournies par l'utilisateur sans aucune vérification et peut être exploitée d'une manière très simple comme suit :

SELECT * FROM Users WHERE Username='1' OR '1' = '1' AND Password='1' OR '1' = '1'

5. L'homme au milieu :

Man in the middle ou l'homme au milieu est une attaque informatique qui a pour objectif d'intercepter les communications entre deux parties sans que ces dernières ne s'en rendent compte**. [Cacciapaglia ,2018]**

Figure 8 l'homme au milieu

[Blogs]

6. Hameçonnage (Phishing) :

Le phishing est un type courant d'attaque de cyber sécurité. Cette technique implique généralement l'envoi d'emails qui semblent authentiques mais qui proviennent en réalité de cybercriminels, demandant généralement des données personnelles. Malheureusement, même si les filtres anti-spam progressent, les cybercriminels continuent de développer des moyens de les contourner. **[Logpoint]**

Figure 9 Hameçonnage

[Miami]

7. L'attaque XSS (Cross-site Scripting):

Certains sites (internet) permettent aux utilisateurs d'interagir avec le site en récupérant les inputs qui feront alors partie du site. Comme exemple, les sites avec des photos et articles qui permettent aux utilisateurs de commenter grâce à des interfaces/formulaires. Le cross-site Scripting est le fait d'exploiter une faille pour y injecter un contenu malveillant provoquant alors d'autres actions que celles déjà existantes sur la page. La faille peut être via un message par un forum ou par de la manipulation d'URL. **[Cacciapaglia ,2018]**

Figure 10 L'attaque XSS

[Detectify]

IV. Injection SQL :

Une injection SQL est l'intégration d'un code malveillant dans des applications web dans le but d'attaquer des sites web et/ou collecter les données des utilisateurs. Les pirates lancent des attaques par injection SQL pour des raisons diverses, mais peuvent aussi, en sus d'enfreindre la sécurité des données, envoyer de fausses informations dans la base de données de l'application, en supprimer des informations importantes ou empêcher l'accès aux propriétaires et créateurs de l'application. Ils doivent alors trouver et exploiter une faille dans la sécurité du logiciel de l'application ciblée.

Abréviation de « Structured Query Language », SQL est un langage spécialement conçu pour saisir des données et modifier le contenu de bases de données. Les sites et applications web utilisent ces bases de données pour stocker toutes leurs données et fournir leurs services aux utilisateurs. **[Software]**

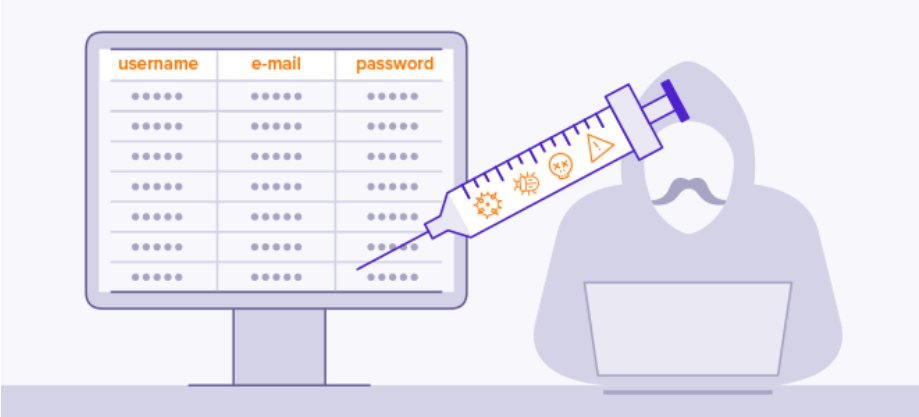

Figure 11 Injection SQL

[Avast]

1. Principe de l'attaque par injection SQL :

La possibilité d'accéder aux applications Web à distance multiplie les points d'attaque potentiels pour les pirates qui veulent attaquer ces applications ainsi que leurs bases de données. Le but essentiel du pirate est de changer la structure (ou la sémantique) des requêtes SQL pour qu'elles soient interprétées différemment de ce qui avait été prévu par le programmeur**.** Pour ce faire, ce pirate injecte des caractères dangereux 1 (‘,/,-, etc.) et des mots clés SQL (union, drop, etc.) dans les champs d'entrée. Intuitivement, si le programmeur ne valide pas les entrées d'utilisateur avant de les employer dans des requêtes SQL dynamiques, alors l'attaque pourra réussir (figure 12). **[Hakim, 2018]**

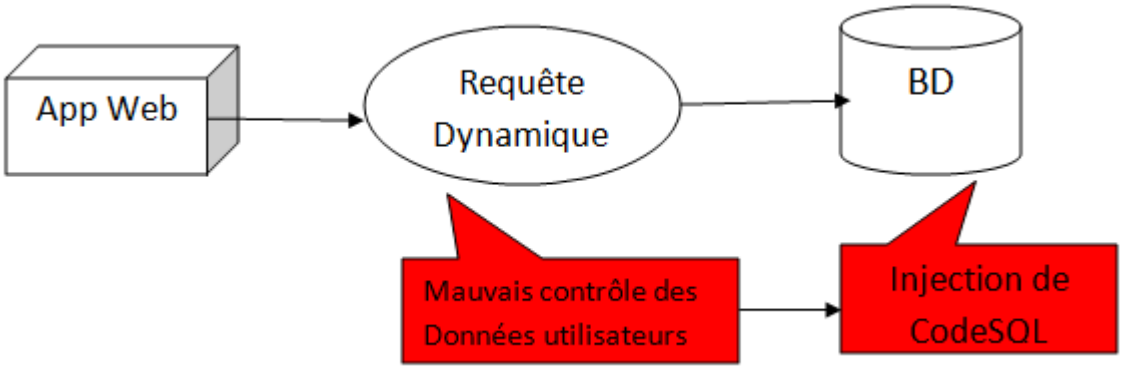

Figure 12 l’attaque par injection SQL

[Hakim]

2. Les mécanismes d'injection :

Il existe trois mécanismes d'injection permettant d'exécuter un code SQL malveillant sur les bases de données d'une application Web.

a. Injection dans les entrées d'utilisateur :

Si le code source d'une application Web contient des requêtes SQL construites avec les entrées d'utilisateur, un pirate peut facilement saisir des caractères dangereux dans ces entrées pour monter une SQLIA. Les entrées d'utilisateur constituent les points d'attaques les plus exploitables par les pirates et les moins détectables par les pare-feu et les IDS.

[Hakim, 2018]

b. Injection dans les témoins de connexion :

Les témoins de connexion (cookies) sont des fichiers placés sur le poste du client, associés à une application Web et contiennent des informations sur les préférences des utilisateurs, etc. Comme les entrées d'utilisateur, les cookies peuvent être exploitables par un attaquant lorsque l'application Web stocke les données dans ces cookies sans les chiffrer. Les nouvelles applications web chiffrent les contenus de cookies pour contrer les attaques provenant de ces derniers. **[Hakim, 2018]**

c. Injection dans les variables du serveur :

La connexion entre le client et le serveur Web se rompt fréquemment, car HTTP (ou HTTPS) est un protocole déconnecté. Pour identifier les clients, le serveur crée des variables, appelées variables de serveur, contenant des informations concernant les clients et qui sont extraites à partir des entêtes http. Un pirate peut donc injecter ses propres valeurs dans l'entête http par l'URL pour perpétrer une SQLIA qui se déclenche automatiquement lorsque le serveur communique les valeurs des variables du serveur avec la base de données aux fins de statistique. **[Hakim, 2018]**

1. Types d'attaque par injections SQL :

Les attaques par injection SQL sont classées en cinq types :

a. Injection SQL en aveugle (Blind SQLi) :

L'injection SQL en aveugle (Blind SQLi), est une attaque SQL qui soumet des modifications de requêtes en aveugle à la base de données. L'objectif de cette approche est de déterminer la réponse en fonction du retour de l'application (absence de données,

message d'erreur ou délais de réponse de la base). Cette attaque est principalement utilisée quand une application est configurée pour afficher des messages d'erreur générique. Il est aisé d'envisager une requête testant l'existence de l'Id d'un utilisateur, puis grâce à une condition if d'exiger du moteur d'attendre quelques secondes s'il n'existe pas. Ce type d'attaque par injection SQL est relativement compliquée, mais pas impossible. Elle sert en règle générale en phase de reconnaissance pour définir le moteur SQL employé ou pour se faire une idée du schéma de la table, de la base ou pour contrôler si un enregistrement existe par des tests sur les clés primaires. **[Analyse]**

b. **Injection SQL par l'erreur (ErrorSQLi) :**

L'injection SQL par l'erreur (ErrorSQLi), est une méthode d'exploitation qui vise au moyen de soumission de requêtes SQL erronées, à récupérer des informations sur le modèle de la base donnée grâce aux messages d'erreurs natifs du moteur de base de données.**[Analyse]**

c. **Injection SQL par union (Union SQLi) :**

L'injection SQL par union (Union SQLi), est une exploitation basée sur l'utilisation de l'opérateur UNION pour combiner différents jeux de résultats de plusieurs instructions SELECT en un seul jeu de données. Ce type d'attaque par injection SQL sert à exfiltrer très rapidement de fortes volumétries de données d'une application.**[Analyse]**

d. **Injection SQL par sous requête et empilement(StackedQueriesSQLi) :**

L'injection SQL par sous-requêtes ou empilement (StackedQueriesSQLi), est de loin l'attaque la plus dangereuse pour une infrastructure. Elle permet d'exécuter plusieurs instructions dans la même requête pour étendre les possibilités de l'injection SQL initiale. Une requête par empilement offre un très grand niveau de contrôle à un attaquant. Ce type d'attaque par injection SQL permet à l'attaquant de ne pas exécuter la requête d'origine et de la remplacer par une nouvelle requête SQL. Il pourra à partir de là faire ce qu'il souhaite. Le risque le plus grand est que l'environnement de la base de données puisse avoir accès à des commandes comme xp shell cmd qui permettent d'exécuter du code Shell directement via une procédure stockée.**[Analyse]**

e. **Injection SQL par requête XPATH :**

L'injection SQL par XPATH est spécifique aux moteurs SQL qui permettent la manipulation d'élément XML (comme le Transac-SQL). Une chaine XPATH constitutive de la requête est alors concaténée avec une donnée non sécurisée, permettant à un

attaquant d'accéder à une donnée autre, ou d'invalider la requête pour en exécuter une nouvelle (par la méthode StackedQueries).**[Analyse]**

Exemple :

Au cours des vingt dernières années, de nombreuses attaques par injection SQL ont visé de gros sites web, sociétés et réseaux sociaux. Plusieurs d'entre-elles ont abouti à d'importantes fuites de données. Nous vous présentons ici certains exemples les plus remarquables :

En 2008, deux pirates russes ont utilisé des techniques d'injection SQL pour attaquer Heart land Payment Systems, un important fournisseur de solutions de traitement de paiements de l'époque. Considérée à l'époque comme la plus importante violation de données de cartes de crédit, cette attaque avait permis aux pirates d'obtenir les détails de plus de 150 millions de cartes de crédit et coûté plus de 300 millions de dollars aux sociétés victimes. Les deux pirates auteurs de l'attaque ont été condamnés à une peine combinée de 16 ans et plus en 2018. **[Software]**

En 2016, un groupe de pirates profite de failles dans vBulletin, un logiciel populaire de messagerie en ligne pour attaquer 11 plateformes de messagerie de jeu, la plupart en russe. Au cours de cette attaque, les pirates ont réussi à voler les identifiants de plus de 27 millions de comptes. **[Software]**

Toujours en 2016, des pirates utilisent des méthodes d'injection SQL pour lancer une cyberattaque contre la banque nationale du Qatar et volent plus de 1,4Go de données qu'ils rendent immédiatement publiques. Ces données contenaient les détails des comptes de milliers de clients, notamment des membres de la famille royale qatari, agents du renseignement, leaders religieux et plusieurs citoyens britanniques, français et américains marqués comme étant des espions dans la base de données de la banque. **[Software]**

V. Conclusion :

Pour organiser une bonne défense, il faut connaître les attaques. Ce chapitre a passé en revue les cyberattaques les plus courantes, que les pirates utilisent pour perturber et compromettre les systèmes informatiques. Comme vous avez pu le constater, les attaquants disposent d'un vaste éventail d'options, telles que les attaques DDoS, les infections malveillantes, les interceptions par l'homme du milieu et, pour tenter d'obtenir un accès non autorisé aux infrastructures critiques et aux données sensibles.

Concernant notre étude on est intéressé des attaques par injection SQL.

Pour détecter les différentes attaques illustrées dans ce chapitre, les chercheurs ont proposé plusieurs solutions. Dans le chapitre suivant, nous allons voir quelques techniques proposées pour défendre contre les SQLIAs.

Chapitre 2: Les systèmes de détection d'intrusion

Les Systèmes de détection d'intrusion

I. Introduction :

Aujourd'hui, avec notre utilisation accrue d'internet, la sécurité informatique n'est pas optionnelle. A l'aide d'outils comme les IDS et IPS, cela nous permet de vérifier l'état de sécurité de notre réseau. Que ceci soit dans un cadre privé, professionnel ou public, un administrateur doit prendre conscience est connaitre les risques qu'encours sont réseau. L'installation d'un IDS n'est pas si coûteuse et permet de réduire les risques et de mieux s'en prémunir. Il faut néanmoins définir correctement ses besoins et attentes afin de choisir le bon outil **[Thom]**

II. IDS (Intrusion Detection System):

Un système de détection d'intrusion (IDS) est une solution logicielle qui surveille un système ou un réseau pour détecter les intrusions, les violations de politique ou les activités malveillantes. Lorsqu'une intrusion ou une violation est détectée, le logiciel avertit l'administrateur ou le responsable de la sécurité. Cela les aide à enquêter sur les incidents signalés et à prendre les mesures appropriées.

Cette solution de surveillance passive peut alerter pour détecter une menace, mais elle ne peut pas agir directement contre celle-ci. C'est comme un système de sécurité installé dans un bâtiment qui peut informer les agents de sécurité d'une menace imminente.
Le système IDS vise à détecter une menace avant qu'elle n'entre dans le réseau. Il permet de scruter le réseau sans entraver le flux du trafic réseau. En plus de détecter les violations de politique, il peut protéger contre les menaces telles que les fuites d'informations, les accès non autorisés, les erreurs de configuration et les virus.
Cela fonctionne mieux lorsque nous ne voulons pas entraver ou ralentir le flux de trafic même si des problèmes surviennent, mais pour protéger vos ressources réseau. **[Geekflare]**

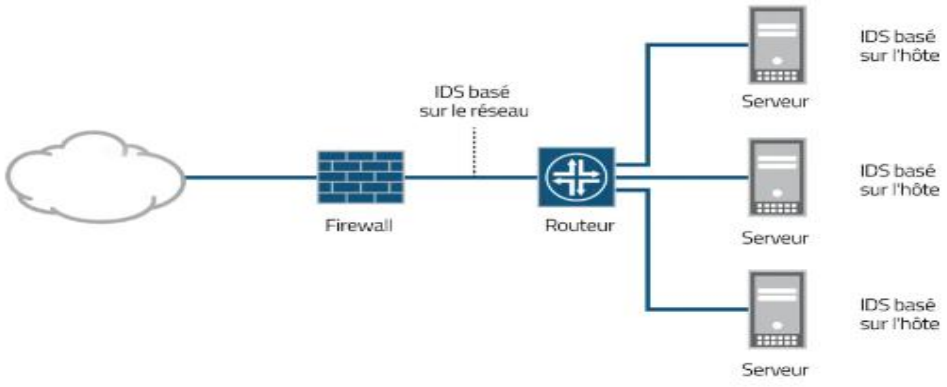

Figure 13 un système de détection d'intrusion

[Glossaire]

1. Les Types d'IDS :

Compte tenu de la variété des attaques pouvant être réalisées, la détection d'intrusion doit se faire à plusieurs niveaux. Il existe différents types d'IDS selon l'endroit où ils surveillent et ce qu'ils contrôlent ("sources d'information") ou selon leur fonction.

a. Les IDS réseaux (Network-based IDS) :

Ils sont également connus sous le nom de NIDS. Network IDS analyse et interprète les paquets circulant sur un réseau (ou un segment de réseau) pour identifier les paquets au contenu malveillant. Le paquet est analysé sur toutes ses couches (réseau, transport, application). Grâce à l'analyse des paquets et à la connaissance des protocoles, NIDS peut détecter les paquets malveillants conçus pour contourner les pare-feux. Ce type d'IDS a l'avantage d'être plus facile à protéger (contre les attaques sur l'IDS lui-même) car il n'observe que le trafic. Cependant, une des limitations des NIDS est que, pour pouvoir écouter tous les paquets, ils nécessitent une bande passante proportionnelle à l'importance du trafic. De plus, l'emplacement du NIDS dans le réseau doit être stratégique pour pouvoir surveiller tout le trafic. De plus, NIDS a des limites dans la protection du réseau contre le trafic crypté. Quelques exemples de NIDS sur le marché sont NetRanger, NFR, Snort, DTK et ISS RealSecure.

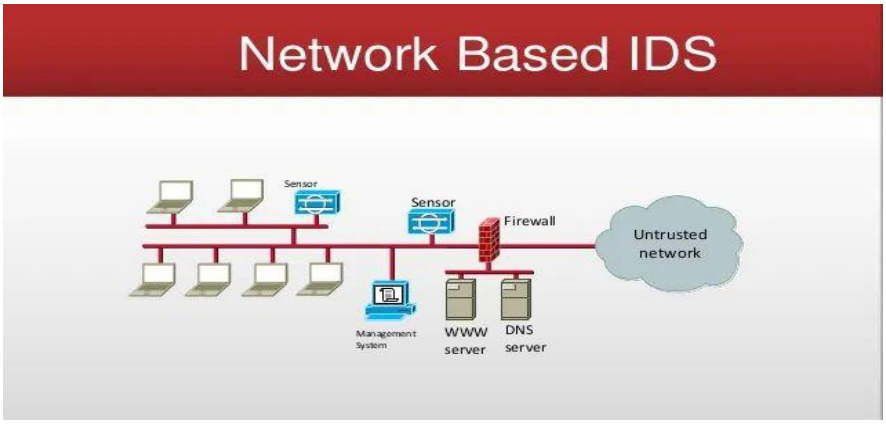

Figure 14 Les IDS Réseaux

[Cyberhoot]

b. Les IDS hôtes (Host-based IDS):

Les systèmes de détection d'intrusion basés sur l'hôte, également appelés HIDS, analysent exclusivement les activités liées aux serveurs sur lesquels ils sont installés (serveurs, postes clients, pare-feu, etc.), recherchent les activités suspectes. La détection peut être effectuée à l'aide des journaux d'audit de sécurité, des journaux système, du trafic réseau

du serveur, des processus en cours d'exécution, de l'accès aux fichiers, des modifications de configuration des applications, etc. Généralement, HIDS est déployé sur des serveurs critiques, tels que des serveurs contenant des informations hautement sensibles et des serveurs accessibles au public. En se concentrant sur la sécurité d'un seul serveur, HIDS a l'avantage d'avoir une plus grande précision contre les types d'attaques. De plus, l'impact d'une attaque peut être vu et permet une meilleure réponse. Les attaques dans le trafic crypté peuvent être détectées (impossible avec l'IDS réseau). Cependant, HIDS est plus vulnérable aux attaques par déni de service. De plus, du fait de leur volume de données, l'analyse des journaux peut nécessiter des ressources importantes (capacité de calcul et de stockage). Des exemples de HIDS sont OSSEC (Open Source Security), Tripwire, Radmind, eXpertBSM AIDE (Enhanced Intrusion Detection Environment) d'EMERALD et PortSentry. **[Illy]**

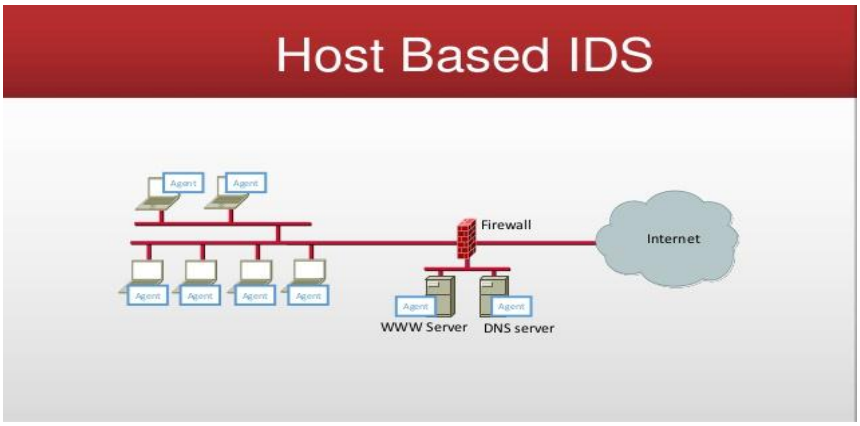

Figure 15 Les IDS Hôtes

[Gbhackers]

c. **Les IDS hybrids (Hybrid-Based IDS):**

La nouvelle tendance en matière de détection d'intrusion est de combiner les NIDS et les HIDS pour concevoir des IDS hybrides. Les systèmes hybrides de détection d'intrusion sont flexibles et augmentent le niveau de sécurité. Ils combinent plusieurs localisations des systèmes IDS et recherchent si bien les attaques visant des éléments particuliers que celles visant l'ensemble du système. Un exemple d'IDS hybride est ISS Real Secure.**[Illy]**

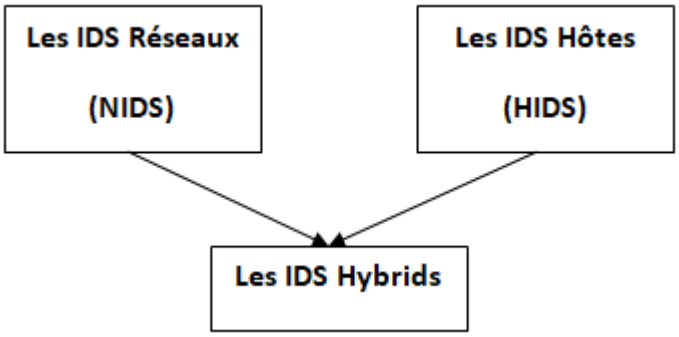

Figure 16 Les IDS Hybrids

[Arxiv]

d. **Les IDS de nœuds réseaux (Network Node IDS) :**

Un système de détection d'intrusion de nœud, connu sous le nom de NNIDS, fonctionne comme un NIDS traditionnel mais n'analyse que les paquets de trafic réseau destinés à un nœud spécifique. Le fait qu'ils n'analysent pas chaque paquet sur le chemin leur permet d'être beaucoup plus rapides et nécessitent moins de ressources. Cela leur permet d'être installés sur la plupart des serveurs. Ils sont également particulièrement adaptés aux segments fortement chargés ou au trafic crypté, domaines où les NIDS traditionnels ont des limites. Cependant, il est nécessaire d'installer plusieurs NNIDS, un NNIDS par serveur pour la sécurité.**[Illy]**

e. **Les IDS basés sur une application (Application-based IDS) :**

L'IDS basé sur les applications (ABIDS) surveille l'interaction entre un utilisateur et un programme en ajoutant des fichiers journaux pour fournir plus d'informations sur les activités. Travaillant entre l'utilisateur et le programme, ABIDS filtre facilement tout comportement notable. Ses principaux avantages sont un fonctionnement clair (contrairement à NIDS, par exemple) d'où une analyse plus facile et la possibilité de détecter et de bloquer des commandes spécifiques que l'utilisateur peut utiliser avec le programme. Deux inconvénients majeurs ont été identifiés : faible chance de détecter, par exemple, un cheval de Troie ; de plus, les fichiers journaux générés par ce type d'IDS sont des cibles faciles pour les attaquants (par exemple, ils ne sont pas aussi sécurisés que les pistes d'audit du système).**[Illy]**

f. **Les IDS basés sur la pile (Stack-Based IDS) :**

Les systèmes de détection d'intrusion basés sur une pile (SBIDS pour Stack-Based IDS), travaillent étroitement avec la pile TCP/IP, octroient la consultation des paquets lorsqu'ils montent à travers les couches OSI et permettent ainsi à l'IDS de retirer les paquets de la pile avant que le système d'exploitation ou l'application n'ait eu la possibilité d'élaborer la charge virale. L'IDS basé sur une pile peut être efficace contre certaines formes de chiffrement en retraçant les paquets après qu'ils aient été déchiffrés par la pile TCP/IP.**[Illy]**

Tableau 1 Tableau comparatif entre NIDS et HIDS

[Meh]

Types	**Avantages**	**Inconvénients**
HIDS	1. Il peut détecter les attaques qui n'impliquent pas le réseau, peut analyser ce que fait une application.	1. Ils sont isolés du réseau d'activités, doit être installé sur chaque héberger.
NIDS	1. Il peut surveiller plusieurs hôtes à la fois. 2. Il peut corréler les attaques contre plusieurs hôtes. 3. Cela n'affecte pas les performances de l'hôte. 4. Il peut détecter les attaques qui ne sont pas visibles à partir d'hôtes uniques.	1. Il peut y avoir un problème avec les canaux cryptés. 2. Il doit pouvoir correspondre à la vitesse du réseau.

2. Les méthodes de détection d'intrusion :

L'idée de base de la détection d'intrusion repose sur l'hypothèse que les comportements d'une activité d'intrusion sont plus ou moins différents des activités normales et qu'ils sont donc détectables. Plusieurs approches de détection d'intrusion ont été proposées dans la littérature. Collectivement, ces approches sont classées en trois catégories : détection des anomalies,

détection basée sur les signatures et La Détection de spécification .Les solutions combinant plusieurs méthodes de détection sont appelées détection hybride.

a. La détection d'anomalie :

Le principe de base de la détection d'anomalie est de modéliser durant une première période, dite phase d'apprentissage, le comportement « normal » du système en définissant une ligne de conduite (dite Baseline ou profil). Ensuite, en une seconde phase, période de détection, il est considéré comme suspect tout comportement inhabituel c'est-à-dire les déviations significatives par rapport au modèle de comportement « normal ». propose un modèle typique de détection d'anomalie illustré par la figure suivante.

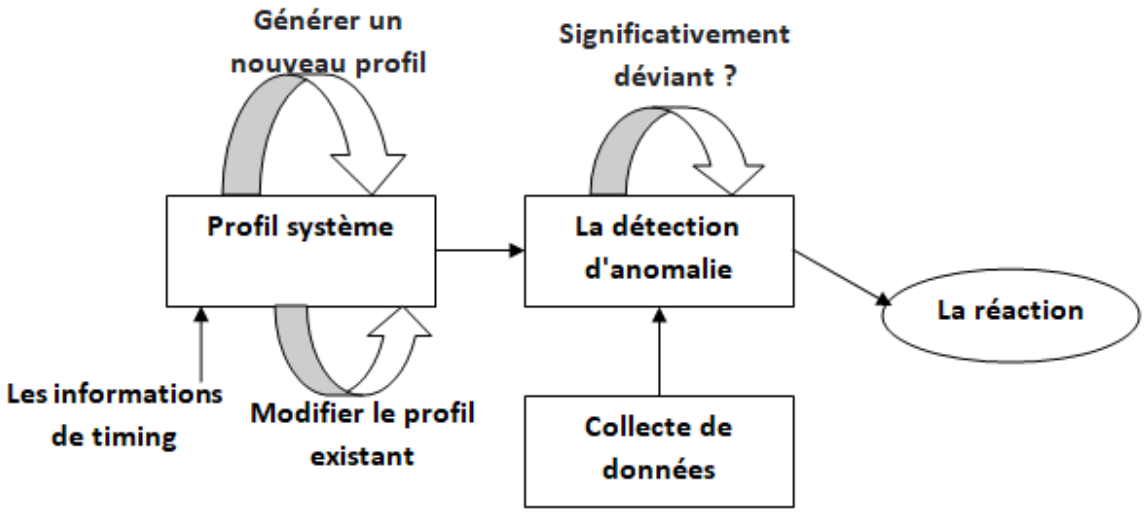

Figure 17 La détection d'anomalie

[Illy]

Le système illustré ici est constitué de quatre composantes à savoir :

- Collecte de données.
- le profil normal du système.
- la détection d'anomalie.
- la réaction.

Les activités normales du système ou les données relatives au trafic sont enregistrées par la composante de collection de données. Des techniques de modélisation spécifiques sont utilisées pour créer les profils normaux du système. La composante de détection d'anomalie détermine combien les activités en cours s'écartent des profils normaux du

système et à quel seuil d'écart ces activités devraient être signalées comme anormales. Enfin, la composante de réaction signale l'intrusion et éventuellement les informations de timing correspondantes.

L'avantage principal de la détection d'anomalie est sa capacité à trouver de nouvelles attaques. Ce qui constituera la plus grande limitation de la détection d'abus. Cependant, en raison des hypothèses sous-jacentes aux mécanismes de détection des anomalies, leurs taux de fausses alarmes sont en général très élevés. De nombreuses techniques de détection anomalie ont été proposées dans la littérature. Ces modèles vont d'autres statistiques avancés à des modèles d'intelligence artificielle et des modèles biologiques basés sur les systèmes immunitaires humains. Bien qu'il soit difficile de classer ces techniques, elles peuvent être divisées en quatre catégories sur la base des enquêtes précédentes sur les systèmes de détection d'anomalie. Il s'agit notamment de modèles statistiques avancés, de modèles fondés sur des règles, de modèles d'apprentissage, de modèles biologiques et de modèles fondés sur des techniques de traitement du signal. **[Illy]**

b. **La détection basée sur les signatures :**

La détection basée sur les signatures est aussi connue sous le nom de détection d'abus.

Cette approche vise à coder les connaissances sur les modèles de flux de données qui correspondent à des procédures intrusives sous la forme de signatures spécifiques. Ainsi, une signature est un modèle qui correspond à une menace spécifique étudiée. Les intrusions sont détectées en faisant une Correspondance entre les évènements du système et les signatures. Les correspondances trouvées sont considérées comme des intrusions. Pour illustrer la détection basée sur les signatures nous avons conçu la figure suivante. Elle est inspirée de l'illustration de la figure 18 suivante. **[Illy]**

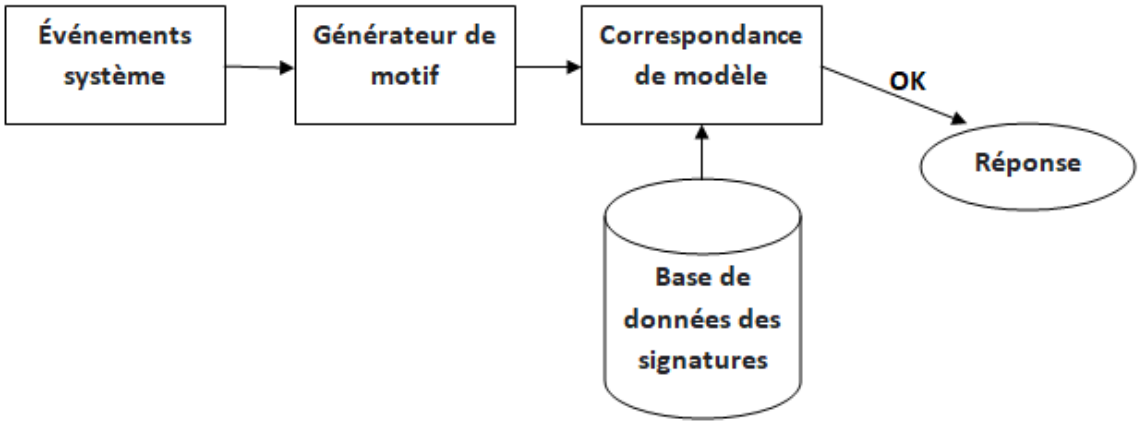

Figure 18 La détection basée sur les signatures

[Illy]

Notre modèle comprend cinq composantes :

- **Événements système** qui collecte les évènements courants du système,
- **Générateur de motif** qui génère les signatures des évènements du système à partir des évènements collectés,
- **Correspondance de modèle** qui compare les signatures générées à partir des évènements courants avec celles des attaques connues,
- **Base de données des signatures** qui est la base de données des signatures connues. La dernière composante,
- **réponse** est la réaction effectuée quand une Correspondance est positive.

Plusieurs catégories de techniques sont couramment utilisées pour mettre en œuvre la détection basée sur les signatures, à savoir la Correspondance de modèle, les techniques basées sur des règles, les techniques basées sur des états et le data mining. **[Illy]**

c. **La Détection de spécification (specification-based detection) :**

Dans les approches de détection basées sur des spécifications, les experts en sécurité prédéfinissent les comportements autorisés du système et les événements qui ne correspondent pas aux spécifications sont étiquetés comme des attaques. Au lieu d'apprendre les profils normaux du système, ici la détection est basée sur la connaissance des experts. Cette approche, en théorie, permet de détecter des attaques invisibles qui pourraient être menées. Cependant, spécifier le comportement d'un grand nombre de programmes s'exécutant dans des environnements d'exploitation réels est une tâche excessivement difficile. **[Illy]**

Tableau 2 Les Avantages et Les Inconvénients d'un IDS

[Note]

Avantages	**Inconvénients**
Pas d'impact sur le réseau (latence, gigue)	Nécessite un bon réglage pour une réaction rapide en cas d'attaque
Pas d'impact en cas de défaillance de la sonde	Nécessite une bonne politique de sécurité
Ne peut pas stopper des paquets d'initiation de connexion	Plus vulnérable aux attaques de type "flooding"
La sonde n'est pas visible pour un attaquant (en théorie)	

III. Conclusion :

Le but d'un système de détection d'intrusion (IDS) est de protéger la confidentialité, l'intégrité et la disponibilité d'un système. Les systèmes de détection d'intrusion (IDS) sont conçus pour détecter des problèmes spécifiques et sont classés en fonction des signatures (SIDS) ou des anomalies (AIDS). **[Saylor]**

Dans ce chapitre, nous avons donné une vision globale sur le système de détection d'intrusion (IDS) en présentant ses types, ses méthodes, ses avantages et ses inconvénients. Par la suite nous allons présenter l'apprentissage automatique et profond et ses algorithmes.

Chapitre 3: Deep Learning

I. Introduction

Avec la sophistication et le volume croissants des cybers attaques, l'intelligence artificielle (IA) aide les analystes des opérations de sécurité non équipés à garder une longueur d'avance sur la menace. En extrayant des renseignements sur les menaces à partir de millions d'articles de recherche, de blogs et d'articles de presse, les technologies d'IA telles que l'apprentissage automatique et réduisent considérablement les temps de réponse rapidement.

L'IA est un changeur de jeu en matière de cyber sécurité qui analyse de grandes quantités de données sur les risques pour réduire les temps de réponse et prendre en charge les opérations de sécurité à ressources limitées. **[Ibm]**

II. Machine Learning

Apprentissage automatique est une discipline scientifique, plus précisément une sous-catégorie de l'intelligence artificielle. Elle consiste à permettre à l'algorithme de détecter des "patterns", ou des patterns répétitifs, dans l'ensemble de données. Ces données comprennent des nombres, des mots, des images, des statistiques, etc. Tout ce qui peut être stocké numériquement peut être utilisé comme données d'apprentissage automatique. En reconnaissant des modèles dans ces données, l'algorithme apprend et améliore les performances lors de l'exécution de certaines tâches. En bref, les algorithmes d'apprentissage automatique apprennent de manière autonome les exécutions de tâches et les prédictions à partir des données, améliorant ainsi les performances au fil du temps. Une fois la formation terminée, l'algorithme sera capable de trouver de nouveaux modèles de données. **[Data]**

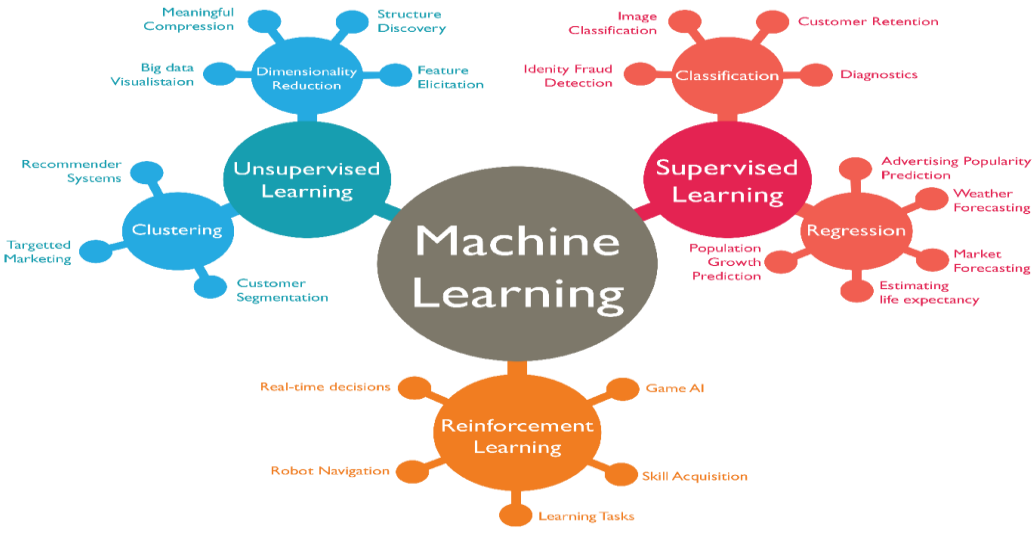

Figure 19 Machine Learning

[Medium]

1. Les Applications de Machine Learning :

L'apprentissage automatique est un mot à la mode pour la technologie d'aujourd'hui, et il se développe très rapidement de jour en jour. L'apprentissage automatique est utilisé dans la vie quotidienne même sans le savoir, comme Google Maps, l'assistant Google, Alexa, etc. les applications réelles les plus tendances de l'apprentissage automatique sont :

a. **Reconnaissance d'images**

La reconnaissance d'images est l'une des applications les plus courantes de l'apprentissage automatique. Elle est utilisée pour identifier des objets, des personnes, des lieux, des images numériques, etc. Le cas d'utilisation populaire de la reconnaissance d'image et de la détection de visage est la suggestion de marquage automatique des amis. **[Java]**

Figure 20 Reconnaissance d'image

[Actualité]

b. **Reconnaissance de la parole :**

Lors de l'utilisation de Google, nous avons une option de "Recherche vocale", cela relève de la reconnaissance vocale et c'est une application populaire de l'apprentissage automatique.

La reconnaissance vocale est un processus de conversion d'instructions vocales en texte, également connu sous le nom de "parole en texte" ou "reconnaissance vocale par ordinateur". À l'heure actuelle, les algorithmes d'apprentissage automatique sont largement utilisés par diverses applications de reconnaissance vocale. L'assistant Google, Siri, Cortana et Alexa utilisent la technologie de reconnaissance vocale pour suivre les instructions vocales. **[Java]**

Figure 21 Reconnaissance de la parole

[Paritel]

c. **Voitures autonomes :**

L'une des applications les plus intéressantes de l'apprentissage automatique est la voiture autonome. L'apprentissage automatique joue un rôle important dans les voitures autonomes. Tesla, le constructeur automobile le plus populaire, travaille sur les voitures autonomes. Utilisez des méthodes d'apprentissage non supervisées pour entraîner le modèle de voiture à reconnaître les personnes et les objets pendant la conduite. **[Java]**

Figure 22 Voiture Autonome

[Automobile]

d. **Négociation en bourse :**

L'apprentissage automatique est largement utilisé dans le trading d'actions. En bourse, les actions comportent toujours des risques de hausse et de baisse. Par conséquent, nous utilisons des réseaux de neurones à mémoire à court terme d'apprentissage automatique pour prédire les tendances du marché boursier. **[Java]**

Figure 23 Bourse

[Press]

e. **Diagnostic médical :**

La médecine utilise l'apprentissage automatique pour diagnostiquer la maladie. Grâce à cela, la technologie médicale évolue très rapidement et il est possible de construire des modèles 3D capables de prédire l'emplacement exact des lésions dans le cerveau. **[Java]**

Figure 24 Médecine

[Switzerland]

3. Le Machine Learning en cyber sécurité :

Les cybers attaques sont de plus en plus nombreux, les données à traiter également et les experts en sécurité informatique peuvent se sentir démuni d'outils pour réussir à détecter ces menaces. Le Machine Learning devient alors un atout majeur dans la détection et le traitement de ces cybers risques. En effet, un des plus grands défis des experts en cyber sécurité est d'anticiper les attaques de demain. **[Httpcs]**

4. Types de machine Learning :

a. Machine Learning supervisé :

Les algorithmes de machine Learning supervisé sont les plus couramment utilisés. Avec ce modèle, un data scientiste sert de guide et enseigne à l'algorithme les conclusions qu'il doit tirer.

- Comme exemples de machine Learning supervisé, on peut citer :

Régression :

Un problème de régression se produit lorsque la variable de sortie est une valeur réelle ou continue, telle que « salaire » ou « poids ». De nombreux modèles différents peuvent être utilisés, le plus simple est la régression linéaire. Il essaie d'ajuster les données avec le meilleur hyper-plan qui passe par les points. **[Geeks]**

Parmi ces applications: Market Forecasting, Weather Forecasting, Estimating life expectancy.

Classification :

La classification est le processus de prédiction de la classe de points de données donnés. Les classes sont parfois appelées cibles/étiquettes ou catégories. La modélisation prédictive de la classification consiste à approximer une fonction de mappage (f) des variables d'entrée (X) aux variables de sortie discrètes (y). **[Towards]**

Parmi ces applications : fraud detection, image classification, diagnostic et customer Retention.

b. Machine Learning non supervisé :

Machine Learning non supervisé utilise une approche plus indépendante dans laquelle un ordinateur apprend à identifier des processus et des schémas complexes sans un quelconque guidage humain constant et rigoureux. Le machine Learning non supervisé implique une formation basée sur des données sans étiquette ni résultat spécifique défini. **[Oracle]**

Clustering :

C'est le processus de division d'une population ou de points de données en groupes afin que les points de données du même groupe soient plus similaires aux autres points de données du même groupe et différents des points de données des autres groupes. Il s'agit

essentiellement d'une collection d'objets basée sur les similitudes et les dissemblances entre les objets.

Application de clustering: Recommender Systems, customer segmentation, targeted marketing…

Réduction de dimension :

La réduction de dimensionnalité est un ensemble de techniques réduisant le nombre de variables prédictives dans les données d'apprentissage. Ceci peut être effectué après le nettoyage et la normalisation et avant l'entraînement afin de détecter les colonnes capturant l'essence de la donnée. **[Inivivoo]**

Ces applications: Big data visualization, structure discovery, feature elicitation…

c. **L'apprentissage par renforcement :**

C'est une méthode d'apprentissage automatique basée sur la récompense des comportements souhaités et/ou la punition des comportements indésirables. En général, un agent d'apprentissage par renforcement est capable de percevoir et d'interpréter son environnement, de prendre des mesures et d'apprendre par essais et erreurs. **[Techtarget]**

Ces applications: Game AI, Learning tasks, Robot navigation.

IV. Deep Learning :

Deep Learning est une méthode qui s'appuie sur le concept de machine Learning. Cela permet à une intelligence artificielle (IA) de s'améliorer en intégrant de nouvelles règles. Leur ajout ne fait l'objet d'aucune intervention humaine. L'apprentissage profond utilise alors différentes couches neuronales qui forment un réseau artificiel. **[Journal]**

Figure 25 Deep Learning

[Ionos]

1. Les types d'algorithmes utilisés en Deep Learning :

a. Réseaux neuronaux convolutifs (CNN) :

Les CNN, également appelés ConvNets, sont constitués de plusieurs couches et sont principalement utilisés pour le traitement d'images et la détection d'objets. Les CNN sont largement utilisés pour identifier des images satellites, traiter des images médicales, prévoir des séries chronologiques et détecter des anomalies. **[Mobiskill]**

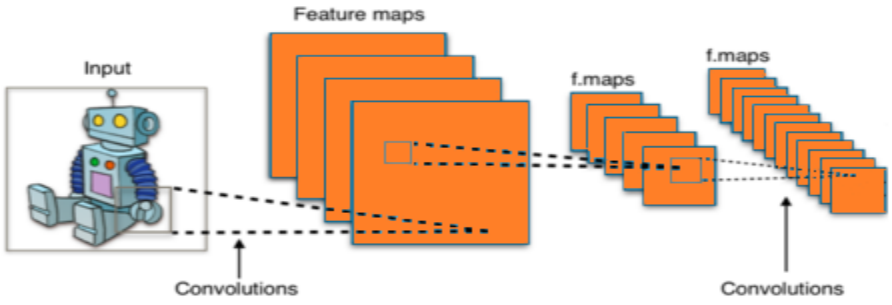

Figure 26 Réseau neuronal convolutif

[Natural]

b. Les couches du réseau de neurone convolutifs :

Le réseau neuronal convolutif est une séquence de couches, Chaque couche reçoit en entrée des données et les renvoie transformées. Pour cela, elle calcule une combinaison linéaire puis applique éventuellement une fonction non-linéaire, appelée fonction d'activation. Les coefficients de la combinaison linéaire définissent les paramètres (ou poids) de la couche. Un réseau de neurones est construit en empilant les couches : la sortie d'une couche correspond à l'entrée de la suivante. Cet empilement de couches définit la sortie finale du réseau comme le résultat d'une fonction différentiable de l'entrée.

Couche de convolution :

La couche de convolution a des noyaux (filtres) et chaque noyau à une largeur, une profondeur et une hauteur. Cette couche produit les cartes de caractéristiques à la suite du calcul du produit scalaire entre les noyaux et les régions locales de l'image.

Couche de Pooling :

L'étape de pooling est une technique de sous-échantillonnage. Généralement, une couche de pooling est insérée régulièrement entre les couches de correction et de convolution. En réduisant la taille des cartes de caractéristiques, donc le nombre de paramètres du réseau, cela accélère le temps de calcul et diminue le risque de sur-apprentissage.

Couche entièrement connectée :

Cette couche est à la fin du réseau. Elle permet la classification de l'image à partir des caractéristiques extraites par la succession de bloc de traitement. Elle est entièrement connectée, car toutes les entrées de la couche sont connectées aux neurones de sorties de celle-ci. Ils ont accès à la totalité des informations d'entrée. Chaque neurone attribue à l'image une valeur de probabilité d'appartenance à la classe i parmi les C classes possibles. Chaque probabilité est calculée à l'aide de la fonction « softmax » dans le cas où les classes sont exclusivement mutuelles. **[Sekkil]**

c. **Réseaux de mémoire à long et à court terme (LSTM) :**

Les LSTM sont des types de réseaux neuronaux récurrents (RNN) qui peuvent apprendre et mémoriser des dépendances à long terme. Se souvenir d'informations passées pendant de longues périodes est le comportement par défaut.

Les LSTM conservent les informations dans le temps. Ils sont utiles pour la prédiction de séries chronologiques car ils se souviennent des entrées précédentes. Les LSTM ont une structure en chaîne dans laquelle quatre couches en interaction communiquent de manière unique. Outre les prédictions de séries chronologiques, les LSTM sont généralement utilisés pour la reconnaissance vocale, la composition musicale et le développement pharmaceutique. **[Mobiskill]**

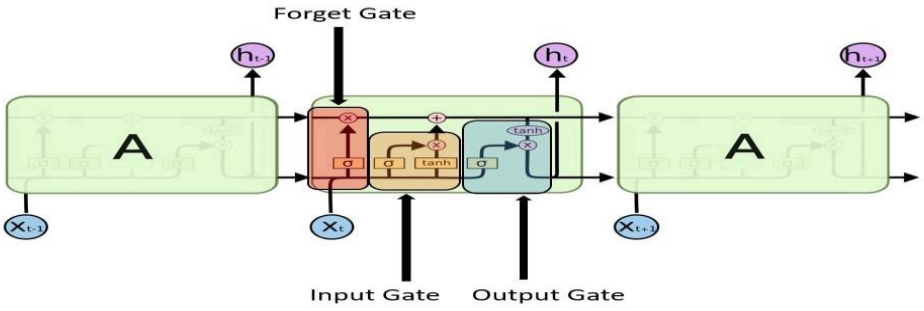

Figure 27 Réseaux de mémoire à long et à court terme

[Medium]

d. **Réseaux neuronaux récurrents (RNN) :**

Les RNN ont des connexions qui forment des cycles dirigés, ce qui permet aux sorties du LSTM d'être utilisées comme entrées dans la phase actuelle.

La sortie du LSTM devient une entrée pour la phase actuelle et peut mémoriser les entrées précédentes grâce à sa mémoire interne. Les RNN sont couramment utilisés pour le sous-titrage d'images, l'analyse de séries temporelles, le traitement du langage naturel, la reconnaissance de l'écriture manuscrite et la traduction automatique. **[Mobiskill]**

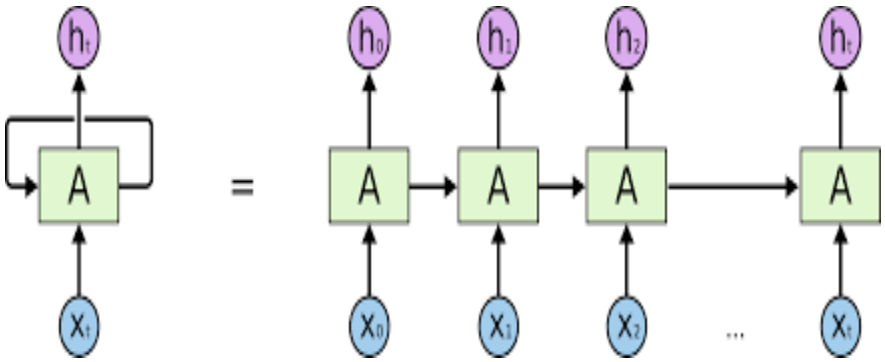

Figure 28 Réseau neuronal récurrent

[Rtavenar]

e. **Auto-Encodeurs :**

Les auto-encodeurs sont des types spécifiques de réseaux neuronaux à anticipation dans lequel l'entrée et la sortie sont identiques. Geoffrey Hinton a conçu les auto-encodeurs dans les années 1980 pour résoudre les problèmes d'apprentissage non supervisé (unsupervised Learning). Il s'agit de réseaux neuronaux formés qui reproduisent les données de la couche d'entrée à la couche de sortie. Les auto-encodeurs sont utilisés à des fins telles que la découverte de produits pharmaceutiques, la prédiction de popularité et le traitement d'images.

Un auto-encodeur se compose de trois éléments principaux : le codeur, le code et le décodeur. **[Mobiskill]**

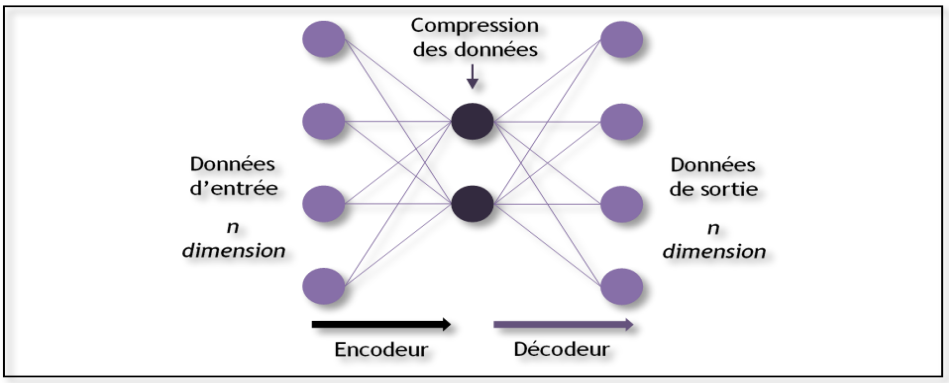

Figure 29 Auto-encodeur

[Big]

Tableau 3 Avantages et Inconvénients des réseaux de neurones

Type	Avantages	Inconvénients
CNN	1. L'exploitation de connaissances empiriques. 2. La robustesse. 3. La dégradation progressive. 3. Le parallélisme massif. **[Adel]**	1. La difficulté de choix de l'architecture et des paramètres. Le problème d'initialisation et de codage. 2. Le manque d'explicabilité. **[Adel]**
LSTM	1. capables de modéliser les dépendances séquentielles à long terme. 2. Ils sont plus robustes au problème de la mémoire courte que les RNN «vanille». **[Science]**	1. Ils augmentent la complexité de calcul par rapport au RNN avec l'introduction de plus de paramètres à apprendre. 2. La mémoire requise est supérieure à celle des RNN « Vanille» en raison de la présence de plusieurs cellules de mémoire. **[Science]**
RNN	1. Capacité à modéliser et prédire à partir de séquences de longueurs variables. 2. Exécution rapide des modèles. 3.Utilisation modérée des ressources **[Blog]**	1. Disparition/Explosion des gradients de la fonction de coût (erreur). 2. Impossibilité de réaliser le traitement de séquences de très grandes tailles. Temps d'entraînement extrêmement long. **[Blog]**
AUTO ENCODEUR	1. Réduire la dimensionnalité des données que nous utilisons 2. la compacité et la rapidité du codage par rétro propagation. **[Researchgate]**	1. Avant de commencer à créer le modèle réel, faites beaucoup de données, de temps de traitement, d'ajustements d'hyper paramètres et de validation de modèle. **[Researchgate]**

2. Types de fonctions d'activation :

Plusieurs types de fonctions d'activation sont utilisés dans Deep Learning. Certains d'entre eux sont expliqués ci-dessous :

a. **Fonction d'étape :**

La fonction d'étape est l'un des types de fonctions d'activation les plus simples. Dans cela, nous considérons une valeur seuil et si la valeur de l'entrée nette disons y est supérieure au seuil alors le neurone est activé. **[Fr]**

$$\boldsymbol{f(x) = 1, if x \geq 0} \quad ; \boldsymbol{f(x) = 0, if x < 0}$$

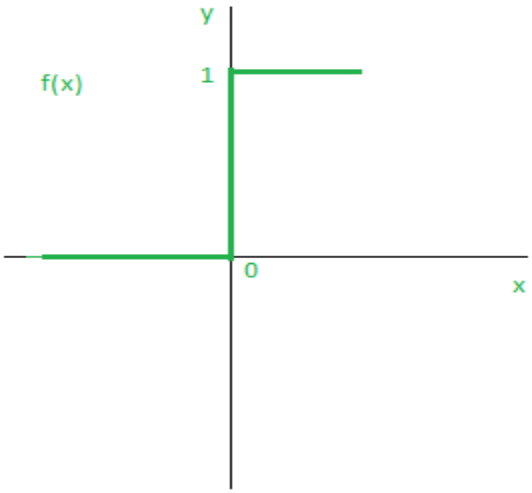

Figure 30 Graphe de la fonction d'étape

[Fr]

b. **Fonction sigmoïde** :

La fonction sigmoïde est une fonction d'activation largement utilisée. Il est défini comme

$$f(x) = \frac{1}{1+e^{-x}} \textbf{[Fr]}$$

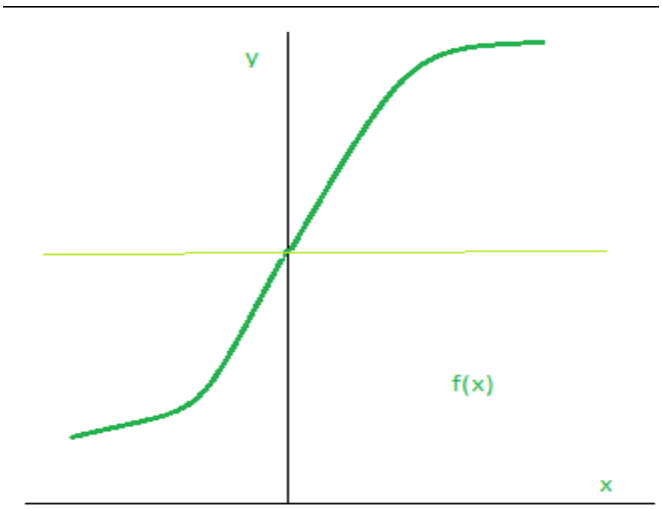

Figure 31 Graphe de la fonction sigmoïd

[Fr]

c. **ReLU:**

La fonction ReLU est l'unité linéaire rectifiée. C'est la fonction d'activation la plus utilisée. Il est défini comme :$f(x) = \max(0, x)$**[Fr]**

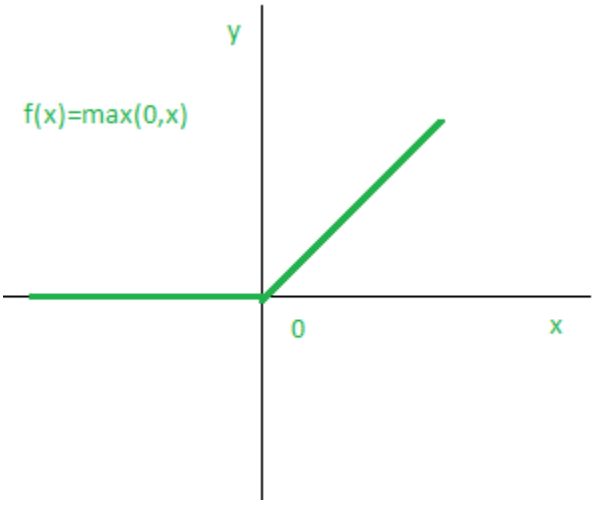

Figure 32 Graphe de la fonction ReLU

[Fr]

Le principal avantage de l'utilisation de la fonction ReLU par rapport aux autres fonctions d'activation est qu'elle n'active pas tous les neurones en même temps.

d. **LeakyReLU :**

La fonction LeakyReLU n'est rien d'autre qu'une version améliorée de la fonction ReLU. Au lieu de définir la fonction Relu comme 0 pour x inférieur à 0, nous la définissons comme un petit composant linéaire de x. Il peut être défini comme:

$$f(x) = ax\ , x < 0 f(x) = x\ otherwise$$**[Fr]**

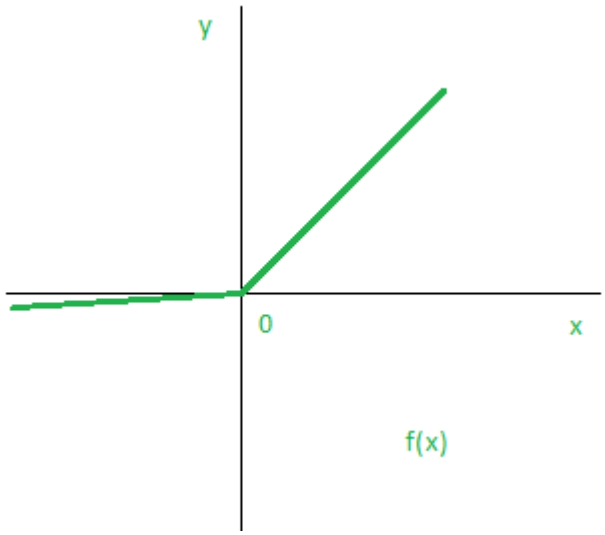

Figure 33 Graphe de la fonction LeakyReLU

[Fr]

V. Deep Learning vs machine Learning:

Comme évoqué précédemment, le Deep Learning est issu de la machine Learning. Les deux approches s'appuient sur le principe de l'apprentissage automatique. On distingue néanmoins des différences notables. La machine Learning se sert de la technique de feature extraction pour effectuer des fonctions prédictives. Ce qui n'est pas le cas du Deep Learning. Pour ce dernier élément, l'algorithme traite des données brutes.

Pour la machine Learning, les valeurs et les variables sont sélectionnées à l'avance. Cela permet d'exploiter uniquement des données tabulaires. Contrairement au Deep Learning qui inclut aussi des systèmes de vision par ordinateur et de langages naturels.
La modélisation des résultats est également sujette à variation.

Il s'agit d'un modèle statistique pour le machine Learning et une optimisation numérique pour le Deep Learning. Enfin, le matériel diffère dans les processus d'application. Il est préférable de

s'appuyer sur l'architecture d'un processeur central (CPU) pour la machine Learning. Le Deep Learning est plus adapté au processeur graphique (GPU). **[Journal]**

VI. Conclusion :

A cette partie, nous avons introduit les concepts de base de l'apprentissage automatique avec ses types et ses applications. Et nous avons détaillé l'apprentissage profond avec ses types et leurs avantages et inconvénients, et à la fin nous avons présenté les fonctions d'activations.

Dans le chapitre suivant nous allons utiliser réseaux neuronaux profonds qu'ils seront intérêt de notre étude.

Chapitre 4: Approche Proposée et Implémentation

Approche Proposée et Implémentation

I. Introduction :

En Informatique l'implémentation désigne la réalisation ou la mise en œuvre de travail demandé, Donc l'objectif de ce chapitre est de présenter des outils, des techniques et des langages utilisés dans notre travail.

II. Matériel :

Le matériel réalisé est PC personnel *hp* I3 avec une capacité mémoire de 4GB, CPU i3-5010U cadencé à 2.10 GHz, tournant sous Windows 10 64 bits

III. Présentation des outils :

1. Sqlmap :

Sqlmap est un outil de test de pénétration open source qui automatise le processus de détection et d'exploitation des failles d'injection SQL.

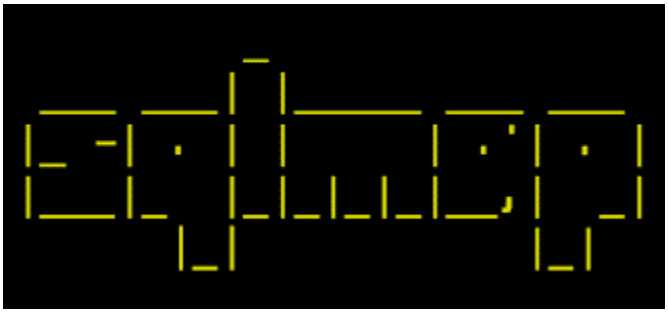

Figure 34 Sqlmap

[Kali]

Il possède un puissant moteur de détection. Parmi ces fonctionnalités :

- Permettre la prise d'empreintes de la base de données.
- La récupération des données de la base de données.
- L'accès au système de fichiers sous-jacent et à l'exécution de commandes sur le système d'exploitation. **[Kali]**

3. Python :

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. Il a libéré les développeurs des contraintes de formes qui occupaient leur temps avec les langages plus anciens. **[Net]**

Figure 35 Python

[Odoo]

Python a d'énorme caractéristique intéressante :

- Il est multiplateforme : fonctionne sur des nombreux systèmes d'exploitation ;
- Il est gratuit.
- C'est un langage interprété : le script python est directement exécuté, il n'a pas besoin d'être compilé avant d'être exécuté.
- C'est un langage orienté objet : on peut créer des programmes qui imite le comportement du monde réel.
- Enfin, il est utilisé en bioinformatique et plus couramment en analyse de données.

Ces caractéristiques faisaient de ce langage un des langages les plus prisés des développeurs, que ce soit dans le domaine de la Data Science ou de la programmation web. **[Transit]**

4. Anaconda :

Anaconda est un outil en distribution libre et open source destinée à la programmation Python et R. Il est véritablement utilisé en science de données, machine Learning et l'intelligence artificielle car il contient plusieurs packages nécessaires dans ce domaine notamment Python, Numpy, Panda, Jupyter, etc. Et comme le langage Python, il est multiplateforme. **[Transit]**

Figure 36 Anaconda

[Anaconda]

5. Jupyter Notebook :

Jupyter Notebook est une plateforme Web open source qui facilite la création et le partage des documents qui contiennent du code. Jupiter Notebook permet de gérer des fichiers, des modules et aussi la présentation du travail. **[Transit]**

Figure 37 Jupyter

[Amis]

IV. Premier Essai d'une attaque :

Nous avons essayé plusieurs tests de vulnérabilité sur différent sites, la première commande analyse le système distant pour voir s'il est vulnérable à l'injection SQL.

```
C:\sqlmap>python sqlmap.py -u http://testphp.vulnweb.com --crawl=2
        ___
       __H__
 ___ ___[,]_____ ___ ___  {1.5.11.5#dev}
|_ -| . [)]     | .'| . |
|___|_  [,]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is
tate and federal laws. Developers assume no liability and are not responsible for any misus

[*] starting @ 17:04:47 /2022-01-24/
```

Figure 38 Test de vulnérabité

Quand le site est vulnérable donc il nous rendre un fichier .csv

Figure 39 le résultat du test

Le fichier csv est comme suit :

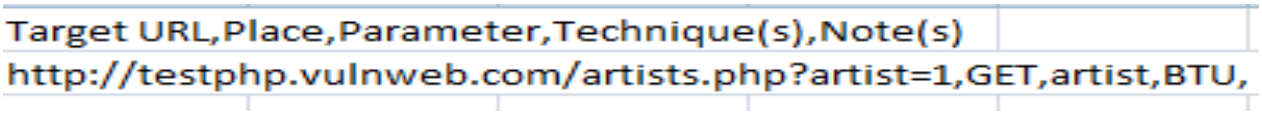

Figure 40 l'emplacement du fichier csv

Plusieurs fichiers csv nous forme un dataset et puisque nous n'avons pas pu a collecté un grand nombre des fichiers csv donc nous avons travaillé avec deux dataset spécialisé dans les injections SQL.

V. Présentation de l'approche proposée :

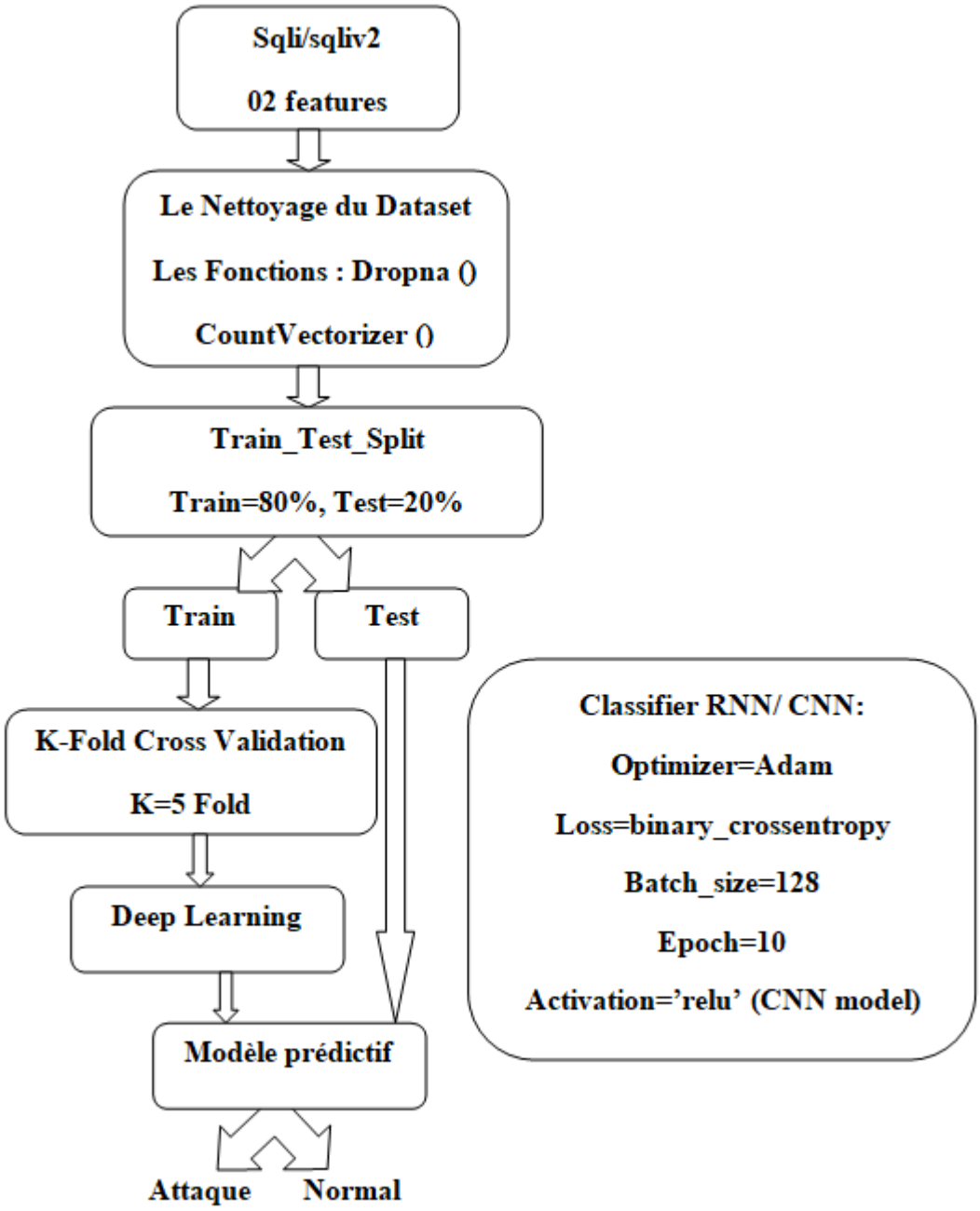

Figure 41 Schéma de la méthode de conception

Approche Proposée et Implémentation

1. Importation des Libraires :

a. **Pandas :**

Pandas est une librairie python qui permet de manipuler facilement des données à analyser :

- Manipuler des tableaux de données avec des étiquettes de variables (colonnes) et d'individus (lignes).
- On peut facilement lire et écrire ces DataFrames à partir ou vers un fichier tabulé. **[Pandas]**

b. **Numpy :**

Le terme Numpy est en fait l'abréviation de **" Numerical Python "**. Il s'agit d'une bibliothèque Open Source en langage Python. On utilise cet outil pour la programmation scientifique en Python, et notamment pour la programmation en Data Science, pour l'ingénierie, les mathématiques ou la science. **[Numpy]**

c. **Scikit-learn :**

Scikit-learn est une bibliothèque clé pour le langage de programmation Python qui est généralement utilisé dans les projets d'apprentissage automatique. **[Scikit]**

d. **Keras :**

Keras est bibliothèque open source de prototypage rapide de modèles de deeplearning. A la portée des débutants en IA, elle s'articule autour d'une API de haut niveau supportant différentes librairies de réseaux de neurones artificiels récurrents ou convolutifs, comme Tensorflow, Microsoft Cognitive Toolkit, PlaidML ou Theano. **[Keras]**

```
import numpy as np
import pandas as pd
```

```
import glob
import time
import pandas as pd
# from xml.dom import minidom
from nltk import ngrams
from nltk.tokenize import sent_tokenize
import nltk
nltk.download('punkt')
nltk.download('stopwords')
nltk.download('wordnet')
from nltk.stem import PorterStemmer
from nltk.stem import PorterStemmer
from nltk.tokenize import sent_tokenize, word_tokenize
from nltk.stem import WordNetLemmatizer
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from sklearn.model_selection import StratifiedKFold
```

Figure 42 Importation des Libraires

e. **Matplotib :**

Matplotlib est une bibliothèque complète pour créer des visualisations statiques, animées et interactives en Python. Matplotlib rend les choses faciles faciles et les choses difficiles possibles.**[Matplo]**

```
import matplotlib.pyplot as plt
```

Figure 43 Importation du Libraire Matplotib

2. Prétraitement du Dataset :

Pour le commencement du travail il faudra charger les données donc on utilise la fonction **read_csv** via la libraire pandas qui import les données des fichiers csv.

```
import pandas as pd
df = pd.read_csv("sqliv2.csv")
df
```

```
import pandas as pd
df = pd.read_csv("sqli.csv")
df
```

Figure 44 Importation des Datasets

Au moment de l'utilisation du Dataset « **sqliv2** » on a remarqué quelle contient des données obsolète qui va l'abimer, Donc on a supprimé les lignes qui ne contient aucune Sentence.

```
df=df.dropna(subset=['Sentence'])
df
```

Figure 45 la suppression des lignes vides

Après, on a remplacé les valeurs infinies par Nan et les éliminer **(sqliv2)**

```
df = df.replace([np.inf, -np.inf], np.nan)
df = df.dropna()
df = df.reset_index()
```

Figure 46 la suppression de valeurs nulles

On a utilisela fonction **Count_Vectorizer**de la libraire **Scikit_Learn**qui va convertir la colonne Sentence des deux Dataseten une matrice de vecteur.

```
from sklearn.feature_extraction.text import CountVectorizer
vectorizer = CountVectorizer( min_df=2, max_df=0.7, stop_words=stopwords.words('english'))
posts = vectorizer.fit_transform(df['Sentence'].values.astype('U')).toarray()
```

Figure 47 Conversion des Sentences

On a créé un Dataframe pour les vecteurs de la colonne Sentence et le nommer « **transformed_posts ».**

```
transformed_posts=pd.DataFrame(posts)
```

Figure 48 création du Dataframe

Ensuite, on a concaténer notre Dataset avec « **transformed_posts ».**

```
df=pd.concat([df,transformed_posts],axis=1)
```

Figure 49 la concaténation des deux Dataframe

Par la suite on a définit nos « inputs » et « outputs»

```
X=df[df.columns[2:]]
```

```
y=df['Label']
```

Figure 50 définition des inputs et outputs

3. La division du Dataset :

On a utilisé la fonction **train_test_split** de la bibliothèque scikit_learn pour la dévision de notre Dataset en deux parties train 80 % et test 20%.

```
from sklearn.model_selection import train_test_split
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Figure 51 split du dataset

Après le **train_test_split** nous avons appliqué la méthode **KFold** sur nos inputs train qui va les deviser en 5 folds.

```
from sklearn.model_selection import KFold
kf = KFold(n_splits=5, random_state=None, shuffle=False)
kf.split(X)
```

```
<generator object _BaseKFold.split at 0x7f75a68d2890>
```

Figure 52 KFold Cross Validation

4. Création du model CNN :

Nous avons utilisé quatre couche**Dense ()** entièrement connectées pour chaque Dataset avec une fonction d'activation **relu** sur les trois couches premiers et la fonction d'activation **sigmoïd**sur la couche de sortie et on a ajouté des fonctions **dropout (0.5)** pour éliminer ½ des nœuds et la fonction **BatchNormalization ()** pour la normalisation **des outputs.**

a. **Première Expérience pour datasetSqliv2 :**

```
input_dim = X_train.shape[1]  # Number of features
model = Sequential()
model.add(layers.Dense(33757 , input_dim=input_dim, activation='relu'))
model.add(layers.Dense(1024,  activation='relu'))
model.add(layers.Dense(1024, activation='relu'))

model.add(layers.BatchNormalization())
model.add(layers.Dropout(0.5))
model.add(layers.Dense(1, activation='sigmoid'))
```

Figure 53modèle CNN pour sqliv2

b. **Deuxième expérience pour dataset sqli :**

```
input_dim = X_train.shape[1]  # Number of features
model = Sequential()
model.add(layers.Dense(4200, input_dim=input_dim, activation='relu'))
model.add(layers.Dense(1024,  activation='relu'))
model.add(layers.Dense(1024, activation='relu'))

model.add(layers.BatchNormalization())
model.add(layers.Dropout(0.5))
model.add(layers.Dense(1, activation='sigmoid'))
```

Figure 54 modèle CNN pour sqli

5. Création du modèle RNN :

Nous avons utilisé quatre couches **« Embedding, SimpleRNN, Dense »**entièrement connectées pour chaque Dataset avec une fonction d'activation **sigmoïd**sur la couche de sortie et on a ajouté des fonctions **dropout (0.5)** pour éliminer ½ des nœuds et la fonction **BatchNormalization ()** pour la normalisation **des outputs.**

a. **Première expérience pour sqliv2 :**

```
model = Sequential()
model.add(layers.Embedding(input_dim=33757, output_dim=64))
model.add(SimpleRNN(1024))
model.add(layers.Dense(1024))
model.add(layers.BatchNormalization())
model.add(layers.Dropout(0.5))
model.add(Dense(1))
model.add(Activation('sigmoid'))
```

Figure 55modèle RNN pour sqliv2

b. **Deuxième expérience pour sqli :**

```
model = Sequential()
model.add(layers.Embedding(input_dim=4200, output_dim=64))
model.add(SimpleRNN(1024))
model.add(layers.Dense(1024))
model.add(layers.BatchNormalization())
model.add(layers.Dropout(0.5))
model.add(Dense(1))
model.add(Activation('sigmoid'))
```

Figure 56 modèle RNN pour sqli

6. Troisième expérience pour LSTM :

Pour l'expérience LSTM nous avons rencontre un problème c'est que le pc plante, Donc on a suffit notre approche pour le CNN et la RNN.

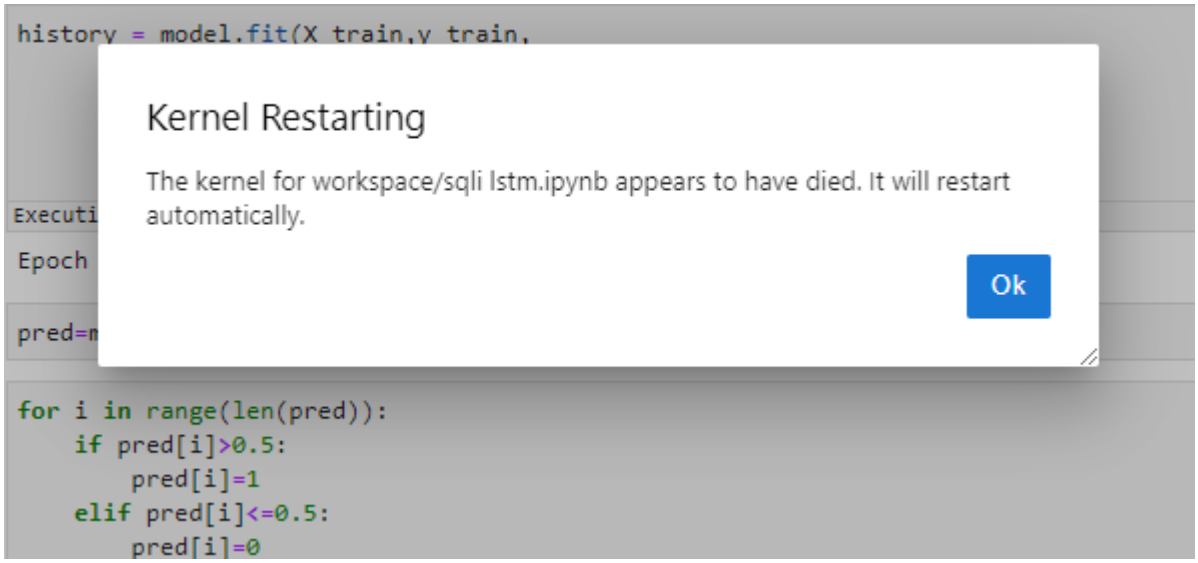

Figure 57 Expérience LSTM

7. Compilation des modèles :

Après la création de nos modèles donc il est nécessaire de les compiler avec le fonction **compile()**. La fonction de perte est spécifiée pour avoir le type **binary_crossentropy,**
Le paramètremetrics est défini sur **accuracy**et enfin nous avons utilisé l'optimiseur **Adam** pour former les réseaux.

```
model.compile(loss='binary_crossentropy',
              optimizer='adam',
              metrics=['accuracy'])
model.summary()
```

Figure 58 la compilation des modèles

8. L'entrainement des modèles :

L'entrainement des modèles se fait en un seul appel de méthode appelé**fit()** qui prend peu de paramètre comme on le voit dans le code ci-dessous.

```
history = model.fit(X_train,y_train,
                    epochs=10,
                    verbose=True,
                    validation_data=(X_test, y_test),
                    batch_size=128)
```

Figure 59 l'entrainement des modèles

9. La Prédiction sur la partie Test:

Dans le domaine de **DataScience**et après le training sur la partie train, Donc il est nécessaire de **prédire** les valeurs sur la partie **Test.**

```
pred=model.predict(X_test)
```

```
for i in range(len(pred)):
    if pred[i]>0.5:
        pred[i]=1
    elif pred[i]<=0.5:
        pred[i]=0
```

Figure 60 Prédiction de la partie Test

10.Evaluation des modèles :

L'évaluation des modèles est faite pour savoir ces performances à l'aide de la **confusion_matrix.**

Tableau 4 Tableau des résultats d'évaluation

Modèles	Accuracy	Precision	Recall
CNN 'sqli'	97.97%	94.33%	99.20%
CNN 'sqliv2'	100%	100%	100%
RNN 'sqli'	30%	30%	100%
RNN 'sqliv2'	33.91%	33.83%	100%

a. **Les Graphes d'accuracy et de perte :**

On a utilisé l'historique enregistré '**history**' pendant notre training pour obtenir un graphique des mesure de précision a l'aide de la fonction 'plot' qui trace la précision sur chaque epoch.

Pour le modèle CNN ‘sqli ‘ :

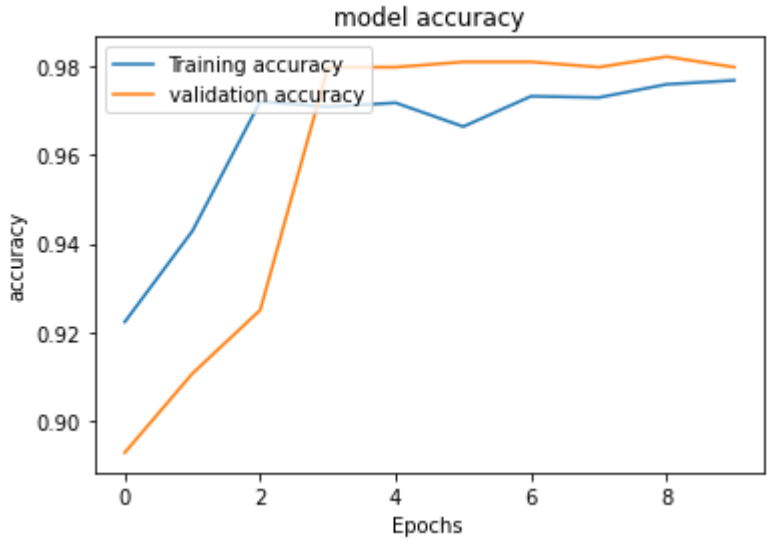

Figure 61 Graphe de la fonction accuracy "CNN sqli"

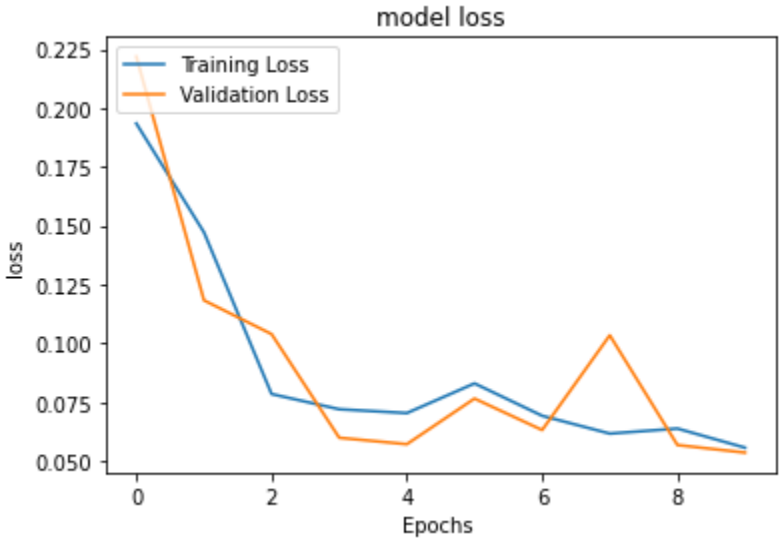

Figure 62 Graphe de la fonction loss "CNN sqli"

Pour le modèle CNN ‘sqliv2 ‘ :

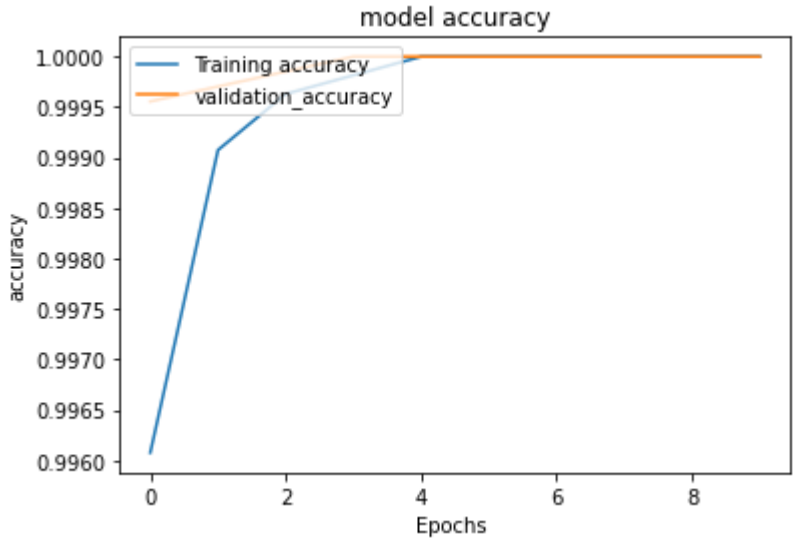

Figure 63 Graphe de la fonction accuracy "CNN sqliv2"

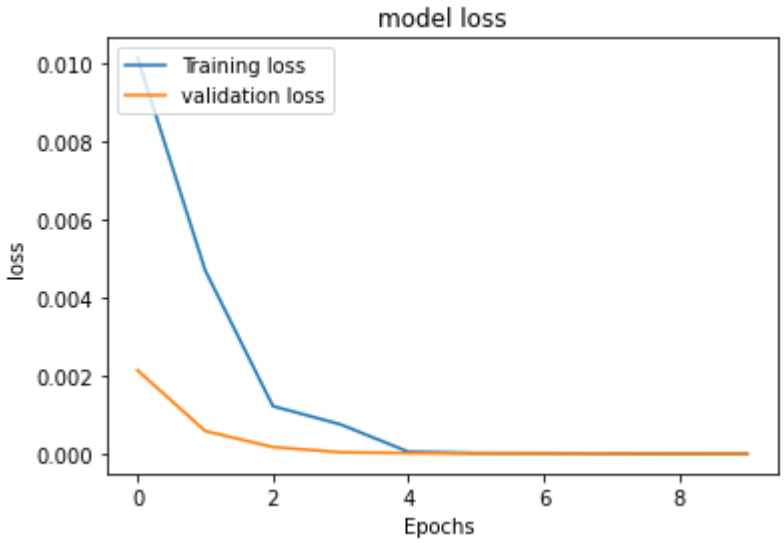

Figure 64 Graphe de la fonction loss "CNN sqliv2"

Pour le modèle RNN ‘sqli ‘ :

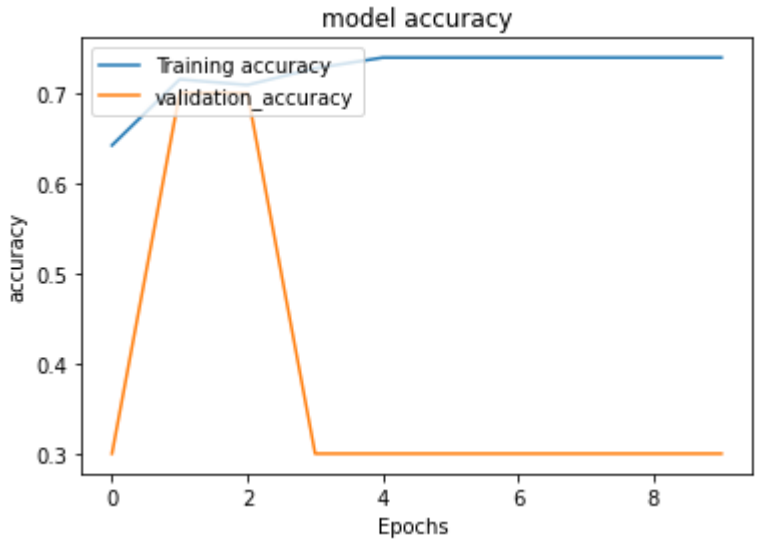

Figure 65 Graphe de la fonction accuracy "RNN sqli"

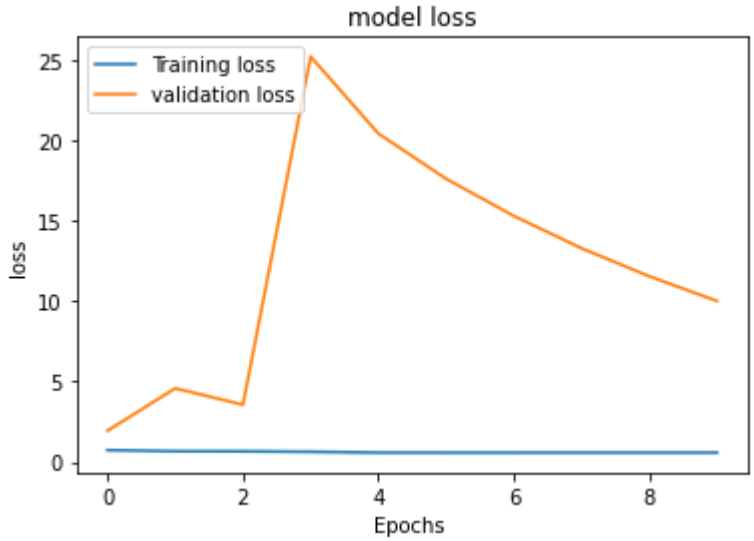

Figure 66 Graphe de la fonction loss "RNN sqli"

Pour le modèle RNN ‘sqliv2 ‘ :

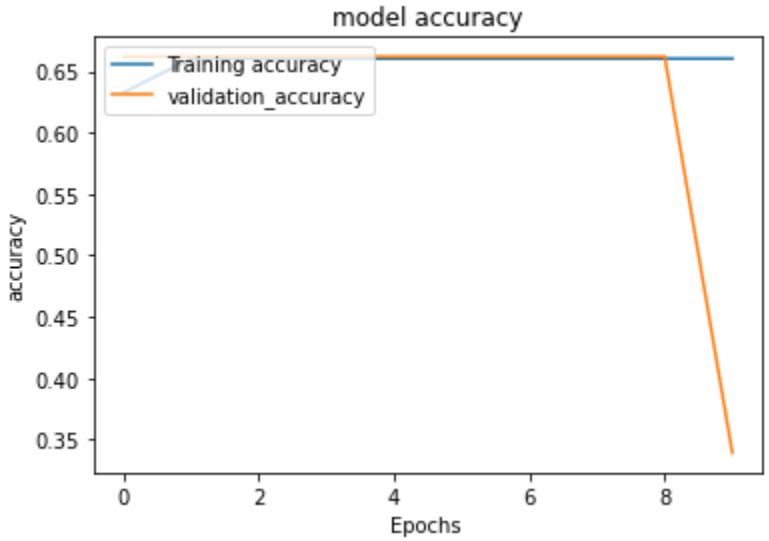

Figure 67 Graphe de la fonction accuracy "RNN sqliv2"

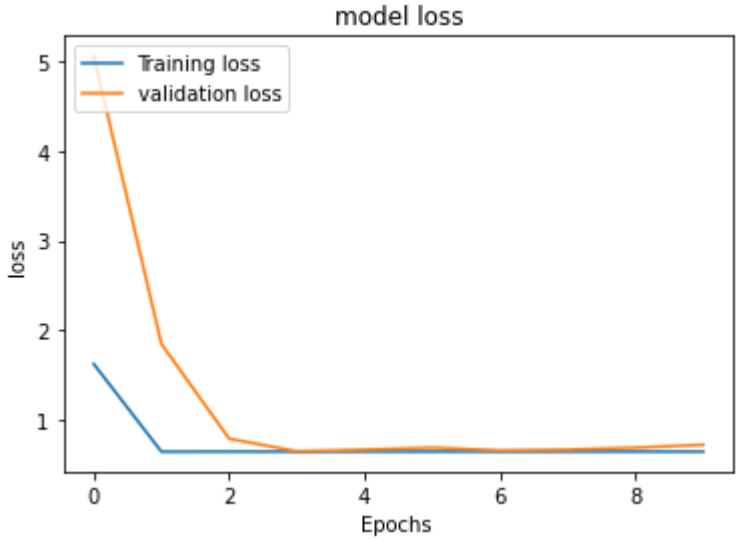

Figure 68 Graphe de la fonction loss "RNN sqliv2"

b. **Matrice de confusion :**

La matrice de confusion est en quelque sorte un résumé des résultats de prédiction pour un problème particulier de classification. Elle compare les données réelles pour une variable cible à celles prédites par un modèle. **[Jedha]**

Tableau 5 La Matrice de confusion

	Predicted : NO	**Predicted : YES**
Actual : NO	**TN**	**FP**
Actual : YES	**FN**	**TP**

- **Définition des termes** :
  Vrai positif (TP) : l'observation est positive et devrait être positive.
  Faux négatif (FN) : l'observation est positive mais prédite négative.
  Vrai négatif (TN) : l'observation est négative et devrait être négative.
  Faux positif (FP) : l'observation est négative mais prédite positive.

- **Ces métriques :**

Accuracy : Ce paramètre fait la somme de tous les vrais positifs et vrais négatifs qu'il divise par le nombre total d'instances.

$$\boldsymbol{Accuracy = \frac{TP + TN}{TP + TN + FP + FN}}$$

La précision : La précision indique le rapport entre les prévisions positives correctes et le nombre total de prévisions positives**.**

$$\boldsymbol{Précision = \frac{TP}{TP + FP}}$$

Le rappel : Le rappel (ou recall) est un paramètre qui permet de mesurer le nombre de prévisions positives correctes sur le nombre total de données positives**.**

$$\boldsymbol{Recall = \frac{TP}{TP + FN}}$$

Le score F1 : Le score F1 (ou F-measure) est une moyenne harmonique de la précision et du rappel**.** Il équivaut au double du produit de ces deux paramètres sur leur somme. Sa valeur est maximale lorsque le rappel et la précision sont équivalents. **[Jedha]**

$$\boldsymbol{Score\ F1 = 2 * \frac{Precession * recall}{precession + recall}}$$

Le support : est le nombre d'occurrences de chaque classe dans y_true.

Pour le modèle CNN 'sqli ' :

```
from sklearn.metrics import confusion_matrix
confusion_matrix(y_test, pred)
```

```
array([[573,  15],
       [  2, 250]])
```

```
from sklearn.metrics import classification_report
print(classification_report(y_test, pred))
```

```
              precision    recall  f1-score   support

           0       1.00      0.97      0.99       588
           1       0.94      0.99      0.97       252

    accuracy                           0.98       840
   macro avg       0.97      0.98      0.98       840
weighted avg       0.98      0.98      0.98       840
```

Figure 69 Matrice de confusion 'CNN sqli'

Pour le modèle CNN 'sqliv2 ' :

```
from sklearn.metrics import confusion_matrix
confusion_matrix(y_test, pred)
```

```
array([[4470,    0],
       [   0, 2282]])
```

```
from sklearn.metrics import classification_report
print(classification_report(y_test, pred))
```

```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      4470
           1       1.00      1.00      1.00      2282

    accuracy                           1.00      6752
   macro avg       1.00      1.00      1.00      6752
weighted avg       1.00      1.00      1.00      6752
```

Figure 70 Matrice de confusion 'CNN sqliv2'

Pour le modèle RNN 'sqli ' :

```
from sklearn.metrics import confusion_matrix
confusion_matrix(y_test, pred)
```

```
array([[  0, 588],
       [  0, 252]])
```

```
from sklearn.metrics import classification_report
print(classification_report(y_test, pred))
```

```
              precision    recall  f1-score   support

           0       0.00      0.00      0.00       588
           1       0.30      1.00      0.46       252

    accuracy                           0.30       840
   macro avg       0.15      0.50      0.23       840
weighted avg       0.09      0.30      0.14       840
```

Figure 71 Matrice de confusion 'RNN sqli'

Pour le modèle RNN 'sqliv2 ' :

```
from sklearn.metrics import confusion_matrix
confusion_matrix(y_test, pred)
```

```
array([[   8, 4462],
       [   0, 2282]])
```

```
from sklearn.metrics import classification_report
print(classification_report(y_test, pred))
```

```
              precision    recall  f1-score   support

           0       1.00      0.00      0.00      4470
           1       0.34      1.00      0.51      2282

    accuracy                           0.34      6752
   macro avg       0.67      0.50      0.25      6752
weighted avg       0.78      0.34      0.17      6752
```

Figure 72 Matrice de confusion 'RNN sqliv2'

Discussion :

- Concernant le modèle CNN on remarque pour les deux graphes : la précision (accuracy) dans la validation est plus élevée que dans l'entrainement et que la perte (loss) dans la validation est plus basse que dans l'entrainement ce qui est un bon signe. Par contre nos graphes ne sont pas stables a cause des époques qu'on a faites, pour cela on doit augmenter le nombre d'époques mais nous n'avons pas la capacité du matériel nécessaire pour le faire.
- Concernant le modèle RNN on remarque qu'on a un problème de sous ajustement et sur ajustement qui est un pire ennemie du data scientiste. Il survient lorsque le modèle essaye de trop s'adapter aux données d'entrainement.

VI. Conclusion :

Dans ce chapitre on a essayé de faire une explication de notre approche et donner les outils nécessaires et les bibliothèques. A la fin nous avons présenté les résultats des expériences effectuées et nous avons conclu à l'aide des résultats et des graphes que le modèle CNN est le meilleur.

VII. Conclusion générale :

Dans le cadre de ce projet, nous avons mis en œuvre une solution pour détecter et catégoriser les attaques SQL injection afin d'améliorer le taux de détection. Pour cette raison, nous avons proposé une solution basée sur les techniques de deep Learning. Cela a été fait par l'étude des caractéristiques de SQL injection et leur relation avec la structure sous-jacente des systèmes Web. Nous avons également discuté du traitement SQL et identifié l'une des parties les plus vulnérables du traitement SQL. Ainsi aux technologies lies aux IDS, par la découverte un de leur architecture, ses types et méthode de détection. Ensuite, nous avons touche les techniques de Deep Learning afin d'améliorer le taux de détection et d'éviter les sur-ajustements et les sous-ajustements découvert par l'application de techniques de machine Learning. A la fin, nous avons présenté une approche proposée à travers a un ensemble des étapes.

Plusieurs expériences ont été effectuées, en utilisant différents modèles de Deep Learning, où elles ont été appliquées sur deux data-set (SQLI, SQLIV2) spécialisées pour les attaques d'injection SQL. Ces expériences ont permis de conclure que les CNN ont donné des bons résultats sur les deux data-set avec la même approche proposée. En général, les résultats expérimentaux de chaque étude du CNN ont un taux de détection élevé pour SQL-Injection. L'apprentissage en profondeur a un grand potentiel dans la détection des renseignements sur les menaces

VIII. Référence :

[Cacciapaglia ,2018] Cacciapaglia, K. (2018). *Analyse sur les différentes cyberattaques informatiques* (Doctoral dissertation, Haute école de gestion de Genève).

[Hakim,2018] Hakim, M. A., Amina, B. A. L. I., &Manele, A. H. MéMoire de fin d'études.

[Software] https://softwarelab.org/fr/injection-sql/

[One] https://www.one.com/fr/securite-de-site-web/injection-sql-definition

[Network] Network-Based Attacks | Introduction to Network Security (flylib.com)

[Logpoint] https://www.logpoint.com/fr/blog/lutter-cyberattaques

[Cisco] https://www.cisco.com/c/fr_ca/products/security/common-cyberattacks.html#~types-de-cyberattaques

[Cyber] https://www.axiomeassocies.fr/cyberattaque/

[Analyse] https://analyse-innovation-solution.fr/publication/fr/hacking/injection-sql-sqli-dorks

[Geekflare] https://geekflare.com/fr/ids-vs-ips-network-security-solutions/dernier accès /06/05

[Web] https://web.maths.unsw.edu.au/~lafaye/CCM/detection/ids.htm dernier accès /12/02

[Varonis] https://www.varonis.com/fr/blog/ids-et-ips-en-quoi-sont-ils-differents dernier accès /12/02

[Illy] Illy, Poulmanogo. (2018). Les systèmes de détection d'intrusion (IDS). 10.13140/RG.2.2.10055.04001. dernier accès /12/02

[Korcak, 2014] Korcak, Michal & Lamer, Jaroslav & Jakab, Frantisek. (2014). Intrusion Prevention/Intrusion Detection System (IPS/IDS) for Wifi Networks. International journal of Computer Networks & Communications. 6. 77-89. 10.5121/ijcnc.2014.6407.

[Thom] https://thomasory.com/r%C3%A9seau/s%C3%A9curit%C3%A9/intrusion/what-is-an-ips-and-ids/

[Saylor] https://learn.saylor.org/mod/book/view.php?id=29755&chapterid=5455

[Data] https://datascientest.com/machine-learning-tout-savoir dernier accès 14/02

[Oracle] https://www.oracle.com/dz/data-science/machine-learning/what-is-machine-learning/ dernier accès 16/02

[Journal] https://www.journaldunet.fr/web-tech/guide-de-l-intelligence-artificielle/1501333-deep-learning-definition-et-principes-de-l-apprentissage-profond/#:~:text=Le%20deep%20learning%20ou%20apprentissage,le%20terme%20de%20machine%20learning.dernier accès 14/02

[Mobiskill] https://mobiskill.fr/blog/conseils-emploi-tech/quels-sont-les-algorithmes-de-deep-learning/ dernier accès 15/02
[Techtarget] https://www.techtarget.com/ dernier accès 14/02
[Geeks] https://www.geeksforgeeks.org/regression-classification-supervised-machine-learning/dernier accès 14/02
[Ibm] https://www.ibm.com/fr-fr/security/artificial-intelligencedernier accès 14/02
[Adel] Gridi Adel, 2019 -2020, Un Outil de Deep Learning pour les données textuelles, UNIVERSITE L'ARBI BEN M'HIDI-OUM EL BOUAGHI.
[Blog] https://blog.ysance.com/aller-plus-loin-en-deep-learning-avec-les-r%C3%A9seaux-de-neurones-r%C3%A9currents-rnns dernier accès 20/02
[Inivivoo] https://www.invivoo.com/reduction-dimensionnalite-machine-learning/#:~:text=La%20r%C3%A9duction%20de%20dimensionnalit%C3%A9%20est%20l'ensemble%20de%20techniques%20r%C3%A9duisant,l'essence%20de%20la%20donn%C3%A9e.dernier accès 20/02
[Java] https://www.javatpoint.com/applications-of-machine-learning#:~:text=Image%20recognition%20is%20one%20of,of%20auto%20friend%20tagging%20suggestion.dernier accès 20/02
[Httpcs] https://www.httpcs.com/fr/machine-learning-cybersecurite
[Researchgate] https://www.researchgate.net/post/Does_anyone_know_what_are_the_pros_and_cons_for_using_autoencoders_instead_of_CNNs_for_features_extraction_in_neural_networks dernier accès 08/04
[Science] https://towardsdatascience.com/recurrent-neural-networks-b7719b362c65
[Fr] https://fr.acervolima.com/fonctions-d-activation/ dernier accès 30/03
[Towards] https://towardsdatascience.com/machine-learning-classifiers-a5cc4e1b0623 19/04
[Sekkil] SEKKIL, Hicham Mohamed; MEBROUKI, Mahmoud (Directeur:M. MEGNAFI Hichem / Co-Directeur: Melle. Imane NEDAJR, 2021-09-26)
[Kali] https://www.kali-linux.fr/conseil/introduction-sql-injection-sqlmap dernier accès 20/04
[Net] https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1445304-python-definition-et-utilisation-de-ce-langage-informatique/ dernier accès 20/04
[Transit] https://www.data-transitionnumerique.com/anaconda-python/ dernier accès 20/04
[Pandas] http://www.python-simple.com/python-pandas/panda-intro.php dernier accès 10/05
[Numpy] https://datascientest.com/numpy dernier accès 10/05
[Scikit] https://www.techopedia.com/definition/33860/scikit-learn dernier accès 10/05
[Keras] https://www.journaldunet.fr/web-tech/guide-de-l-intelligence-artificielle/1501863-keras-bibliotheque-de-deep-learning/#keras-cest-quoi dernier accès 10/05
[Matplo] https://matplotlib.org/ dernier accès 22/05
[Jedha] https://www.jedha.co/formation-ia/matrice-confusion#:~:text=La%20matrice%20de%20confusion%20est,celles%20pr%C3%A9dites%20par%20un%20mod%C3%A8le. dernier accès 28/05

IX. Webographie

[Wekimedia] :https://commons.wikimedia.org/wiki/File:Tcp_synflood.png

[Cyber] : https://www.axiomeassocies.fr/cyberattaque/

[Cacciapaglia]: Cacciapaglia, K. (2018). Analyse sur les différentes cyberattaques informatiques (Doctoral dissertation, Haute école de gestion de Genève).

[Open] : https://iq.opengenus.org/udp-flood-attack/

[Network] : Network-Based Attacks | Introduction to Network Security (flylib.com)

[Blogs]: https://blog.malwarebytes.com/101/2018/07/when-three-isnt-a-crowd-man-in-the-middle-mitm-attacks-explained/

[Miami]: https://www.it.miami.edu/about-umit/it-news/phishing/phishing-at-the-u/index.html

[Detectify] :https://blog.detectify.com/2015/12/16/what-is-cross-site-scripting-and-how-can-you-fix-it/

[Avast]: https://www.avast.com/fr-fr/c-sql-injection

[Hakim] : Hakim, M. A., Amina, B. A. L. I., &Manele, A. H. MéMoire de fin d'études.

[Glossaire]: https://iotindustriel.com/glossaire-iiot/systeme-de-detection-dintrusion-ids/

[Cyberhoot]: https://cyberhoot.com/cybrary/network-based-intrusion-detection-system-nids/

[Gbhackers]: https://gbhackers.com/intrusion-detection-system-ids-2/

[Arxiv]: https://arxiv.org/ftp/arxiv/papers/1312/1312.2052.pdf

[Meh]: Mehra, Lekhraj & Gupta, Mukesh & BHATT, MONIKA. (2014). An Effectual and Secure Approach for the Detection and Efficient Searching of Network Intrusion Detection System (NIDS). International Journal of Computer Applications. 108. 37-41. 10.5120/18990-0442

[Medium]: https://medium.com/@redouanechafi/data-science-0-0-quest-ce-que-le-machine-learning-fde2b3c5f19f

[Actualité]: https://actualiteinformatique.fr/intelligence-artificielle/reconnaissance-image-definition

[Paritel]: https://anticip.paritel.fr/se-projeter/5-outils-de-reconnaissance-vocale-pour-booster-votre-productivite/

[Automobile]: https://www.lepoint.fr/automobile/innovations/les-fausses-promesses-de-la-voiture-autonome-12-12-2019-2352818_652.php

[Press]: https://thepressfree.com/negociation-avant-bourse-et-apres-les-heures-douverture-nyse-et-le-nasdaq/

[Switzerland]: https://www.ggba-switzerland.ch/luniversite-de-berne-linselspital-creent-center-for-artificial-intelligence-in-medicine/

[Ionos]: https://www.ionos.fr/digitalguide/web-marketing/search-engine-marketing/deep-learning-vs-machine-learning/

[Natural]: https://www.natural-solutions.eu/blog/la-reconnaissance-dimage-avec-les-rseaux-de-

[Rtavenar]:https://rtavenar.github.io/teaching/neuralnets_td/html/rnn.html

[Big]: https://bigdatablog.skapane.com/les-auto-encodeurs/

[Odoo]: https://www.odoo.com/fr_FR/forum/aide-1/odoo-call-method-written-in-old-api-from-method-written-in-new-api-90151

[Anaconda]: https://anaconda.org/

[Amis]: https://technology.amis.nl/data-analytics/quickest-way-to-try-out-jupyter-notebook-zero-install-3-cli-commands-and-5-minutes-to-action/

[Note]: https://www.n0tes.fr/2019/04/17/Differences-entre-IDS-IPS-et-Firewall/