REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Saad Dahlab de Blida 1

Faculté des sciences

Département Informatique

Mémoire de fin d'étude pour l'obtention du diplôme de Master 2 en Informatique

Option : Sécurité des systèmes d'information

Thème

Développement d'une application d'analyse des fichiers logs et prédiction des attaques

Organisme d'Accueil :

Centre de Recherche sur l'Information Scientifique et Technique

Encadré par :

Mme. Boulekrinat Nour El Houda

Promotrice : Mme. Boustia Narhimene

Réalisé par :

Mlle. Taguelmint Ikram

Mlle. Hadj Mohammed Mariya

Soutenu le 29/09/2019 devant le jury composé de :

Mr. Bala Mahfoud, Maître de conférence, U.Blida 1, **Président**

Mme. Ghebghoub , Maître assistant, U.Blida 1, **Examinateur**

Mme. Boustia Narhimene , Professeur, U.Blida 1, **Promoteur**

Promotion 2019

Session Septembre

Remerciements

Nos premiers remerciements vont à ALLAH le tout Puissant qui nous a guidé et qui nous a donné la force et la volonté de réaliser ce travail

C'est avec grand plaisir que nous réservons cette page, en signe de gratitude et de reconnaissance à tous ceux qui nous ont aidés à la réalisation de ce travail.

Nous remercions Mme Boulkrinat notre encadreur pour sa grande disponibilité, sa rigueur et professionnalisme qui n'a eu de cesse de nous inspirer et aussi pour la confiance qu'elle nous a accordée en proposant ce travail.

Nous remercions, notre promoteur Mme Boustia pour sa rigueur et la pertinence de ses jugements qui ont été très constructifs et nous ont permis de faire ce travail.

Nous remercions vivement les membres de jury pour nous avoir fait l'honneur d'accepter d'examiner notre travail.

Nous voudrons exprimer à nos proches toute notre gratitude : nos très chers parents, nos frères et nos sœurs. Sans leur amour, leur soutien, leur confiance et leurs encouragements, nous n'y serons peut à être pas arrivés.

Résumé

Les applications Web sont l'épine dorsale des systèmes d'information modernes. L'exposition sur internet de ces applications engendre continuellement de nouvelles formes de menaces qui peuvent mettre en péril la sécurité de l'ensemble des systèmes d'information.

La sécurité des systèmes d'information est une problématique d'une importance majeure pour les individus ainsi que pour les entreprises. Elle repose sur la mise en place d'une politique de sécurité autour de ces systèmes, pour compléter cette politique de sécurité, il est devenu nécessaire d'avoir des outils de surveillance pour auditer le système d'information et détecter d'éventuelles intrusions.

Aujourd'hui, l'analyse des fichiers logs est devenue le moyen idéal pour détecter les tentatives d'attaques et aider les administrateurs à identifier les éventuels failles de sécurité. Aussi, le traitement des données de logs peut se faire manuellement, mais cela nécessite beaucoup de temps et peut devenir pratiquement impossible lorsque la taille des fichiers logs est grande.

Dans notre travail nous proposons une solution de prédiction des attaques au travers l'analyse des fichiers logs. Les entrées du fichier journal de serveur web sont utilisées pour la prédiction des intrusions en se basant sur de bonnes techniques d'analyse prédictive et de classification qui permettent de traiter les données et de détecter les anomalies. Nous avons implémenté notre solution en utilisant plusieurs techniques d'apprentissage automatique et nous avons testé chaque technique. Les résultats des tests montrent que ces techniques présentent des résultats intéressants. Aussi l'utilisation de spark l'un des outils Big Data pour l'exécution de notre solution a montré que le temps d'analyse était très court par rapport à l'analyse normale, alors spark à assurer la rapidité du traitement.

Mots Clés : Analyse des fichiers logs, Fichier journal, prédictions des attaques, analyse prédictive, apprentissage automatique.

Abstract

Web applications are the backbone of modern information systems. The Internet exposure of these applications continually generates new forms of threats that can jeopardize the security of the entire information system.

The security of information systems is a problem of major importance for individuals as well as for companies. It is based on the implementation of a security policy around these systems, to complete this security policy, it has become necessary to have monitoring tools to audit the information system and detect possible intrusions.

Today, log file analysis has become the ideal way to detect attack attempts and help administrators to identify security vulnerabilities. Also, the processing of log data can be done manually, but it requires a lot of time and can become practically impossible when the size of log files is large.

In our work we propose a solution of attacks prediction through the analysis of log files. Web server log file entries are used for intrusion prediction based on predictive analytics and classification techniques to process data and detect anomalies. We implemented our solution using several machine learning techniques and tested each technique. Tests show that these techniques have interesting results. Also the use of spark one of the big data tools for the execution of our solution showed that the analysis time was very short in comparison with the normal analysis, so spark ensures the speed of the analysis.

Keywords : Log files analysis, log file, attacks prediction, predictive analysis, machine learning.

Table des matières

Table des figures

Liste des tableaux

Introduction générale

1 Contexte

Avec l'évolution des technologies de l'information et des communications (TIC), les systèmes d'information sont aujourd'hui de plus en plus ouverts sur le monde extérieur notamment Internet. Cette ouverture qui a simplifie la vie pour l'homme en lui offrant plusieurs services, a permis également aux utilisateurs malveillants d'utiliser ces ressources à des fins abusives et de lancer des attaques de divers types à l'encontre des serveurs web. Afin de garantir la sécurité des systèmes d'information, il est devenu vital de développer de nouveaux outils pour la surveillance.

Les fichiers logs sont des ensembles séquentiels de messages produits par un programme dont le rôle est de conserver un historique de l'exécution qui contient toutes les informations détaillées sur l'état et le comportement du système. Ce caractère les rend précieux, et peut être un trésor pour les entreprises qui exploitent les fichiers logs convenablement.

Dans le domaine de la sécurité informatique, l'analyse des fichiers logs constitue le point de départ pour détecter les attaques et faire face aux problèmes de sécurité.

2 Problématique

L'utilisation de l'internet augmente et s'accentue de plus en plus et les attaques malicieuses se multiplient en volume et en complexité, ce qui oblige les administrateurs de vouloir analyser le trafic web sur leur réseau.

L'analyse des fichiers logs constitue un moyen très efficace pour surveiller la performance des systèmes, détecter les disfonctionnements et les problèmes de sécurité.

Cependant, l'analyse des fichiers logs peut se faire manuellement mais la plupart des administrateurs n'osent pas les ouvrir car les logs peuvent avoir une structure complexe et ne sont pas aisés à déchiffrer. Les fichiers journaux sont souvent très volumineux et cela conduit à un traitement trop long.

3 Objectifs

Notre objectif principal consiste à faire la conception et le développement d'une application pour l'analyse des fichiers logs (serveur Web). Cette application doit assurer la prédiction des attaques, pour aider l'administrateur du site Web à identifier les éventuelles failles de sécurité.
L'analyse doit se faire en temps relativement acceptable en se basant sur les algorithmes prédictifs d'apprentissage automatique, à savoir : la régression logistique, la méthode des k plus proches voisins et le classificateur d'arbres de décision pour une bonne prédiction des attaques et l'utilisation du moteur d'exécution de spark qui est l'un des solution Big Data pour assurer la rapidité du traitement.

4 Organisation du mémoire

Notre travail s'organise autour de cinq chapitres principaux :
Les trois premiers chapitres permettent d'introduire les différents concepts théoriques liés à notre travail.

• Le premier chapitre est consacré à présenter l'analyse des fichiers logs, les intérêts ainsi que les contraintes et les difficultés liées aux fichiers logs.
• Dans le deuxième chapitre, nous allons présenter la technologie de Big Data, ces caractéristiques, ces domaines d'application ainsi que son importance dans le domaine de la sécurité informatique.
• Pour le troisième chapitre, nous allons présenter le concept d'analyse prédictive et les techniques dédiés à la prédiction, nous allons parler aussi sur le rôle de la prédiction d'intrusions et les travaux de recherche sur ce domaine.
• Dans le quatrième chapitre, nous allons appliquer les algorithmes de machine Learning pour l'analyse du fichier journal, nous présentons ainsi la conception de notre travail, les différentes étapes nécessaires à l'implémentation de notre conception comme (le prétraitement, le nettoyage, l'exploration et l'analyse du fichier journal) afin de détecter et prédire les attaques.
• Le dernier chapitre est dédié à l'implémentation et la mise en œuvre de notre application avec les différents outils utilisés.

Pour finir, une conclusion générale du mémoire sera nécessaire pour récapituler notre travail et évaluer les résultats obtenus, ainsi que quelques perspectives pour améliorer notre solution.

Chapitre 1

Analyse des fichiers log

1 Introduction

Dans le domaine informatique, pour qu'un système informatique puisse fonctionner correctement, le système d'exploitation doit généralement exécuter plusieurs processus en même temps. La majorité de ces tâches se déroulent en arrière-plan, sans que l'utilisateur ne s'en rende compte.
Pour permettre l'enregistrement des actions des programmes, des protocoles et des fichiers log sont générés automatiquement, ils sont aussi appelés fichiers journaux ou fichiers de traces. Il s'agit des activités et des évènements spécifiques à chaque application comme les messages d'erreur, les modifications de paramètres, les accès aux programmes ou les rapports d'incident. Si vous examinez les fichiers log des serveurs Web, vous pouvez aussi obtenir des informations sur le comportement général des visiteurs d'un site Internet.

2 Définition

L'expression « fichier log » signifie « le journal de bord des connexions », c'est l'historique des requêtes adressées à un système.
Un fichier log est un fichier créé par un logiciel spécifique installé sur un système. Il contient des informations concernant l'activité du système. Il a une structure ASCII [1] qui est lisible par les humains. Le contenu d'un fichier log dépend du niveau d'enregistrement et du type d'activité du système.
Un fichier log contient des entrées. Chaque entrée est une ligne qui représente une requête que le système a reçue, la réponse et le temps de traitement de cette requête. Une entrée d'un fichier log est composée de champs élémentaires de données. Chaque champ désigne une information concernant la requête telle que le nom d'utilisateur, son adresse IP, la requête adressée par l'utilisateur au système, la réponse du système, la date et le temps de soumission de la requête, le protocole utilisé et d'autres informations spécifiques à la requête. Un fichier log représente une base de données textuelle listée par le champ temps [**1**].

1. ASCII :American Standard Code for Information and Interchange

3 Types des fichiers logs

Afin de fournir les informations essentielles sur le comportement d'un système, Les systèmes informatiques génèrent plusieurs types de fichiers de journalisation qui sont généralement classés selon leur emplacement dans le système. Parmi ces types, nous citons les exemples suivants :

3.1 Coté serveur (Server side log files)

Un fichier journal de serveur (Serveur web) est un fichier texte qui comprend les activités exécutées sur le serveur Web. Ces fichiers journaux collectent et stockent les types de données suivants : date, heure, adresse IP du client, référent, agent d'utilisateur, nom du service, nom du serveur, adresse IP du serveur, etc [**2**].
Par conséquent, ces fichiers sont utiles pour analyser les performances et le rendement des serveurs et l'optimisation des moteurs de recherche.

3.2 Coté client (Client side log files)

Les fichiers journaux client contiennent les données collectées côté client lors de l'exécution d'un script sur la machine du client. Ce script est envoyé par le serveur avec le document Web. L'étiquetage ou le marquage des pages (page tagging) est l'une des méthodes les plus utilisées pour la collecte des données coté client [**3**].

3.3 Coté proxy (Proxy side log files)

Un serveur proxy est présent entre la machine cliente et la machine serveur. Il réduit la charge du serveur Web en répondant à la demande des pages disponibles. Il élimine l'inconvénient des fichiers journaux du serveur Web, ce qui permet d'analyser uniquement le comportement de l'utilisateur pour un site particulier [**4**].

3.4 Coté pare-feu (Firewall side log files)

Le Pare-feu enregistre seulement les évènements qui sont refusés par le système. Par conséquent, l'examen des fichiers journaux du registre du pare-feu est essentiel pour la détection des intrusions. Il est également nécessaire de déterminer et d'apprécier la force des mesures de sécurité mises en œuvre sur un pare-feu par le moyen de la vérification et de l'audit du pare-feu [**5**].

3.5 Coté réseau (Network side log files)

Les fichiers journaux réseau peuvent être obtenus à partir des composants du réseau, tels qu'un pare-feu réseau, des routeurs et des filtres de paquets à des fins d'analyse. Le fichier journal réseau a un avantage par rapport aux autres fichiers journaux vu qu'il ne comporte pas de problèmes juridiques concernant la violation de la confidentialité des données des utilisateurs [**6**].

3.6 Coté système (System side log files)

Les fichiers journaux côté système traitent les informations générées par le noyau et les utilitaires systèmes. Des informations détaillées sur les activités du système d'exploitation sont capturées dans les fichiers journaux du système. De plus, ils libèrent le programmeur de la tâche de rédaction des fichiers journaux [**2**].

4 Contenu des fichiers logs web

Lorsque nous utilisons des applications Web, nous sommes exposés à des menaces et des pirates. Contre ces menaces, les responsables de sécurité analysent les fichiers journaux qui répertorient toutes les actions effectuées. Ces fichiers journaux résident sur le serveur Web.
Les fichiers journaux de différents serveurs Web gèrent différents types d'informations. Les informations de base présentes dans le fichier journal sont [**7**] :

- **Nom d'utilisateur (User Name)** : Identifie qui a visité le site Web. L'identification de l'utilisateur est principalement l'adresse IP attribuée par le fournisseur de services Internet (ISP [2]).
- **Chemin de visite (Visiting Path)** : Chemin emprunté par l'utilisateur lors de la visite du site Web. Cela peut être en utilisant l'URL directement ou en cliquant sur un lien ou par le biais d'un moteur de recherche.
- **Chemin parcouru (Path Traversed)** : Identifie le chemin emprunté par l'utilisateur sur le site Web par l'utilisation de divers liens.
- **Horodatage (Time Stamp)** : Le temps passé par l'utilisateur dans chaque page Web lors de la navigation sur le site Web. Ceci est identifié comme session.
- **Dernière page visitée (Page last visited)** : La page visitée par l'utilisateur avant qu'il quitte le site Web.
- **Taux de réussite (Success rate)** : Le taux de réussite du site Web peut être déterminé par le nombre de téléchargements effectués et par le nombre de copies effectuées par l'utilisateur. Si tout achat d'objets ou de logiciels est réalisé, cela correspond également au taux de réussite.
- **Agent utilisateur (User Agent)** : Ce n'est rien d'autre que le navigateur à partir duquel l'utilisateur envoie la demande au serveur Web. Il s'agit simplement d'une chaîne décrivant le type et la version du logiciel de navigateur utilisé.
- **URL** : La ressource consultée par l'utilisateur. Elle peut être une page HTML [3], un programme CGI [4] ou un script.
- **Type de demande (Request Type)** : La méthode utilisée pour le transfert d'informations est indiquée. Comme les méthodes GET et POST.

2. ISP : Internet Service Provider
3. HTML :HyperText Markup Language
4. CGI :Common Gateway Interface

5 Format des fichiers logs

La plupart des systèmes informatiques génèrent des fichiers logs dans un format propriétaire qui est mystérieux et difficile à déchiffrer. Cependant, seulement les développeurs peuvent comprendre leur contenu. L'analyse du contenu d'un fichier log exige une bonne compréhension du format et une structure appropriée facilitant son analyse. En effet, des formats standards ont été développés [**1**].
Parmi les quels, nous citons : le format W3C [5] Etendu, le format NCSA [6] log commun et le format Microsoft IIS [7].

5.1 Le format log Etendu W3C (W3C Extended Log File Format)

Le format log W3C (Word Wide Web Consortium) Etendu est un format ASCII personnalisé avec une variété de champs. Avec l'utilisation de ce format, des champs peuvent être inclus lorsque cela est nécessaire, et la limitation de la taille du journal peut être faite tout en omettant les champs non désirés. Les champs sont séparés par des espaces, l'heure est enregistrée en UTC [8] aussi appelé GMT [9] qui est le temps moyen de Greenwich [**8**]. Ce format est disponible pour les serveurs Web et les serveurs FTP [10][**1**].

L'exemple ci-dessous montre les lignes d'un fichier log au format W3C Etendu utilisant les champs suivants : Heure, Adresse IP du client, Méthode, Stem d'URI, état ou statut du protocole et Version du protocole.

```
#Software: Microsoft Internet Information Services 5.1
#Version: 1.0
#Date: 1998-05-02 17:42:15
#Fields: time c-ip cs-method cs-uri-stem sc-status cs-version
17:42:15 172.16.255.255 GET /default.htm 200 HTTP/1.0
```

FIGURE 1.1: Format W3C Etendu [**8**]

L'entrée précédente indique que :

— Le 2 mai 1998 à 17h42 UTC, un utilisateur avec la version 1.0 du HTTP et l'adresse IP 172.16.255.255 a émis une commande HTTP GET pour le fichier Default.htm.

— La demande a été renvoyée sans erreur.

— Le champ Date : indique quand la première entrée du journal a été effectuée, c'est-à-dire quand le journal a été créé.

— Le champ Version : indique que le format de journalisation W3C 1.0 a été utilisé.

5. https ://www.w3.org/
6. http ://www.ncsa.illinois.edu/
7. https ://www.iis.net/
8. UTC :Universal Time Coordinated
9. GMT :Greenwich Mean Time
10. FTP :File Transfer Protocol

Tous les champs peuvent être sélectionnés, mais certains champs peuvent ne pas avoir d'informations disponibles pour certaines demandes.
Pour les champs sélectionnés, mais pour lesquels il n'y a pas d'information, un tiret (-) apparaît dans le champ en tant qu'espace réservé.
Si le site est configuré pour l'authentification d'utilisateur anonyme, l'utilisateur s'affiche sous forme de tiret (-).

5.2 Le format log commun du NCSA (Common Log File Format)

Le format NCSA Commun est un format ASCII fixe, il est utilisable pour les serveurs Web uniquement et pas pour les serveurs FTP. Il a été développé par NCSA [11] à l'université d'Illinois à Urbana-Champaign [**1**]. Il contient des informations de base sur les demandes des utilisateurs telles que[**8**] :

• Le nom de l'hôte ou l'adresse IP de l'hôte.

• Le nom d'utilisateur.

• La date et l'heure de soumission de la requête.

• Le type de demande.

• Le contenu de la requête envoyée par le client.

• Le code d'état HTTP retourné à l'utilisateur : c'est le code de la réponse http envoyé par le serveur au client.

• La taille en octets des informations envoyés par le serveur. Les champs sont séparés par des espaces ; l'heure est enregistrée comme heure locale.

Lorsque vous ouvrez un fichier au format NCSA log Commun dans un éditeur de texte, les entrées sont similaires à l'exemple suivant [**8**] :

```
172.21.13.45 - REDMOND\fred [08/Apr/1998:17:39:04 -0800] "GET
/scripts/iisadmin/ism.dll?http/serv HTTP/1.0" 200 3401
```

FIGURE 1.2: Format NCSA commun [**8**]

L'entrée précédente indique que :

— Un utilisateur nommé Fred dans le domaine REDMOND, dont l'adresse IP est 172.21.13.45, a émis une commande HTTP GET (c'est-à-dire, téléchargé un fichier) à 17h39 le 8 avril 1997.

— La demande a été renvoyée, sans erreur.

— 3401 octets de données envoyées à l'utilisateur nommé Fred.

Dans l'entrée précédente, le deuxième champ (qui indiquerait le nom de l'hôte de l'utilisateur) est vide, il est représenté par un tiret après l'adresse IP 172.21.13.45.

11. NCSA :National Center for Super computing Applications

5.3 Le format Microsoft IIS

Le format Microsoft IIS est un format ASCII fixe. Il enregistre plus d'informations que le format NCSA log Commun.

Le format Microsoft IIS comprend des éléments de base tels que :

• L'adresse IP de l'utilisateur.

• Le nom d'utilisateur.

• La date et l'heure de la demande.

• Le code d'état du protocole HTTP et le nombre d'octets reçus.

En outre, il inclut des éléments détaillés tels que :

• Le temps écoulé de la demande.

• Le nombre d'octets envoyés.

• L'action (par exemple, un téléchargement effectué par une commande GET) et le fichier cible.

Les éléments sont séparés par des virgules, ce qui facilite la lecture du format par rapport aux autres formats ASCII qui utilisent des espaces pour la séparation des champs. L'heure est enregistrée comme heure locale [**8**]. Lorsque vous ouvrez un fichier log au format Microsoft IIS dans un éditeur de texte, les entrées sont similaires à l'exemple suivant [**9**] :

```
192.168.114.201, -, 03/20/98, 7:55:20, W3SVC2, SALES1, 192.168.114.201,
4502, 163, 3223, 200, 0, GET, /DeptLogo.gif, -,

172.16.255.255, anonymous, 03/20/98, 23:58:11, MSFTPSVC, SALES1,
192.168.114.201, 60, 275, 0, 0, 0, PASS, /intro.htm, -,
```

FIGURE 1.3: Format Microsoft IIS [**9**]

Dans l'exemple précédent la première entrée indique que :

— Un utilisateur anonyme avec l'adresse IP 192.168.114.201 a émis une commande HTTP GET pour le fichier image /DeptLogo.gif à 7 h 55 le 20 mars 1998 à partir d'un serveur nommé SALES1 à l'adresse IP 172.21.13.45.

— Le temps de traitement écoulé de la requête HTTP (de 163 octets) était de 4502 millisecondes (4,5 secondes) et renvoyait sans erreur 3223 octets de données à l'utilisateur anonyme.

6 Analyse d'un fichier log

L'analyse des fichiers log est le processus d'inspection ciblé et d'évaluation d'un fichier journal. L'analyse est notamment utilisée par exemple pour tracer les erreurs de transmissions de données et de courriers électroniques, ou pour vérifier les activités du pare-feu. Mais plus généralement cette méthode est utile dans l'optique de l'optimisation pour les moteurs de recherche (le référencement).

En plus d'enregistrer des données concernant les visiteurs du site Web, le fichier journal sert également à

déterminer les erreurs techniques (concernant le réseau, les programmes ou encore les composants individuels), les problèmes de sécurité ainsi que les accès automatisés par les robots [**10**]. L'analyse manuelle des fichiers logs est tout simplement impossible en raison du nombre important de fichiers et de leur taille. Cependant il existe différents outils d'analyse qui permet d'extraire les informations pertinentes des fichiers journaux. Il suffit ainsi de tirer les bonnes conclusions des données collectées.

7 Intérêt d'analyse des fichiers log

Les fichiers logs sont généralement utilisés pour plusieurs raisons, nous avons :

1. **La surveillance des systèmes :** Lorsqu'un système ou une application est en cours d'exécution, c'est comme une boite noir et il est donc impossible de comprendre le comportement du système. En effet, les logs sont générés en temps réel. Il est donc possible de s'en servir pour surveiller un programme pendant son exécution.
2. **Le dépannage :** Lorsqu'un bug ou un problème est signalé, le fichier log est, sans exception, le meilleur endroit où chercher pour savoir exactement où se situe le problème.
3. **Audit :** De nombreuses organisations doivent se conformer à certaines procédures de conformité et sont tenues de gérer les logs. Par exemple, l'activité de connexion ou les activités de transaction effectuées par un utilisateur sont généralement capturées et conservées dans des logs pendant une certaine durée, à des fins d'audit ou d'analyse d'activités malveillantes par des utilisateurs ou des pirates [**12**].
4. **Identification des attaquants :** Les fichiers logs sont un composant vital et critique pour Network Forensics [**1**]. Les informations pertinentes contenues dans les fichiers logs représentent la preuve qui est le besoin indispensable pour l'investigation [**13**]. C'est le seul moyen pour identifier l'attaquant a fin de le poursuivre judiciairement.
5. **La détection d'anomalies :** Un log contient des informations sur un évènement s'étant produit au sein du programme et l'ensemble des logs révèle donc l'histoire du programme. Ce caractère historique est très précieux quand il s'agit d'identifier un problème qui est survenu. Il suffit d'analyser le fichier log pour diagnostiquer le problème et prendre les mesures nécessaires. Les fichiers logs sont universellement utilisés pour de la détection d'anomalies [**11**].
6. **Analyse prédictive :** C'est une branche de l'analyse avancée qui est utilisée pour prédire les évènements inconnus qui pourraient survenir dans le futur. Les modèles qui résultent en données historiques et transactionnelles peuvent ensuite être utilisés pour identifier les opportunités ainsi que les risques pour l'avenir [**12**].

8 Les outils d'analyse des fichiers logs

Il existe de nombreux outils d'analyse des fichiers logs qui aident à mieux comprendre les données du journal et à les analyser de manière plus efficace. Cela peut aider à identifier la cause première de toute erreur d'application ou de logiciel, et donc faire gagner du temps la prochaine fois qu'un problème survient. La même chose peut être faite pour les problèmes liés à la sécurité, afin de détecter les intrusions et les vulnérabilités du réseau et de prévenir les attaques avant même qu'elles ne se produisent. Nous détaillons dans ce qui suit les outils traditionnels et les solutions Big Data d'analyse des logs :

8.1 Les outils traditionnels d'analyse des fichiers logs

Parmi les outils standards d'analyse des fichiers logs, nous citons :

8.1.1 GoAccess

GoAccess [12] est un open source conçu pour être un analyseur de fichiers logs rapide basé sur un terminal. Son idée principale est d'analyser et d'afficher rapidement les statistiques des serveurs Web en temps réel sans avoir à utiliser votre navigateur[**14**].

8.1.2 Scalp

Scalp [13] est un analyseur des fichiers logs serveur Web Apache qui vise à rechercher les problèmes de sécurité. L'idée principale est de parcourir d'énormes fichiers journaux et d'extraire les éventuelles attaques envoyées via HTTP / GET (par défaut, Apache n'enregistre pas la variable HTTP / POST)[**15**].

8.1.3 Logstash

Logstash [14] un outil open source gratuit pour la gestion des fichiers logs. Il est utilisé pour collecter des journaux, les analyser et les stocker pour une utilisation ultérieure. Cet outil va de pair avec Elasticsearch [15] et Kibana [16]. L'utilisation conjointe de ces éléments peut constituer une puissante combinaison pour un outil d'analyse de journal [**16**].

9 Les problèmes liés aux fichiers logs

Malgré l'importance des fichiers logs, néamoins certains problèmes demeurent posés [**1**]. Nous citons :

• Les fichiers logs consomment un espace disque très grand.

• Les fichiers logs contiennent beaucoup d'information, ils sont immenses et par conséquent l'analyse de leurs contenus devient une tâche très difficile.

12. https ://goaccess.io/download
13. https ://github.com/nanopony/apache-scalp
14. https ://www.elastic.co/fr/downloads/logstashest
15. Elasticsearch :un moteur de recherche distribué, intégrant une base de données NoSQL
16. Kibana :une interface web permettant de rechercher des informations stockées par Logstash dans ElasticSearch

• Les fichiers logs menacent la vie privée(privacy) de l'utilisateur. Un utilisateur refuse l'idée que toutes ses activités soient enregistrées.

• Les fichiers logs peuvent être menacés comme d'autres formes de données dans le réseau ou dans un système. Un attaquant qualifié pénétrant dans un système peut effacer les fichiers logs ou modifier leur contenu. Il peut même arrêter le mécanisme d'enregistrement.

10 Conclusion

Dans ce premier chapitre, nous avons introduit et défini les logs tout en donnant un apperçu sur les informations qu'ils peuvent contenir et de la façon don't ils sont enregistrés ainsi que les types et les outils d'analyse des logs. Nous avons présenté aussi l'analyse des fichiers log qui est devenue une tache très importante pour la surveillance des systèmes, la détection d'anomalies , la prévention de fraude, audit ...

Nous avons vu en fin du chapitre les problèmes liées au fichiers logs lorsqu'ils sont immenses et par conséquent l'analyse de leurs contenus devient une tâche très difficile et consomme un espace disque très grand. L'arrivée du Big Data fait face à ce problème par ça capacité de collecter, traiter et analyser une énorme quantité de données. Enfin, La sécurité informatique a besoin d'un moyen de surveillance des systèmes et des réseaux cependant l'analyse des logs représente ce moyen de surveillance. Dans le chapitre suivant nous allons présenter la technologie Big Data.

Chapitre 2

Big Data

1 Introduction

Depuis la révolution numérique, la quantité de données produites chaque jour dans ou en dehors d'un Système d'Information a pris de telles proportions qu'il est difficile de continuer à utiliser les outils traditionnels pour les manipuler de façon performante. Il est devenu nécessaire de développer de nouvelles méthodes pour gérer et analyser cette énorme quantité d'informations. Ainsi, est né le Big Data ; un concept qui porte sur la recherche, l'analyse, la capture, le stockage, le partage et la présentation de ces données.

2 Définitions

Littéralement, le terme Big Data signifie mégadonnées, grosses données ou encore données massives. Il désigne un ensemble très volumineux de données qu'aucun outil classique de gestion de base de données ou de gestion de l'information ne peut les traiter. En effet, on procrée environ 2,5 trillions d'octets de données tous les jours. Ce sont les informations provenant de partout : messages que nous envoyons, vidéos que nous publions, informations climatiques, signaux GPS, enregistrements transactionnels d'achats en ligne et bien d'autres encore.
Cependant, aucune définition précise ou universelle ne peut être donnée au Big Data. Etant un objet complexe, sa définition varie selon les communautés qui s'y intéressent en tant qu'usager ou fournisseur de services. Parmi les autres définitions nous citons :

1. **Selon Gartner [17] :** les Big Data sont des ressources d'information volumineuses, à grande vélocité et à grande variété qui exigent des formes innovantes et rentables de traitement de l'information pour améliorer la compréhension et la prise de décision.
2. **Selon Lisa Diforti [18] :** le Big Data est un ensemble de technologies, d'architecture, d'outils et de procédures permettant à une organisation de très rapidement capter, traiter et analyser de

larges quantités et contenus hétérogènes et changeants, et d'en extraire les informations pertinentes à un coût accessible.

3 Caractéristiques du Big Data

La caractérisation de Big Data est généralement faite selon trois « V » : Volume, Variété et Vélocité, auxquels s'ajoutent d'autres « V » complémentaires : Valeur et Véracité [**19**].

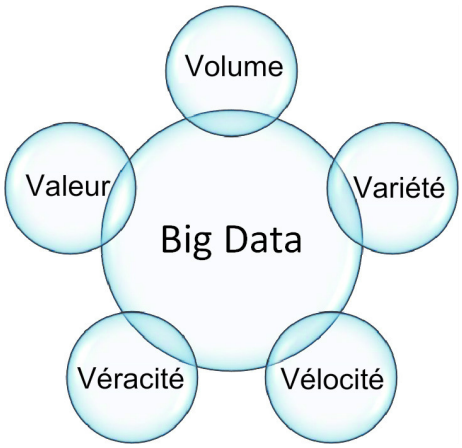

FIGURE 2.1: Caractéristiques du Big Data

— **Volume** : Le caractère « volume » est certainement celui qui est le mieux décrit par le terme « Big » de l'expression. Volume fait référence à la quantité d'informations, trop volumineuse pour être acquise, stockée, traitée, analysée et diffusée par des outils standards. Ce caractère peut s'interpréter comme le traitement d'objets informationnels de grande taille ou de grandes collections d'objets [**20**].

— **Variété** : Fait référence à l'hétérogénéité des formats, de types et de qualité des informations. Il est lié au fait que ces données peuvent présenter des formes complexes du fait qu'elles trouvent leurs origines dans des capteurs divers et variés (température, vitesse du vent, hygrométrie, tours/mn, luminosité...), dans des messages échangés (e-mails, médias sociaux, échanges d'images, de vidéos, musique), dans des textes, des publications en ligne (bibliothèques numériques, sites web, blogs...), des enregistrements de transactions d'achats, des plans numérisés, des annuaires, des informations issues des téléphones mobiles, etc [**21**].

— **Vélocité** : C'est l'aspect dynamique et/ ou temporel des données, à leur délai d'actualisation et d'analyse. Les données ne sont plus traitées, analysées, en différé, mais en temps réel ou quasi réel. Elles sont produites en flots continus, sur lesquels des décisions en temps réel peuvent être prises. Ce sont les données notamment issues de capteurs, nécessitant un traitement rapide pour

une réaction en temps réel. Dans le cas de telles données de grande vélocité engendrant des volumes très importants, il n'est plus possible de les stocker en l'état, mais seulement de les analyser en flux (streaming), voire de les résumer [**22**].

— **Valeur** : Le caractère complémentaire « valeur » fait référence à la potentialité des données, en particulier en termes économiques. Il est ainsi associé à l'usage qui peut être fait de ces mégadonnées, de leur analyse, notamment d'un point de vue économique. L'analyse de ces mégadonnées demande une certaine expertise tant liée à des méthodes et techniques en statistique, en analyse de données, que de domaine pour l'interprétation de ces analyses [**21**].

— **Véracité** : Enfin, le caractère complémentaire « véracité ou validité » fait référence à la qualité des données et/ou aux problèmes éthiques liés à leur utilisation. Il comprend les problèmes de valeurs aberrantes ou manquantes (ces problèmes pouvant être résolus par le volume de données), mais aussi à la confiance que l'on peut avoir dans les données. S'il existe des critères permettant de qualifier la qualité des données, dans le cas de Big data, cette vérification de la qualité est rendue difficile voire impossible du fait du volume, de la variété et de la vélocité spécifiques au Big Data [**23**].

4 Domaines d'application du Big Data

Le domaine d'application du Big Data est très vaste. Ce concept peut être appliqué dans plusieurs secteurs qui manipulent quotidiennement des volumes de données très importants , avec des problématiques de vitesse associées. Nous citons les exemples suivant :

4.1 Marketing

Le secteur du marketing a toujours été un secteur très ouvert aux nouvelles pratiques et les technologies émergentes y ont leur place.
Les métiers éprouvent des difficultés à concevoir une stratégie marketing efficace pour leur compagnie. La bonne nouvelle est que le Big Data ouvre de nombreuses portes pour améliorer ces stratégies. Par le biais de plusieurs logiciels, les compagnies peuvent utiliser la donnée pour être davantage informées sur leurs clients. En chiffre, le marché du Big Data a été estimé autour de 203 milliards de dollars en 2020, et le progrès va encore plus loin en ce qui concerna la prédiction [**24**].

4.2 Surveillance

Le terrorisme est l'un des principaux problèmes du 21ème siècle à l'échelle mondiale. La menace est présente et les attentats peuvent survenir à tout moment. Pour lutter contre ce fléau, le Big Data se présente comme un atout précieux.

Le Big Data aide à prévenir un attentat (suivre les déplacements d'un suspect, reconnaissance faciale sur des vidéos, interception de correspondance par mails entre terroristes) et résoudre efficacement des enquêtes policières (analyser des indices, trouver une corrélation entre plusieurs affaires). Il permet de réduire le temps de résolution des affaires et d'en augmenter le taux de résolution. Alors nous pouvons prédire jusqu'à 90 pourcent des attaques des terroristes [**25**].

4.3 Sécurite

Selon une étude réalisée par Marketsand Markets récemment, le marché de l'analytique de sécurité pourrait atteindre les 9 milliards de dollars d'ici 2020. A cet effet, le Big Data va jouer de plus en plus de rôles dans les années à venir dans ce domaine. Il est capable de capturer, de filtrer et d'analyser des millions d'évènements réseau par seconde. Ces solutions peuvent intervenir sur des données issues de sources de tout horizon, même des fichiers journaux et d'audit. Le cabinet Gartner, lui, estime que les entreprises ayant des difficultés en matière de cyber sécurité devront s'investir dans l'analytique pour une meilleure détection des menaces. Les différentes entreprises doivent bien profiter de ces solutions pour déceler et contrer les menaces plus complexes, grâce à des informations automatisées et exploitables qu'elles fournissent [**26**].

FIGURE 2.2: Les applications de Big Data en résumé [**27**]

5 Les avantages et les inconvénients du Big Data

La technologie Big Data offre un nombre considérable d'avantages et marque une avancée révolutionnaire en plus d'être pleine de promesses pour les entreprises. Mais, comme toute avancée technologique, le Big Data a ses risques et cela nous amène à nous questionner sur les avantages et les inconvénients majeurs de son utilisation.

a) Avantages

• Résoudre les problèmes de lenteurs de traitement de requête, le stockage est beaucoup plus robuste et permet un stockage en masse des données externes ou internes à l'entreprise.

• Faciliter le tri des données pour extraire certaines données non compatibles ou non acceptées par les bases de données traditionnelles, vu que le Big Data gère tout type de données qu'elles soient structurées ou pas (Vidéo, musique, fichier csv, json, xml, etc...).

• Améliorer la sécurité par le diagnostique des anomalies, l'aide à la détection de fraude et à l'évaluation des risques en analysant toutes les informations prévenant des transactions possibles avec l'entreprise ou l'individu.

• Optimiser l'offre d'une entreprise : il permet en effet une analyse complète du comportement et des attentes du consommateur. Google Analytics permet par exemple d'optimiser son site Web par une analyse en temps réel des données liées : nombre de visites, comportement de navigation, taux de rebond, nombre de pages lues, taux de clics... [**28**]

• Le Big Data a eu un succès certains chez Target et Amazon notamment, qui ont utilisés les Big Data dans le but de gagner des avantages compétitifs dans la vente. Dans son livre [**29**], Nate Silver mentionne comment l'utilisation massive des données, en informatique, a permis d'améliorer la qualité des prévisions météorologiques, en particulier au cours des vingt-cinq dernières années.

b) Inconvénients

• Le cout lié à l'implémentation des technologies Big Data et la formation du personnel n'est pas accessible à beaucoup d'entreprise. Etant donné que c'est une nouvelle technologie en pleine expansion il y'a une forte demande de personnel qualifié ayant de l'expérience pour l'implémentation et la manipulation des outils Big Data au sein des entreprises.

• La mauvaise utilisation des technologies Big Data, mène à des failles et cela est considéré comme inconvénient majeur ; il y a le problème de la sécurité des données puisqu'il est très difficile de stocker une masse de données aussi conséquente sans aucun risque d'intrusion. Pour cela il faut bien utiliser les outils de Big Data en assurant toute politique de sécurité.

En résumé, nous ne pouvons pas nier que le Big Data peut se révéler dangereux pour les individus et la société en raison des dérives et des failles qu'il peut induire. En revanche, le Big Data marque une avancée incontestable et il convient de renforcer la surveillance de l'exploitation de ces données pour en éviter les dérives.

6 Architecture Big Data (Lambda)

Actuellement, il existe plusieurs architectures Big Data : l'architecture Lambda, l'architecture kappa et l'architecture Zeta. Dans ce qui suit, nous allons nous intéresser à l'architecture Lambda qui est la plus répandue en ce moment.

L'architecture Lambda permet de fournir un modèle de traitement presque temps réel sur des volumes importants de données, en proposant un nouveau modèle de calcul. Ce modèle essaie de trouver l'équilibre entre la tolérance aux pannes, les contraintes de latence (latence très faible pour les lectures/écritures) et le débit des disques durs en se basant à la fois sur les traitements batch qui fournissent des vues batch et les traitements temps réel qui fournissent des vues, puis les joint avant leur présentation [**30**].

Cette architecture est indépendante de la technologie, et se base sur le pré-calcul des résultats, puis à les récupérer dans une base et les envoyant au demandeur. Elle est composée de trois couches suivantes (Figure 1.3) :

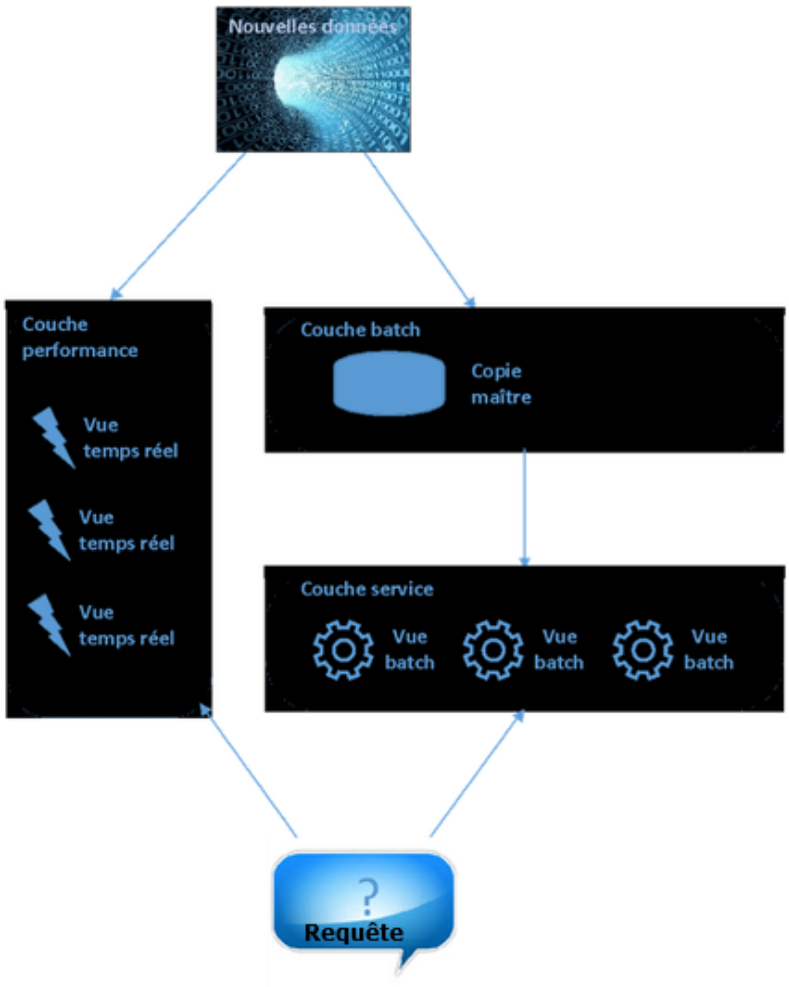

FIGURE 2.3: Architecture Lambda [**31**]

— **Couche Batch** : La couche Batch stocke de façon distribuée une copie de l'ensemble des données. Elle comprend des données structurées et non structurées brutes. Les données sont stockées telles quelles, sans dérivation ni transformation. Le fait de garder les données dans un format brut fait en sorte qu'il n'y a aucune perte de données.
Par ailleurs, c'est la couche batch qui s'occupe du calcul des vues batch. Les vues batch sont le résultat de dérivations et de transformations de données brutes.

— **Couche service** : La couche service accède aux vues batch dès qu'elles sont disponibles, c'est-à-dire dès que le calcul (ou le recalcul) par la couche batch est complété. Comme pour la couche batch, il y a distribution sur des grappes de machine pour assurer l'évolutivité.

— **Couche performance** : La couche performance se veut une sorte de couche batch, mais seulement pour les données récentes. Elle produit des vues en temps réel qui sont mises à jour au fur et à mesure que des nouvelles données sont reçues. Elle utilise des algorithmes incrémentaux et des structures de stockage avec des contraintes de latence plus complexes. Mais cette complexité est isolée pour quelques heures de données seulement. Une fois la latence passée pour une nouvelle donnée dans les couches batch et service, la donnée correspondante dans la couche performance n'est plus requise.
Les requêtes de la couche service qui requièrent une faible latence vont alors typiquement combiner les vues batch et temps réel pour s'exécuter.

7 Big data et la sécurité informatique (Cyber sécurité)

Le Big Data suscite énormément de débats dans le domaine de sécurité, mais de quoi discutons-nous vraiment ? En termes de cyber sécurité, le Big Data représente à la fois une opportunité et une menace pour les entreprises. Alors deux problèmes différents se posent : d'une part la sécurité des informations de l'entreprise et de ses clients dans un contexte de Big Data, d'autre part l'utilisation des techniques du Big Data pour analyser, ou prévoir, les incidents de sécurité.

7.1 La sécurité des Big Data

Un grand nombre d'entreprises utilisent le Big Data pour le marketing et les recherches, mais ne maîtrisent pas forcément les concepts de base, en particulier la sécurité.
Ces entreprises utilisent cette technologie pour stocker et analyser des pétaoctets de données, notamment les journaux Web, les données sur le parcours de navigation et le contenu des réseaux sociaux, et ce, dans le but de mieux connaître leurs clients et leurs activités.
Cependant, les cybercriminels peuvent eux aussi profiter de ces opportunités pour accéder à des quantités massives d'informations sensibles en utilisant les technologies les plus avancées.
Par conséquent, la classification des informations devient encore plus critique et il convient de déterminer

la propriété des informations pour permettre une classification acceptable.
Pour cela, les entreprises doivent faire recours à des techniques telles que le chiffrement pour protéger les données sensibles et appliquer les contrôles d'accès et les outils analytiques du Big Data pour détecter et prévenir les cyber attaques.

7.2 Solution de sécurité pour les environnements Big Data

Comme nous avons déjà mentionné les environnements Big Data doivent être dotés d'une protection des données sensibles. A cet effet une solution de IBM® Security Guardium® [**32**] sécurise les environnements Big Data en :
• Surveillant étroitement l'activité des bases de données Hadoop et NoSQL par les applications et les utilisateurs en temps réel ; déclenchant des alertes en cas de violation de règles ; en faisant le suivi des tentatives d'accès et d'utilisation des données afin de déceler tout comportement inhabituel parmi les utilisateurs privilégiés et externes ; et en notifiant les tableaux de bord SIEM [1] pour engager les mesures correctives adéquates (alerte, blocage, résiliation de connexion).
• Implémentant des contrôles automatisés et centralisés au sein de l'entreprise (bases de données, applications, fichiers, Big Data, etc.).
• Protégeant les données sensibles au moyen de techniques de chiffrement, de masquage et d'occultation.
• Évaluant et résolvant les faiblesses de l'environnement de manière à sécuriser l'ensemble du Big Data.
En résumé, l'objectif de la solution Guardium est d'améliorer la sécurité des données et les décisions en matière de sécurité en se fondant sur des informations exploitables et priorisées, issues du contrôle et de la surveillance de l'ensemble de l'environnement.

7.3 Le Big Data au service de la sécurité

A l'heure des Big Data, ces nouvelles ressources doivent permettre aux entreprises de dépasser les niveaux de sécurité classiques. Désormais les Big Data vont permettre aux entreprises d'accéder à un troisième niveau de protection contre les cyber attaques.
En ce qui concerne les paliers de sécurité informatique, on peut distinguer trois niveaux :
• Le premier niveau, qui consiste à sécuriser l'entreprise des attaques provenant de l'extérieur.
• Le deuxième niveau est cependant plus complexe, puisqu'il s'agit de protéger l'entreprise de l'intérieur, vis-à-vis de ses propres utilisateurs qui peuvent être une source de vulnérabilités au sein du système d'information.
• Le troisième niveau de sécurité consiste à mesurer l'impact que la menace détectée a eu sur l'infrastructure, ce qui suppose de pouvoir remonter dans le temps, donc il s'agit d'enregistrer le trafic qui transite par les réseaux de l'entreprise afin de traquer le chemin emprunté par le logiciel malveillant dès qu'il a

1. Security Information and Event Management

été repéré et de prendre des mesures efficaces en conséquence.

- **Les technologies analytiques au service de la cyber sécurité** :

Les outils analytiques du Big Data offrent aux professionnels de la cyber sécurité la capacité d'analyser différents types de données en provenance de sources diverses et de réagir en temps réel. Ces outils ne permettent pas seulement de rassembler des informations, mais aussi de connecter ces données, et d'établir des corrélations et des connexions.

Parmi les outils analytiques qui assure la sécurité :

1. **Apache Spark [33] :** Apache Spark est un moteur rapide pour le traitement des données à grande échelle. C'est un framework de calcul en cluster open source.

   Apache Spark peut aider les responsables de la cyber sécurité à analyser des données et à répondre à des questions :

   • Quels serveurs internes de l'entreprise tentent de se connecter à des serveurs internationaux ?

   • Est-ce que le modèle d'accès des utilisateurs aux ressources internes a été changé au fil du temps ?

   • Quels utilisateurs présentent des schémas de comportement irréguliers, tels que la connexion en utilisant des ports non standards ?

   Les solutions de découverte Big Data optimisées par Spark peuvent être utilisées pour détecter des anomalies et des valeurs aberrantes dans de grands ensembles de données. Les techniques de visualisation aident lorsque des pétaoctets de données doivent être analysés.

2. **IBM Security QRadar [34] :** Cet outil utilise les capacités Big Data pour suivre le rythme des menaces avancées et prévenir les attaques de manière proactive. Il aide à révéler les relations cachées dans de grandes quantités de données de sécurité, en utilisant des analyses pour réduire des milliards d'événements de sécurité à un ensemble contrôlable d'incidents hiérarchisés. Il utilise les fonctionnalités suivantes de la solution Big Data :

   • Détection en temps réel de la corrélation et des anomalies des données de sécurité, de nature diverse.

   • Interrogation à grande vitesse de données d'intelligence de sécurité.

   • Une analyse flexible des données volumineuses, structurée ou non structurée cela comprend les données de sécurité, les courriers électroniques, les contenus de documents et de médias sociaux, les données de processus métier ; et d'autres informations.

   • Outil graphique frontal permettant de visualiser et d'explorer des données volumineuses.

8 Les outils Big Data d'analyse des fichiers logs

Face à l'explosion du volume d'informations et aux problèmes de la taille énorme et du temps d'analyse des fichiers de journalisation, le Big Data vise à proposer une alternative aux solutions traditionnelles d'analyse des fichiers logs tout en répondant à trois critères : un Volume de données important à traiter, une grande Variété et un certain niveau de Vélocité à atteindre qui veut dire la fréquence de creation, collecte, traitement, analyse et partage des données [**35**]. Parmi ces outils nous citons :

8.1 Splunk

Splunk [2] est une plate-forme logicielle utilisée pour rechercher, analyser et visualiser des données générées par machine pour une meilleure représentation des données.Splunk est conçu pour prendre en charge le processus d'indexation et de déchiffrement des journaux de tout type, qu'ils soient structurés, non structurés ou sophistiqués, sur la base d'une approche multiligne [**36**]. Splunk permet le traitement en temps réel des données et la creation des alertes ou des notifications d'événements en fonction de l'état de la machine [**37**].

8.2 Sumo Logic

Sumo [3] Logic est une plate-forme unifiée de journaux et de mesures qui aide à analyser vos données en temps réel à l'aide de l'apprentissage automatique. Sumo Logic peut rapidement décrire la cause première d'une erreur ou d'un événement particulier. Le point fort de Sumo Logic est sa capacité à manipuler les données à un rythme rapide, éliminant ainsi le besoin d'outils externes d'analyse et de gestion des données [**36**].

8.3 Apache Metron

Apache Metron [4] est framework d'application de cyber sécurité qui permet d'avoir une vue unique de diverses données de sécurité en continu pour aider les centres d'opérations de sécurité à détecter rapidement les menaces et à y répondre. Apache Metron intègre des technologies Big Data open source dans un outil centralisé de surveillance et d'analyse de la sécurité [**38**].

9 Conclusion

Nous avons abordé dans ce deuxième chapitre la technologie Big Data, nous l'avons détaillé par ses définitions, caractéristiques ainsi que les différents domaines dans les quels cette technologie est utilisée. Nous avons présenté en fin du chapitre la relation entre le Big Data et la sécurité informatique qui est une liaison à double sens. La cyber sécurité est devenue un problème de Big Data car la taille et la complexité

2. https ://www.splunk.com/
3. https ://www.sumologic.com/
4. http ://metron.apache.org/

des données liées à la sécurité sont devenues trop importantes pour être gérées et analysées par les outils de sécurité traditionnels. L'arrivé de Big Data et les outils analytiques offrent la possibilité d'améliorer la connaissance de la situation, ainsi que la sécurité de l'information en détectant et prévenant les attaques en temps réel. Aussi, l'analyse de données offerte par le Big Data permet de prédire les attaques et les menaces dans un contexte de sécurité.

Finalement, nous avons pu voir comment le Big Data marque une avancée révolutionnaire aux différents domaines et surtout en ce qui concerne la cyber sécurité qui est le sujet de notre travail.

Dans le chapitre suivant nous présentons une étude sur l'analyse prédictive.

Chapitre 3

Analyse prédictive

1 Introduction

Prédire l'avenir est l'un des rêves de l'humanité et personne ne peut le faire. Aujourd'hui, il existe des méthodes et des algorithmes informatiques qui permettent de tracer les tendances et les évolutions à venir. C'est la mission et le but de l'analyse prédictive (Predictive Analytics). Cette méthode d'analyse de données est un sous-ensemble du Big Data et une technique de Data Mining qui révèle les modèles et les relations entre les données, et rend possible des prévisions de comportement futur sur la base de l'historique des bases de données, notamment dans le secteur du marketing, des finances, des assurances et même la sécurité informatique. L'apport que prodigue l'analyse prédictive se manifeste par une aide à la prise de décisions intelligentes, rapides et économiques.

2 Data Mining

Le data mining, ou fouille de données, est l'ensemble des méthodes et techniques destinées à l'exploration et l'analyse de bases de données informatiques (souvent grandes), de façon automatique ou semi-automatique, en vue de détecter dans ces données des règles, des associations, des tendances inconnues ou cachées, des structures particulières restituant l'essentiel de l'information utile tout en réduisant la quantité de données [**39**].

2.1 Principales tâches de Data Mining

De nombreuses tâches peuvent être associées au Data Mining, parmi elles nous citons :

— **La classification**

Dans ce modèle les groupes ou classes cibles sont connus dès le départ. La classification vise à expliquer une caractéristique qualitative à partir d'autre, et à examiner les caractéristiques d'un objet et lui attribuer une classe prédéfinie [**41**].

— **L'estimation**

Ce modèles est similaire à des modèles de classification, mais avec une différence majeure. Le modèle d'estimation est utilisé pour prédire et estimer la valeur d'un champ continu en fonction des valeurs observées des attributs d'entrée [**41**].

— **La prédiction**

La prédiction est similaire à la classification et à l'estimation, et donc les méthodes de classification et d'estimation peuvent être utilisées en prédiction, sauf que pour la prédiction, les résultats se situent dans le futur.

En effet, la prédiction consiste à estimer et prédire une valeur future d'un champ à partir des valeurs connues qui sont historiées [**40**].

— **L'association**

Cette tâche, plus connue comme l'analyse du panier de la ménagère, consiste à déterminer les variables qui sont associées. L'exemple type est la détermination des articles (le pain et le lait, la tomate, les carottes et les oignons) qui se retrouvent ensemble sur un même ticket de supermarché. Cette tâche peut être effectuée pour identifier des opportunités de vente croisée et concevoir des groupements attractifs de produit [**43**].

— **La segmentation ou clustering**

Le clustering est un processus de partitionnement d'un ensemble de données (ou d'objets) en un ensemble de sous-classes significatives, appelées clusters.

Pour cette tâche, il n'y a pas de classe à expliquer ou de valeur à prédire définie a priori, il s'agit de créer des groupes homogènes dans la population (l'ensemble des enregistrements) [**41**].

— **La régression**

L'analyse de régression est une méthodologie statistique qui est la plus souvent utilisée pour la prédiction numérique, bien que d'autres méthodes existent également. La régression englobe également l'identification des tendances de distribution à partir des données disponibles. La classification et la régression doivent être précédées d'une analyse de la pertinence, qui tente d'identifier les attributs significativement pertinents pour le processus de classification et de régression. Ces attributs seront sélectionnés pour le processus de classification et de régression. D'autres attributs, non pertinents, peuvent alors être exclus de la considération [**42**].

2.2 Techniques et algorithmes de Data Mining

Différents algorithmes et techniques tels que la classification, le regroupement, la régression, l'intelligence artificielle, les réseaux de neurones, les règles d'association, les arbres de décision, l'algorithme génétique et la méthode du plus proche voisin , etc., sont utilisés pour la découverte de connaissances à partir de bases de données. Cependant, tous les types d'algorithmes peuvent être classés en deux grandes catégories : apprentissage supervisé et apprentissage non supervisé [**44**].

2.2.1 Techniques supervisées

Dans la modélisation supervisée ou prédictive, dirigée ou ciblée, l'objectif est de prédire un événement ou d'estimer les valeurs d'un attribut numérique continu. Dans ces modèles, il y a des champs d'entrée ou des attributs et un champ de sortie ou cible. Les champs d'entrée sont également appelés prédicteurs, car ils sont utilisés par le modèle pour identifier une fonction de prédiction pour le champ de sortie **[45]**. L'apprentissage supervisé englobe les modèles d'estimation, de régression et de classification.
Le tableau suivant illustre quelques techniques supervisées de Data Mining :

Algorithmes	**Description**
K plus proches voisins(knn)	• KNN un algorithme simple, utilisé dans l'estimation statistique et la reconnaissance de formes. • Son principe est le suivant : Une donnée de classe inconnue est comparée à toutes les données stockées. On choisit pour la nouvelle donnée la classe majoritaire parmi ses K plus proches voisins en fonction d'une mesure de similarité (par exemple, des fonctions de distance) **[44]**.
Naïve de bayes	• Les modèles de classification naïve bayésienne sont des modèles de probabilité pouvant être utilisés dans les problèmes de classification pour estimer la probabilité d'occurrences. Ils sont des modèles graphiques qui fournissent une représentation visuelle des relations entre les attributs.
Machines à vecteurs supports (SVM)	• SVM est un algorithme qui peut modéliser les profils de données non linéaires hautement complexes. SVM fonctionne en données cartographiques à un espace de grande dimension caractéristique dans lequel les enregistrements deviennent plus facilement séparables (séparés par des fonctions linéaires) à l'égard des catégories cibles.

TABLE 3.1: Exemples d'algorithmes d'apprentissage supervisé de Data Mining

2.2.2 Techniques non supervisées

Dans les modèles non supervisés ou non dirigés, il n'y a pas de champ de sortie, juste des entrées. La reconnaissance de motif est non dirigée ; il n'est pas guidé par un attribut cible spécifique.Le but de ces modèles est de découvrir des modèles de données dans l'ensemble des champs de saisie **[45]**. L'apprentissage non supervisé englobe les modèles d'association et de clustering.
Le tableau suivant illustre quelques techniques non supervisées de Data Mining :

Algorithmes	Description
K moyennes (K means)	• K-Means est une technique de regroupement basée dans laquelle l'utilisateur commence par un ensemble d'échantillons et tente de regroupez-les en ≪k≫ nombre de clusters basés sur certaines mesures de distance spécifiques. • K-Means clustering génère un nombre spécifique de clusters disjoints et plats (non hiérarchiques) [**44**].
Clustring hiérarchique	• Il est considéré comme la "mère" de tous les modèles de clustering. Il est appelé hiérarchique ou d'agglomération, car il commence avec une solution dans laquelle chaque enregistrement comprend un cluster et regroupe progressivement les enregistrements jusqu'au point où tous tombent dans un super-cluster. À chaque étape, il calcule les distances entre toutes les paires d'enregistrements et les groupes les plus similaires.
Carte auto organitrice de Kohonen	• Les réseaux de Kohonen sont basés sur des réseaux de neurones et produisent généralement une grille ou une carte bidimensionnelle des clusters, d'où le nom "auto-organisation".Les réseaux Kohonen prennent généralement plus de temps à s'entraîner que les algorithmes K-means, mais ils fournissent un clustering d'être essayée.

TABLE 3.2: Exemples d'algorithmes d'apprentissage non supervisé de Data Mining

3 Analyse de données

Le Data Mining offre trois grands types d'analyse de données : l'analyse descriptive, l'analyse prédictive et l'analyse prescriptive.

— **L'analyse descriptive**

En réalité, ce type a (presque) toujours existé. Il s'agit de transformer des données en connaissance, pas à pas, en suivant des principes statistiques. L'analyse descriptive a pour but de résumer les données en leur assignant une nouvelle représentation, de synthétiser en faisant ressortir ce qui est dissimulé par le volume [**40**].

— **L'analyse prédictive**

Ce type est beaucoup plus récent, il consiste à analyser les données actuelles afin de faire des hypothèses sur des comportements futurs et de faciliter la prise de décision. Son but est de réduire les risque ainsi que d'identifier des opportunités [**41**].

— **L'analyse prescriptive**

Ce type a pour but d'optimiser le résultat des prédictions en orientant les décideurs vers le meilleur scénario. L'analyse prescriptive permet à répondre à la question "Que devrions-nous faire ?" [**40**].

4 Analyse prédictive

L'analyse prédictive, parfois appelée analyse avancée, est un terme utilisé pour décrire une série de techniques analytiques et statistiques permettant de prédire des actions ou des comportements futurs. Dans les entreprises, l'analyse prédictive est utilisée pour prendre des décisions proactives et déterminer des actions, au moyen de modèles statistiques permettant de découvrir des schémas dans des données

historiques et transactionnelles, dans le but d'identifier des risques potentiels et des opportunités [**46**].

4.1 Processus de l'analyse prédictive

Le processus de l'analyse prédictive [**?**] est composé des six étapes suivantes (Figure 1.3) :

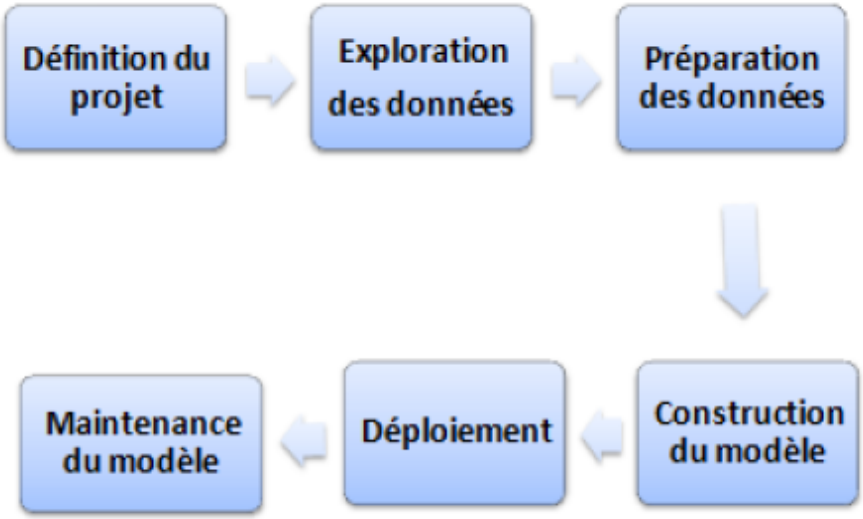

FIGURE 3.1: Processus de l'analyse prédictive

a) **Définition du projet :** Cette étape consiste à définir les objectifs à atteindre et les résultats attendus du projet et en déduire les spécifications et les tâches à réaliser.

b) **Exploration des données :** Cette phase consiste à analyser les sources de données afin de déterminer les données et l'approche du modèle les plus appropriés.

c) **Préparation des données :** C'est choisir, extraire et transformer les données avec lesquels les modèles seront créés.

d) **Construction du modèle :** Cette étape consiste à créer, tester et valider les modèles et confirmer qu'ils permettent l'aboutissement des objectifs du projet.

e) **Déploiement :** C'est l'intégration du modèle prédictif dans les processus existants afin de faciliter ou d'automatiser la prise de décision.

f) **Maintenance du modèle :** Enfin, cette étape consiste à entretenir les modèles pour en améliorer les performances, contrôler les accès, permettre la réutilisation et en standardiser la forme et l'utilisation.

5 Les techniques de prédiction

Il existe plusieurs techniques de prédiction, dans ce qui suit nous allons citer quelques unes.

5.1 Les arbres de décision

Un arbre de décision est un schéma représentant les résultats possibles d'une série de choix inter connectés. Il permet à une personne ou une organisation d'évaluer différentes actions possibles en fonction de leur coût, leur probabilité et leurs bénéfices. Il peut être utilisé pour alimenter une discussion informelle ou pour générer un algorithme qui détermine le meilleur choix de façon mathématique [**47**]. Un arbre de décision commence généralement par un nœud d'où découlent plusieurs résultats possibles. Chacun de ces résultats mène à d'autres nœuds, d'où émanent d'autres possibilités. Le schéma ainsi obtenu rappelle la forme d'un arbre.

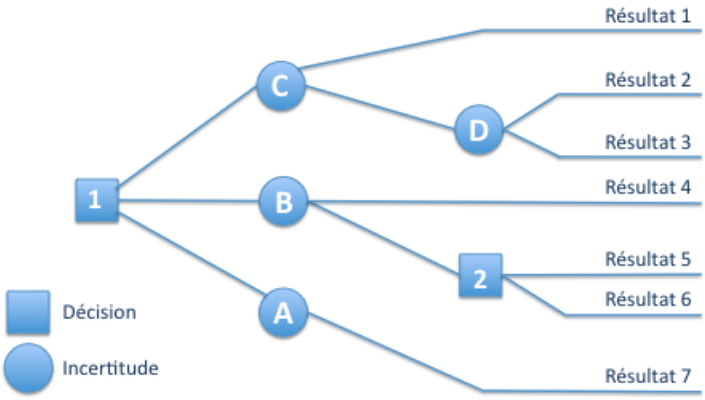

FIGURE 3.2: Architecture d'un arbre de décision

5.2 Les réseaux de neurones

Un réseau de neurones est un modèle de calcul dont le fonctionnement schématique est inspiré du fonctionnement des neurones biologique. Chaque neurone fait une somme pondérée de ses entrées (ou synapses) et retourne une valeur en fonction de sa fonction d'activation. Cette valeur peut être utilisée soit comme une des entrées d'une nouvelle couche de neurones, soit comme un résultat qu'il appartient à l'utilisateur d'interpréter (classe, résultat d'un calcul, etc.) [**48**].

La phase d'apprentissage d'un réseau de neurones permet de régler le poids associé à chaque synapse d'entrée (on parle également de coefficient synaptique). C'est un processus long qui doit être réitéré à chaque modification structurelle de la base de données traitée.

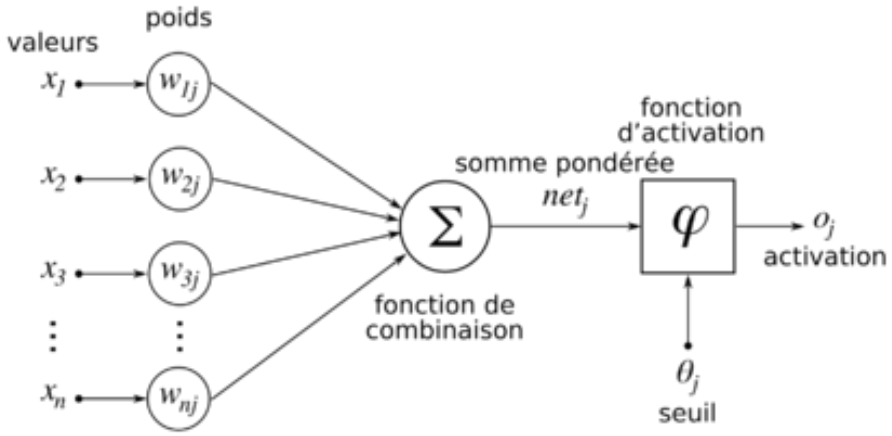

FIGURE 3.3: Structure d'un réseau de neurone

5.3 Les K plus proches voisins

L'algorithme des k-plus proches voisins (k-ppv) ou k-nearest neighbors en anglais (K-NN) est une méthode d'apprentissage supervisé dédiée à la classification, elle a été considérée parmi les plus simples algorithmes d'apprentissage artificiel.

Pour prédire la classe d'un exemple donné, l'algorithme cherche les K plus proches voisins de ce nouveau cas et prédit la réponse la plus fréquente de ces K plus proches voisins. Le principe de décision consiste tout simplement donc a calcules la distance de l'exemple inconnu a tous les échantillons fournis. L'exemple est alors affecté à la classe majoritaire représenté parmi ces K échantillons. La méthode utilise deux paramètres : le nombre K et la fonction de similarité pour comparer le nouvel exemple aux exemples déjà classés [**49**].

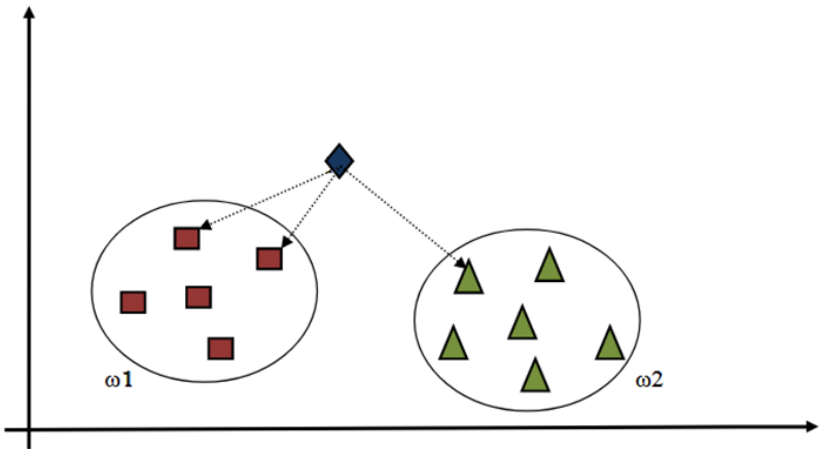

FIGURE 3.4: Principe de fonctionnement de l'algorithme K-ppv

- **Le choix de k :**

Le paramètre k doit être déterminé par l'utilisateur : k ∈ N, il est utile de choisir k impair pour éviter les votes égalitaires. Le meilleur choix de k dépend du jeu de donnée. La fixation du paramètre k est délicate,

une valeur très faible va engendrer une forte sensibilité. Un k trop grand va engendrer un phénomène d'uniformisation des décisions.
Pour remédier à ce problème, il faut tester plusieurs valeurs de k et choisir le k optimal qui donne un meilleur taux de classification

5.4 La régression logistique

La régression logistique est une méthode d'analyse statistique qui consiste à prédire une valeur de données d'après les observations réelles d'un jeu de données.
La régression logistique est devenue un outil important dans la discipline de l'apprentissage automatique. Cette approche permet d'utiliser un algorithme dans l'application d'apprentissage automatique pour classer les données entrantes en fonction des données historiques. Plus il y a de données pertinentes en entrée, plus l'algorithme est en mesure de prédire des classifications au sein des jeux de données **[50]**.

6 Prédiction et sécurité informatique

Au cours des dernières années, les cyberattaques ont connu une croissance rapide et considérable. Malgré l'existence de systèmes de cyber-défense avancés, les attaques et les intrusions se produisent encore. Les systèmes de défense ont tenté de bloquer les attaques déjà connues, d'arrêter les attaques en cours et de détecter les attaques survenues. Cependant, les dégâts causés par une attaque sont souvent catastrophique. Par conséquent, la nécessité d'améliorer les systèmes de détection d'intrusion et de proposé un système de prédiction robuste est plus urgente aujourd'hui.
Dans cette partie, nous allons parler brièvement des systèmes de prévision d'intrusion pour montrer la nécessité d'un tel système, l'insuffisance des systèmes de détection d'intrusion actuels et la manière dont la prédiction permettra d'améliorer les capacités de sécurité pour les systèmes de défense.

6.1 Le rôle de la prédiction d'intrusion

La prédiction est utilisée dans de nombreux domaines tels que le marché boursier, les prévisions météorologiques, le secteur de la santé et autres. Bien que les systèmes de prévision d'intrusions ne sont pas encore largement utilisés dans l'industrie du cyber. A ce titre, nous clarifierons l'importance de la prédiction d'intrusions à partir de termes apparentés, à savoir détection et prévention.
La détection d'intrusion est «le processus de surveillance des événements survenant dans un système informatique ou un réseau et de les analyser pour rechercher des signes d'intrusion, définis comme des tentatives de compromettre la confidentialité, l'intégrité, la disponibilité ou de contourner les mécanismes de sécurité d'un ordinateur ou d'un réseau».
La prévention des intrusions est le processus consistant à détecter les intrusions et à essayer de les arrêter. Par conséquent, selon ces définitions, la prévention dépend de la détection et toutes les deux reposent sur des incidents de sécurité surveillés et identifiés, c'est à dire que ces incidents se sont déjà produits ou sur

le point de se produire.
Il est important de noter que les systèmes de détection et de prévention ont besoin d'informations directes pour déclencher une alarme de violation de la sécurité ou interdire les attaques connues. De plus, ces systèmes n'ont pas pu identifier les attaques en plusieurs étapes, car le concept de détection consiste à capturer un seul incident à la fois, et non l'ensemble de la série. Dans ce contexte, le besoin de systèmes de prédiction est apparu comme un outil permettant de prendre en charge les systèmes de détection et de prévention. L'idée de base du concept de prédiction est la tentative de fournir des informations sur des événements qui ne se sont pas encore produits en fonction d'informations historiques et de connaissances acquises d'événements similaires ou des mêmes événements survenus dans le passé [**51**].

6.2 Travail connexe

Dans son travail R.P. Menon [**52**] a proposé un système de prédiction d'intrusion dans lequel des entrées de fichier journal réseau sont utilisées pour la prédiction d'attaques. Les techniques utilisées dans ce travail sont les algorithmes Naive Bayes et Adaboost Cost Sensitive Learning. Les fichiers journaux réseau obtenus à partir de périphériques réseau tels que IDS et Firewalls sont collectés, normalisés et corrélés à l'aide d'Alienvault SIEM afin d'obtenir plus d'informations pour un certain type d'attaques. Les fichiers journaux corrélés sont prétraités pour extraire et les champs importants pour la classification. Ensuite, les données d'entraînement sont classées à l'aide de Naive Bayes et les entrées mal classées sont transmises à la variante Cost Sensitive d'Adaboost, qui améliore le taux de classification. Avec l'aide de ces données d'entrainement, le système crée un modèle d'attaque à l'aide duquel il prédit si une attaque est sur le point de se produire ou non.

7 Conclusion

Dans ce chapitre nous avons présenté les concepts de Data Mining et de l'analyse prédictive. Nous avons abordé dans un premier temps le Data Mining pour montrer l'utilité de ce dernier dans l'extraction de l'information à partir de grande quantité de données.
Nous avons ensuite présenté l'analyse prédictive comme étant une approche très puissante pour l'étude et l'estimation de futurs phénomènes susceptible d'avoir un mauvais impact sur le bon déroulement des activités d'un domaine précis.
Finalement nous avons montré comment l'analyse prédictive constitue un outil très important dans le domaine de la sécurité informatique et une véritable opportunité pour les décideur afin de détecter et prévoir les attaques informatiques.
Dans le chapitre suivant nous allons détailler la conception de notre système pour l'analyse des fichiers log et la prédiction des attaques.

Chapitre 4

Conception de la solution

1 Introduction

Les équipements réseaux, les logiciels de sécurité, les serveurs Web et les systèmes d'exploitation sont tous à l'exposition des cybers menaces et comme nous avons déjà mentionné dans les chapitres précédents, l'analyse des fichiers logs constitue le moyen idéal pour la détection d'anomalies et la surveillance des systèmes.
Lors de ce chapitre, nous allons présenter notre solution proposée qui est une expérimentation de l'utilisation des algorithmes de machine learning dans l'analyse des fichiers log des serveurs web afin de prédire les attaques.

2 Architecture générale du système

Le schéma suivant représente l'architecture générale de notre système :

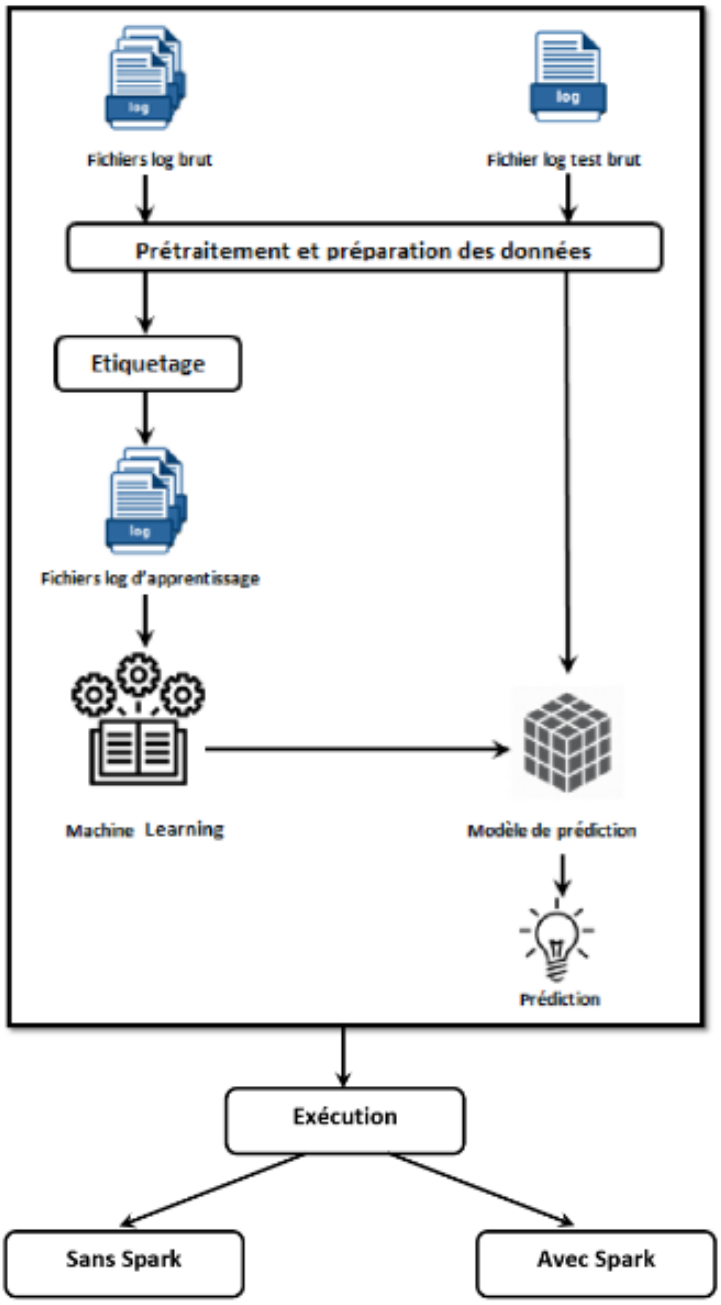

FIGURE 4.1: Architecture générale du système

Au début de l'analyse les évènements enregistrés par un serveur web sont traités et nettoyés, ensuite ces données préparées vont être utilisé dans la phase d'étiquetage afin d'identifier l'état de chaque ligne du fichier log si elle est considérée comme comportement normal ou anormal et attribuer une étiquette pour cette valeur, une fois les données sont étiquetés, on obtient ce qu'on appelle le corpus training ou les données d'apprentissage. Ces données sont prêtes maintenant à former nos classificateurs pour obtenir le modèle prédictif.

A la fin le corpus de test ou bien le fichier log de test est prêt pour être appliqué au travers le modèle

de prédiction pour prédire les attaques. Lorsque ce programme d'analyse est prêt, l'administrateur peut choisir ou ne pas choisir spark comme moteur d'exécution et visualiser les résultats d'analyse.

3 Démarche suivie pour la conception du système

Dans cette partie, nous abordons l'aspect technique de la solution proposée, nous expliquons entre autres les algorithmes de Machine Learning utilisés pour prédire les attaques, le traitement qui précède la prédiction et d'autres points en relation avec notre solution.

3.1 Préparation des données

Notre travail consiste à faire l'analyse des fichiers log pour la prédiction des attaques, pour cela nous avons choisis d'utiliser les algorithmes prédictives d'apprentissage automatique.
Dans notre solution nous avons proposé l'utilisation d'algorithmes supervisés, ce qui signifie qu'il doivent être formé aux données étiquetées avant de pouvoir être utilisé pour la prédiction. Ainsi, les données d'entraînement doivent être étiquetées et nous devons choisir les caractéristiques que nous utiliserons pour la prédiction.
La préparation des données consiste a extraire les caractéristiques utiles des fichiers journaux bruts du serveur http et à les étiqueter à l'aide de deux étiquettes

— **Nettoyage des champs**

Notre travail consiste à faire l'analyse des fichiers logs d'un serveur web. Ce type de fichier contient plusieurs champs comme l'adresse ip de l'utilisateur, l'horodatage, l'url, le type de retour, la méthode utilisée ... etc. Pour analyser ces fichiers il est nécessaire d'enlever les champs inutiles et qui ne vont pas nous aider dans notre cas d'étude.

— **Extraction de caractéristiques**

L'extraction de caractéristiques est la transformation de données originales en un ensemble de données avec un nombre réduit de variables, qui contient les informations les plus discriminantes.
Dans notre cas, les entités suivantes sont extraites du fichier journal brut :

• Code de retour HTTP.

• L'URL de la requête.

Les fonctionnalités suivantes sont ajoutées et calculées :

• Longueur de l'URL.

• Nombre de paramètres dans la requête.

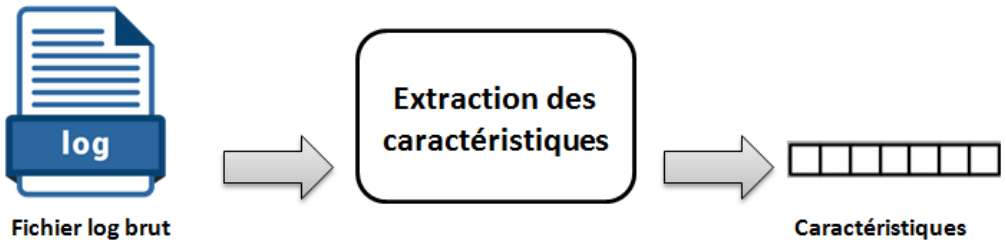

FIGURE 4.2: Extraction de caractéristiques

— **Étiquetage des données :**

L'étiquetage consiste à attribuer une étiquette à chaque unité de données (dans notre cas chaque ligne du fichiers log). Cette étiquette indiquera si la ligne de journal concernée est considérée ou non comme une attaque.

L'étiquetage doit normalement être effectué manuellement par un ingénieur en sécurité expérimenté. Dans notre travail de recherche, il est effectué automatiquement à l'aide d'un système de détection qui recherche des modèles spécifiques dans chaque URL et décide s'il s'agit d'une attaque. Pour cela,, nous avons ajouté un champ qui indiquera 1 si la ligne du log est considérée comme attaque et 0 si la ligne représente un comportement normal.

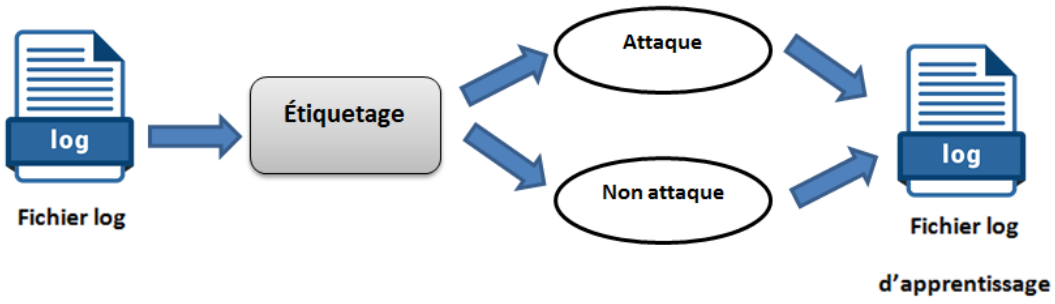

FIGURE 4.3: Étiquetage des données

3.2 Analyse prédictive

Lors de la conception d'un système prédictif, on se trouve généralement face à une multitude de techniques et méthodes prédictives, qui semblent à première vue toutes rapides et efficaces. Cependant, si l'on regarde de plus prés, chaque méthode présente des avantages qui la rendent imbattable sur une catégorie de prédictions, et des inconvénients la rendant inutilisable dans d'autres catégories.

Pour notre système nous avons choisi d'utiliser trois algorithmes d'apprentissage automatique (K-plus proches voisins, arbre de décision, régression logistique) afin d'aboutir à de meilleurs résultats de prédiction.

— **Construction du modèle prédictif :**

Après avoir traiter, nettoyer et étiqueter nos fichiers log, nous passons maintenant à trainer et former nos classificateurs pour obtenir le modèle prédictif.

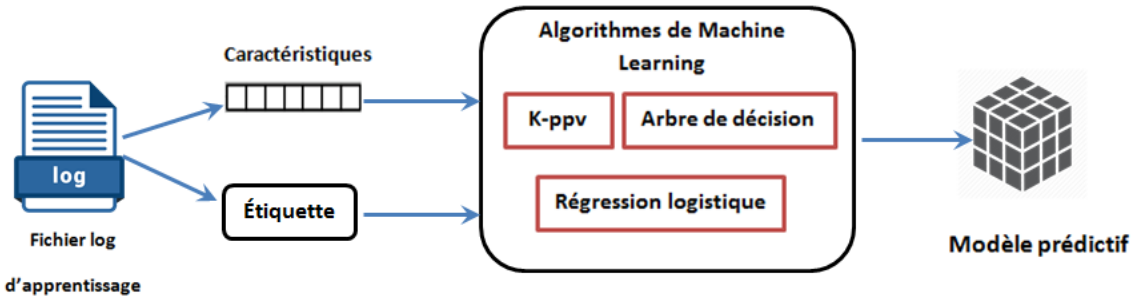

FIGURE 4.4: Construction du modèle prédictif

Un modèle définit la relation entre les caractéristiques et l'étiquette.

Dans cette étape nous avons présenté les fichiers log étiquetés au différents algorithmes pour entraîner ces classificateurs et prendre progressivement les relations entre les caractéristiques et l'étiquette, cette phase permet de construire le modèle prédictif.

— **Application du modèle prédictif :**

Cette étape consiste à appliquer le modèle prédictif à des fichiers log test (Sans étiquettes) pour faire de la prédiction.

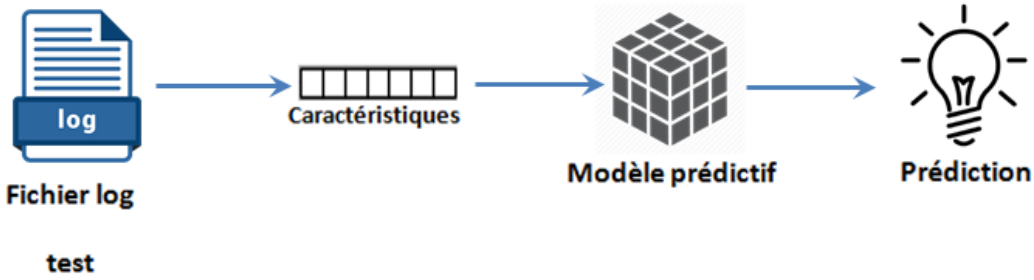

FIGURE 4.5: Application du modèle prédictif

— **Évaluation du modèle prédictif :**

Nous devons toujours évaluer un modèle pour déterminer s'il contribuera à prédire correctement la cible dans le cadre de nouvelles données à venir. Comme les instances futures ont des valeurs cibles inconnues, nous devons vérifier la métrique de précision du modèle d'apprentissage-machine sur des données dont on connais déjà la réponse cible, puis utiliser cette évaluation comme indicateur de la précision prédictive des données futures.

Parmi les métriques d'évaluation, nous avons choisis la précision.

• La précision est le nombre de données pertinents retrouvés rapporté au nombre de données total proposé par le moteur de recherche pour une requête donnée.

4 Lancement du système avec Spark

Pour accélérer l'analyse nous avons pensé à lancer notre programme sous Spark.
Spark est un moteur de traitements Big Data open source construit pour effectuer un traitement de larges volumes de données de manière distribuée et conçu pour la rapidité et la facilité d'utilisation.

— **Comment lancer un programme dans Spark ? :**
Malgré l'utilisation d'un gestionnaire de cluster, Spark est doté d'un script unique permettant de soumettre un programme appelé spark-submit . Il lance l'application sur le cluster. Il existe diverses options permettant à spark-submit de se connecter à différents gestionnaires de cluster et de contrôler le nombre de ressources que notre application reçoit.

5 Conclusion

L'architecture proposée comporte toutes les étapes nécessaires d'analyse d'un fichier log, qui sont : le pré-traitement, la préparation des données, le traitement qui consiste à faire la détection et la prédiction des attaques et enfin la présentation des résultats aux administrateurs de sécurité. Dans cette solution nous avons utilisé les K plus proches voisins, la régression logistique ainsi que l'arbre de décision afin de concevoir notre modèle prédictif.
Dans le chapitre suivant nous allons présenter les différents outils qui vont servir à l'implémentation de notre projet, ainsi que l'implémentation de notre solution et les discussions sur la réalisation.

Chapitre 5

Implémentation et Réalisation

1 Introduction

Après avoir présenté dans les chapitres précédents, les différents concepts théoriques liés à notre travail et en se basant sur l'architecture présentée dans le chapitre précédent, nous avons développé un système d'analyse des fichiers logs et prédiction des attaques. Au cours de ce chapitre, nous définirons d'abord nos choix logiciels et matériels pour la réalisation de notre projet. Ensuite, nous présenterons les différentes étapes d'installation et de préparation suivie par le déroulement du programme d'analyse et prédiction. Enfin, nous terminerons par des tests de performances de notre solution.

2 Les ressources matérielles et logicielles

Dans cette section, nous vous présentons les ressources matérielles et logicielles que nous avons utilisées pour mettre en œuvre le cas d'utilisation proposé :

2.1 Matériels utilisés

L'implémentation de notre système a été réalisée sur une machine virtuelle possédant les caractéristiques suivantes :

- Processeur : 1 core.
- Mémoire : 2 Go de RAM.
- Disque dur : 100 Go.

2.2 Logiciels utilisés

a) Système d'exploitation :

— **Ubuntu 16.02** : Ubuntu est un système d'exploitation linux complet, il est adapté pour être utilisé comme poste de travail ou serveur.

b) Outils de développement :

— **Python3** : Python est un langage de programmation interprété à usage général, interactif, orienté objet et de haut niveau. Python combine une puissance remarquable avec une syntaxe claire. Il comporte des modules, des classes, des exceptions, des types de données dynamiques de très haut niveau et un typage dynamique. Il existe des interfaces pour de nombreux appels système et bibliothèques, ainsi que pour divers système de fenêtrage.

— **Bibliothèque Python3 Sckit-Learn** : Scikit-learn est une bibliothèque libre développée en Python destinée à l'apprentissage automatique (machine learning), elle propose plusieurs types d'algorithmes de classification , régression et regroupement et peut être utilisée comme middleware, notamment pour des tâches de prédiction.

c) Cluster de traitement :

— **Apache spark** : Apache Spark est un moteur de traitement de données rapide dédié au Big Data. Il permet d'effectuer un traitement de larges volumes de données de manière distribuée. Ses principaux avantages sont sa vitesse, sa simplicité d'usage, et sa polyvalence.

3 Installation de l'environnement (Spark)

Nous allons maintenant passer à l'installation de spark, le Framework qui a été utilisé pour accélérer le traitement. Cette installation est appliquée pour les systèmes d'exploitation linux.

- **Installation java**

Spark a besoin de Java pour fonctionner. Sur le terminal nous avons exécuté les commandes suivantes :

```
$ sudo apt-get update
$ sudo apt-get upgrade
$ sudo apt-get install openjdk-8-jdk
```

Ensuite, il faut tester l'installation de java en tapant :

```
$ java -version
```

Le résultat doit être similaire à ceci :

```
openjdk version "1.8.0_131"
OpenJDK Runtime Environment (build 1.8.0_131-8u131-b11-2ubuntu1.17.0
4.3-b11)
OpenJDK 64-Bit Server VM (build 25.131-b11, mixed mode)
```

- **Installation de Spark**

Tout d'abord il faut télécharger une version de spark, dans notre cas nous avons téléchargé spark 2.2.0 pre-built disponible sur le site suivant :

```
https://spark.apache.org/downloads.html
```

Maintenant nous avons la source compressée de spark, il faut la décompresser et déplacer le dossier résultant dans le bureau :

```
$ tar zxvf spark-2.2.0-bin-hadoop2.7.tgz
$ mv spark-2.2.0-bin-hadoop2.7 ~
```

Finalement, il faut définir la variable d'environnement nécessaire SPARK_HOME.
Pour ce faire, il faut aller sur le répertoire personnel et ouvrir le fichier '.bashrc' :

```
$ cd ~
$ sudo nano .bashrc
```

Et Ajoutez les lignes suivantes à la fin du fichier :

```
# Spark
export SPARK_HOME="/home/<your_username>/spark-2.2.0-bin-hadoop2.7/"
```

Enregistrez le fichier : ctrl+ x, écrivez y et appuyez sur return pour le faire.
Nous pouvons maintenant vérifier notre installation de spark :

```
$ cd spark-2.2.0-bin-hadoop2.7
$ bin/pyspark
```

Si tout est correctement installé, la sortie doit être semblable à ceci :

```
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /__ / .__/\_,_/_/ /_/\_\   version 2.2.0
      /_/
Using Python version 3.6.1 (default, May 11 2017 13:09:58)
SparkSession available as 'spark'.
```

4 Préparation du système

Après avoir installé les logiciels nécessaires, nous allons présenter dans cette partie la préparation de nos données d'analyse et l'implémentation des fonctions et des algorithmes de notre application.

4.1 Préparation de données

Afin tester notre application, nous avons utilisé des fichiers journaux du serveur web (fichier d'accès) disponibles sur http ://www.secrepo.com/self.logs/.

La figure ci-dessous représente les dix premières lignes d'un fichier log brut.

```
77.75.78.160 - - [15/Dec/2018:02:18:31 -0800] "GET /robots.txt HTTP/1.1" 200 334 "-" "Mozilla/5.0 (compatible; SeznamBot/3.2; +http://napoveda.seznam.cz/en/seznambot-intro/)"
77.75.78.160 - - [15/Dec/2018:02:18:34 -0800] "GET /self.logs/access.log.2018-09-18.gz HTTP/1.1" 200 26471 "-" "Mozilla/5.0 (compatible; SeznamBot/3.2; +http://napoveda.seznam.cz/en/seznambot-intro/)"
77.75.78.160 - - [15/Dec/2018:02:18:37 -0800] "GET /self.logs/access.log.2018-06-04.gz HTTP/1.1" 200 19964 "-" "Mozilla/5.0 (compatible; SeznamBot/3.2; +http://napoveda.seznam.cz/en/seznambot-intro/)"
77.75.78.160 - - [15/Dec/2018:02:18:38 -0800] "GET /self.logs/error.log.2017-01-28.gz HTTP/1.1" 200 1436 "-" "Mozilla/5.0 (compatible; SeznamBot/3.2; +http://napoveda.seznam.cz/en/seznambot-intro/)"
77.75.78.160 - - [15/Dec/2018:02:18:38 -0800] "GET /self.logs/error.log.2018-11-10.gz HTTP/1.1" 200 4470 "-" "Mozilla/5.0 (compatible; SeznamBot/3.2; +http://napoveda.seznam.cz/en/seznambot-intro/)"
77.75.78.160 - - [15/Dec/2018:02:18:38 -0800] "GET /self.logs/error.log.2018-02-22.gz HTTP/1.1" 200 4302 "-" "Mozilla/5.0 (compatible; SeznamBot/3.2; +http://napoveda.seznam.cz/en/seznambot-intro/)"
3.16.186.19 - - [15/Dec/2018:02:32:45 -0800] "GET / HTTP/1.1" 200 12470 "-" "Mozilla/5.0 (Windows NT 6.1; Win64; x64; rv:47.0) Trident/5.0 Safari/602.1"
157.55.39.93 - - [15/Dec/2018:02:32:49 -0800] "GET /self.logs/access.log.2017-08-03.gz HTTP/1.1" 200 20556 "-" "Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.htm)"
211.209.217.125 - - [15/Dec/2018:02:36:40 -0800] "GET /wp-login.php HTTP/1.1" 404 295 "-" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:40.0) Gecko/20100101 Firefox/D1BE"
211.209.217.125 - - [15/Dec/2018:02:36:46 -0800] "GET /wp-login.php HTTP/1.1" 404 295 "-" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:40.0) Gecko/20100101 Firefox/D1BE"
```

FIGURE 5.1: Fichier log serveur web brut

4.2 Prétraitement et nettoyage

La figure suivante montre comment nous avons fait le prétraitement du fichier log.

```
def extract_data(log_file):
        regex = '([(\d\.)]+) - - \[(.*?)\] "(.*?)" (\d+) (.+) "(.*?)" "(.*?)"'
        data = {}
        log_file = open(log_file, 'r')
        for log_line in log_file:
                log_line=log_line.replace(',','_')
                log_line = re.match(regex,log_line).groups()
                size = str(log_line[4]).rstrip('\n')
                return_code = log_line[3]
                url = log_line[2]
                param_number = len(url.split('&'))
                url_length = len(url)
                if '-' in size:
                        size = 0
                else:
                        size = int(size)
                if (int(return_code) > 0):
                        charcs = {}
                        charcs['size'] = int(size)
                        charcs['param_number'] = int(param_number)
                        charcs['length'] = int(url_length)
                        charcs['return_code'] = int(return_code)
                        data[url] = charcs
        return data
```

FIGURE 5.2: Méthode de prétraitement et nettoyage

Les fichiers logs possèdent des données qui ne vont rien apporter à notre analyse, elles seront donc filtrées. Pour cela nous sommes amenés à créer cette fonction qui supprime par exemple des caractères comme ”,” et ”-” et qui fait l'extraction des champs et des données qui vont être utiles pour notre travail. La figure ci-dessous est un extrait du fichier log résultant sous format CSV.

24	1	200	0	GET /robots.txt HTTP/1.1
48	1	200	0	GET /self.logs/access.log.2018-09-18.gz HTTP/1.1
48	1	200	0	GET /self.logs/access.log.2018-06-04.gz HTTP/1.1
47	1	200	0	GET /self.logs/error.log.2017-01-28.gz HTTP/1.1
47	1	200	0	GET /self.logs/error.log.2018-11-10.gz HTTP/1.1
47	1	200	0	GET /self.logs/error.log.2018-02-22.gz HTTP/1.1
14	1	200	0	GET / HTTP/1.1
48	1	200	0	GET /self.logs/access.log.2017-08-03.gz HTTP/1.1
26	1	418	0	GET /wp-login.php HTTP/1.1
47	1	200	1	GET /honeypot/Honeypot%20-%20Howto.pdf HTTP/1.1

FIGURE 5.3: Résultat du prétraitement du fichier log

Ce fichier contient les champs suivants :

- Taille de la requête.
- Nombre de paramètre dans la requête.
- Code du retour http.

- Champ qui indique si la ligne est une attaque ou non.
- La requête (URL)

Ces données d'apprentissage vont être utilisées pour trainer les algorithmes de prédiction.

4.3 Les algorithmes de prédiction

Dans notre application, nous avons utilisé trois algorithmes de classifications, à savoir : l'arbre de décision, la régression logistique et l'algorithme k plus proches voisins. Ces algorithmes ont besoin des données d'apprentissage pour construire un modèle prédictif et des données de test pour effectuer des prévisions. L'implémentation de ces classificateurs a été faite à l'aide de la bibliothèque Sckit-Learn de Python.
La figure suivante représente l'implémentation de la méthode des K plus proches voisins (k-nearest neighbors).

```
args = get_args()
traning_data = args['traning_data']
testing_data = args['testing_data']

training_features, traning_labels = get_data_details(traning_data)

testing_features, testing_labels = get_data_details(testing_data)

print ("\n\n=-=-=-=-=-=-=-= K plus proches voisins -=-=-=-=-=-=-=-\n")

attack_classifier = neighbors.KNeighborsClassifier(n_neighbors=5)

attack_classifier = attack_classifier.fit(training_features, traning_labels)

predictions = attack_classifier.predict(testing_features)

print ("La precision de K-ppv est: " + str(get_occuracy(testing_labels,predictions, 1)) + "%")

print ("temps d'execution : %s secondes " % ((time.time() - start_time)))
```

FIGURE 5.4: Implémentation du K-ppv

Cette implémentation permet de faire les fonctionnalités suivantes :

- Obtenir les caractéristiques et l'étiquette de chaque ensemble de données (données d'apprentissage, données de tests).
- Instancier le classificateur K-ppv.
- Trainer le classificateur en utilisant les données d'apprentissage (training data) et construire le modèle prédictif.
- Et finalement appliquer le model prédictif sur les données de test et obtenir les prédictions.

Comme le K-ppv, l'arbre de décision et la régression logistique sont mis en œuvre de la même façon.

5 Mise en marche du système

Dans cette section, nous allons mettre en marche le système d'analyse des fichiers logs, nous allons présenter les commandes d'exécution de notre application ainsi que le visionnage des résultats obtenus.

5.1 Lancement du prétraitement et nettoyage

Tout d'abord, nous allons commencer par le nettoyage et le prétraitement du fichier log en exécutant la commande suivante :

$ python pretraitement.py -l ./fichiers-logs/training-log.log -d ./fichiers/training-log.csv

Cette commande permet de prendre en entrée le fichier log brut et donne en sortie un fichier log prétraité et étiqueté sous format csv.

La figure ci-dessous représente un l'affichage des 10 premières lignes d'un fichier log avant et après le prétraitement.

.

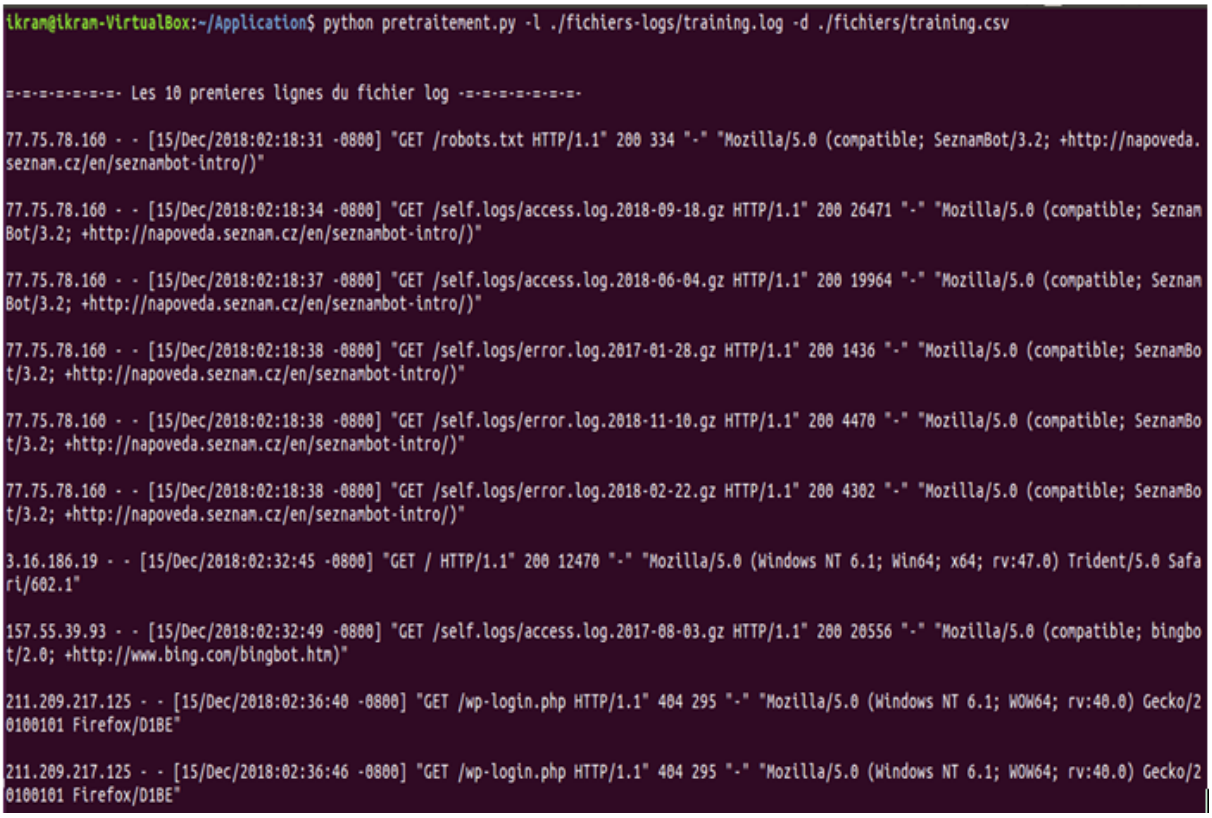

FIGURE 5.5: Affichage du log avant le prétraitement

```
les lignes ont ete enregistrees avec succes dans les fichier
./fichiers/training.csv

=-=-=-=-=-=-=- Les memes lignes du fichier log aprés les prétraitement -=-=-=-=-=-=-=-
24,1,200,0,GET /robots.txt HTTP/1.1
48,1,200,0,GET /self.logs/access.log.2018-09-18.gz HTTP/1.1
48,1,200,0,GET /self.logs/access.log.2018-06-04.gz HTTP/1.1
47,1,200,0,GET /self.logs/error.log.2017-01-28.gz HTTP/1.1
47,1,200,0,GET /self.logs/error.log.2018-11-10.gz HTTP/1.1
47,1,200,0,GET /self.logs/error.log.2018-02-22.gz HTTP/1.1
14,1,200,0,GET / HTTP/1.1
48,1,200,0,GET /self.logs/access.log.2017-08-03.gz HTTP/1.1
26,1,418,0,GET /wp-login.php HTTP/1.1
47,1,200,1,GET /honeypot/Honeypot%20-%20Howto.pdf HTTP/1.1
```

FIGURE 5.6: Affichage du log aprés le prétraitement

5.2 Lancement des programmes de prédiction

Une fois nous avons les données d'apprentissage, nous pouvons maintenant exécuter nos programmes de prédictions :

Chacun de ces algorithmes prend en entrée les données fichiers log de d'apprentissage et les fichiers log de test et donne en sortie un affichage qui contient :

- Le nombre réel des attaques.
- Le nombre prévu d'attaques.
- La précision de l'algorithme.
- Le temps d'exécution de l'algorithme.

— **Lancement du programme K-ppv** :

```
ikram@ikram-VirtualBox:~/Application$ python knn.py -t ./fichiers/training.csv -v ./fichiers/test_log.csv


=-=-=-=-=-=-=- K plus proches voisins -=-=-=-=-=-=-=-

Nombre reel d'attaques: 257.0
Nombre prevu d'attaques: 218.0
La precision de K-ppv est: 84.8249027237354%
temps d'execution : 19.030494689941406 secondes
```

FIGURE 5.7: Résultat de prédiction avec K-ppv

.

— **Lancement du programme d'arbre de décision** :

```
ikram@ikram-VirtualBox:~/Application$ python arbre-decision.py -t ./fichiers/training.csv -v ./fichiers/test_log.csv


=-=-=-=-=-=-=- Arbre de decision -=-=-=-=-=-=-=-

Nombre reel d'attaques: 257.0
Nombre prevu d'attaques: 184.0
La precision de l'arbre de decision est: 71.59533073929961%
temps d'execution : 3.190889596939087 secondes
```

FIGURE 5.8: Résultat de prédiction avec l'arbre de décision

— **Lancement du programme de régression logistique** :

```
ikram@ikram-VirtualBox:~/Application$ python regression-logistique.py -t ./fichiers/training.csv -v ./fichiers/test_log.csv


=-=-=-=-=-=-=- Regression logistic -=-=-=-=-=-

Nombre reel d'attaques: 257.0
Nombre prevu d'attaques: 32.0
La precision de la regression logistique est: 12.45136186770428%
temps d'execution : 4.542421102523804 secondes
```

FIGURE 5.9: Résultat de prédiction avec la régression logistique

5.3 Lancement des programmes de prédiction avec Spark

Comme nous avons déjà mentionné dans le chapitre précédent, nous pouvons exécuter notre programme d'analyse sur le cluster de traitement spark.

Afin de lancer les différents programmes sous spark, nous avons exécuté la commande suivante :

$ ./bin/spark-submit nom-du programme.py -t ./fichiers /training.csv -v ./fichiers/test-log.csv

Les figures suivantes représentent les résultats d'exécution des algorithmes sous spark.

```
ikram@ikram-VirtualBox:~/spark-2.2.0-bin-hadoop2.7$ ./bin/spark-submit knn.py -t ./fichiers/training.csv -v ./fichiers/test_log.csv
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
19/09/17 22:08:32 WARN Utils: Your hostname, ikram-VirtualBox resolves to a loopback address: 127.0.1.1; using 10.0.2.15 instead (on interface
enp0s3)
19/09/17 22:08:32 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another address


=-=-=-=-=-=-=- K plus proches voisins -=-=-=-=-=-=-=-

Nombre reel d'attaques: 257.0
Nombre prevu d'attaques: 218.0
La precision de K-ppv est: 84.8249027237354%
temps d'execution : 5.907429933547974 secondes
```

FIGURE 5.10: Résultat d'exécution du K-ppv sous spark

.

```
ikram@ikram-VirtualBox:~/spark-2.2.0-bin-hadoop2.7$ ./bin/spark-submit arbre-decision.py -t ./fichiers/training.csv -v ./fichiers/test_log.csv
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
19/09/17 22:10:47 WARN Utils: Your hostname, ikram-VirtualBox resolves to a loopback address: 127.0.1.1; using 10.0.2.15 instead (on interface
enp0s3)
19/09/17 22:10:47 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another address


=-=-=-=-=-=-=- Arbre de decision -=-=-=-=-=-=-=-

Nombre reel d'attaques: 257.0
Nombre prevu d'attaques: 184.0
La precision de l'arbre de decision est: 71.59533073929961%
temps d'execution : 0.6709232330322266 secondes
```

FIGURE 5.11: Résultat d'exécution d'arbre de décision sous spark

```
ikram@ikram-VirtualBox:~/spark-2.2.0-bin-hadoop2.7$ ./bin/spark-submit regression-logistique.py -t ./fichiers/training.csv -v ./fichiers/test_l
og.csv
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
19/09/17 22:07:08 WARN Utils: Your hostname, ikram-VirtualBox resolves to a loopback address: 127.0.1.1; using 10.0.2.15 instead (on interface
enp0s3)
19/09/17 22:07:08 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another address


=-=-=-=-=-=-=- Regression logistic -=-=-=-=-=-

Nombre reel d'attaques: 257.0
Nombre prevu d'attaques: 32.0
La precision de la regression logistique est: 12.45136186770428%
temps d'execution : 1.3665032386779785 secondes
```

FIGURE 5.12: Résultat d'exécution de la régression logistique sous spark

6 Analyse de la performance du système

Vu que notre solution comporte l'implémentation de plusieurs algorithmes et offre le choix d'utilisation du cluster de traitement Spark, nous allons analyser la performance de notre système en prenant en considération deux critères : L'algorithme prédictif employé et le moteur d'exécution utilisé.

6.1 Analyse de la performance selon l'algorithme prédictif utilisé

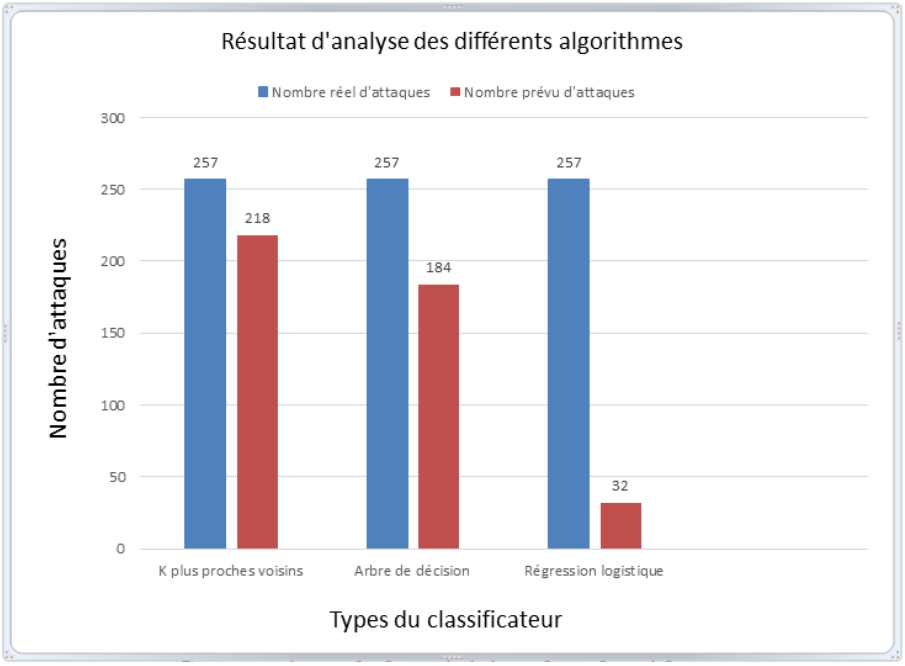

FIGURE 5.13: Comparaison de la précision des algorithmes

En analysant le graphe de la figure ci-dessus, nous remarquons que :
Avec l'utilisation de l'algorithme knn : entre 257 attaques, nous avons pu prévoir 218 attaques et que l'utilisation de l'arbre de décision : entre 257 attaques, nous avons pu prévoir 184 attaques ; alors que l'utilisation de la régression logistique a permis de prédire 32 attaques.
En résumé, l'algorithme knn et l'arbre de décision donnent une bonne prédiction, par contre la régression logistique donne une mauvaise prédiction. De plus, le k plus proches voisins est très fort par rapport aux autres algorithmes.

6.2 Analyse de la performance selon le cluster de traitement utilisé

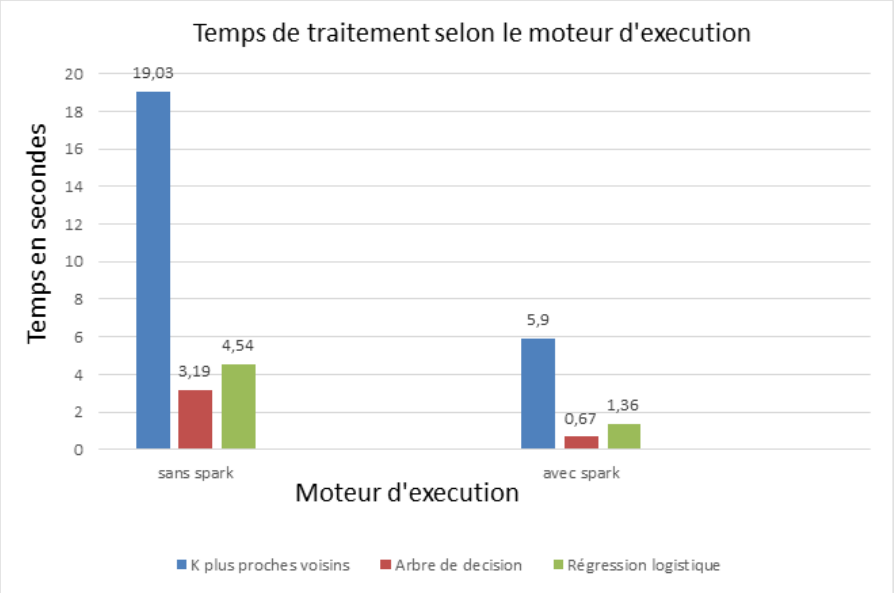

FIGURE 5.14: Comparaison du temps de traitement des algorithmes

En analysant le graphe de la figure ci-dessus, nous remarquons que :
La durée que l'algorithme k plus proches voisins a prise est trop longue par rapport aux autres algorithmes. Par contre, l'arbre de décision a pris le plus court temps d'exécution.
Aussi, nous remarquons que l'exécution de nos programmes sous spark a diminué le temps de réponse de chaque algorithme.
Finalement, d'après les résultats obtenus nous pouvons conclure que l'algorithme des k plus proches voisins donne la meilleur précision, et que l'utilisation du moteur d'exécution du spark rend le temps de réponse au minimum deux fois plus rapide.

7 Conclusion

Dans ce chapitre nous avons présenté les outils matériels et logiciels nécessaire à la mise en place et l'implémentation de notre système d'analyse des fichiers logs. Puis, nous avons décrit pas à pas les

différentes étapes d'installation et de préparation du système conçu ainsi que son fonctionnement. Finalement, nous avons terminé par l'étude de performance de notre système. D'après les résultats obtenus, nous pouvons conclure que notre système donne des résultats intéressants.

Conclusion générale

L'évolution des réseaux informatiques, notamment Internet, au cours de ces dernières années a engendré une augmentation considérable du nombre d'utilisateurs. Ceci est dû essentiellement à la facilité d'accès et la diversité des services utiles offerts.

L'ouverture de ces réseaux rend l'accès aux informations plus simple et plus rapide, et les rend plus vulnérables et plus exposés aux menaces. Ainsi, la mise en place d'une politique de sécurité permettant de garantir la protection de ces réseaux des risques est plus qu'indispensable.

En tant que réponse à ce défi, une nouvelle génération de solution d'analyse de sécurité a émergée ces dernières années, capable de surveiller la performance des systèmes et détecter les problèmes de sécurité. C'est l'analyse des fichiers logs.

En matière de sécurité informatique, l'analyse des fichiers logs est une tâche très essentielle pour la détection des attaques. Cependant, lorsque la taille des fichiers logs est très grande, la tache de leur analyse devient pratiquement impossible, ce qui généralement le cas dans les systèmes en production.

Dans le cadre de ce mémoire, nous avons développé une application d'analyse des fichiers logs du serveur web basée sur l'utilisation des algorithmes de machine learning pour la prédiction des attaques. D'après l'étude comparative que nous avons réalisée, nous remarquons que l'analyse en utilisant l'algorithme k plus proches voisins donne le meilleur résultat de prédiction, et que l'exécution sous spark rend le système plus rapide.

Notre solution est une modeste expérimentation pour l'utilisation des algorithmes d'apprentissage automatique dans l'analyse des fichiers logs, cette solution peut être améliorée de plusieurs façons. Parmi les améliorations possibles nous citons :

• Automatisation de l'analyse en temps réel en utilisant spark streaming qui est destiné à traiter des données qui arrivent en continu.

• L'utilisation de spark mllib qui est une librairie de machine learning qui contient tous les algorithmes d'apprentissage automatique.

Bibliographie

[1] H. Bensefia, « Fichiers logs : preuves judicaires et composant vital pour Forensics », Revue de l'Information Scientifique et Technique Volume 15 - No 1, p. 48-61, 2005.

[2] B. Deokar, A. Hazarnis, « Intrusion Detection System using Log Files and Reinforcement Learning », International Journal of Computer Applications (0975 – 8887) Volume 45– No.19, Mai 2012.

[3] D. Booth, B.J. Jansen, « A Review of Methodologies for Analyzing Websites », Web technologies : Concepts, methodologies, tools, and applications p. 145-166, IGI Global, 2010.

[4] J. Kerkhofs, K. Vanhoof, D. Pannemans, « Web Usage Mining on Proxy Servers : A Case Study », Rapport de recherche : Limburg University Centre, Belgique, 2001.

[5] C. Winter, « Firewall Best Practices », Future Internet (FI) and Innovative Internet Technologies and Mobile Communications (IITM) volume 1, 2016.

[6] S.T. Brugger, « Data Mining Methods for Network Intrusion Detection », Rapport de recherche : Université de Californie, Etat-Unis, 2004.

[7] R. K.Jain, R. S. Kasana, S. Jain, « Efficient Web Log Mining using Doubly Linked Tree », International Journal of Computer Science and Information Security Volume 3 - No.1, 2009.

[8] M. H. Abd Wahab, M. N. Haji Mohd, et al. « Data Pre-processing on Web Server Logs for Generalized Association Rules Mining », World Academy of Science, Engineering and Technology, 2008.

[9] About Logging Site Activity,
http ://www.nsi.bg/nrnm/Help/iisHelp/iis/htm/core/iiabtlg.htm [Consulté le : 03/03/2019]

[10] Analyse des fichiers logs et statistiques utilisateurs,
https ://www.ionos.fr/digitalguide/web-marketing/analyse-web/analyse-des-fichiers-log-et-statistiques-utilisateurs/ [Consulté le : 03/03/2019]

[11] E. Bourget, « De nouvelles perspectives d'utilisation des logs dans un contexte de sécurité informatique », Thèse de doctorat : École Polytechnique de Montréal, Canada, 2016.

[12] P. Shukla, S. Kumar, « Learning Elastic Stack 6.0 », Packt Publishing, Inde, 2017.

[13] A. Yasinsac, Y. Manzano, « Policies to Enhance Computer and Network Forensics »,In Proceedings of the 2001 IEEE workshop on information assurance and security (p. 289-295), États Unis, 5-6 Juin 2001.

[14] GoAccess, https ://goaccess.io/ [Consulté le : 15/03/2019]

[15] apache-scalp, https ://github.com/nanopony/apache-scalp [Consulté le : 17/03/2019]

[16] Top 10+ Log Analysis Tools - Making Data-Driven Decisions , https ://www.keycdn.com/blog/log-analysis-tools [Consulté le : 17/03/2019]

[17] M. Beyer, D. Laney, « The importance of big data : A definition », CT : Gartner, États Unis, 2012.

[18] Data Analyse - Qu'est-ce que le Big Data ?, https ://www.redsen-consulting.com/fr/inspired/data-analyse/big-data [Consulté le : 25/03/2019]

[19] T. Erl, W. Khattak, et al. « Big Data Fundamentals », Prentice Hall, États Unis, 2016.

[20] B. Espinasse, P. Bellot, « Introduction au Big Data :Opportunités, stockage et analyse des mégadonnées », Techniques de l'Ingénieur, France, 2017.

[21] P. Bensabat, D. Gaultier, et al. « Du Big Data au Big Business », Livre Blanc, France, 2014.

[22] A. Sathi, « Big Data Analytics », IBM Corporation., Canada, 2012.

[23] P. Zikopoulos, K. Parasuraman, et al. « Harness the Power of Big Data », The McGraw-Hill Companies, États Unis, 2013.

[24] Comment le Big Data révolutionne le marketing ?, https ://www.datavalue-consulting.com/fr/big-data/comment-le-big-data-revolutionne-le-marketing [Consulté le : 02/04/2019]

[25] E .Hadjipavlou, « Big Data, Surveillance et Confiance : La question de la traçabilité dans le milieu aéroportuaire », Thèse de Doctorat : Université Côte d'Azur, France, 2016.

[26] Le Big data entre en scène dans l'analytique de sécurité, https ://www.sekurigi.com/2017/01/big-data-entre-scene-lanalytique-de-securite/ [Consulté le : 16/04/2019]

[27] Technologie, méthode et applications du Big Data, http ://www.economistesquebecois.com/files/documents/ft/f2/technologie-m-thode-et-applications-du-big-data-v3-ml.pdf [Consulté le : 20/04/2019]

[28] Révolution Big Data – Les enjeux et les risques du Big Data, https ://www.ludosln.net/revolution-big-data-les-enjeux-et-les-risques-du-big-data/ [Consulté le : 20/04/2019]

[29] N. Silver, «The signal and the noise : why most predictions fail but some don't », The Penguin Press, États Unis, 2012.

[30] N. Marz, J. Warren, « Big Data : Principles and best practices of scalable realtime data systems », Manning Publications, États Unis, 2015.

[31] Architecture Big Data Lambda : une approche agnostique,
https ://www.agiledss.com/fr/blogue/architecture-big-data-lambda-approche-agnostique [Consulté le : 28/04/2019]

[32] IBM, « Conseils pour sécuriser le Big Data », Rapport de recherche, 2015.
http ://www.infomania-services.fr/controle/file/180509044559000000_9561476186_1525884359.pdf [Consulté le : 02/05/2019]

[33] P.Joglekar, N.Pise, « Solving Cyber Security Challenges using Big Data », International Journal of Computer Applications (0975 – 8887) Volume 154 – No.4, November 2016.

[34] H.Teymourlouei, L.Jackson, « How Big Data Can Improve Cyber Security », The 4th International Conference on Advances in Big Data Analytics, USA, 2017.

[35] S. SUGUNA, M. VITHYA, « ANALYSIS OF WEB LOGS USING BIG DATA TOOLS », International Journal of Advanced Research Trends in Engineering and Technology (IJARTET) Vol. 3, Special Issue 20, Avril 2016.

[36] Best Log Management Tools : 51 Useful Tools for Log Management, Monitoring, Analytics, and More ,
https ://stackify.com/best-log-management-tools/ [Consulté le : 09/05/2019]

[37] Tools for Log Analysis,
https ://medium.com/tensult/tools-for-log-analysis-461eb07c2d6b ?fbclid=IwAR2O3IwHNZeVlTp7pY7a48PCKfCnQV4ih1JvryHWNT7qwXsZSyi08kMe10c [Consulté le : 09/05/2019]

[38] Apache Metron,
https ://fr.hortonworks.com/apache/metron/ [Consulté le : 09/05/2019]

[39] M. Kantardzic, « Data mining : concepts, models, methods,and algorithms », John Wiley & Sons, États-Unis, 2011.

[40] J. Han, M. Kamber, J. Pei, « Data Mining : Concepts and Techniques », Elsevier, États-Unis, 2012.

[41] D.T. Larose, C.D. Larose, « Data Mining and Predictive Analytics », John Wiley & Sons, États-Unis, 2015.

[42] S. Tufféry, « Data mining et statistique décisionnelle : l'intelligence des données », Editions Technip, France, 2012.

[43] B. Belgacem, « Extraction de connaissances à partir de données incomplètes et imprécises », Mémoire de Magister : niversité de M'sila, Algérie, 2011.

[44] SR. Joseph, H. Hlomani, K. Letsholo, « Data mining algorithms : an overview », Neuroscience, Volume 12– No.3, p. 719-743, 2016.

[45] K. Tsiptsis, A. Chorianopoulos, « Data mining techniques in CRM : inside customer segmentation », John Wiley & Sons, États-Unis, 2011.

[46] InsideBIGDATA Guide to Predictive Analytics,
https ://www.analyticstoday.nl/atd/tibco/insideBIGDATAGuidetoPredictiveAnalytics.pdf
[Consulté le : 25/05/2019]

[47] W. Eckerson, « Predictive analytics », Extending the Value of Your Data Warehousing Investment. TDWI Best Practices Report, Volume 1, p. 1-36, 2007.

[48] L. Rokach, O. Maimon, « Data mining with decision trees : theory and applications », World scientific, États-Unis, 2014.

[49] S. Haykin, « Neural Networks : A Comprehensive Foundation », Prentice Hall, États-Unis, 1998.

[50] M. Koudri, « Modèle de mélange Gaussien.Application sur image cytologique », Mémoire de Master : Université Abou Bakr Belkaid–Tlemcen, Algérie, 2011.

[51] D. Hosmer, S. Lemeshow, R. Sturdivant, « Applied logistic regression », John Wiley & Sons, États-Unis, 2013.

[52] M. Abdlhamed, K. Kifayat,Shi, et al. « Information Fusion for Cyber-Security Analytics », Springer, États-Unis, 2017.

[53] M. Rakesh, « Log analysis based intrusion prediction system », Springer, États-Unis, 2015.