République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université A. Mira de Bejaïa

Faculté des Sciences Exactes

Département d'Informatique

Mémoire de Fin d'Etude

En vue de l'obtention du diplôme de master professionnel en Informatique

Option : Génie Logiciel

Thème

Authentification dans un environnement BIG DATA Sous Hadoop

Réalisé par :

- SANA Mounir

Devant le jury composé de

Examinatrice :	Dr YAICI Malika	M.C.B - Université de Béjaïa.
Examinatrice :	Mme AIT HACENE Souhila	M.A.A - Université de Béjaïa.
Encadrante :	Dr BATTAT Nadia	M.C.B - Université de Béjaïa.

promotion 2021 - 2022

Remerciements

En premier, je tiens à remercier Dieu le tout puissant de m'avoir donné le courage, la patience, la santé et la volanté d'entamer et de terminer ce mémoire.

Que nos chers parents et familles, trouvent ici l'expression de nos remerciements les plus sincères et les plus profonds en reconnaissance de leurs sacrifices, aides, soutien et encouragement.

J'ai l'honneur et le plaisir de présenter ma profonde gratitude et mes sincères remerciements à ma encadreuse Madame Battat Nadia, pour ses précieuses aides, ces orientations et le temps qu'elle m'a accordé pour mon encadrement.

Je remercie l'ensemble des enseignants qui m'ont suivi durant mon cycle d'étude.

Mon respect aux membres du jury Madame Yaici Malika et Madame Ait hacene Souhila qui nous feront l'honneur d'apprécier et d'avoir accepté d'examiner et de juger ce travail.

Je tiens à exprimer tous mes reconnaissances à tous ceux qui ont contribué de prêt ou de loin à la réalisation de ce travail.

Je remercie ALLAH en début et en dernier.

Dédicace :

Je tiens c'est avec grande plaisir que je dédie ce modeste travail :

A l'être le plus cher de ma vie, ma mère Farida.

A celui qui m'a fait de moi un homme, mon père Hamid.

A mon cher frère Samir.

A Mes chers Sœurs Leila, Katia, et Mounia.

A tous mes amis de promotion de 2ème année Master en Génie Logiciel.

Toute personne qui occupe une place dans mon cœur.

A tous les membres de ma famille et toutes personne qui porte le nom ***SANA****, je dédie ce travail à tous ceux qui ont participé à ma réussite.*

Mounir.

Table des matières

Tables des figures

Liste des Tableaux

Liste des abréviations

AMF	Authentication Multi-Factor
AC	Autorité de certification
CERN	Organisation européenne pour la recherche nucléaire
DCAuth	Data-Centric Authentication for Secure In-Network Big-Data Retrieval
FPR	False Positive Rate
FHE	Fully homomorphic encryption
IDA	Innovative data authentication model
IA	Intelligence Artificielle
IPE	Intermediate physical entities
IBM	International Business Machines Corporation
IDC	International Data Corporation
IoT	Internet of Things
IP	Internet Protocol
KBA	Knowledge-based authentication
MACA	A Privacy-Preserving Multi-factor Cloud Authentication System
MFA	Multi-Factor Authentication
NDN	Named-Data Network
NoSQL	Not Only Structured Query Language
OSN	Online Social Network
OTP	One Time Pad
PCR	Platform Configuration Register
PKI	Public Key Infrastructure
PKI-NDN	Public Key Infrastructure for Named Data Networks

RAID	Redundant Array of Independent Disks
RH	Resource Humaine
SSO	Single Sign-On
TCP	Transmission Control Protocol
TPM	Trusted Platform Module
2FA	Two-Factor Authentication
US	United States

Introduction générale

L'augmentation des données massives générée par l'internet, les réseaux sociaux, les sites, les déférents domaines d'applications tel que le domaine scientifique, domaine informatique, et domaine électrique…etc, a conduit au développement d'outils de stockage et d'analyse d'ensemble de données massifs et complexes et de volumes de données qui incluent d'énormes quantités de données, les capacités de gestion des données, l'analyse des médias sociaux et les données en temps réel. Ce phénomène est appelé Big Data [84].

En raison de la complexité des données variées et de la quantité importante accrue du Big Data, ces grandes données ne peuvent pas être stoker et gérer à l'aide de requêtes SQL traditionnelles et le système de gestion de base de données relationnelle (SGBDR). Cependant, une grande variété d'outils et de techniques de bases de données évolutives a été évolué parmi lesquels elle apparait la plateforme Hadoop (développer en 2006) afin de faciliter la gestion de données massives issue de toute source, ainsi que leurs tailles et leurs structures et de réduire considérablement leurs coûts de stockage[85].

Hadoop est un cadre évolutif, open source, basé sur Java, pour le traitement distribué à grande échelle. Il s'étend d'une seule machine à plusieurs milliers de serveurs pour mettre en place une plateforme d'exécution avec calcul local, stockage et haute disponibilité [85]. Le framework Hadoop peut comprend trois composants essentiels à savoir : le HDFS qui est un système de gestion de fichier distribué, MapReduce qui est un paradigme de programmation sur lequel sont effectués les calculs parallèles et distribués de grandes masses de données et Yarn qui est une technologie qui gère l'utilisation des ressources dans un cluster [1].

Initialement Hadoop ne possède aucun système de sécurité, même s'il disposait de certains contrôles d'autorisation comme les permissions d'accès aux fichiers, dont un cyber attaquant ou hacker peut diminuer les propriétés des autres processus de Hadoop à des fins malveillantes. En outre les utilisateurs ont le même droits d'accès aux données du cluster, cette égalité d'accès permet à un utilisateur malveillant de lire et de modifier les données dans le cluster de l'autre utilisateur, il pourrait aussi supprimer ou tuer les autres calculs en cours d'exécution afin d'exécuter son calcul[85].

Problématique :

Aujourd'hui, nous avons le choix entre plusieurs protocoles d'authentification. Ces protocoles peuvent être basés sur le protocole Kerberos ou sur un autre mécanisme (les chaînes de hachage, les certificats, etc.). Il est facile de voir comment les acteurs intéressés (chef d'une entreprise)

pourraient être frustré en essayant de choisir celui à adopter. Chacun de ces protocoles offre des avantages et atouts, mais aussi ils présentent des limites. Il est essentiel de comparer ces protocoles en considérant un ensemble de critères (comme, temps de traitement) afin de fournir des indicateurs et des mesures aux personnes intéressées. Ces indicateurs et des mesures peuvent aider ces personnes à prendre de décisions sur le protocole d'authentification à mettre en place.

Contribution :

L'objectif initial de ce projet était de comparer des protocoles d'authentification qui utilisent des mécanisme différents. Cependant nous avons pu comparer deux versions différentes du protocole Kerberos. La première se base sur chiffrement symétrique et la deuxième utilise le chiffrement asymétrique.

Structure de mémoire :

Ce mémoire est composé de trois chapitres :

Le premier chapitre : sera consacré à des définitions et à des généralités sur les Big Data et Hadoop.

Le deuxième chapitre : dans ce chapitre, nous introduirons la sécurité des Big Data, nous présenterons ensuite les types d'authentification et nous citerons quelques travaux relatifs à l'authentification des Big Data.

Dans le dernier chapitre : nous allons comparer deux protocoles d'authentification basés sur l'utilisation du protocole kerberos (versions symétrique et asymétrique) afin d'estimer le temps de traitement de chaque protocole.

Enfin, nous allons conclure notre travail par une conclusions générales et quelques perspectives.

Chapitre 1 BIG DATA

1.1 Introduction :

Le Big Data est un phénomène qui a vu le jour avec l'émergence des données volumineuses qu'ils ne peuvent pas être traiter avec des techniques traditionnelles.[1]

Le Big Data est devenu une tendance incontournable pour beaucoup d'acteurs industriels du fait de l'apport qu'il offre en qualité de stockage, de traitement et d'analyse de données[1].

Les premiers projets de Big Data sont ceux des acteurs de la recherche d'information sur le web « moteurs de recherche » tel que Google et Yahoo. En effet, ces acteurs étaient confrontés aux problèmes de la scalabilité (passage à l'échelle) des systèmes et du temps de réponse aux requêtes utilisateurs. Très rapidement, d'autres sociétés ont suivis le même chemin comme Amazon et Facebook[29].

Dans ce chapitre nous présenterons le terme Big Data, nous citerons ses caractéristiques, ainsi que ses classifications en termes de : sources de données, format du contenu, mise à disposition des données, traitement des données, les magasins de données, le type de données et la méthode d'analyse, et nous parlerons sur les domaines d'applications des Big Data ce terme, et en finale nous allons aborder quelques avantages et limites liés à ce domaine.

1.2 Historique :

La première trace de Big Data remonte à 1663 lorsque John Graunt a traité des quantités écrasantes d'informations alors qu'il étudiait la peste bubonique, qui hantait l'Europe à l'époque. Graunt a été la toute première personne à utiliser l'analyse de données statistiques. Plus tard, au début des années 1800, le domaine des statistiques s'est élargi pour inclure la collecte et l'analyse de données[3].

Le monde a vu pour la première fois le problème de la surabondance de données en 1880. Le US Census Bureau a annoncé qu'il estimait qu'il faudrait huit ans pour gérer et traiter les données recueillies au cours du programme de recensement cette année-là. En 1881, un homme du Bureau nommé Herman Hollerith a inventé Hollerith Tabulating Machine qui a réduit le travail de calcul[3].

Tout au long du XXe siècle, les données ont évolué à une vitesse inattendue. Les Big Data sont devenues le cœur de l'évolution. Des machines pour stocker des informations magnétiquement et numériser des motifs dans des messages, ainsi que des ordinateurs ont également été créés à

cette époque. En 1965, le gouvernement américain a construit le premier centre de données, avec l'intention de stocker des millions d'empreintes digitales et de déclarations de revenus[3].

En 1969, Advanced Research Projects Agency Network (ARPANET), le **réseau étendu** comprenant des protocoles de contrôle distribué et **TCP / IP** , a été créé. Cela a formé la base de l'Internet d'aujourd'hui[4].

Alors que les ordinateurs commencent à partager des informations à des taux exponentiellement plus élevés grâce à Internet, la prochaine étape de l'histoire du Big Data prend forme. En 1989, Tim Berners-Lee et Robert Cailliau ont découvert le World Wide Web et développé HTML, URL et HTTP tout en travaillant pour le CERN. L'ère d'Internet avec un accès généralisé et facile aux données débute[4]. En 1997, Le domaine google.com est enregistré un an avant son lancement, déclenchant la montée en puissance du moteur de recherche et le développement de nombreuses autres innovations technologiques, notamment dans les domaines de l'apprentissage automatique, du Big Data et de l'analyse (la capacité de stockage est 4GB [5]). En 1998, Carlo Strozzi développe **NoSQL** , une base de données relationnelle open source qui fournit un moyen de stocker et de récupérer des données modélisées différemment des méthodes tabulaires traditionnelles trouvées dans les bases de données relationnelles[4].

Le Big Data tel que nous le connaissons arrive enfin, et l'explosion d'ingéniosité qu'il apporte avec lui ne peut être surestimée. En 2001, Doug Laney du cabinet d'analystes Gartner invente les **3V** (volume, variété et vitesse), définissant les dimensions et les propriétés du Big Data. Les V résument la véritable définition du Big Data et inaugurent une nouvelle période où le Big Data peut être considéré comme une caractéristique dominante du 21e siècle. Des V supplémentaires - tels que la véracité, la valeur et la variabilité ont depuis été ajoutés à la liste[4], (capacité de stockage est 80GB [5]).

En 2005, Les informaticiens Doug Cutting et Mike Cafarella créent Apache **Hadoop** , le **framework open source** utilisé pour stocker et traiter de grands ensembles de données, avec une équipe d'ingénieurs issue de Yahoo(capacité de stockage est 250GB [5]). En 2006, Amazon Web Services (**AWS**) commence à proposer des services d'infrastructure informatique basés sur le Web, désormais **connus sous le nom de cloud computing** . Cependant, AWS domine l'industrie des services cloud avec environ un tiers de la part de marché mondiale[4] (capacité de stockage est 500GB [5]). En 2011, Facebook lance le **projet Open Compute** pour partager les spécifications des centres de données économes en énergie. L'objectif de l'initiative est de fournir une augmentation de 38 % de l'efficacité énergétique à un coût inférieur de 24 %[4]

(capacité de stockage est 4TB [5]). En 2014, Pour la première fois, plus d'appareils mobiles accèdent à Internet que d'ordinateurs de bureau aux États-Unis. Le reste du monde a emboîté le pas deux ans plus tard (la capacité de stockage est 6TB [5]). En 2016, Quatre-vingt-dix pour cent des données mondiales ont été créées au cours des deux dernières années seulement, et IBM rapporte que 2,5 quintillions d'octets de données sont créés chaque jour.

Des technologies telles que l'apprentissage automatique, l'IA et l' analyse IoT améliorent considérablement les capacités à traiter, analyser et agir sur les données afin de repousser les limites liés aux Big Data[4].

1.3 Définition du Big Data :

Le Big Data a été définie par plusieurs manière qui défère d'une définition à une autre. A savoir :

- The McKinsey Global Institute (MGI) définit le Big Data comme un ensemble de données dont la taille dépasse la capacité des logiciels de base de données traditionnelles à capturer, stocker, gérer et analyser[6]. Et dans d'autre terme International Data Corporation (IDC) qui à définit le Big Data comme Les technologies qui décrivent une nouvelle génération de technologies et d'architectures, conçues pour extraire économiquement la valeur à partir de très grands volumes et d'une grande variété de données, en permettant une très grande vitesse de capture, une découverte et/ou une analyse[7].

Le terme de Big data (parfois appelées « mégadonnées » en français) désigne une nouvelle discipline qui se situe au croisement de plusieurs domaines : statistiques, technologie, base de données et métiers (marketing, finance, RH, etc).[1]

Le mot Big Data est définie comme suit « un ensemble de solutions, technologiques qui permettent d'analyser et de traiter le contenu audio, vidéo et textuel » [9].

1.4 Architecture du Big Data :

Le succès du fonctionnement de le Big data dépend de son architecture, son infrastructure correcte et de son utilité que l'on fait (Data into Information into Value)[2].

La plupart des architectures Big Data incluent tout ou partie des éléments suivants :

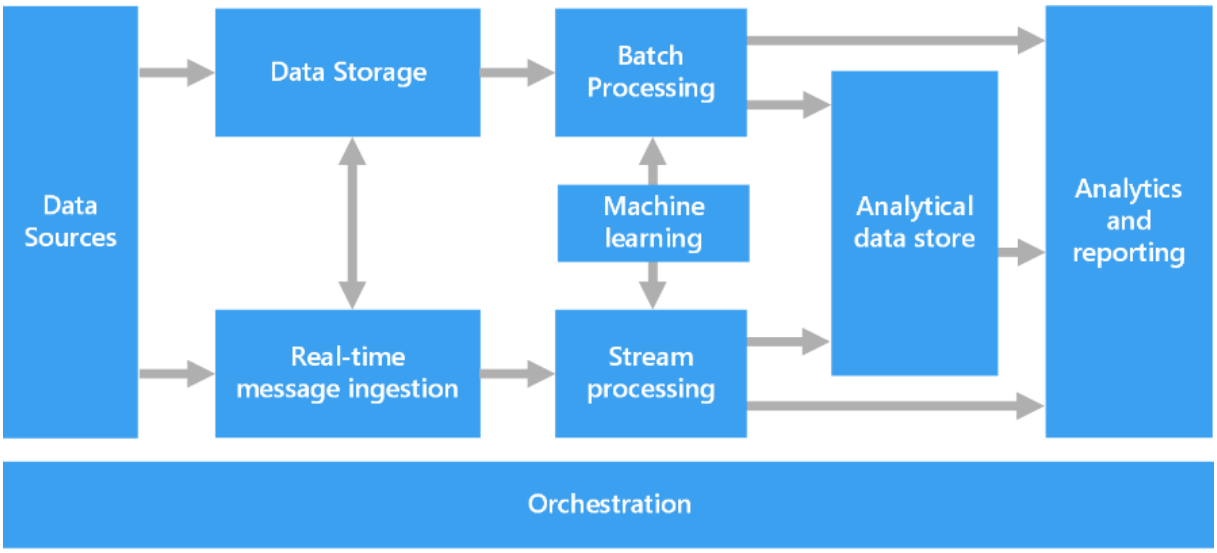


Figure 1.1: Architecture du Big Data[10].

- **Sources des données (Data sources) :**

Toutes les solutions Big Data commencent par une ou plusieurs sources de données[10]. Les exemples comprennent:

- Magasins de données d'application, tels que les bases de données relationnelles.
- Fichiers statiques produits par les applications, tels que les fichiers journaux du serveur Web.
- Sources de données en temps réel, telles que les appareils IoT.

- **Stockage des données *(*Data storage*)* :**

Les données pour les opérations de traitement par lots sont généralement stockées dans un magasin de fichiers distribué qui peut contenir de gros volumes de fichiers volumineux dans divers formats. Ce type de magasin est souvent appelé un lac de données . Les options d'implémentation de ce stockage incluent Azure Data Lake Store ou des conteneurs d'objets blob dans Azure Storage[10].

- **Traitement par lots (Batch processing) :**

Étant donné que les ensembles de données sont si volumineux, une solution Big Data doit souvent traiter des fichiers de données à l'aide de travaux par lots de longue durée pour filtrer, agréger et autrement préparer les données pour l'analyse. Habituellement, ces travaux impliquent la lecture de fichiers source, leur traitement et l'écriture de la sortie dans de nouveaux

fichiers. Les options incluent l'exécution de tâches U-SQL dans Azure Data Lake Analytics, l'utilisation de tâches Hive, Pig ou Map/Reduce personnalisées dans un cluster HDInsight Hadoop, ou l'utilisation de programmes Java, Scala ou Python dans un cluster HDInsight Spark[10].

- **Ingestion de messages en temps réel (Real-time message ingestion) :**

Si la solution inclut des sources en temps réel, l'architecture doit inclure un moyen de capturer et de stocker des messages en temps réel pour le traitement des flux. Il peut s'agir d'un simple magasin de données, où les messages entrants sont déposés dans un dossier pour être traités. Cependant, de nombreuses solutions ont besoin d'un magasin d'ingestion de messages pour agir comme tampon pour les messages et pour prendre en charge le traitement évolutif, la livraison fiable et d'autres sémantiques de mise en file d'attente des messages. Cette partie d'une architecture de diffusion en continu est souvent appelée mise en mémoire tampon de flux. Les options incluent Azure Event Hubs, Azure IoT Hub et Kafka[10].

- **Traitement de flux (Stream processing) :**

Après avoir capturé les messages en temps réel, la solution doit les traiter en filtrant, en agrégeant et en préparant autrement les données pour analyse. Les données de flux traitées sont ensuite écrites dans un puits de sortie. Azure Stream Analytics fournit un service de traitement de flux géré basé sur des requêtes SQL en cours d'exécution qui fonctionnent sur des flux illimités. Des technologies de streaming Apache open source telles que Storm et Spark Streaming dans un cluster HDInsight peuvent également être utilisées[10].

- **Magasin de données analytiques (Analytical data store) :**

De nombreuses solutions de Big Data préparent les données pour l'analyse, puis fournissent les données traitées dans un format structuré qui peut être interrogé à l'aide d'outils analytiques. Le magasin de données analytiques utilisé pour répondre à ces requêtes peut être un entrepôt de données relationnelles de type Kimball, comme on le voit dans la plupart des solutions de Business Intelligence (BI) traditionnelles. Alternativement, les données peuvent être présentées via une technologie NoSQL à faible latence telle que HBase, ou une base de données Hive interactive qui fournit une abstraction des métadonnées sur les fichiers de données dans le magasin de données distribué. Azure Synapse Analytics fournit un service géré pour l'entreposage de données à grande échelle basé sur le cloud. HDInsight prend en charge

Interactive Hive, HBase et Spark SQL, qui peuvent également être utilisés pour fournir des données à des fins d'analyse[10].

- **Analyse et rapport (Analysis and reporting):**

L'objectif de la plupart des solutions Big Data est de fournir des informations sur les données par le biais d'analyses et de rapports. Pour permettre aux utilisateurs d'analyser les données, l'architecture peut inclure une couche de modélisation des données, telle qu'un cube OLAP multidimensionnel ou un modèle de données tabulaire dans Azure Analysis Services. Il peut également prendre en charge la BI en libre-service, en utilisant les technologies de modélisation et de visualisation de Microsoft Power BI ou Microsoft Excel. L'analyse et le reporting peuvent également prendre la forme d'une exploration interactive des données par des data scientists ou des data analysts. Pour ces scénarios, de nombreux services Azure prennent en charge les blocs-notes analytiques, tels que Jupyter, permettant à ces utilisateurs de tirer parti de leurs compétences existantes avec Python [10].

- **Orchestration :**

La plupart des solutions de Big Data consistent en des opérations de traitement de données répétées, encapsulées dans des flux de travail, qui transforment les données sources, déplacent les données entre plusieurs sources et puits, chargent les données traitées dans un magasin de données analytiques ou poussent les résultats directement vers un rapport ou un tableau de bord. Pour automatiser ces workflows, une technologie d'orchestration telle qu'Azure Data Factory ou Apache Oozie et Sqoop peut être utilisée[10].

1.5 Caractéristiques du Big Data :

Le volume élevé, la vitesse élevée et la grande variété des Big Data ont révolutionné de nombreux aspects des systèmes traditionnels de stockage, de traitement et d'analyse des données et créé de nombreux nouveaux défis. Ces 3 V (volume, vitesse et variété) ont été définis comme les principales dimensions et caractéristiques du Big Data qui le rendent différent des données traditionnelles. Dans cette section, la discussion se concentre sur les caractéristiques du Big Data, car les données d'origine jouent un rôle central dans la capture, le stockage et l'analyse des données[13].

La figure2 décrit les 10Vs de Big Data.

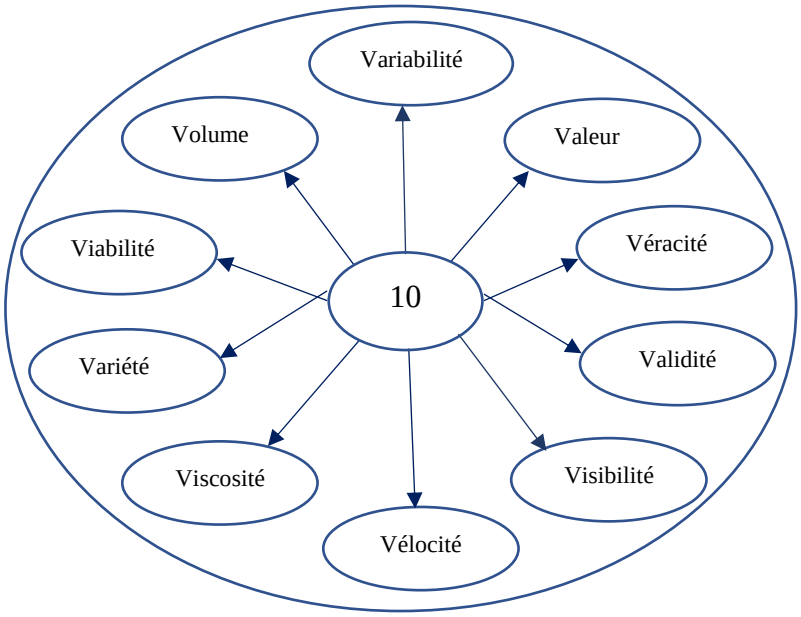


Figure 1.2 : Caractéristique de Big Data[8][13].

- **Variabilité :**

Les données sont générées par diverses sources de données et stockées dans une installation de stockage à des vitesses, des formats ou des types variables[9].

- **Valeur :**

Il s'agit de s'assurer que l'organisation acquiert de la valeur pour ces données. Il s'agit de mesurer l'utilité des données pour la prise de décision[9].

- **Véracité :**

Fait référence à la nécessité de données correctes et précises qui doivent être traitées pour obtenir les meilleurs résultats[9].

- **Validité :**

Cela signifie que les données doivent être correctes et précises pour l'usage utilisation prévue. La source des Big Data doit être exacte, surtout lorsque les résultats sont utilisés pour la prise de décision[9].

- **Visibilité :**

Il s'agit d'améliorer les données et à les rendre faciles à comprendre[9].

- **Vélocité :**

Fait référence à la vitesse de génération des données[9].

- **Viscosité :**

il s'agit d'une différence de temps entre l'événement s'est produit et l'événement qui est décrit[10].

- **Variété :**

Représente les différents format et type de donnée, Nous avons des données structurées, semi-structurées, non-structurées.

- **Viabilité :**

Le sens de la viabilité est que le Big Data devrait avoir la capacité d'être vivant et actif pour toujours, et capable de se développer, et de produire plus de données quand c'est nécessaire[13].

- **Volume :**

Fait référence à la quantité massive des données qui sont chaque année en croissance exponentielle.

1.6 Classification des Big Data :

Les Big Data se présentent sous différents formats, chacun ayant ses propres caractéristiques. Pour comprendre les avantages et les inconvénients des applications qui traitent des volumes importants de données, il est nécessaire de classer les Big Data. Le stockage des données, les types de contenu et la mise en scène des données sont autant de catégories qui peuvent être utilisées pour classer les Big Data. Chacun de ces groupes possède son propre ensemble de caractéristiques et d'interdépendances[14].

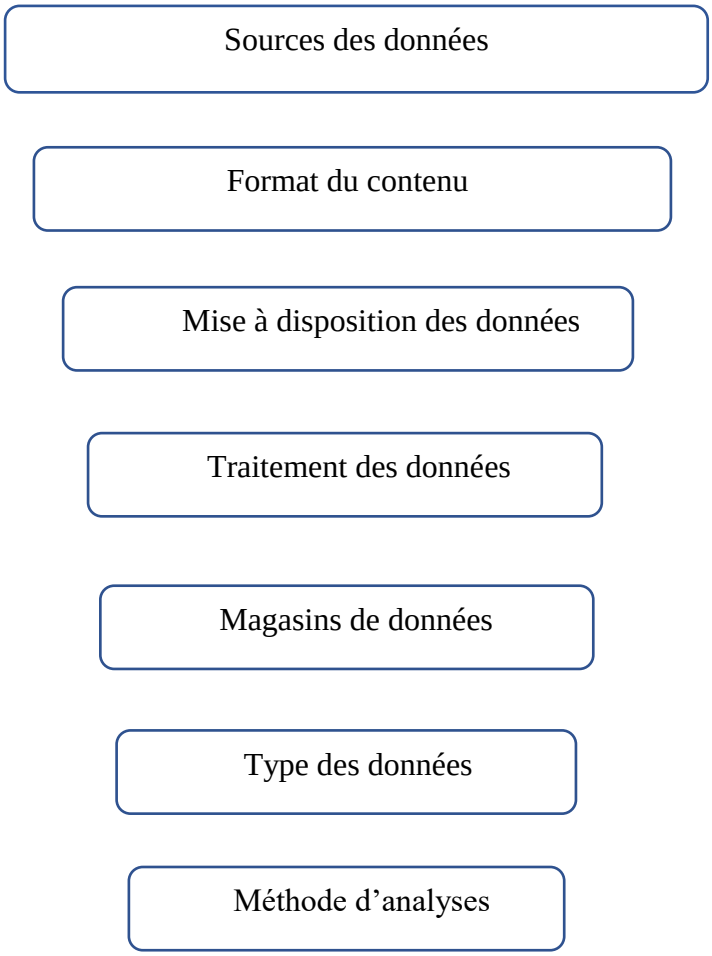


Figure 1.3: Catégories utilisées pour classifier les Big Data [14].

- **Sources des données :**

Les données sociales, les données machine et les données transactionnelles sont les trois principales sources de Big Data. En outre, les entreprises doivent faire la distinction entre les données générées en interne, ou celles qui restent derrière le pare-feu de l'entreprise, et les données générées en externe, ou celles qui doivent être importées dans un système. Il est également important de considérer si les informations sont non structurées ou organisées. En l'absence d'un modèle de données prédéfini, les données non structurées sont plus longues à comprendre. Sur les sites de réseaux sociaux les plus importants du monde, les Likes, Tweets et Retweets, les commentaires, les téléchargements de vidéos et les médias en général sont tous des sources de données sociales. Ce type de données peut être extrêmement utile dans l'analyse marketing car il offre un aperçu inégalé du comportement et du sentiment des clients. Les informations générées par les machines industrielles, les capteurs montés sur les machines et

même les journaux Web qui suivent le comportement des utilisateurs sont autant d'exemples de données machine. À mesure que l'internet des objets se popularise et s'étend dans le monde entier, ce type de données devrait connaître une croissance exponentielle. Les appareils médicaux, les compteurs intelligents, les caméras routières, les satellites, les sports et l'Internet, qui se développe rapidement, sont tous des exemples de capteurs. Les données transactionnelles comprennent les factures, les ordres de paiement, les registres de stockage et les reçus de livraison, mais les données en elles-mêmes n'ont pratiquement aucune valeur et la plupart des entreprises n'ont aucune idée des données qu'elles génèrent ni de la manière de les utiliser efficacement[14].

- **Format du contenu :**

Les formats de contenu des Big Data peuvent également être utilisés pour la classification. Voici les différentes formes de Big Data basées sur le format de contenu [14]:

Données structurées : Dans une base de données, les données structurées sont généralement des données tabulaires exprimées par des colonnes et des lignes. les bases de données relationnelles sont celles qui stockent les tableaux dans ce format. Les bases de données relationnelles sont celles qui stockent des tableaux dans ce format. Le terme mathématique "relation" fait référence à une table qui contient une collection construite de données. Chaque ligne d'un tableau de données structurées possède le même ensemble de colonnes. SQL (Structured Query Language) est un langage de programmation de données structurées[14].

Données semi-structurées : Les données semi-organisées sont des informations qui ne sont pas structurées (comme une base de données relationnelle), mais qui présentent également une certaine structure. Les documents au format JavaScript Object Notation (JSON) constituent des données semi-structurées. On y trouve également des bases de données graphiques et des magasins de valeurs clés. De telles données nécessitent des processus dynamiques pour des lois complexes lors des opérations sur les données. Par conséquent, le défi de travailler avec une source de données aussi complexe est toujours à l'étude[14].

Données non structurées : Les données non structurées sont des informations qui n'ont de structure prédéfinie ou de modèle de données. Les données non structurées sont une collection de documents à forte teneur en texte qui peuvent également inclure des données numériques, chronologiques et factuelles. Il n'est pas garanti que les vidéos, les fichiers audio ou les fichiers de données binaires aient une structure particulière. Les données non structurées sont le nom qui leur est donné. Comme l'ampleur de cette forme de données ne cesse de croître en raison du

nombre de téléphones mobiles et d'applications de médias sociaux, leur gestion constitue un défi de taille[14].

- **Mise à disposition des données :**

Traditionnellement, nous disposons d'une série de zones/structures de stockage de données intermédiaires ou de transit dans nos architectures de l'information. Les répertoires de publication sur les réseaux sources, les zones de transit dans les centres de données, les coffres-forts de données et, surtout, les hubs de fichiers de données ont tous été utilisés dans le passé. Ces méthodes de stockage des fichiers de données présentent en général deux inconvénients majeurs : La conservation des données était normalement limitée à quelques mois en raison des coûts de stockage. Les utilisateurs finaux n'avaient pas accès aux données de mise à disposition pour l'analyse ou la découverte de données, car ces systèmes étaient conçus pour publier des données à des fins d'intégration de systèmes[14].

- **Traitement des données :**

Le type de traitement qui produit les données peut être utilisé pour classer les données. Voici quelques exemples de méthodes de traitement[14] :

Traitement par lots : Le traitement par lots est le traitement simultané d'une grande quantité de données. Pour une seule journée, les données peuvent facilement consister en des millions d'enregistrements et peuvent être traitées de plusieurs façons (fichier, enregistrement, etc.). Dans la plupart des cas, les tâches sont effectuées dans un ordre séquentiel sans interruption[14].

Traitement des flux : Le traitement en continu est la capacité d'interpréter les données qui circulent d'un ordinateur à un autre en temps quasi réel. Ce type de calcul continu se produit au fur et à mesure que les données circulent dans le système, sans contrainte de temps. Les systèmes ne nécessitent pas de gros volumes de données à traiter en raison du flux quasi instantané. Si les événements que vous souhaitez surveiller se produisent régulièrement et de manière rapprochée dans le temps, le traitement en flux est une excellente option. C'est également la meilleure option si l'événement doit être identifié et traité rapidement. Ainsi, des tâches telles que la détection des fraudes et la cybersécurité bénéficient du traitement en continu. Les transactions frauduleuses peuvent être détectées et arrêtées avant qu'elles ne soient achevées si les données relatives aux transactions sont traitées en continu[14].

En temps réel : Les réactions aux données sont communément appelées "traitement des données en temps réel". Un système est en temps réel s'il peut garantir que la réaction se produira dans

un court laps de temps dans le monde réel, généralement quelques secondes ou millisecondes. L'un des meilleurs exemples de systèmes en temps réel est celui des systèmes boursiers, qui produisent des données en temps réel[14].

- **Magasins de données :**

Des grappes de stockage de données sont nécessaires pour obtenir des performances efficaces et rapides dans le cadre de l'analyse des données volumineuses. Les modèles classiques de bases de données relationnelles n'étant pas adaptés aux bases de données à très grande échelle, des problèmes de performance se posent tout au long de l'analyse des données volumineuses. En raison de leur capacité à partitionner horizontalement les données, de leur puissance de traitement étendue et de leurs meilleures performances, les bases de données No-SQL sont préférées aux bases de données SQL pour le traitement. NoSQL est utilisé par des entreprises telles que Google, Facebook, Amazon et LinkedIn pour traiter des flux de données toujours plus importants. Les dix principaux magasins de données volumineuses sont Cassandra, Hbase, MongoDB, Neo4j, CouchDB, OrientDB, Terrstore, FlockDB, Hibari et Riak[14].

- **Type de données :**

L'ère des Big Data a produit une variété d'ensemble de données provenant de différentes sources dans différents domaines. Ces ensembles de données se composent de plusieurs modalités, chacune ayant une représentation, une distribution, une échelle et une densité différentes[15].

Voici quelque type de données :

- Données du réseau en ligne :

Les données des réseaux sociaux en ligne (OSN), tels que Facebook [16], constituent l'un des principaux centres d'intérêt du Big Data. Ce centre d'intérêt s'est élargi avec l'évolution de l'analyse des données[17].

- Données mobile et IoT :

Une autre tendance du Big Data réseau est l'analyse des données mobiles et IoT. Avec le développement de la technologie 5G, les réseaux mobiles convergents ont permis d'améliorer considérablement les performances des communications de machine à machine. Les réseaux mobiles intégrés partagent des bandes de spectre sans licence dans des réseaux cellulitiques, tels que l'évolution à long terme-avancée, en utilisant la technologie radio cognitive. Ce réseau

génère de grands volumes de données, par rapport aux anciens réseaux mobiles [18]. Outre l'augmentation du volume de données mobiles, l'IoT génère également une grande quantité de données dans ce nouveau contexte[19],[20].

- Données géographiques :

Les **données géographiques** permettent de préciser les localisations et descriptions de particularités géographiques par un composite de données spatiales et de données descriptives. Une carte de la végétation est un outil souvent indispensable. Par exemple, des zones boisées peuvent être localisées par les références d'un système en quadrillage de coordonnées et les données qui s'y réfèrent, tels que le type d'arbres qui la caractérise ou leur hauteur moyenne, ils peuvent également être enregistrées[21].

- Données visuelles :

La visualisation des données désigne le fait de représenter visuellement ses données pour pouvoir déceler et comprendre des informations, les données brutes étant difficilement interprétables et exploitables. Ce processus se fait par des outils analytiques spécifiques et se matérialise par des tableaux (type Excel), des graphiques, des cartes visuelles ou même des infographies regroupées dans des dashboards (tableaux de bord)[22].

- **Méthodes d'analyses :**

Actuellement, les méthodes utilisées pour l'analyse des Big Data sont liées à MapReduce. Pour le contrôle des données dans le passé, les instruments d'analyse des données étaient insuffisants dans les systèmes de dépôt et d'exploration. Les modèles utilisés par les chercheurs en Big Data sont généralement inspirés par la facilité d'exposition mathématique [23]. En vertu de l'essence du Big Data, il est mémorisé dans un cadre de système de documents dispersés. Hadoop et HDFS d'Apache sont largement utilisés pour la mémorisation et le contrôle des Big Data. L'exploration des Big Data est semée d'obstacles, car elle est liée à des systèmes documentaires dispersés de grande taille qui sont censés se caractériser par leur résistance aux erreurs, leur agilité et leur capacité d'extension [24].

1.7 Domaines d'application de Big Data :

Le Big Data a été conçu pour la prise de décision, et donc la réduction des risques, en exploitant les fabuleuses capacités de classification automatique désormais disponibles et en établissant des liens exploitables entre de grandes quantités de données disparates au premier abord. Être assisté dans ses choix en vue d'un bénéfice à court terme est au centre des préoccupations de tout être humain, cette raison expliquant à elle seule pourquoi le champ d'application du Big Data est aussi large.[25]

- **La santé et les sciences :**

Le Big Data favorise une médecine préventive et personnalisée. à l'aide de ces outils statistiques puissants, Nous pouvons identifier très vite et de manière très fiable les origines des maladies chez les patients (étiologie), notamment avec des outils d'analyse d'imagerie (IRM, scanner, radio), et de suivre en temps réel et de manière non intrusive les paramètres physiologiques vitaux de n'importe quelle personne, et même de détecter des allergies ou des phénomènes épidémiologiques (grippe) grâce à des modèles prédictifs adaptés.

Il est possible aujourd'hui de mettre en évidence des rapprochements subtils entre ces origines (avec des techniques de regroupement issues de l'apprentissage non supervisé) et d'intégrer dans le diagnostic final un nombre considérable de facteurs environnementaux jusqu'ici négligés, ce qui rend la médecine plus prédictive et moins risquée.

Les sciences profitent, elle aussi, de cette avancée majeure et des capacités exponentielles de calcul. Grâce aux centres de données, il est possible de modéliser et de simuler des phénomènes multiphysiques trop complexes pour être directement appréhendés par l'homme, aussi bien à l'échelle submicronique qu'à celle du système solaire[25].

- **Les services :**

Sur un plan sociétal, les services à la personne, entre les êtres humains et entre les hommes et les machines profitent aussi du Big Data. Les secteurs bancaires, financiers et de l'assurance sont constitutivement intéressés par la réduction de la prise de risque pour leur entité comme pour leurs clients. On peut par exemple citer l'utilisation des Big Data pour lutter contre les fraudes et la cybercriminalité, en déterminant en temps réel les anomalies sur les comptes des usagers. Dans le même ordre d'idée, l'analyse à la microseconde près des transactions financières, des flux boursiers, des ventes et achats d'actions, et de la réaction immédiate des marchés permet au trading haute fréquence (high frequency trading) de croître de manière très

importante (des ordinateurs se substituent à l'intervention humaine et, à partir d'algorithmes puissants, arbitrent et transmettent les ordres sur les marchés avec une vitesse d'exécution de quelques microsecondes)[25].

- **Les transports :**

l'analyse des données du Big Data (données provenant des pass de transport en commun, géolocalisation des personnes et des voitures, etc.) permet de modéliser les déplacements des populations afin d'adapter les infrastructures et les services (horaires et fréquence des trains, par exemple).[26]

- **La gestion énergétique :**

l'analyse des données issues du Big Data intervient dans la gestion de réseaux énergétiques complexes via les réseaux électriques intelligents (smartgrids) qui utilisent des technologies informatiques pour optimiser la production, la distribution et la consommation de l'électricité.

De la même manière, l'analyse des données provenant de capteurs sur les avions (données de vol) associées à des données météo permet de modifier les couloirs aériens afin de réaliser des économies de carburant et d'améliorer la conception et la maintenance des avions.[26]

- **L'assurance :**

Le manque de services personnalisés ciblés et de tarification sur-mesure sont préjudiciables pour les assureurs. En effet, pallier ces lacunes permettrait de nettes améliorations. En premier lieu, les assureurs gagneraient de nouveaux segments et parts de marché. Dans un second lieu, une enquête menée par Marketforce, va plus loin dans cette analyse. La sous-utilisation des données recueillies par les experts en sinistres représente une mine d'or trop peu utilisée.

Le Big Data est ici utilisé pour fournir des informations sur les clients. On cherche à proposer des produits plus simples, en analysant et en prédisant le comportement des clients. Cela est notamment permis grâce aux données issues des médias sociaux, des appareils équipés de GPS et des images de vidéosurveillance. Le Big Data permet également de mieux fidéliser les clients en proposant des personnalisations[27].

- **Socio-économique :**

D'une manière générale, dans le domaine socio-économique, en écoutant mieux les opinions des utilisateurs et en comprenant comment les utilisateurs utilisent ces services[28], les Big Data peuvent être utilisées pour simplifier ou ajuster les services fournis. Par exemple, Google

Analytics offre aux entreprises et aux administrations publiques la possibilité d'améliorer la conception de leurs sites Web en analysant les visites des internautes. Dans le domaine de l'éducation, à travers l'enseignement à distance (en particulier les cours publics en ligne à grande échelle-MOOC), le traitement des Big Data permet d'analyser les activités des étudiants (temps passé, méthode de suivi, temps d'arrêt) - regarder des vidéos éducatives sur Internet, recherche parallèle, etc.[29]

Santé & sciences

Services

transports

Gestion énergétique

Assurance

Socio-économique

Figure 1.4: Domaines d'application de Big Data.

1.8 Technique d'analyse des données :

Il existe trois principales méthodes d'analyse des données pour le Big Data :

1. La méthode descriptive vise à mettre en évidence les informations présentes dans les données. Mais il est masqué par une grande quantité de données [30].Certaines techniques et algorithmes utilisés dans l'analyse descriptive comprennent :

   - Analyse factorielle (PCA et ACM)
   - Méthode du centre mobile
   - Classification hiérarchique
   - Classification des neurones
   - Recherche d'association

2. La méthode de prédiction vise à déduire de nouvelles informations à partir d'informations actuelles [31] Cette technologie utilise l'intelligence artificielle, les principales méthodes sont :

- Arbre de décision
- Les réseaux de neurones
- Classification bayésienne
- Support Vector Machine (SVM)
- Voisin le plus proche (KNN)

3. Les méthodes prescriptives visent à identifier et anticiper les actions / décisions les plus appropriées.

- Le meilleur choix pour atteindre la situation souhaitée.[31]

1.9 Technologies du Big Data :

La figure 5 montre les différentes technologies du Big Data qui sont des solutions pour traiter et analyser de très gros volumes de données, cependant la technologie la plus répandue est Apache Hadoop qui est un framework largement utilisé aujourd'hui [1].

Figure 1.5: Technologies du Big Data[32].

❖ Hadoop

Hadoop est un framework libre et open source écrit en Java destiné à faciliter la création d'applications distribuées (au niveau du stockage des données et de leur traitement) et échelonnables (scalables) permettant aux applications de travailler avec des milliers de nœuds

et des pétaoctets de données. Ainsi chaque nœud est constitué de machines standard regroupées en grappe [33]. Hadoop a été créé en 2005, basé sur des travaux de Google publié en 2004 sur le Map/Reduce et sur GoogleFS, un système de fichier distribué. C'est Doug Cutting qui l'a créé et qui a choisi le nom et le logo grâce à la peluche de son fils, un éléphant jaune qu'il appelait Hadoop[34]. Hadoop est composé de plusieurs éléments : un système de stockage (HDFS), un système de planification des traitements (YARN) et le framework de traitement (MapReduce) [1][35].

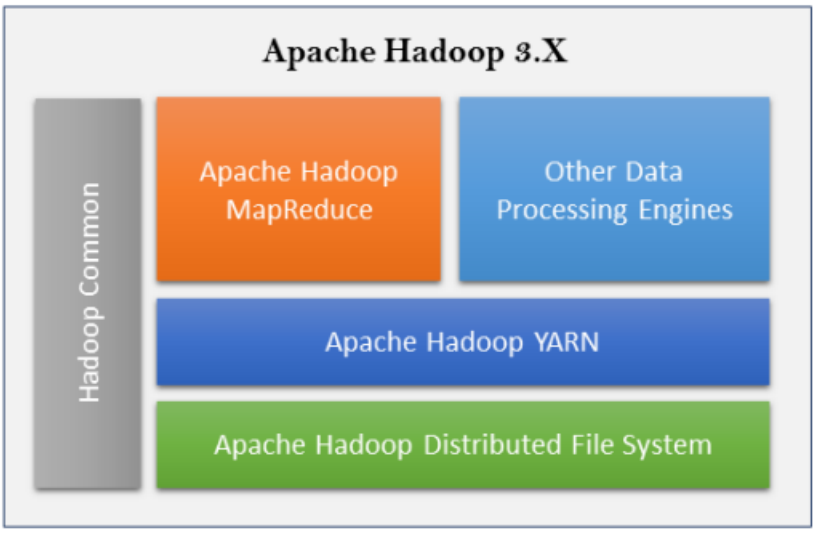


Figure 1.6: Architecture d'apache Hadoop 3.X [1].

- **Apache HDFS :**

est un système de fichier distribué conçu pour le stockage de gros volumes de données et peut fonctionner sur plusieurs machines de faible coût. Il offre aux applications un accès rapide aux données. HDFS est adapté pour les applications qui traitent une grande quantité de données. Il est largement inspiré de GoogleFS [1] [36].

L'architecture est faite de la façon suivante. HDFS se compose d'un nœud principal, appelé le NameNode. Ce nœud est très important car c'est lui qui va gérer l'emplacement de l'ensemble des données. Il fait la correspondance entre un fichier et ses blocs associés (les metadata d'un fichier), et il sait également sur quels nœuds chaque bloc se situe. Sur les autres nœuds se trouvent les DataNode. Un DataNode va gérer les blocs de données présent sur son nœud. Le DataNode tiens très souvent le NameNode au courant des blocs qu'il gère, et c'est avec ce principe qu'il est possible au NameNode de détecter des problèmes et de demander la

réplication des blocs. Les DataNodes ne gèrent pas de fichiers, mais seulement des blocs. La notion de fichier est géré par le NameNode. Il va pouvoir ouvrir, fermer, supprimer des fichiers, et répercuter ces changements aux DataNodes concernés. Il va donc demander aux DataNodes de créer des blocs, de les supprimer, de les lire ou écrire dedans. Le NameNode peut poser problème en cas de défaillance de son nœud, c'est pour cela qu'il existe un Secondary NameNode, qui va recevoir de temps en temps les données du NameNode, et qui va pouvoir, en cas de défaillance du NameNode, de prendre sa place [1] [34].

Le rack est une collection physique de nœuds dans le cluster Hadoop, un rack peut avoir plusieurs DataNodes stockant les blocs de fichiers et leurs répliques. Cependant à l'aide des informations du rack, le NameNode choisit le DataNode le plus proche pour atteindre des performances maximales tout en effectuant les informations de lecture/écriture qui réduisent le trafic réseau [37] (Voir la Figure 7).

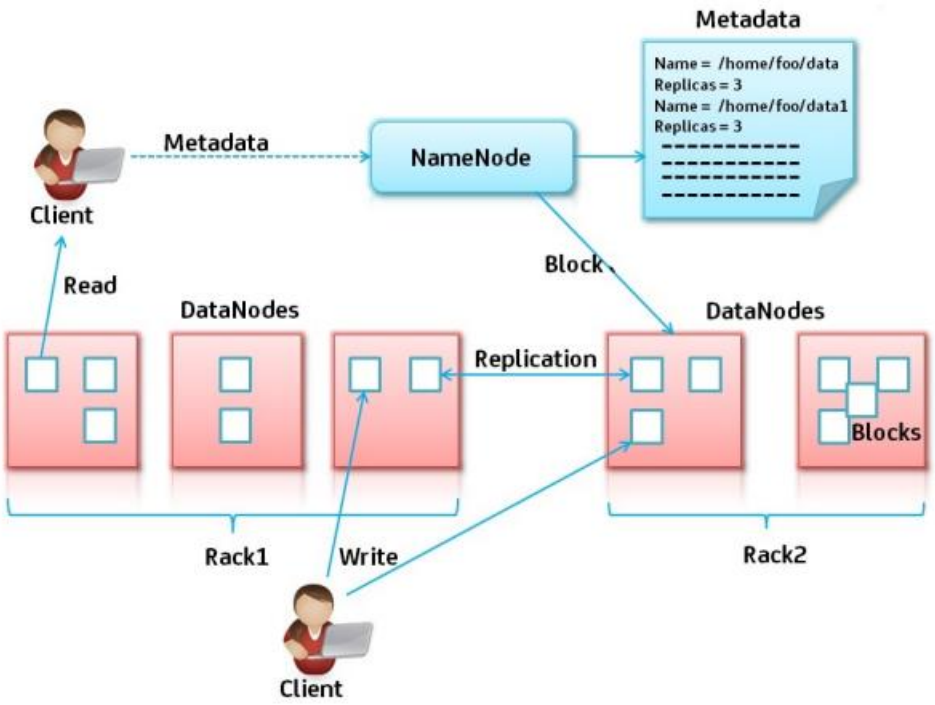


Figure 1.7: Architecture de HDFS[38].

- **Apache Hadoop YARN :**

est une technologie de gestion de clusters. Elle rend l'environnement Hadoop mieux adapté aux applications opérationnelles qui ne peuvent pas attendre la fin des traitements par lots[39]. Le fonctionnement de son architecture se fait avec un système maîtres/esclaves. Le RessourceManager représente l'autorité suprême du cluster. Il est composé de deux rôles : la

gestion des ressources du cluster et la gestion des applications. Il va donc gérer la soumission des applications sur le cluster, et va donc assigner à chaque application des ressources d'un nœud (ce qu'on appelle un conteneur ou Container) qui pourra gérer l'exécution de cette application. L'exécution des applications n'est donc pas centralisé sur un seul nœud. Chaque application aura donc son ApplicationMaster tournant sur un nœud du cluster. La gestion des ressources du cluster se fait avec le Scheduler, qui fait partie du ResourceManager. Il va devoir assigner aux ApplicationMaster des ressources venant de nœuds suivant la demande de ces-derniers, et suivant le type d'ordonnancement. Les applications peuvent être différentes, mais elles ne sont traitées selon leurs types par le Scheduler, elles sont traitées par leurs demande en ressources sur le cluster. La Figure 1.8 schématise un exemple de distribution des ressources d'un cluster pour deux applications[34]. Chaque nœud du cluster est composé d'un NodeManager, qui va gérer les demandes de ressources sur ce nœud. Il va tenir le ResourceManager au courant grâce au heartbeat. Le heartbeat est envoyé par tous les nœuds au ResourceManager pour donner ses informations. Les ressources demandées, regroupées en conteneurs, sont des ressources d'une machine : la mémoire, le disque et le réseau, etc [1] [34].

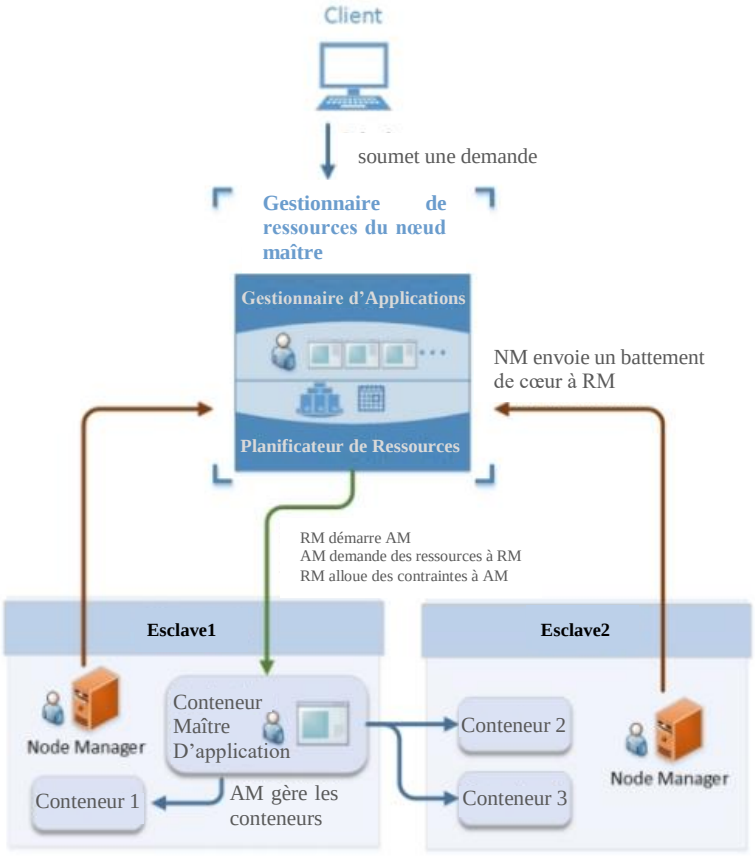


Figure 1.8: Architecture YARN[40].

- **Apache Hadoop MapReduce :**

est un modèle de programmation créé par Google pour le traitement et la génération de larges ensembles de données sur des clusters d'ordinateurs dans lequel sont effectués des calculs parallèles, et souvent distribués, de données potentiellement très volumineuses, typiquement supérieures en taille à 1 téraoctet. Il s'agit d'un composant central du Framework logiciel Apache Hadoop, qui permet le traitement résilient et distribué d'ensembles de données non structurées massifs sur des clusters d'ordinateurs, au sein desquels chaque nœud possède son propre espace de stockage. MapReduce est composé de deux processus essentiels qui sont JobTracker et des nœuds appelés TaskTracker. Le JobTracker doit savoir l'état du cluster et l'ensemble des nœuds qui sont actifs et c'est lui qui recevra les différentes tâches à effectuer pour les distribuer par la suite aux autres nœuds ou TaskTracker qui vont réaliser ces tâches. Un TaskTracker est chargé d'exécuter des tâches de Map ou de Reduce [1] [2].

Le fonctionnement de MapReduce s'articule principalement autour de deux fonctions : Map, et Reduce. Map sert à décomposer et à cartographier les données, Reduce mélange et réduit les données[41].

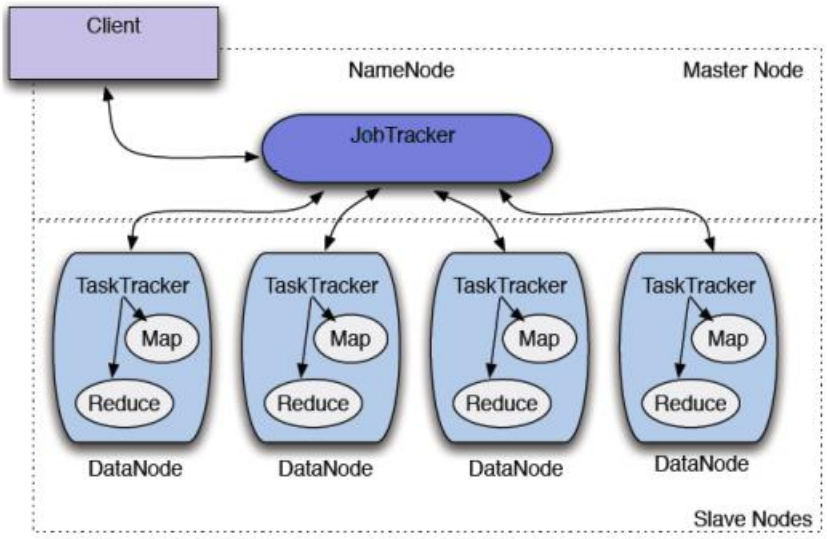


Figure 1.9: Architecture de MapReduce[42].

1.10 Les avantages du Big Data :

Les principaux avantage sont :

1. **Obtenir un feedback en temps réel :**

La collections des informations en temps réel effectués par les outils de Big Data permet d'obtenir un retour d'information continu sur les actions mises en place et de réagir rapidement aux problèmes qui surviennent ou aux changements radicaux du marché[43].

2. **Optimisation des coûts :**

La rédaction des coûts du stockage, du traitement et de l'analyse de données massives est l'un des avantages les plus importants du Big Data pour les entreprises. Les outils de Big Data permettent aussi d'identifier des manières efficaces et plus économiques de faire des affaires[44].

3. **Efficacité :**

Les outils des Big Data permet d'améliorer l'efficacité au quotidien de manière significative, et d'accumuler une grande quantité de données utiles. Celles-ci peuvent ensuite être analysées et interprétées pour définir des patrons significatifs, qui permettent aux entreprises de concevoir des produits et des services personnalisés[44].

4. **Innovation :**

Les outils d'analyse de données nous permettent d'extraire des informations précieuses qui peuvent se transformer en stratégies et en décisions commerciales fondamentales pour innover. Modifier des stratégies, développer de nouveaux produits ou services pour résoudre les problèmes concrets de clients, améliorer les techniques de marketing, le service client ou la productivité des employés[44].

5. **Extensibilité (scalabilité) :**

le concept Big Data apporte une architecture scalable qui peut prévenir la taille de l'infrastructure et l'espace disque nécessaire[1].

6. **Disponibilité :**

on a plus besoin des RAID disques, souvent coûteux. L'architecture Big Data apporte ses propres mécanismes de haute disponibilité [1].

1.11 Les limites du Big Data :

1. **Stockage :** les ensembles de données peuvent nécessiter des ressources considérables pour être stockés[45].
2. **Mise en forme et nettoyage des données** : il peut être nécessaire de faire appel à des connaissances informatiques avancées avant que les données ne soient analysables[45].
3. **Contrôle de la qualité :** peut être difficile et doit souvent être effectué à partir de petits échantillons représentatifs[45].
4. **Les questions de sécurité :** souvent plus complexes que pour les ensembles de données traditionnels[45].
5. **Précision et cohérence des méthodes :** de nombreuses approches tel que : Security Analysis and Improvement for Kerberos Based on Dynamic Password and Diffile-Hellman Algorithm[63] et Research on Hadoop Identity Authentication Based on Improved Kerberos Protocol[64]. sont relativement nouvelles et imparfaites, même si elles peuvent continuer à s'améliorer avec le temps[45].

1.12 Conclusion :

Les données collectées dans le cadre d'une étude Big Data peuvent avoir des origines et des formes très différentes, elles sont ensuite traités à travers des algorithmes et des structures adaptés dans le but de produire une information. Aux fondements des algorithmes des solutions de Big Data se trouvent des lois de statistiques et de probabilités, telles que la loi normale où les moindres carrés, d'un autre côté il y'a des risque de sécurité qui peut menacer les données massive utilisée par la population tel que l'authentification, la confidentialité, autorisation…etc.

Dans le deuxième chapitre, nous allons aborder et détailler le concept d'authentification des Big Data. Nous allons aussi définir ses types ainsi que étudier des travaux relatifs à ce sujet.

Chapitre 2 Authentification dans un environnement BIG DATA Sous Hadoop

2.1 Introduction :

Dans ce chapitre nous allons parler sur la sécurité des Big Data d'une manière générale, et définir quelque type d'authentification comme l'authentification par mot de passe, l'authentification cryptographique, et l'authentification biométrique. Nous allons ensuite présenter les travaux relatifs à ce sujet.

2.2 La sécurité dans le Big Data :

La sécurité de l'information est définie par six éléments et si nous omettons l'un d'entre eux, elle sera diminuée. Ce sont les six scénarios de pertes d'informations[46].

- **Disponibilité :**

La conservation de la disponibilité doit être considérée comme un objectif de la sécurité de l'information, et sa perte est un problème majeur. Et la preuve de son importance est ce qui se passe avec la coopérative de crédit[46].

- **Utilité :**

Dans cet élément, nous parlons des informations qui sont disponibles mais pas utiles. Comme la clé de chiffrement est détruite ou manquante, pour préserver l'utilité de l'information, nous avons besoin d'un mécanisme de protection robotisé tel que la cryptographie, des tests de sécurité précis lors du développement de l'application. Le mécanisme le plus important est de disposer de copies de sauvegarde obligatoires de toutes les informations critiques[46].

- **Intégrité :**

Il existe différents types de perte d'intégrité de l'information, comme des parties importantes de l'information qui peuvent être manquantes ou mal ordonnées (mais toujours disponibles) et nous ne pouvons pas les restaurer. Nous pouvons appliquer plusieurs contrôles pour prévenir la perte d'intégrité de l'information, notamment des tests manuels et automatiques, la vérification des numéros de séquence, des sommes de contrôle et/ou des totaux de hachage pour garantir l'exhaustivité et l'intégralité d'une série d'éléments[46].

- **Authentification :**

La gravité de la perte d'authenticité peut prendre plusieurs formes : absence de conformité à la réalité sans possibilité de récupération, informations modérément fausses ou trompeuses avec

récupération tardive à un coût modéré, ou informations factuellement correctes avec seulement des divergences gênantes[46].

L'industrie informatique comprend donc la nécessité de prouver l'authenticité des mises à jour des systèmes d'exploitation et des sites Web[46].

- **Confidentialité :**

Selon la plupart des experts en sécurité, la confidentialité concerne la divulgation, mais la confidentialité peut être perdue par l'observation, que cette observation soit volontaire ou involontaire. Le contrôle du maintien de la confidentialité nécessite l'utilisation de la cryptographie. Le pire scénario de perte de confidentialité qui coûte les dommages irrécupérables est lorsqu'une partie ayant l'intention et la capacité de causer des dommages observe les informations sensibles d'une victime[46].

- **Possession :**

La définition de la confidentialité ne concerne que les informations secrètes que les personnes peuvent posséder. Mais toutes les informations doivent être détenues, qu'elles soient confidentielles ou non, et perdre la possession des informations de l'entreprise lui coûtera très cher. Avec cette perte, nous pouvons perdre la confidentialité de l'information parce que nous ne contrôlons pas cette information[46].

Nous pouvons protéger la possession en appliquant différents types de contrôles, notamment en mettant en œuvre des limitations d'utilisation physiques et logiques, en utilisant les lois sur les droits d'auteur, en préservant et en examinant les journaux d'audit informatique pour trouver des preuves de vol, en inventoriant les actifs tangibles et intangibles, en utilisant des couleurs et des étiquettes distinctives sur les supports et en attribuant la propriété pour assurer la responsabilité des actifs informationnels de l'organisation[46].

2.3 Types d’authentification :

On distingue trois type d’authentification :

2.3.1 Authentification par mot de passe (Password-based authentication) :

Le mécanisme d’authentification par mot de passe est un processus qui permet à un utilisateur d’avoir les droit d’accès aux ressources en introduisant un ensemble d’information contenant un nom d’utilisateur et un mot de passe, cette méthode d’authentification est largement utilisée connue par sa simplicité, sa rentabilité, sa facilité d’utilisation et son aspect pratique.[49]

2.3.2 L'authentification cryptographique (Cryptographic-based authentication) :

L'authentification cryptographique concerne la reconnaissance d'une entité comme étant en possession d'une clé cryptographique secrète. L'entité peut être un dispositif contenant la clé, ou un utilisateur possédant un tel dispositif. L'authentification cryptographique peut comprendre la vérification de l'identité ou de certains attributs du propriétaire de l'appareil revendiqués par un tiers faisant autorité, ou la reconnaissance de l'utilisateur en tant que visiteur récurrent ou en tant que propriétaire d'un compte précédemment établi[47].

L'authentification cryptographique peut être combinée avec d'autres facteurs d'authentification pour la vérification d'identité à distance à l'aide d'informations d'identification enrichies et pour l'authentification du titulaire de la carte[47].

2.3.2.1 Authentification par clé symétrique :

l'authentification par clé symétrique, l'utilisateur partage une seule clé secrète avec un serveur d'authentification (normalement la clé est intégrée dans un jeton) [48]. L'utilisateur peut être authentifié en envoyant au serveur d'authentification son nom d'utilisateur ainsi qu'un message de défi aléatoire chiffré par la clé secrète. Ainsi, l'utilisateur est considéré comme un utilisateur authentifié si le serveur peut faire correspondre le message chiffré reçu à l'aide de sa clé secrète partagée [49].

Exemple d'un algorithme pour établir une clé secrète (Algorithme Diffie-Hellman) :

Supposons qu'Alice et Bob souhaitent se mettre d'accord sur la clé secrète.

L'algorithme Diffie-Hellman permet l'établissement de cette clé secrète, via les étapes présentés dans le tableau 1 :

	Alice	Bob
Etape1	Alice et Bob choisissent deux grands nombres premiers p et g (générateur).	
Etape2	Alice choisis une clé secrète x.	Bob choisis une clé secrète y.
Etape3	Alice calcule X = g^x mod p.	Bob calcule Y = g^y mod p.
Etape4	Alice envoie le résultat X à Bob.	Bob envoie le résultat Y à Alice.
Etape5	Alice calcule Z1 = Y^x mod p.	Bob calcule Z2 = X^y mod p.

Tableau 2.1: Algorithme de Diffie-Hellman

2.3.2.2 Authentification par clé publique :

La cryptographie à clé publique utilise un algorithme mathématique avec une paire de clés (clé publique/ clé privée) pour chiffrer et déchiffrer les données. L'une des clés est une clé publique,

qui peut être librement distribuée aux parties communicantes, tandis que l'autre est une clé privée, que son propriétaire doit garder secrète. Les données chiffrées avec la clé privée peuvent être déchiffrées uniquement avec la clé publique et inversement[50].

Lorsque les clés sont utilisées pour l'authentification, la partie authentifiée crée une signature numérique à l'aide de la clé privée d'une paire de clés, la clé privée peut donc être utilisée pour chiffrer le code de hachage afin d'établir une signature numérique[52]. Le hachage est la transformation d'une chaîne de caractères en valeur ou en clé de longueur fixe, généralement plus courte, représentant la chaîne d'origine[51]. Les Hachages garantissent l'intégrité des données qui empêchent les attaquants de l'altération des informations transmise entre les deux entités autorisés.

Le destinataire doit utiliser la clé publique correspondante pour vérifier l'authenticité de la signature numérique. Cela signifie que le destinataire doit posséder une copie de la clé publique de l'autre partie et être certain de l'authenticité de cette clé[50]. Dans ce cas l'émetteur ne peut pas nier toute action effectué par lui-même, ce qui permet d'assurer la non-répudiation des informations.

2.3.3 Authentification biométrique (Biometric authentication) :

Une authentification biométrique est une numérisation des mesures d'une caractéristique physiologique ou comportementale pour l'humain. Un système d'authentification biométrique peut théoriquement être utilisé pour déterminer une identité d'une personne. Cependant, de nombreux systèmes d'authentification biométrique ont été proposés qui sont classés comme : système d'authentification par détection de visage, système d'authentification par empreintes digitales, système d'authentification Iris et système d'authentification vocale [53].

2.3.3.1 Reconnaissance d'empreintes digitales :

Un système d'empreintes digitales utilise un dispositif électronique pour capturer une image numérique du motif d'empreintes digitales. Cette image capturée dans le système d'empreintes digitales est appelée une analyse en direct qui est traitée numériquement pour créer un modèle biométrique (caractéristiques des doigts). Les caractéristiques biométriques seront ensuite stockées et utilisées pour le processus de correspondance[49].

Figure 2.10: Reconnaissance d'empreintes digitales.

2.3.3.2 Authentification biométrique vocale :

L'authentification biométrique vocale est l'utilisation du modèle vocal pour reconnaître l'identité de la personne. Pendant ce temps, l'authentification vocale est désormais considérée comme un large forme déployée d'authentification biométrique. Cependant, c'est l'une des meilleures méthodes pour déterminer l'efficacité de la méthode biométrique. Cependant, la reconnaissance vocale est classée en cinq types: système dépendant du locuteur, système indépendant du locuteur, reconnaissance vocale discrète, reconnaissance vocale continue et langage naturel[49].

Figure 2.11: Authentification biométrique vocale.

2.3.3.3 Détection de visage :

Les systèmes de détection et de reconnaissance faciale sont deux scénarios complémentaires [53]. La détection des visages est telle que la technologie utilise des algorithmes d'apprentissage pour attribuer des visages humains dans des images numériques. Comme le montre la figure 8, l'algorithme de détection de visage se concentre et détermine les traits du visage et ignore tout le reste dans les images numériques [53, 54].

Par ailleurs, de nombreuses techniques de détection de visage ont été présentées telles que ; Détection de visage Viola et Jones [55], Adaboost basé sur la détection de visage [56], apprentissage semi-supervisé pour la reconnaissance des expressions faciales [57], etc.

Figure 2.12: Détection de visage.

2.3.3.4 Authentification de l'iris :

En fait, l'iris et les empreintes digitales sont parallèles dans leur technologie d'unicité. Dans le monde, le résultat statistique de l'utilisation de l'iris dans l'authentification est que l'iris est l'un des meilleurs moyens de faire face aux situations à haut risque. Le logiciel de reconnaissance de l'iris est actuellement largement utilisé aux frontières des aéroports. De plus, il est également largement utilisé dans de nombreux autres secteurs pour effectuer l'authentification[49].

Figure 2.13: Authentification de l'iris.

2.4 L'authentification dans Hadoop :

Plusieurs travaux ont été proposés et réalisés dans le but d'assurer la sécurité de l'environnement Hadoop.

Dans ce qui suit, nous allons se concentrer seulement sur les travaux relatifs à l'authentification Big Data.

Ces protocoles peuvent être basés sur l'utilisation du protocole d'authentification Kerberos dans hadoop.

Kerberos peut se basé sur le chiffrement symétrique ou asymétrique et il repose sur trois serveurs pour assurer l'authentification :

- Un serveur d'authentification (AS : Authentication Server) qui prend en charge toute la partie authentification pur du client. C'est lui seul qui peut permettre au client de communique au TGS (grâce à un ticket d'accès).
- Le serveur de distribution de tickets TGS : (Ticket Granting Server) prend en charge les demandes d'accès aux services des clients déjà authentifiés. L'ensemble des infrastructures serveur de Kerberos AS et TGS est appelé le centre de distribution de clés (KDC : Key Distribution Center).
- Serveur d'application : C'est un serveur qui exécuter un service particulier.

Exemple d'une version asymétrique du protocole kerberos :

Yao Yao et al. Ont proposé un nouveau modèle d'authentification de domaine hétérogène croisé principalement basé sur PKI, et ils ont conçu les détails des processus d'authentification dans différentes situations. Le modèle réalise efficacement l'authentification inter domaine entre le domaine PKI et le domaine Kerberos et prend en charge les authentifications mutuelles. L'analyse théorique montre que le schéma proposé a une bonne compatibilité, extensibilité et fiabilité. Par conséquent, ce modèle est adapté à une utilisation dans un environnement de réseau à grande échelle principalement basé sur PKI [83].

2.5 Travaux Relatifs :

Kerberos est un protocole d'authentification réseau qui repose sur un mécanisme de clés secrètes (chiffrement symétrique) et l'utilisation de tickets, et non de mot de passe en clair, évitant ainsi le risque d'interception frauduleuse des mots de passe des utilisateurs [58].

El-Emam et al. ont présentés des modifications simple à la base de données du protocole d'authentification Kerberos largement déployé. La clé secrète à long terme du principe sera indépendante du mot de passe de l'utilisateur dans le but de surmonter les mots de passe faibles choisis par le principal du réseau qui sont sensibles aux attaques par devinette de mot de passe, le principal inconvénient du protocole Kerberos. Au lieu de cela, le centre de distribution Kerberos enregistrera un profil pour chaque instance du domaine qu'il gère et la clé secrète sera générée en fonction de ce profil. Ce dernier sera haché, puis le résumé de sortie sera chiffré

pour générer la clé secrète. En outre, la durée de vie de la clé secrète sera contrôlée à l'aide de la durée de vie du système. Ils ont utilisé 3DES comme algorithme de chiffrement, SHA-256 comme algorithme de hachage et Blum Blum Shub comme algorithme générateur de nombres aléatoires[59].

Le protocole Secure Remote Password (SRP) est un protocole d'authentification cryptographiquement fort pour l'authentification mutuelle basée sur un mot de passe sur une connexion réseau non sécurisée. Une authentification SRP réussie nécessite que les deux côtés de la connexion connaissent le mot de passe de l'utilisateur. En plus de la vérification du mot de passe, le protocole SRP effectue également un échange de clé sécurisé pendant le processus d'authentification. Cette clé peut être utilisée pour protéger le trafic réseau via un chiffrement à clé symétrique [60].

Miti Jhaveri et al. Ont proposé une version modifiée du protocole SRP qui fusionne les processus d'authentification et d'autorisation, cependant ils ont proposé un modèle pour fournir un contrôle d'accès dynamique basé sur les attributs en adoptant la recherche effectuée dans un modèle statique basé sur les rôles et en l'intégrant au protocole SRP. Ils ont également élaboré les attaques repoussées par le protocole LV-SRP en prouvant mathématiquement la répulsion. Ce modèle est flexible pour s'adapter à différentes exigences d'attributs selon les besoins de l'organisation. Il s'efforce de fournir une mécanisme de contrôle d'accès plus fort et complexe en raison des cas d'utilisation qui sont généralement activés par eux. Il s'efforce également d'augmenter la vitesse à laquelle le processus fusionné est effectué par rapport aux méthodes traditionnelles [61].

S. Ramasamy et RK Gnanamurthy ont proposé un mécanisme d'authentification utilisateur multicouche Single Sign-On (SSO), Two-Factor Authentication (2FA), Multi-Factor Authentication (MFA) pour sécuriser les mégadonnées dans le cloud computing. Tout d'abord ils ont traité des mesures de sécurité et des défis de cloud computing, après cela ils ont proposé une conception de stockage de centre de données d'authentification utilisateur multicouche basée sur un cluster pour défendre le Big Data dans une atmosphère de cloud computing. Cette solution garantisse la sécurité des données massives grâce à l'authentification dans le cloud, d'ailleurs ses avantages est d'augmenter la flexibilité et l'efficacité et la simplification du processus de connexion[62].

Chundong Wang et Chaoran Feng ont étudié le déroulement de processus du protocole d'authentification Kerberos, après une analyse ils ont trouvé les problèmes liés à ce protocole

tel que : l'attaque par rejeu, attaque par dictionnaire, problème de stockage des clés, et enfin les attaques de logiciels malveillants, à partir de ces derniers ils ont mis en avant les pensées et les méthodes utilisant le mot de passe dynamique pour améliorer la sécurité de cryptage pendant le processus d'interaction entre le client et le KDC, et rend les mots de passe échangés de manière sécurisée en utilisant l'algorithme de clé Diffile-Hellman[63].

Daming Hu et al. Ont amélioré le processus d'authentification du protocol kerberos dans un environnement du cluster HDFS, tout d'abord ils ont étudié le mécanisme d'authentification du protocole Kerberos sous HDFS, et soulignent les problèmes auxquels ce mécanisme est confronté tel que : synchronisation temporelle, sécurité du KDC, attaques par dictionnaire et mécanisme de déni. à partir de ces problèmes de sécurité , les auteurs ont pu modifier le protocole Kerberos en utilisant le chiffrement à clé publique et le mécanisme de signature des données. Une analyse complète montre que la sécurité et l'efficacité temporelle du protocole Kerberos amélioré sont améliorées par rapport au mécanisme d'authentification d'identité existant. Il fournit une solution d'authentification de l'identité plus fiable et plus efficace pour le cluster HDFS[64].

M. Hena et N. Jeyanthi ont proposé un nouveau cadre d'authentification pour les clusters Hadoop kerberisés qui utilisent un mot de passe à usage unique One Time Pad (OTP) amélioré qui peut résoudre certains problèmes de sécurité tel que le point de défaillance unique, le point de vulnérabilité unique, et la menace interne, ainsi que le problème de synchronisation de l'heure. Cette méthode améliorée est une solution fiable avec moins de surcharge de communication et de calcul. Des résultats de la simulation dans Riverbed Modeler prouvent que le modèle proposé fonctionne aussi bien que le mécanisme d'authentification kerberos traditionnel[65].

Le département Big Data d'Intel a développé un service d'authentification Apache Hadoop* (HAS) qui est compatible avec le mode d'authentification du protocole Kerberos, de sorte que tous les composants de l'écosystème Hadoop peuvent utiliser le mode d'authentification Kerberos d'origine fourni par HAS. HAS étend Kerberos avec une intégration d'identité plus flexible. Le nouveau mécanisme d'authentification (authentification par jeton basée sur Kerberos) prend en charge la plupart des composants de l'écosystème Hadoop et n'apporte que peu ou pas de modifications aux composants. HAS fournit une série d'interfaces et d'outils pour aider à simplifier le déploiement. Il fournit également des interfaces pour aider les utilisateurs à implémenter des plug-ins pour intégrer Kerberos à d'autres systèmes de gestion de l'identité des utilisateurs[66].

Ashok Kumar, ont proposé un nouveau mécanisme d'authentification d'identité qui est suggéré sous HDFS pour modifier le protocole Kerberos avec moins de puissance de calcul et une valeur de débit élevée de l'image efficace de l'algorithme cryptographique à clé symétrique en tant que clé secrète. Ainsi, le mécanisme d'authentification Kerberos sous HDFS est amélioré avec de bonnes pratiques de sécurité et surmonte les limitations de Kerberos. Cette modification suggérée pour l'authentification Kerberos est applicable à toutes les implémentations open source de systèmes d'exploitation comme Debian et BOSS (Bharat Operating System Solutions). Le chiffrement et le déchiffrement des données au repos est la bonne pratique pour assurer la sécurité et en dehors de cela, GnuPG (GNU Privacy Guard) prend en charge la sécurité des données, la confidentialité des utilisateurs et le contrôle d'accès pour Hadoop Cluster. Ainsi, les données au repos et les données en transit sont sécurisées avec un processus de cryptage et de décryptage qui protège les données de l'utilisateur et de l'administrateur malveillants [67].

Wenyi Liu et al. Ont développé un système d'authentification multi-facteur qui préserve la confidentialité sans introduire de dispositif physique supplémentaire pour les systèmes en nuage avec l'utilisation des caractéristiques du Big Data. Dans ce systèmes d'authentification appelé MACA, le premier facteur est un mot de passe et le second facteur est un profil utilisateur hybride qui résume le comportement de l'utilisateur. A Privacy-Preserving Multi-factor Cloud Authentication System (MACA) se concentre sur la préservation de la confidentialité du second facteur, ce qui présente deux avantages par rapport aux systèmes précédemment proposé. Premièrement, la confidentialité de l'utilisateur n'est pas perdue dans un environnement omniprésent de cloud computing avec le Fully homomorphic encryption (FHE) et le hachage flou. Deuxièmement, le modèle hybride de profilage de l'utilisateur est hautement utilisable et configurable et intègre un grand nombre de caractéristiques et de données correspondantes, ce qui permet des opérations d'AMF (Authentication Multi-Factor) simples préservant la vie privée avec des calculs Fully homomorphic encryption (FHE) et de hachage flou. Ils ont évalué les performances du système via une série d'expériences avec l'utilisation de quatre jeux de données différents. Ils ont obtenu un rappel optimal de 80,8% et un False Positive Rate (FPR) de 14,7 %. En outre, le surcoût du système et l'utilisation des ressources se situent dans une fourchette acceptable, ce qui prouve la faisabilité de leur système[68].

Anas Ibrahim a proposé un modèle qui vise à révolutionner le processus d'authentification en adoptant l'approche «quelque chose que vous faites». Cette approche fournit tous les moyens nécessaires pour prendre avantage du facteur «quelque chose que vous faites» et offre un moyen

plus pratique et un processus d'authentification plus sûr. Le modèle d'authentification proposé tente de générer des profils d'utilisateurs, et générer un ensemble de questions difficiles en temps réel. Ces questions couvrent toutes les actions qui représentent le comportement instantané de l'utilisateur spécifique. La particularité de cette approche est que les questions seront choisis de manière à ce que les exigences de sécurité et de convivialité soient maintenues. Les questions utilisées ne seront émises qu'une seule fois ; qui assure leur unicité, pour protéger la réponse des utilisateurs d'être compromis. Cela permet de s'affranchir des écueils des méthodes Knowledge-based authentication (KBA). Et les sources d'informations infinies garantiront que les données seront fraîches pour aider à légitimer l'utilisateur à se souvenir facilement et à relever avec succès le défi. En plus des avantages que cette approche offre, elle fait face à des défis qui pourraient entraver sa progression. Ces défis sont les préoccupations de confidentialité des utilisateurs en ce qui concerne les données qui sont utilisées et la légalité de l'utilisation de données provenant de différentes sources[69].

Ruidong Li et al. Ont proposé DCAuth (Data-Centric Authentication for Secure In-Network Big-Data Retrieval), qui fournit une authentification centrée sur les données en fusionnant la confiance basée sur l'AC (Autorité d'authentification) et la confiance basée sur le voisinage. Elle permet l'authentification entre des entités telles que les utilisateurs, les IPE (Intermediate physical entities), les détenteurs de copies et les éditeurs, indépendamment de leur imprévisibilité. Des simulations approfondies ont été réalisées et montrent que DCAuth peut réduire le délai de collecte des certificats par rapport à PKI-NDN (Public Key Infrastructure for Named Data Networks) et peut prévenir efficacement les attaques par requête malveillante et empoisonnement des données[70].

Risha Tabasuum et Dr. Nidhi Tyagi. Ont proposé une stratégie améliorée du protocole kerberos basée sur un système de cryptage à clé publique et le mécanisme de signature de données avec un canal de communication sécurisé à l'aide de RSA. Enfin, ils ont fourni une solution d'authentification d'identité plus fiable et efficace pour HDFS [71].

Une blockchain est un grand livre numérique public distribué et décentralisé qui est utilisé pour stocker les enregistrements de transactions financières en termes de blocs sur plusieurs ordinateurs connectés en réseau afin que tout enregistrement impliqué qui est stocké dans des blocs et placé sur des ordinateurs ne puisse pas être modifié rétrospectivement[72].

Dr. Pramod Patil et al. ont voulu présenter les faiblesses des implémentations de Kerberos et identifier les besoins d'authentification susceptibles d'améliorer la sécurité des informations

volumineuses dans les environnements distribués. Le mécanisme de développement sera une nouvelle perspective d'utilisation de la blockchain dans Hadoop pour l'authentification au lieu de Kerberos. Le mécanisme repose sur la technologie montante de la blockchain qui surmonte les lacunes de Kerberos. L'auteur a utilisé les concepts de base de la blockchain et créé un modèle client HDFS de mécanisme d'authentification basé sur la blockchain pour le cadre de données volumineuses qui peut coexister avec la configuration Hadoop, où il décrit et implémente une méthodologie de blockchain privée qui pourrait être implémentée pour une organisation privée dans la configuration Hadoop. En outre, il fournit les diverses fonctionnalités opérationnelles de base pour l'administrateur de la blockchain et le client HDFS pour l'utilisateur final, ainsi qu'un mécanisme d'authentification locale distribuée utilisant la blockchain [72].

Issa Khalil et al. Ont conçu et mis en œuvre un protocole d'authentification pour les systèmes Hadoop basé sur Trusted Technologies de module de plate-forme (TPM) qui fournit une authentification mutuelle forte entre toutes les entités Hadoop interagissant en interne, en plus de s'authentifier mutuellement avec des clients externes. Les opérations de liaison et de scellement prises en charge par le TPM pour assurer leur protection contre les initiés malveillants, car les initiés ne peuvent pas modifier l'état de la machine sans affecter les valeurs PCR. De plus, le protocole fournit des services d'attestation de plate-forme à distance aux clients de fournisseurs Hadoop tiers, éventuellement non fiables. Ainsi que, le sceau de la clé de session protège contre la possibilité de divulguer les données chiffrées dans toute autre plate-forme que celle qui correspond aux configurations de confiance spécifiées par les entités communicantes. Enfin, leur protocole élimine l'exigence d'un tiers de confiance (comme Kerberos KDC) avec tous ses problèmes associés tels que le point de défaillance unique, la disponibilité en ligne et la concentration de la confiance et des informations d'identification [73].

2.6 Conclusion :

L’authentification est un processus qui permet de comparer les informations fournit par l’utilisateur à celle des utilisateurs autorisés pour leur accorder les droits d’accès.

Dans ce chapitre nous avons cité les types d’authentification les plus récemment utilisés ainsi que les travaux relatifs.

Chapitre 3 Etude comparative

3.1 Introduction :

Kerberos est l'un des protocoles largement appliqué dans le mécanisme d'authentification dans l'environnement du cluster Hadoop. Il permet d'autoriser les utilisateurs à l'aide d'un ticket distribué par le centre d'authentification kerberos.

Dans ce chapitre, nous allons comparer deux versions du protocole kerberos (la version symétrique et la version asymétrique) afin d'estimer le temps de traitement de chaque protocole.

3.2 Mise en place des protocoles à comparer:

La versions symétrique du protocole kerberos cité dans [1] est comparée avec celle cité dans [86] afin d'estimer leur temps de traitement. Les étapes de la mise en place de la première versions du protocole kerberos sont bien détaillé dans [1]. Ces étapes sont aussi présenté dans l'annexe B, C et D.

3.2.1 Configuration de réseaux :

Tous d'abord, nous allons créer trois machine qui seront reliées sur le même réseau, par la suite nous allons éditer le fichier /etc/hostname sur chaque machine de notre cluster afin de leur donnée le nom d'hôte le plus significatif comme il est montré dans le tableaux 4 suivant :

Nom de la machine	Nom d'hôte
MASTER	krb5.abc.io
SLAVE1	client1.abc.io
SLAVE2	client2.abc.io

Tableau 3.1: Le nom d'hôte de chaque machine.

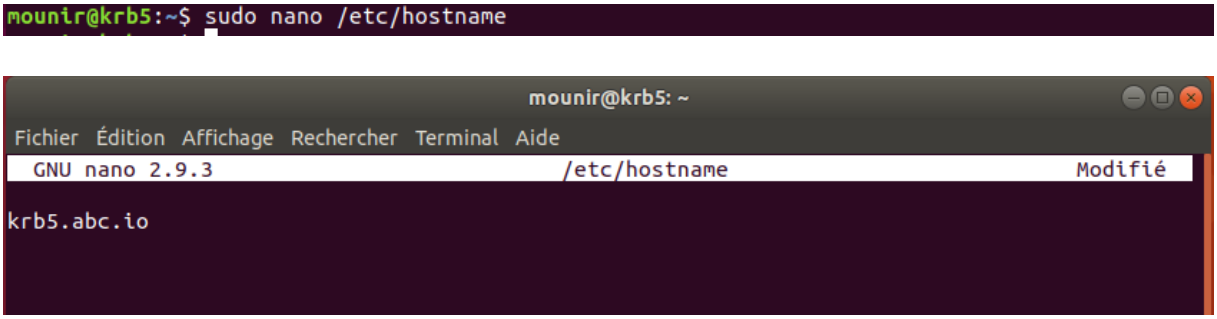


Figure 3.1: le fichier hostname.

Maintenant nous éditons le fichier **/etc/hosts** sur toutes les machines en spécifient l'adresse IP de chaque machine suivie de leurs noms d'hôtes comme illustrer dans la figure 20.

```
mounir@krb5:~$ sudo nano /etc/hosts
```

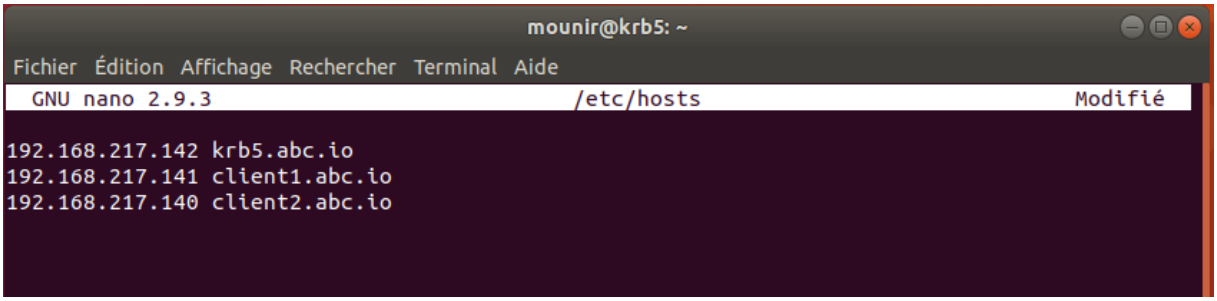


Figure 3.2: L'adresse IP de chaque machine.

La mise en place de l'autre protocole nécessite aussi l'installation d'openssl et la configuration de ce dernier avec la version adéquate de kerberos.

3.2.2 Installation openssl :

Avant de commencer l'installation d'openssl, nous devons installer le package **build-essential** sur **ubuntu**, ainsi que les packages **checkinstall** et **zlib1g-dev** qui permet aux applications de lire et d'écrire facilement des fichiers compatibles **gzip**[7].

```
mounir@master:~$ sudo apt install build-essential checkinstall zlib1g-dev -y
```

La commande ci-dessous nous permet de télécharger openssl à partir de la source. Dans notre cas nous utilisons openssl 1.1.1.

```
mounir@master:~$ cd /usr/local/src/
mounir@master:/usr/local/src$ sudo wget https://www.openssl.org/source/openssl-
1.1.1q.tar.gz
```

Maintenant que nous avons téléchargé le code source et installé toutes les dépendances nécessaires, nous utilisons la commande **tar** pour extraire le fichier **openssl-1.1.1q .tar.gz**, pour que nous puissions accédez à son contenu.

```
mounir@master:/usr/local/src$ sudo tar -xf openssl-1.1.1q.tar.gz
```

Ensuite, nous allons installer openssl que nous avons téléchargé à l'aide des commandes ci-dessous :

```
mounir@master:/usr/local/src/openssl-1.1.1q$ sudo ./config --prefix=/usr/local/
ssl --openssldir=/usr/local/ssl shared zlib
```

Dans la commande ci-dessus, les propriétés **--prefix** et **--openssldir** définissent le chemin de sortie d'openssl, cependant l'option **shared** force la construction à créer une bibliothèque partagée, et l'option **zlib** permet d'activer la compression[7].

```
mounir@master:/usr/local/src/openssl-1.1.1q$ sudo make
```

```
mounir@master:/usr/local/src/openssl-1.1.1q$ sudo make test
make depend && make _tests
make[1]: Entering directory '/usr/local/src/openssl-1.1.1q'
make[1]: Leaving directory '/usr/local/src/openssl-1.1.1q'
make[1]: Entering directory '/usr/local/src/openssl-1.1.1q'
```

Le résultat du test :

```
All tests successful.
Files=158, Tests=2644, 99 wallclock secs ( 1.93 usr  0.65 sys + 79.36 cusr 18.6
2 csys = 100.56 CPU)
Result: PASS
make[1]: Leaving directory '/usr/local/src/openssl-1.1.1q'
```

```
mounir@master:/usr/local/src/openssl-1.1.1q$ sudo make install
```

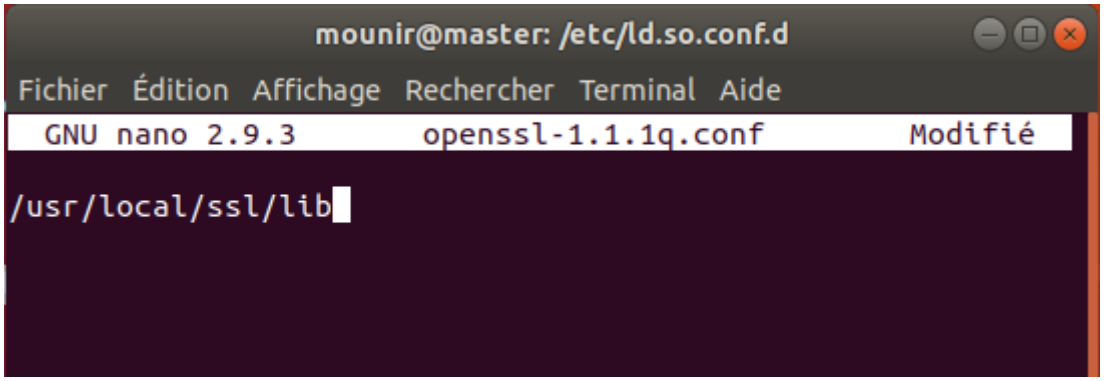
3.2.3 Configuration d'openssl :

Nous allons configurer les bibliothèques openssl partagées afin qu'elles se chargent au moment de l'exécution. Le binaire openssl nouvellement installé chargera les fichiers de bibliothèque à partir du répertoire **/usr/local/ssl/lib**[7].

Tout d'abord nous devons crée un nouveau fichier de configuration **openssl-1.1.1q.conf** dans le répertoire **/etc/ld.so.conf.d** à l'aide de la commande **nano**.

```
mounir@master:/usr/local/src/openssl-1.1.1q$ cd /etc/ld.so.conf.d
mounir@master:/etc/ld.so.conf.d$ sudo nano openssl-1.1.1q.conf
```

En ajoute dans le fichier de configuration openssl le chemin d'accès à la bibliothèque openssl comme suit :



Maintenant, nous enregistrons et quittons l'éditeur, et par la suite nous devons recharger le lien dynamique avec la sorite de débogage complète en utilisons la commande suivante :

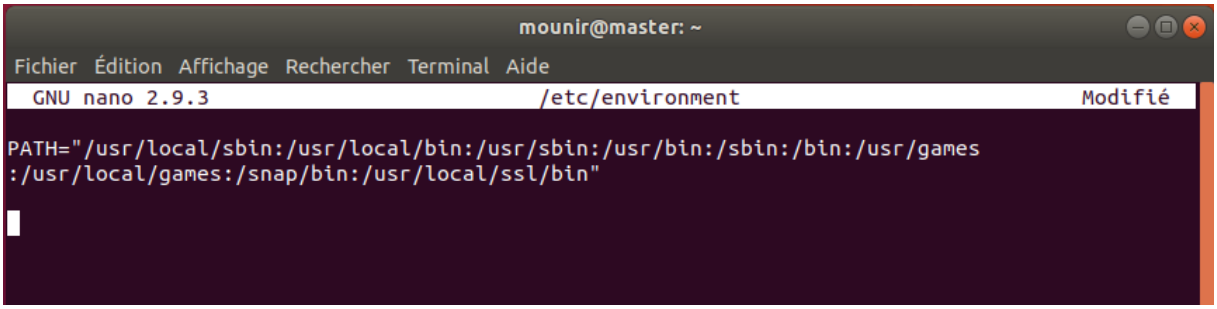
```
mounir@master:/etc/ld.so.conf.d$ sudo ldconfig -v
```

Nous sauvegardons les fichiers binaires comme suit :

```
mounir@master:/etc/ld.so.conf.d$ sudo mv /usr/bin/c_rehash /usr/bin/c_rehash.ba
ckup
mounir@master:/etc/ld.so.conf.d$ sudo mv /usr/bin/openssl /usr/bin/openssl.back
up
```

Ensuite, nous éditons le fichier **/etc/environment** pour ajouter le répertoire **/usr/local/ssl/bin** dans la variable d'environnement **PATH**. Cela nous permettra de configurer l'environnement système afin qu'il reconnaisse l'openssl nouvellement installé.

```
mounir@master:/etc/ld.so.conf.d$ sudo nano /etc/environment
```



Voici la version d'openssl:

```
mounir@master:~$ openssl version
OpenSSL 1.1.1  11 Sep 2018 (Library: OpenSSL 1.1.1q  5 Jul 2022)
```

3.3 Test :

Afin de tester les deux protocoles, nous allons exécuter la requête 'demande d'authentification' pour accéder au système de fichier Hadoop (HDFS).

Protocoles	Temps de traitement	
	Authentification échouée	Authentification réussie
Protocole 1[1] (versions symétrique)	3.126 s	3.289 s
Protocole 2[86] (versions asymétrique)	4.356 s	7.719 s

Tableau 3.2: Test

3.4 Conclusion :

Dans ce chapitre, nous avons mis en place la version symétrique du protocole kerberos en suivant les étapes nécessaires citées dans [1]. Ensuite, nous avons présenté les différentes étapes à suivre pour mettre en place la version asymétrique du protocole Kerberos. A la fin nous avons terminé par donner un exemple de fonctionnement de chaque version du protocole.

Conclusion générale et perspectives

Aujourd'hui, La sécurité des Big Data est importante pour la protection des données sensibles. Ces derniers sont recueillies à partir de diverses sources autonomes, où elles sont souvent entrelacées et décomposées pour produire des connaissances.

Au fur et à mesure que les données et les risques liés au Big Data augmentent, l'importance de prendre des mesures de sécurité est devenue une nécessité majeure. Les préoccupations des chercheurs en sécurité se sont concentrées sur la sécurité et l'assurance des données délicates. Cependant Apache Hadoop est l'une des plates-formes de traitement Big Data la plus utilisée pour l'analyse et le traitement des données. Hadoop prend en charge l'authentification de ses clients et utilisateurs à l'aide des différents mécanisme et protocole (comme kerberos) pour la sécurité.

Dans ce travail, nous avons présenté quelques généralités sur les Big Data, où nous avons défini ses caractéristiques, ses classification ainsi que ses domaines d'applications. Puis nous avons défini la sécurité des Big Data et les types d'authentification utilisées. Nous avons aussi présenté quelques travaux relatifs permettant d'assurer l'authentification.

Nous avons comparé deux versions (symétrique et asymétrique) du protocole kerberos afin d'estimer le temps de traitement de chaque version.

En guise de perspective pour les futurs travaux, nous envisagerons de réaliser l'objectif initial de ce projet qui consiste à faire une étude comparative des protocoles d'authentification basant sur l'utilisation des divers outils et méthode d'authentification (kerberos, certificats, chaîne de hachage, etc.) afin d'évaluer leurs performances en considérant d'autres critères d'évaluation (comme le coût et le nombre de ressources consommées).

Bibliographie & Webographie

Bibliographie :

[1] **Mlle MEDJRI Kahina, Mlle HADDOUCHE Soumia.** Mise en place d'un système d'authentification pour hadoop. Mémoire de master . Bouira : UNIVERSITE AKLI MOHAND OULHADJ-BOUIRA, 2019

[2] **Medfouni Hayet**. Validation de clustering des données dans un contexte big data, mémoire de master. Université Larbi Ben M'hidi Oum El Bouaghi, 2018.

[3] **Beatrice Adilin.** Analytics Insight. *ALL ABOUT THE BASICS OF BIG DATA: HISTORY, TYPES AND APPLICATIONS.* [En ligne] 2 Mars 2021. [Citation : 1 Avril 2022.] https://www.analyticsinsight.net/all-about-the-basics-of-big-data-history-types-and-applications/.

[4] **Phillips Andres.** TechTarget. A history and timeline of big data. [En ligne] 1 Avril 2021. [Citation : 27 Juin 2022.] https://www.techtarget.com/whatis/feature/A-history-and-timeline-of-big-data.

[5] **Philippe Laflamme.** Big Data et ses technologies. *Cours.* 2017.

[6] **Manyika, J., Chui, M., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C., and Byers,** A. H. Big data: The next frontier for innovation, competition, and productivity. McKinsey Global Institute, May 2011.

[7] **J. Gantz and D. Reinsel**, ―Extracting Value from Chaos State of the Universe: An Executive Summary,‖ IDCiView , no. June, pp. 1 – 12, 2011.

[8] BIG DATA ANALYTICS FOR IOT. **Preeti Gulia, Ayushi Chahal.** India : International Journal of Advanced Research in Engineering and Technology (IJARET), 2020, Vol. 11. 0976-6499.

[9] **Maxime VIGIER**. Les big data : une mine d'informations pour les entreprises, University of Economics and Finance Faculty of Business, Mémoire de Master. 2014

[10] **Tejada, Zoiner.** Microsoft. Big data architectures. [En ligne] [Citation : 27 Juin 2022.] https://docs.microsoft.com/en-us/azure/architecture/data-guide/big-data/.

[11] Big Data Security and Privacy: A Taxonomy with Some HPC and Blockchain Perspectives. **Khalil Alsulbi, Maher Khemakhem, Abdullah Basuhail, Fathy Eassa, Kamal Mansur Jambi, Khalid Almarhabi.** 5, 2021.

[12] The 17 V's Of Big Data. **Arockia Panimalar.S, Varnekha Shree.S, Veneshia Kathrine.A.** Tamilnadu, India : s.n., 2017, Vol. 04. e-ISSN: 2395-0056.

[13] The 10 Vs, Issues and Challenges of Big Data. **Nawsher Khan, Mohammed Alsaqer, Habib Shah, Gran Badsha, Aftab Ahmad Abbasi, Soulmaz Salehian.** 2018.

[14] BIG DATA CHARACTERISTICS, CLASSIFICATION AND CHALLENGES . **K S Ananda Kumar, Sisay Muleta Hababa Bekele Worku, Gizaw Tadele, Yihenew GebruMengistu and Prasad A Y.** 12, India, Ethiopia : Turkish Journal of Computer and Mathematics Education, 2021, Vol. 12. 4236-4243.

[15] **Zheng, Y.** (2015). Methodologies for Cross-Domain Data Fusion: An Overview. IEEE Transactions on Big Data, 1(1), 16–34. doi:10.1109/tbdata.2015.2465959

[16] **A. Menon**, "Big data @ facebook," in Proceedings of the 2012 workshop on Management of big data systems, ser. MBDS '12, 2012, pp. 31–32.

[17] *Next-Generation Big Data Analytics: State of the Art, Challenges, and Future Research Topics.* **Zhihan Lv, Houbing Song, Pablo Basanta-Val, Anthony Steed, Minho Jo.** 4, s.l. : EEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, 2017, Vol. 13.

[18] **M. Jo, T. Maksymyuk, R. L. Batista, T. F. Maciel, A. L. F. de Almeida and M. Klymash,** "A survey of converging solutions for heterogeneous mobile networks," in *IEEE Wireless Communications*, vol. 21, no. 6, pp. 54-62, December 2014, doi: 10.1109/MWC.2014.7000972.

[19] **Lin, C., Song, Z., Song, H., Zhou, Y., Wang, Y., & Wu, G.** (2016). *Differential Privacy Preserving in Big Data Analytics for Connected Health. Journal of Medical Systems, 40(4).* doi:10.1007/s10916-016-0446-0

[20] **Zhang, Y., Sun, L., Song, H., & Cao, X.** (2014). *Ubiquitous WSN for Healthcare: Recent Advances and Future Prospects. IEEE Internet of Things Journal, 1(4), 311–318.* doi:10.1109/jiot.2014.2329462

[21] aquaportail. *Données géographiques : définition, explications.* [En ligne] 23 Mars 2022. [Citation : 28 Juin 2022.] https://www.aquaportail.com/definition-6169-donnees-geographiques.html.

[22] talend. *Tout savoir sur la visualisation des data.* [En ligne] [Citation : 28 Juin 2022.] https://www.talend.com/fr/resources/visualisation-donnees/.

[23] **C. Snijders, U. Matzat, and U. D. Reips,** "Big data: Big gaps of knowledgein the field of internet science," Int. J. Internet Sci., vol. 7, no. 1, pp. 1–5,2012.

[24] **R. Kitchin**, "The real-time city? Big data and smart urbanism," Geo J.,vol. 79, no. 1, pp. 1–14, 2014.

[25] **universalis.** *Applications du big data.* [En ligne] [Citation : 14 Avril 2022.] https://www.universalis.fr/encyclopedie/big-data/5-applications-du-big-data/.

[26] **LCL.** *Big Data : définition, enjeux et applications.* [En ligne] 16 Mai 2019. [Citation : 14 Avril 2022.] https://www.lcl.fr/mag/tendances/big-data-definition-enjeux-et-applications.

[27] **IKIGAÏ.** *Big Data : applications dans différents secteurs d'activité.* [En ligne] 22 Avril 2021. [Citation : 14 Avril 2022.] https://ikigai-groupe.fr/big-data-applications-dans-differents-secteurs/.

[28] **HAMEL (M.-P.) et MARGUERIT (D.).** – Analyse des big data usages, quels défis ?Note d'analyse du Commissariat général à la stratégie et à la prospective, N 8, nov. 2013 (2013). consulté :17 Avril 2022

[29] La protection de la vie privée dans le Big Data. **BELKHAMSA Amel, YAHIAOUI Manel.** Bouira : Université AMO de Bouira, 2020.

[30] S. Tuffery, Cours de Data Mining , université de Rennes 1, 2014. consulté :19 Avril 2022

[31] **D. Gaultier**, Data Science Big Data – Etat de l'art , 2015. consulté :20 Avril 2022.

[32] Apache Flink – A Big Data Processing Platform. *Data Flair*. [En ligne] [Citation : 04 août 2022.] https://data-flair.training/blogs/apache-flink-big-data-unified-platform/.

[33] Mlle MOUZAIA Chahinas épouse REDJDAL Mlle ABBAS Célia. Le Contrôle d'Accès au Big Data. Cas d'Etude : Internet des Objets, Université A/Mira de Béjaia, Mémoire de Master. 2017.

[34] CLOAREC Erwann. Hadoop : Optimisation et Ordonnancement, Université François-Rabelais, Tours. 2014.

[35] **saagie.** Qu'est-ce que le Big Data et quelles sont ses Applications ? *DataOps.Rocks.* [En ligne] 14 12 2016. [Citation : 04 août 2022.] https://www.saagie.com/fr/blog/qu-est-ce-que-le-big-data-definition/.

[36] Brighen Assia. Conception de bases de données volumineuses sur le Cloud, Universite Béjaia, Mémoire de Master. 2012.

[37] . Sparrow, «Hadoop – Sensibilisation des racks et des racks,» 5 Juilliet 2022. [En ligne]. Available: https://fr.acervolima.com/hadoop-rack-et-rack-awareness/. [Accès le 3 août 2022].

[38] **Ekanayke, Shehan.** Hadoop - The Elephant in the Big Data Room. *medium.* [En ligne] 3 février 2018. [Citation : 4 août 2022.] https://medium.com/@y2kshehan/hadoop-the-elephant-in-the-big-data-room-e983892c1936.

[39] Margaret Rouse. YARN (Yet Another Resource Negotiator), TechTarget. 2015.

[40] The YARN architecture. *Packt.* [En ligne] [Citation : 4 août 2022.] https://subscription.packtpub.com/book/bigdataandbusinessintelligence/9781784393960/1/ch01lvl1sec11/the-yarn-architecture.

[41]https://datascientest.com/mapreduce#:~:text=Pour%20faire%20simple%2C%20Map%20sert,termes%20de%20Mappers%20et%20Reducers

[42] BIGDATA & HADOOP (I) - DEFINICIONES BÁSICAS Y ESQUEMA DE LA CONFIGURACIÓN TÍPICA. franciscojavierpulido. [En ligne] 18 juillet 2013. [Citation : 4 août 2022.] https://www.franciscojavierpulido.com/2013/07/bigdata-hadoop-i-definiciones-basicas-y.html.

[43] **Vieja, Angela de la.** Minderest. *10 avantages du big data pour votre commerce.* [En ligne] [Citation : 20 Avril 2022.] https://www.minderest.com/fr/blog/avantages-bigdata-pour-votre-commerce.

[44] COSTA DEL SOL MALAGA. *Les Avantages du Big Data pour les Entreprises : La Révolution des Données.* [En ligne] [Citation : 20 Avril 2022.] https://blog.visitacostadelsol.com/fr/big-data-entreprises.

[45] Research Techniques Made Simple: An Introduction to Use and Analysis of Big Data in Dermatology. **Mackenzie R. Wehner, Katherine A. Levandoski, Martin Kulldorff, Maryam Asgari.** 137, 2017, Vol. 8. e153-e158.

[46] **DOUNYA, KASSIMI.** A Big Data Security Approach in Cloud Computing . THESIS. Biskra : Mohamed Khider University of Biskra, 2020.

[47] pomcor. Cryptographic Authentication for Web Applications. [En ligne] [Citation : 16 Mai 2022.] https://pomcor.com/cryptographic-authentication/.

[48] Government Chief Information Officer. "e-Authntication". http://www.e-authentication.gov.hk/en/professional/skey.htm , 2014.

[49] Cryptography Based Authentication Methods. **Mohammad A. Alia, Abdelfatah Aref Tamimi, and Omaima N. A. AL-Allaf.** San Francisco, USA : s.n., 2014, Vol. I.

[50] Authentification par clé publique. www.attachmate.com. [En ligne] [Citation : 21 Mai 2022.]

[51] **TechTarget.** LEMAGIT. *Hachage (hashing).* [En ligne] août 2016. [Citation : 21 Mai 2022.] https://www.lemagit.fr/definition/Hachage.

[52] LEMAGIT. *Signature numérique (signature digitale).* [En ligne] Juillet 2016. [Citation : 21 Mai 2022.] https://www.lemagit.fr/definition/Signature-numerique-signature-electronique-ou-e-signature.

[53] **M. Alia A. Tamimi O. AL-Allaf**. "Integrated System For Monitoring And Recognizing Students During Class Session". The International Journal of Multimedia & Its Applications (IJMA), Vol.5, No.6. 2013.

[54] **I. Marqués, and M. Graña**. Face Recognition Algorithms. Proyectos Fin de Carrera, Universidad Carlos III de Madrid.2010.

[55] **P. Viola , and M. Jones**.Rapid object detection using a boosted cascade of simple features. Accepted Conference On Computer Vision And Pattern Recognition, 2001.

[56] **Yan-Wen Wu , and Xue-Yi Ai**. Face Detection in Color Images Using AdaBoost Algorithm Based on Skin Color Information. First International Workshop on Knowledge Discovery and Data Mining, 2008.

[57] **I. Cohen , N. Sebe , F. G. Cozman ,Thomas S. Huang** . Semi-Supervised Learning for Facial Expression Recognition. Proceedings of the 5th ACM SIGMM international workshop on Multimedia information retrieval. ACM New York, NY, USA Pages 17 – 22.

[58] «Kerberos (protocole),» 19 Mai 2022. [En ligne]. Available: https://fr.wikipedia.org/wiki/Kerberos_(protocole). [Accès le 9 août 2022].

[59] **E. El-Emam, M. Koutb, H. Kelash and O. Farag Allah,** "An optimized Kerberos authentication protocol," *2009 International Conference on Computer Engineering & Systems*, 2009, pp. 508-513, doi: 10.1109/ICCES.2009.5383213.

[60] «Secure Remote Password,» [En ligne]. Available: https://pythonhosted.org/srp/srp.html. [Accès le 5 Juin 2022].

[61] Big Data Authentication and Authorization using SRP Protocol. **Miti Jhaveri, Devang Jhaveri, Narendra Shekokar.** 1, s.l. : International Journal of Computer Applications (0975 – 8887), 2015, Vol. 130.

[62] Cluster Based Multi Layer User Authentication Data Center . **S. Ramasamy, R. K. Gnanamurthy.** India : s.n.

[63] **C. Wang and C. Feng,** "Security Analysis and Improvement for Kerberos Based on Dynamic Password and Diffie-Hellman Algorithm," *2013 Fourth International Conference on Emerging Intelligent Data and Web Technologies*, 2013, pp. 256-260, doi: 10.1109/EIDWT.2013.49.

[64] Research on Hadoop Identity Authentication Based on Improved Kerberos Protocol . **Daming Hu, Deyun Chen, Yuanxu Zhang and Shujun Pei.** 11, china : International Journal of Security and Its Applications, 2015, Vol. 9.

[65] Authentication Framework for Kerberos Enabled . **M. Hena, N. Jeyanthi.** 1, 2019, Vol. 9. 2249 - 8958.

[66] **department, Intel big data.** Intel. *Big Data Security Solution Based on Kerberos.* [En ligne] 8 Mars 2018. [Citation : 2 Juin 2022.] https://www.intel.com/content/www/us/en/developer/articles/technical/big-data-security-solution-based-on-kerberos.html.

[67] *A study on authentication challenges for Big Data in public cloud.* **Ashok Kumar.** 2017, Vol. 6. 2277-8160.

[68] **W. Liu, A. S. Uluagac and R. Beyah,** "MACA: A privacy-preserving multi-factor cloud authentication system utilizing big data," *2014 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2014, pp. 518-523, doi: 10.1109/INFCOMW.2014.6849285.

[69] **Anas Ibrahim.** Data Science Solution for User Authentication . *Electronic Thesis and Dissertation Repository.* s.l. : The University of Western Ontario, 2017.

[70] **R. Li, H. Asaeda and J. Wu,** "DCAuth: Data-Centric Authentication for Secure In-Network Big-Data Retrieval," in *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 1, pp. 15-27, 1 Jan.-March 2020, doi: 10.1109/TNSE.2018.2872049.

[71] D. N. T. Risha Tabassum, «Hadoop Identity Authentication using PublicPrivate Key Concept,» vol. 45, n° %19, 2017.

[72] *Developing Blockchain Authentication for Hadoop using.* **Dr. Pramod Patil, Dr. Jyoti Rao, Mithun Kankal.** 7, s.l. : International Journal of Research in Advent Technology, 2019, Vol. 7. E-ISSN: 2321-9637.

[73] **Khalil, I., Dou, Z., & Khreishah, A.** (2015). *TPM-Based Authentication Mechanism for Apache Hadoop. International Conference on Security and Privacy in Communication Networks, 105–122.* doi:10.1007/978-3-319-23829-6_8

[74] Hadoop. wikipedia. [En ligne] [Citation : 15 août 2022.] https://fr.wikipedia.org/wiki/Hadoop.

[75] Présentation de l'authentification Kerberos. Digital Guide. [En ligne] 04 nouvembre 2021. https://www.ionos.fr/digitalguide/serveur/securite/kerberos/.

[76]Rahul, P. K., & GireeshKumar, T. (2014). *A Novel Authentication Framework for Hadoop. Artificial Intelligence and Evolutionary Algorithms in Engineering Systems, 333–340.* doi:10.1007/978-81-322-2126-5_37

[77] Jeong, Y.-S., Shin, S.-S., & Han, K.-H. (2015). *High-dimentional data authentication protocol based on hash chain for Hadoop systems. Cluster Computing, 19(1), 475–484.* doi:10.1007/s10586-015-0508-y

[78] Dou, Z., Khalil, I., Khreishah, A., & Al-Fuqaha, A. (2018). Robust Insider Attacks Countermeasure for Hadoop: Design and Implementation. IEEE Systems Journal, 12(2), 1874–1885. doi:10.1109/jsyst.2017.2669908

[79] Jeong, Y.-S., & Kim, Y.-T. (2015). *A token-based authentication security scheme for Hadoop distributed file system using elliptic curve cryptography. Journal of Computer Virology and Hacking Techniques, 11(3), 137–142.* doi:10.1007/s11416-014-0236-5

[80] Chattaraj, D., Sarma, M., Das, A. K., Kumar, N., Rodrigues, J. J. P. C., & Park, Y. (2018). *HEAP: An Efficient and Fault-tolerant Authentication and Key Exchange Protocol for Hadoop-assisted Big Data Platform. IEEE Access, 1–1.* doi:10.1109/access.2018.2883105

[81] Guiyuan, W., & Hongyun, N. (2018). *The Improvement of HDFS Authentication Model Based on Token Push Mechanism. Proceedings of the 2018 International Conference on Big Data and Computing - ICBDC ’18.* doi:10.1145/3220199.3220222

[82] 19 exemples d’utilisation d’OpenSSL. malekal.com. [En ligne] 30 Juin 2021. [Citation : 15 août 2022.] https://www.malekal.com/19-exemples-utilisation-openssl/.

[83] Yao Yao, Wang Xingwei, Sun Xiaoguang «A Cross Heterogeneous Domain Authentication Model Based on PKI» 2011.

[84] Ishwarappa, J.Anuradha. «A Brief Introduction on Big Data 5Vs Characteristics and Hadoop Technology,» vol. 48, pp. 319-324, 2015.

[85] N. Somu, A.Gangaa and V. S. Shankar Sriram, «Authentication Service in Hadoop using One Time Pad,» vol. 7, pp. 56-62, 2014.

[86]

Webographie :

[W1]http://dspace.univ-bouira.dz:8080/jspui/bitstream/123456789/10907/1/memoire.pdf [Citation : 25 Mars 2022.]

[W2]http://bib.univ-oeb.dz:8080/jspui/bitstream/123456789/6933/1/m%c3%a9moire%20final.pdf [Citation : 27 Mars 2022]

[W3]https://www.analyticsinsight.net/all-about-the-basics-of-big-data-history-types-and-applications/ [Citation : 1 Avril 2022.]

[W4] https://www.techtarget.com/whatis/feature/A-history-and-timeline-of-big-data [Citation : 27 Juin 2022.]

[W5]https://cours.etsmtl.ca/log660/public_docs/acetates/BigData_Technologies_PL.pdf

[Citation : 27 Juin2022.]

[W6]https://www.mckinsey.com/~/media/mckinsey/business%20functions/mckinsey%20digital/our%20insights/big%20data%20the%20next%20frontier%20for%20innovation/mgi_big_data_exec_summary.pdf [Citation : 27 Juin2022.]

[W7]https://documents.pub/document/idc-report-extracting-value-from-chaos.html [Citation : 27 Juin2022.]

[W8]https://www.researchgate.net/publication/342946040_Big_Data_Analytics_for_IoT [Citation : 27 Juin2022.]

[W9]https://www.licence-mci.fr/espace-etudiants/les-memoires/les-big-data-une-mine-d-information/ [Citation : 27 Juin2022.]

[W10] https://docs.microsoft.com/en-us/azure/architecture/data-guide/big-data/ [Citation : 27 Juin 2022.]

[W11]https://www.researchgate.net/publication/353693055_Big_Data_Security_and_Privacy_A_Taxonomy_with_Some_HPC_and_Blockchain_Perspectives [Citation : 28 Juin2022.]

[W12]https://www.irjet.net/archives/V4/i9/IRJET-V4I957.pdf [Citation : 28 Juin2022.]

[W13]https://www.researchgate.net/profile/NawsherKhan/publication/325480960_The_10_Vs_Issues_and_Challenges_of_Big_Data/links/5b8ba7bb299bf1d5a738092f/The-10-Vs-Issues-and-Challenges-of-Big-Data.pdf [Citation : 28 Juin2022.]

[W14]https://www.turcomat.org/index.php/turkbilmat/article/download/8316/6490/14910 [Citation : 28 Juin2022.]

[W15] https://ieeexplore.ieee.org/document/7230259 [Citation : 28 Juin2022.]

[W16] https://dl.acm.org/doi/10.1145/2378356.2378364 [Citation : 28 Juin2022.]

[W17]https://www.researchgate.net/publication/314201124_Next-Generation_Big_Data_Analytics_State_of_the_Art_Challenges_and_Future_Research_Topis [Citation : 20 Juin2022.]

[W18] https://ieeexplore.ieee.org/abstract/document/7000972 [Citation : 20 Juin2022.]

[W19]https://www.researchgate.net/publication/294289065_Differential_Privacy_Preserving_in_Big_Data_Analytics_for_Connected_Health [Citation : 20 Juin2022.]

[W20] https://ieeexplore.ieee.org/document/6827212

[W21] https://www.aquaportail.com/definition-6169-donnees-geographiques.html [Citation : 20 Juin 2022.]

[W22] https://www.talend.com/fr/resources/visualisation-donnees/ [Citation : 20 Juin 2022.]

[W23] https://www.ijis.net/ijis7_1/ijis7_1_editorial.pdf [Citation : 22 Juin2022.]

[W24]https://www.researchgate.net/publication/262008155_The_real-time_city_Big_data_and_smart_urbanism [Citation : 22 Juin2022.]

[W25]https://www.universalis.fr/encyclopedie/big-data/5-applications-du-big-data/ [Citation : 14 Avril 2022.]

[W26]https://www.lcl.fr/mag/tendances/big-data-definition-enjeux-et-applications [Citation : 14 Avril 2022.]

[W27]https://ikigai-groupe.fr/big-data-applications-dans-differents-secteurs/ [Citation : 14 Avril 2022.]

[W28]https://www.strategie.gouv.fr/espace-presse/analyse-big-data-usages-defis [Citation : 17 Avril 2022.]

[W29]http://dspace.univ-bouira.dz:8080/jspui/bitstream/123456789/10839/1/memoire%20%281%29.pdf [Citation : 17 Avril 2022.]

[W30]https://docplayer.fr/2854462-Cours-de-data-mining.html [Citation : 19 Avril 2022.]

[W31]http://simulation.armees.free.fr/adis//reunionsadis/20150401_reunion14//15_20150401_B&D_etat-de-l-art-Big-Data-et-Analyse-des-donnees.pdf [Citation : 20 Avril 2022.]

[W32] https://data-flair.training/blogs/apache-flink-big-data-unified-platform/ [Citation : 04 août 2022.]

[W33]https://www.theses-algerie.com/1931801080693699/memoire-de-master/universite-abderrahmane-mira---bejaia/le-contr%C3%B4le-d-acc%C3%A8s-au-big-data-cas-d-etude [Citation : 04 août 2022.]

[W34] http://www.applis.univ-tours.fr/scd/EPU_DI/2014_PFEDI_CLOAREC.Erwann.pdf [Citation : 04 août 2022.]

[W35] https://www.saagie.com/fr/blog/les-technologies-et-applications-du-big-data/ [Citation : 04 août 2022.]

[W36]http://www.univ-bejaia.dz/xmlui/bitstream/handle/123456789/9565/Conception%20de%20bases%20de%20donn%C3%A9es%20volumineuses%20sur%20le%20cloud.pdf?sequence=1&isAllowed=y [Accès le 3 août 2022].

[W37] https://fr.acervolima.com/hadoop-rack-et-rack-awareness/ [Accès le 3 août 2022].

[W38]https://medium.com/@y2kshehan/hadoop-the-elephant-in-the-big-data-room-e983892c1936 [Citation : 4 août 2022.]

[W39]https://www.techtarget.com/whatis/fr/definition/YARN-Yet-Another-Resource-Negotiator [Citation : 4 août 2022.]

[W40]https://subscription.packtpub.com/book/bigdataandbusinessintelligence/9781784393960/1/ch01lvl1sec11/the-yarn-architecture [Citation : 4 août 2022.]

Webographie

[W41]https://datascientest.com/mapreduce#:~:text=Pour%20faire%20simple%2C%20Map%20sert,termes%20de%20Mappers%20et%20Reducers [Citation : 4 août 2022.]

[W42]https://www.franciscojavierpulido.com/2013/07/bigdata-hadoop-i-definiciones-basicas-y.html [Citation : 4 août 2022.]

[W43] https://www.minderest.com/fr/blog/avantages-bigdata-pour-votre-commerce [Citation : 20 Avril 2022.]

[W44] https://blog.visitacostadelsol.com/fr/big-data-entreprises [Citation : 20 Avril 2022.]

[W45]https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5600502/ [Citation : 20 Avril 2022.]

[W46]http://thesis.univ-biskra.dz/5132/1/Thesis-Kassimi-D.pdf [Citation : 20 Avril 2022.]

[W47]https://pomcor.com/cryptographic-authentication/[Citation : 16 Mai 2022.]

[W48]http://www.e-authentication.gov.hk/en/professional/skey.htm [Citation : 16 Mai 2022.]

[W49]http://www.iaeng.org/publication/WCECS2014/WCECS2014_pp199-204.pdf [Citation : 16 Mai 2022.]

[W50]https://www.attachmate.com/fr-fr/documentation/reflection-desktop-v16/rdesktop-guide/data/rsitclient_client_public_key_auth_ch.htm [Citation : 21 Mai 2022.]

[W51]https://www.lemagit.fr/definition/Hachage [Citation : 21 Mai 2022.]

[W52]https://www.lemagit.fr/definition/Signature-numerique-signature-electronique-ou-e-signature [Citation : 21 Mai 2022.]

[W53]https://aircconline.com/ijma/V5N6/5613ijma04.pdf [Citation : 25 Mai 2022.]

[W54]https://www.ehu.eus/ccwintco/uploads/d/d2/PFC-IonMarqu%C3%A9s.pdf [Citation : 27 Mai 2022.]

[W55]https://www.cs.cmu.edu/~efros/courses/LBMV07/Papers/viola-cvpr-01.pdf [Citation : le 9 août 2022]

[W56]https://sci-hub.se/10.1109/WKDD.2008.148 [Citation : 9 août 2022]

[W57]http://disi.unitn.it/~sebe/publications/MIR03.pdf [Citation : 9 août 2022]

[W58] https://fr.wikipedia.org/wiki/Kerberos_(protocole) [Citation : 9 août 2022]

[W59]https://sci-hub.se/10.1109/ICCES.2009.5383213 [Citation : 3 Juin 2022]

[W60] https://pythonhosted.org/srp/srp.html [Citation : 5 Juin 2022]

[W61]https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.742.665&rep=rep1&type=pdf [Citation : 5 Juin 2022]

[W62]https://jit.ndhu.edu.tw/article/viewFile/2232/2245 [Citation : 5 Juin 2022]

[W63]https://sci-hub.se/10.1109/EIDWT.2013.49 [Citation : 4 Juin 2022]

[W64]http://article.nadiapub.com/IJSIA/vol9_no11/39.pdf [Citation : 3 Juin 2022]

[W65]https://www.ijeat.org/wp-content/uploads/papers/v9i1/A9638109119.pdf [Citation : 3 Juin 2022]

[W66]https://www.intel.com/content/www/us/en/developer/articles/technical/big-data-security-solution-based-on-kerberos.html. [Citation : 2 Juin 2022.]

[W67]https://www.researchgate.net/profile/Ashok-Kumar-256/publication/325390560_A_STUDY_ON_AUTHENTICATION_CHALLENGES_FOR_BIG_DATA_IN_PUBLIC_CLOUD/links/5b0a7bf2aca2725783e8c5f6/A-STUDY-ON-AUTHENTICATION-CHALLENGES-FOR-BIG-DATA-IN-PUBLIC-CLOUD.pdf [Citation : 3 Juin 2022]

[W68] https://ieeexplore.ieee.org/document/6849285 [Citation : 6 Juillet 2022]

[W69] https://ir.lib.uwo.ca/cgi/viewcontent.cgi?article=6408&context=etd [Citation : 7 Juillet 2022]

[W70] https://ieeexplore.ieee.org/document/8471195 [Citation : 9 Juillet 2022]

[W71]https://www.researchgate.net/publication/317546759_Hadoop_Identity_Authentication_using_Public_Private_Key_Concept [Citation : 10 Juillet 2022]

[W72] https://ijrat.org/downloads/Vol-7/july-2019/77201924.pdf [Citation : 11 Juillet 2022]

[W73] https://web.njit.edu/~abdallah/lnicst.pdf [Citation : 13 Juillet 2022]

[W74] https://fr.wikipedia.org/wiki/Hadoop [Citation : 16 Juillet 2022]

[W75] https://www.ionos.fr/digitalguide/serveur/securite/kerberos/ [Citation : 24 Juillet 2022]

[W76] https://sci-hub.se/10.1007/978-81-322-2126-5_37 [Citation : 15 août 2022.]

Webographie

[W77] https://sci-hub.se/10.1007/s10586-015-0508-y [Citation : 1 août 2022.]

[W78] https://sci-hub.se/10.1109/jsyst.2017.2669908 [Citation : 17 août 2022.]

[W79] https://sci-hub.se/10.1007/s11416-014-0236-5 [Citation : 20 août 2022.]

[W80] https://sci-hub.se/10.1109/access.2018.2883105 [Citation : 21 août 2022.]

[W81] https://sci-hub.se/10.1145/3220199.3220222 [Citation : 21 août 2022.]

[W82] https://www.malekal.com/19-exemples-utilisation-openssl/ [Citation : 27 août 2022.]

[W83] https://ieeexplore.ieee.org/document/6128526 [Citation : 27 août 2022.]

[W84] https://www.sciencedirect.com/science/article/pii/S1877050915006973#! [Citation : 27 août 2022.]

[W85]https://sciresol.s3.us-east-2.amazonaws.com/IJST/Articles/2014/Issue-Supplementary-4/Article11.pdf [Citation : 30 août 2022.]

[W86]

Annexes A

Le tableau ci-dessous représente une étude comparative entre les déférentes approches d'authentifications tel que A Novel Authentication framework for Hadoop, High-dimensional data authentication protocol based on hash chain for Hadoop systems, Robust Insider Attacks Countermeasure for Hadoop : Design and Implementation, A token-based authentication security scheme for Hadoop distributed file system using elliptic curve cryptography, HEAP : An Efficient and Fault-tolerant Authentication and Key Exchange Protocol for Hadoop-assisted Big Data Platform, et The improvement of HDFS Authentication Model Based on Token Push Mechanism en spécifiant les déférentes methodologies utilisées et l'idée introduite par chaque approche amélioré, ainsi que les désavantages de chaque amélioration.

Etude comparative [65]**:**

Titre	Méthodologie	Idée introduite	Désavantage
A Novel Authentication framework for Hadoop[76].	-Cryptographie à clé publique. -Cryptographie à clé privée. -Hachage. -Génération de nombres aléatoires.	-Deux serveurs : serveur utilisateur (pour l'authentification de l'utilisateur) et serveur de données (pour une communication sécurisée). -Un nombre aléatoire généré par le système [SHA-2(nom d'utilisateur et ce nombre aléatoire)] est utilisé pour toutes les authentifications client. Les ID de bloc des données sécurisées sont transmises sous	-La cryptographie à clé publique ralentit le système. -Calculateur frais généraux. -Problème de certification de la clé publique, non mentionné.

		forme cryptée (clé symétrique) .	
High-dimensional data authentication protocol based on hash chain for Hadoop systems[77].	-Athentification basé sur la chaîne de hachage.	-Le NameNode génère une clé secrète à partager avec chaque client. Un jeton de délégation est généré à partir d'une valeur de départ à l'aide d'un mécanisme de chaîne de hachage et est partagé avec les clients. Les clients qui soumettent les jetons de délégation corrects reçoivent des jetons d'accès aux blocs pour accéder aux blocs de données dans le DataNode.	-Soit une longue chaîne de hachage doit être maintenue => mémoire. -Ou la chaîne de hachage doit être reconstruite très souvent -Charge complète sur le NameNode.
Robust Insider Attacks Countermeasure for Hadoop : Design and Implementation[78].	Plateforme de confiance Modulaire(TPM) . Combine des solutions de sécurité matérielles et logicielles. . Chiffrement RSA.	-Les clés de session sont chiffrées à l'aide des clés d'authentification dans le TPM et des valeurs dans le registres de configuration de la plate-forme(PCR). Les entités sont	-L'attaque contre le TPM non traitée. -Les Hadoop NameNode sont supposés être partiellement sécurisé -La clé d'endossement n'est pas mise à la

		scellées et liées à des configurations de plate-forme de confiance spécifiques. Les services d'attestation de plate-forme à distance sont également activés via un mécanisme de vérification périodique des empreintes digitales (mécanisme Heartbeat).	disposition des utilisateurs finaux. Seul le fabricant la sait.
A token-based authentication security scheme for Hadoop distributed file system using elliptic curve cryptography[79].	-Courbe elliptique Cryptographie(ECC) -bloque de jeton d'accès. -chaîne de hachage de clés	-Deux jetons sont utilisés :- jeton de délégation (DT) et jeton d'accès au bloc (BAT) générés à l'aide d'ECC. -Les clients authentifiés à l'aide de DT reçoivent BAT pour accéder aux blocs de données sécurisés dans le DataNode.	-Algorithme complexe. -Risque élevé d'erreur de mise en œuvre et peut donc compromettre la sécurité.
HEAP : An Efficient and Fault-tolerant Authentication and Key Exchange Protocol for Hadoop-	-Signature numérique -Norme de chiffrement Avancé(AES).	-HEAP-KDC a introduit trois serveurs : deux publiques(Client Management Server	-Long. -Calculs complexes. -Exigences supplémentaires en matière de gestion de

assisted Big Data Platform[80].	-Courbe elliptique cryptographie(ECC).	& NameNode Management Server). Les entités enregistrent d'abord elles-mêmes l'ES et obtiennent des identifiants factices à usage unique. Ensuite, ils s'inscrivent auprès des serveurs publique correspondants. Pour l'authentification mutuelle, des signatures numériques sont utilisées.	serveur et augmentation de la zone d'attaques.
The improvement of HDFS Authentication Model Based on Token Push Mechanism[81].	-Basé sur l'agent poussée de jeton mécanisme.	-Les agents génèrent des jetons de nœuds croisés après la première connexion réussie des utilisateurs via KDC. Ceux-ci sont poussés vers tous les DataNodes impliqués. DataNode utilise ces jetons de nœuds croisés pour	-La sécurité de l'agent doit être assurée. -problème de Synchronisation de l'heure dans un environnement distribués.

		s’authentifier l’utilisateur. -Paramètre T-nonce généré par le traitement de hachage de l’horodatage et de l’IP pour empêcher les attaques par relecture.	

Tableau 4: Etude comparative déjà cité

Annexes B

Mise en place :

Les outils de développement :

Hadoop : est un framework libre et open source écrit en Java destiné à faciliter la création d'applications distribuées (au niveau du stockage des données et de leur traitement) et échelonnables (scalables) permettant aux applications de travailler avec des milliers de nœuds et des pétaoctets de données. Ainsi chaque nœud est constitué de machines standard regroupées en grappe. Tous les modules de Hadoop sont conçus selon l'idée que les pannes matérielles sont fréquentes et qu'en conséquence elles doivent être gérées automatiquement par le framework[74].

Kerberos : est un service d'authentification, très utilisé dans les réseaux informatiques ouverts ou non sécurisés. Le protocole de sécurité authentifie les demandes de service entre deux hôtes de confiance, ou plus, via un réseau non approuvé comme Internet. Pour l'authentification d'applications client-serveur et la vérification de l'identité de l'utilisateur, il utilise le chiffrement cryptographique et un tiers de confiance[75].

Openssl : est une librairie libre qui permet générer un fichier CSR de demande signature de certificat, créer des clés privés, des certificats d'autorités auto-signés ou encore récupérer un certificat SSL distant et vérifier la connexion TLS [82].

Les étapes de simulation :

Installation Hadoop :

Sur la première machine nous avons installé Hadoop en mode master (maître) et sur les deux autres machines nous avons installé Hadoop en mode slave (esclave).

Les étapes d'installation de Hadoop sont comme suivant:

- Avant de commencer l'installation de Hadoop nous devons tout d'abord installer **java 8** et **ssh**, par la suite nous configurons le fichier **/etc/ssh/sshd_config**. Pour installer **java 8** nous exécutons la commande suivante :

```
mounir@krb5:~$ sudo apt install openjdk-8-jdk-headless
```

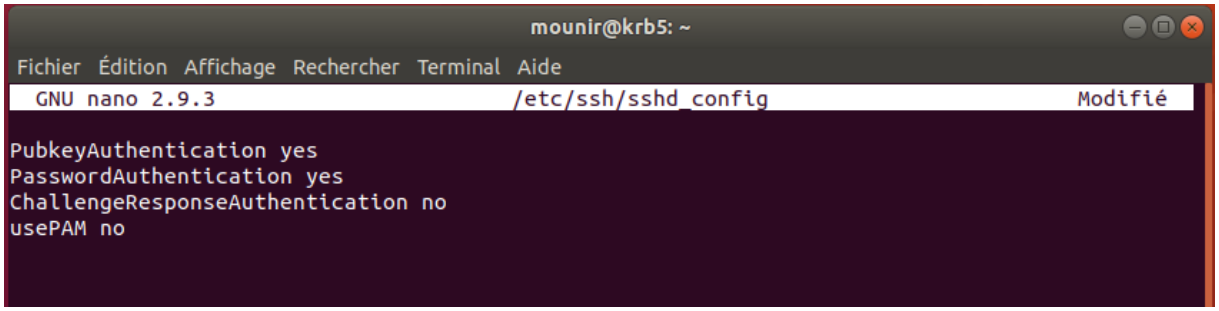
```
mounir@krb5:~$ java -version
openjdk version "1.8.0_342"
OpenJDK Runtime Environment (build 1.8.0_342-8u342-b07-0ubuntu1~18.04-b07)
OpenJDK 64-Bit Server VM (build 25.342-b07, mixed mode)
```

Installation de **ssh** qui permet d'établir la communication entre les différents nœuds :

```
mounir@krb5:~$ sudo apt install ssh
```

configuration de fichier **sshd_config** :

```
mounir@krb5:~$ sudo nano /etc/ssh/sshd_config
```



-Nous passons maintenant au téléchargement de **Hadoop-3.2.4.tar.gz** en saisissent la commande suivante :

```
mounir@krb5:~$ wget https://dlcdn.apache.org/hadoop/common/hadoop-3.2.4/hadoop-3.2.4.tar.gz
```

Après le téléchargement nous coupions l'archive de **Hadoop** dans les 2 nœuds esclaves :

```
mounir@krb5:~$ scp hadoop-3.2.4.tar.gz slave1@client1.abc.io:/home/slave1/
```

```
mounir@krb5:~$ scp hadoop-3.2.4.tar.gz slave2@client2.abc.io:/home/slave2/
```

Ensuite, nous décompressons l'archive **Hadoop-3.2.4** et nous le déplaçons vers le répertoire **/etc/hadoop/** avec la commande **« mv »** :

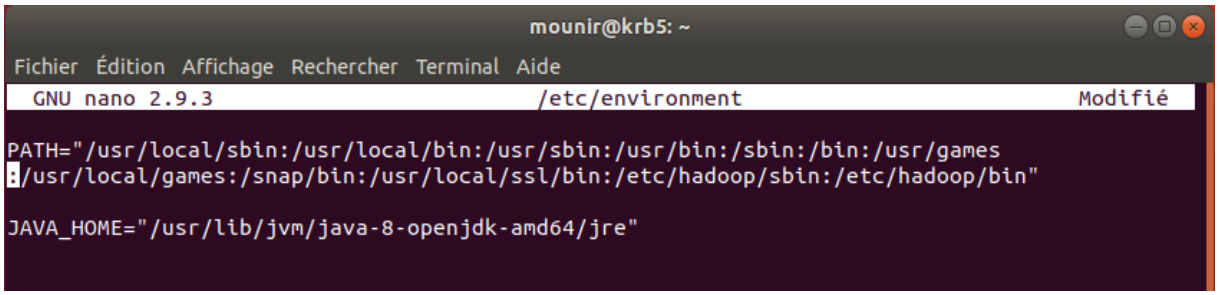
```
mounir@krb5:~$ tar -xzvf hadoop-3.2.4.tar.gz
```

```
mounir@krb5:~$ sudo mv hadoop-3.2.4 /etc/hadoop/
```

L'étape ci-dessus sera faite sur toutes les machines de cluster.

En outre, nous devons mettre à jours nos variables d'environnement pour inclure les répertoires binaires **JAVA_HOME** et **Hadoop** avec la commande ci-dessous:

```
mounir@krb5:~$ sudo nano /etc/environment
```



Nous ajoutons ensuite un nouveaux utilisateur **hdoop** avec la commande **« adduser »** et nous lui donnons les autorisations appropriées sur le répertoire **/etc/hadoop/** . En effet, la commande **« adduser »** permet d'ajouter un utilisateur à un système.

```
mounir@krb5:~$ sudo adduser hdoop
```

Avec la commande **« usermod »** nous modifions les paramètres du compte utilisateur **hdoop** ajouté [8] et nous l'ajoutons au groupe **hadoop** sans supprimer **hdoop** de ses groupes d'origine par l'option **« -aG ».**

```
mounir@krb5:~$ sudo usermod -aG hadoop hdoop
```

Nous changeons le propriétaire et le groupe propriétaire du répertoire **/etc/hadoop/** et tout ce qu'il contient en utilisant la commande **« chown »** avec l'option **« –R »**.

```
mounir@krb5:~$ sudo chown hdoop:hadoop -R /etc/hadoop/
```

Ensuite, nous procédons par modifier les permissions d'accès au répertoire **/etc/hadoop** par la commande **« chmod »** en accordent les droits de permission **« rwx »** (Read, Write, Execution) au groupe propriétaire du fichier.

```
mounir@krb5:~$ sudo chmod g+rwx -R /etc/hadoop/
```

De plus, nous ajoutons l'utilisateur **hdoop** au groupe **sudo** par la commande **« adduser »** pour qu'il puisse exécuter les commandes autant que super-utilisateur (comme le root).

```
mounir@krb5:~$ sudo adduser hdoop sudo
```

Après nous configurons une connexion **ssh** basée sur une clé dans chaque nœud pour qu'ils puissent ce communiquer entre eux sans aucune demande de mot de passe.

```
mounir@krb5:~$ su - hdoop
Mot de passe :
hdoop@krb5:~$
```

```
hdoop@krb5:~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/hdoop/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/hdoop/.ssh/id_rsa.
Your public key has been saved in /home/hdoop/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:ZN5sgmCw+bY5ihKqsDA5sHigVdpmqvscUULaaXDPa7M hdoop@krb5.abc.io
The key's randomart image is:
+---[RSA 2048]----+
|. +              |
| * B             |
|. B B   o        |
| . O o = o       |
|o + X . S +      |
|== B =   o       |
|@.+ E            |
|=O o .           |
|Oo+              |
+----[SHA256]-----+
```

Nous utilisons la commande **«cat»** pour concaténer le fichier id_rsa.pub avec le fichier authorized_keys, ici l'utilisation de symbole >> permet d'ajouter le contenu du fichier id_rsa.pub vers la fin du fichier authorized_keys.

```
hdoop@krb5:~$ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
```

Nous donnons les autorisations de lecture et d'écriture au propriétaire du fichier authorized_keys.

```
hdoop@krb5:~$ chmod 0600 ~/.ssh/authorized_keys
```

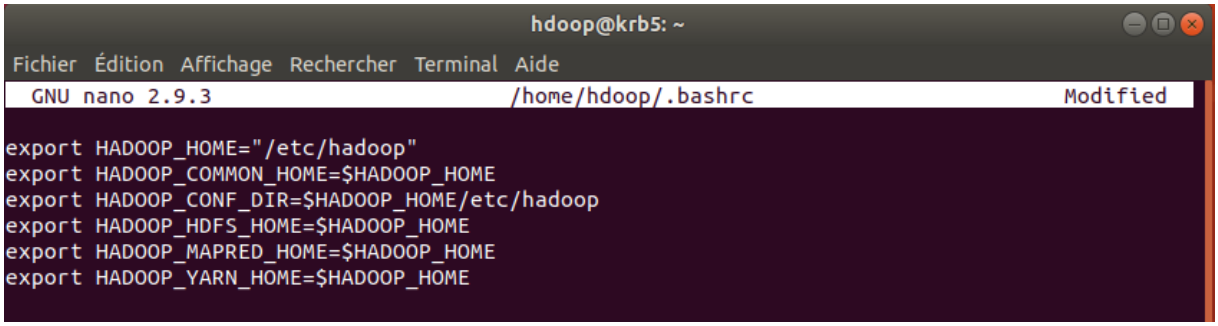
Nous coupions la clé générer vers tous les nœuds esclaves avec la commande **«ssh-copy-id»**.

```
hdoop@krb5:~$ ssh-copy-id -i ~/.ssh/id_rsa.pub hdoop@client1.abc.io
```

```
hdoop@krb5:~$ ssh-copy-id -i ~/.ssh/id_rsa.pub hdoop@client2.abc.io
```

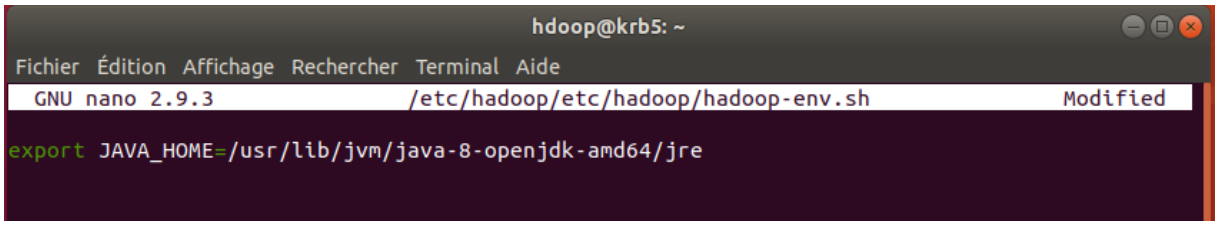
Nous éditons le fichier **~/.bashrc** et nous ajoutons les commandes suivante afin de définir les variables d'environnement **Hadoop** pour l'utilisateur **hdoop**.

```
hdoop@krb5:~$ sudo nano ~/.bashrc
```



Nous éditons aussi le fichier **/etc/hadoop/etc/hadoop/hadoop-env.sh** pour inclure le répertoire binaire **JAVA_HOME**.

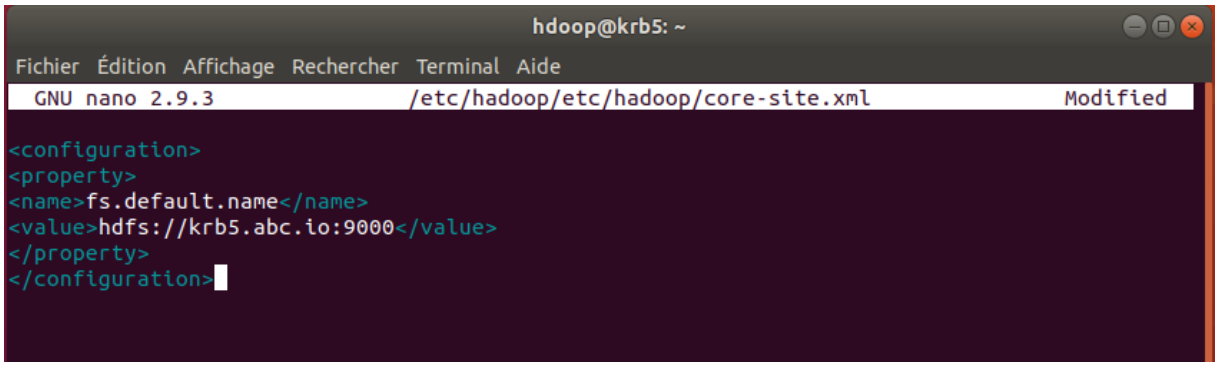
```
hdoop@krb5:~$ sudo nano /etc/hadoop/etc/hadoop/hadoop-env.sh
```



Configuration de Hadoop :

Nous ouvrons le fichier **/etc/hadoop/etc/hadoop/core-site.xml** et nous saisissons la propriété suivante :

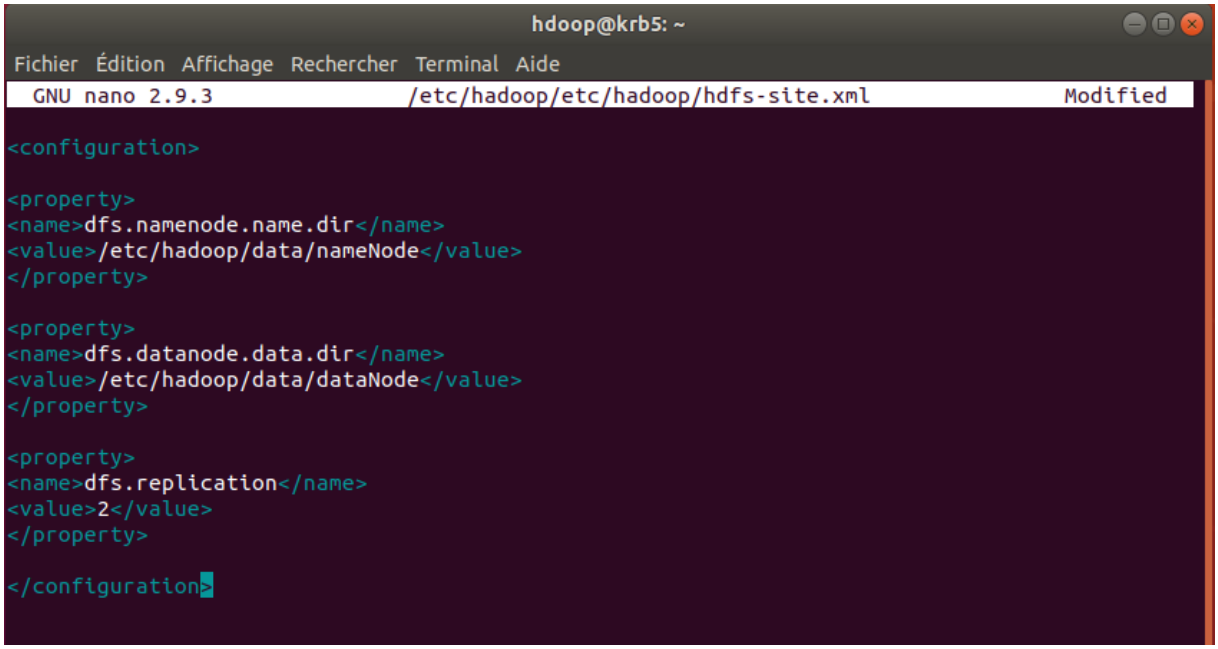
```
hdoop@krb5:~$ sudo nano /etc/hadoop/etc/hadoop/core-site.xml
```



La propriété **fs.default.name** permet d'indiquer le nœud et le port du système de fichier **HDFS**. Nous utilisons la machine **krb5.abc.io** comme le nœud et **9000** comme le port d'accès à ce nœud.

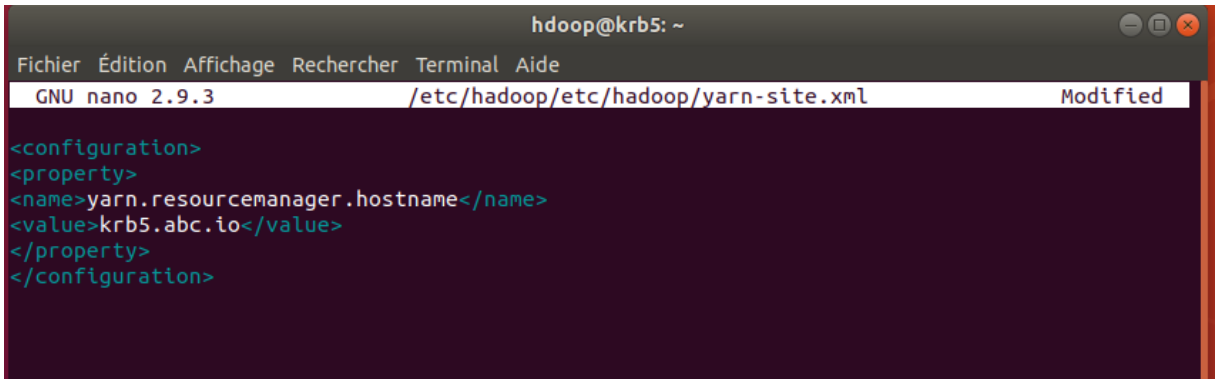
Le fichier **hdfs-site.xml** permet de configurer où le NameNode va stocker l'historique des transactions et où les DataNode vont stocker leurs blocks. C'est également ici où le coefficient de réplication est configuré. Dans notre cas le coefficient de réplication est égale à 2.

```
hdoop@krb5:~$ sudo nano /etc/hadoop/etc/hadoop/hdfs-site.xml
```



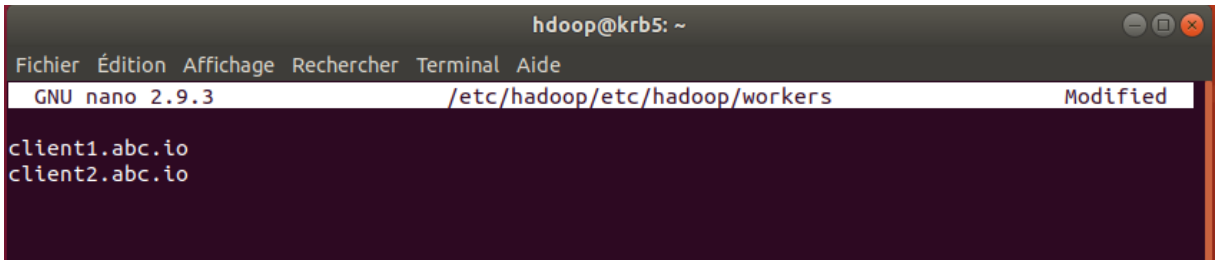
Le fichier **yarn-site.xml** contient des informations de configuration qui remplacent les valeurs par défaut des paramètres **YARN**. Les remplacements des valeurs par défaut pour les propriétés de configuration principales sont stockés dans le fichier Paramètres **YARN** par défaut .

```
hdoop@krb5:~$ sudo nano /etc/hadoop/etc/hadoop/yarn-site.xml
```



Nous éditons par la suite le fichier **/etc/hadoop/etc/hadoop/workers** sur le nœud maître et nous ajoutons les noms d'hôtes des autres nœuds esclaves **(client1.abc.io ; client2.abc.io)**.

```
hdoop@krb5:~$ sudo nano /etc/hadoop/etc/hadoop/workers
```



Une fois que la configuration de **hadoop** sur le nœud maître est terminé, nous allons la copier sur les autres nœuds esclaves avec la commande **« scp »** :

```
hdoop@krb5:~$ scp /etc/hadoop/etc/hadoop/* client1.abc.io:/etc/hadoop/etc/hadoop/
Enter passphrase for key '/home/hdoop/.ssh/id_rsa':
```

```
hdoop@krb5:~$ scp /etc/hadoop/etc/hadoop/* client2.abc.io:/etc/hadoop/etc/hadoop/
Enter passphrase for key '/home/hdoop/.ssh/id_rsa':
```

Maintenant nous allons formater le système de fichiers **HDFS** comme suit :

```
hdoop@krb5:~$ hdfs namenode -format
2022-09-13 14:34:09,136 INFO namenode.NameNode: STARTUP_MSG:
/************************************************************
STARTUP_MSG: Starting NameNode
STARTUP_MSG:   host = krb5.abc.io/192.168.217.142
STARTUP_MSG:   args = [-format]
STARTUP_MSG:   version = 3.2.4
```

Le résultat de formatage est montré ci-dessous:

```
2022-09-13 14:34:20,735 INFO common.Storage: Storage directory /etc/hadoop/data/nameNode has been successfully formatted.
2022-09-13 14:34:20,865 INFO namenode.FSImageFormatProtobuf: Saving image file /etc/hadoop/data/nameNode/current/fsimage.ckpt_0000000000000000000
 using no compression
2022-09-13 14:34:21,737 INFO namenode.FSImageFormatProtobuf: Image file /etc/hadoop/data/nameNode/current/fsimage.ckpt_0000000000000000000 of siz
e 400 bytes saved in 0 seconds .
2022-09-13 14:34:21,765 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 0
2022-09-13 14:34:21,857 INFO namenode.FSNamesystem: Stopping services started for active state
2022-09-13 14:34:21,858 INFO namenode.FSNamesystem: Stopping services started for standby state
2022-09-13 14:34:21,874 INFO namenode.FSImage: FSImageSaver clean checkpoint: txid=0 when meet shutdown.
2022-09-13 14:34:21,875 INFO namenode.NameNode: SHUTDOWN_MSG:
/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at krb5.abc.io/192.168.217.142
************************************************************/
```

- **Démarrage de Hadoop sur le nœud maître :**

Nous utilisons la commande **« start-dfs.sh »** pour lancer le **NameNode**, le **DataNode** et le **Secondary NameNode**.

```
hdoop@krb5:~$ start-dfs.sh
Starting namenodes on [krb5.abc.io]
Starting datanodes
Starting secondary namenodes [krb5.abc.io]
```

Avec la commande **« start-yarn.sh »** Nous lançons le **ResourceManager** et le **NodeManager** du service **YARN**.

```
hdoop@krb5:~$ start-yarn.sh
Starting resourcemanager
Starting nodemanagers
```

Pour vérifier si les processus **Hadoop** tels que **NameNode**, **DataNode**, **ResourceManager**, **NodeManager** et **secondary NameNode** sont actifs et en cours d'exécution, nous utilisons la commande **« jps »** (Java Virtual Machine Process Status Tool) :

```
hdoop@krb5:~$ jps
12854 SecondaryNameNode
12634 NameNode
13407 Jps
13087 ResourceManager
```

Nous utilisons aussi la même commande **« jps »** sur les autres nœuds esclaves. Nous remarquons que les services des nœuds esclaves sont lancés automatiquement à distance par le nœud maître.

```
hdoop@client1:~$ jps
12052 DataNode
12260 Jps
12171 NodeManager
```

```
hdoop@client2:~$ jps
12355 DataNode
12566 Jps
12477 NodeManager
```

- **Les interfaces utilisateurs d'administration Hadoop :**

La distribution **Hadoop** par Apache fournit des interfaces utilisateurs pour l'administration. Ceux-ci sont accessibles via des applications Web. La première application concerne l'état du cluster et elle est accessible via l'adresse **http://krb5.abc.io:8088**. Il est possible d'avoir une vue globale sur les nœuds du cluster et sur les jobs en cours d'exécution comme il est montré dans la figure 21 :

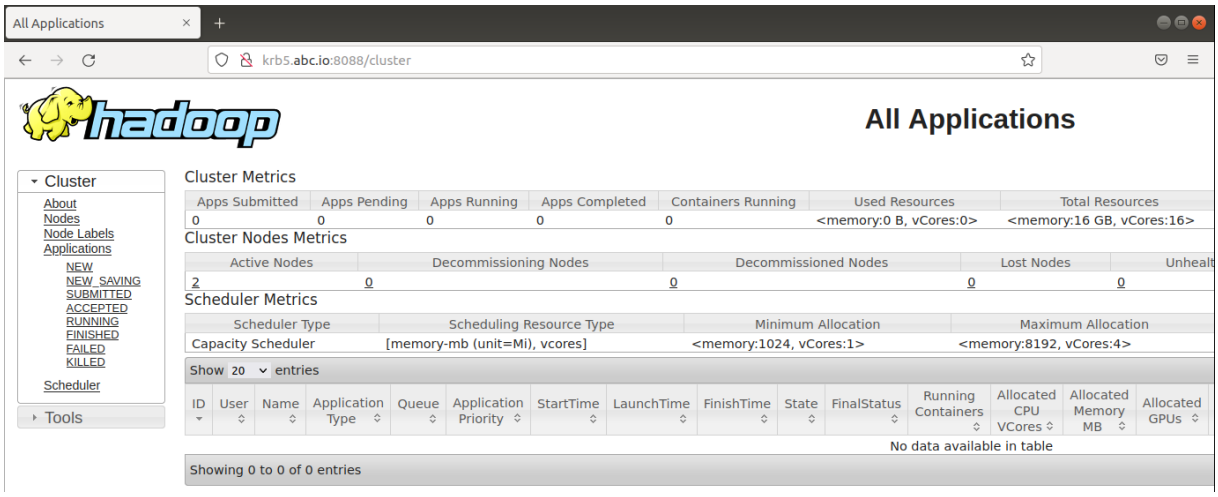


Figure 4: Interface utilisateur d'administration Hadoop.

La deuxième interface utilisateur concerne l'accès aux données contenues dans le nœud **NameNode** et elle est accessible via l'adresse **http://krb5.abc.io:9870**. Elle permet d'obtenir des informations sur la capacité totale et connaître l'état de disponibilité des nœuds[8]. Elle permet également d'avoir des informations sur les fichiers et de naviguer dans le **HDFS** du cluster.

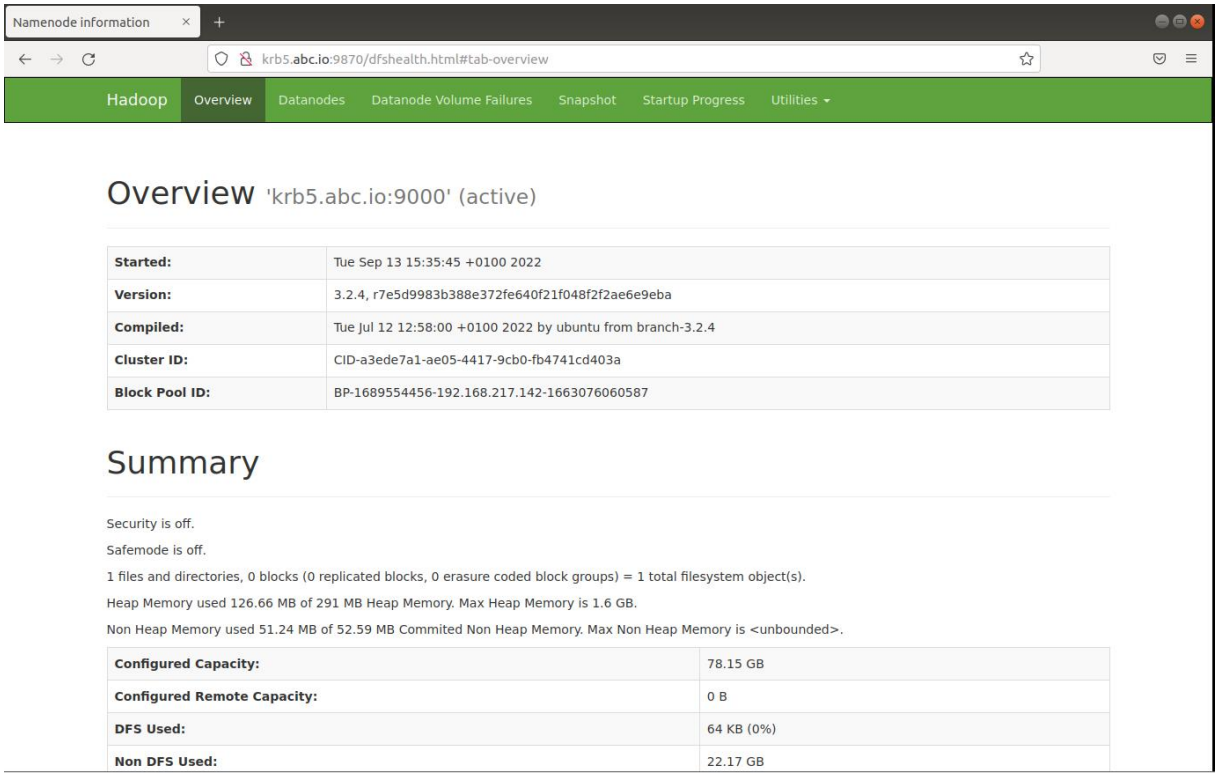


Figure 5: Interface d'accès aux données contenues dans le nœud NameNode.

Annexes C

Installation kerberos :

Nous allons installer le serveur kerberos **KDC** uniquement sur le nœud maître, ce serveur contient le serveur d'authentification (**AS**) et le serveur d'octroi de tickets (**TGS**). De même nous installons le client kerberos sur tous les autres nœuds de notre cluster (nœuds esclaves), cela nous permettra d'obtenir un ticket auprés du **KDC** afin que nous puissions utiliser les différents services et ressources de notre cluster.

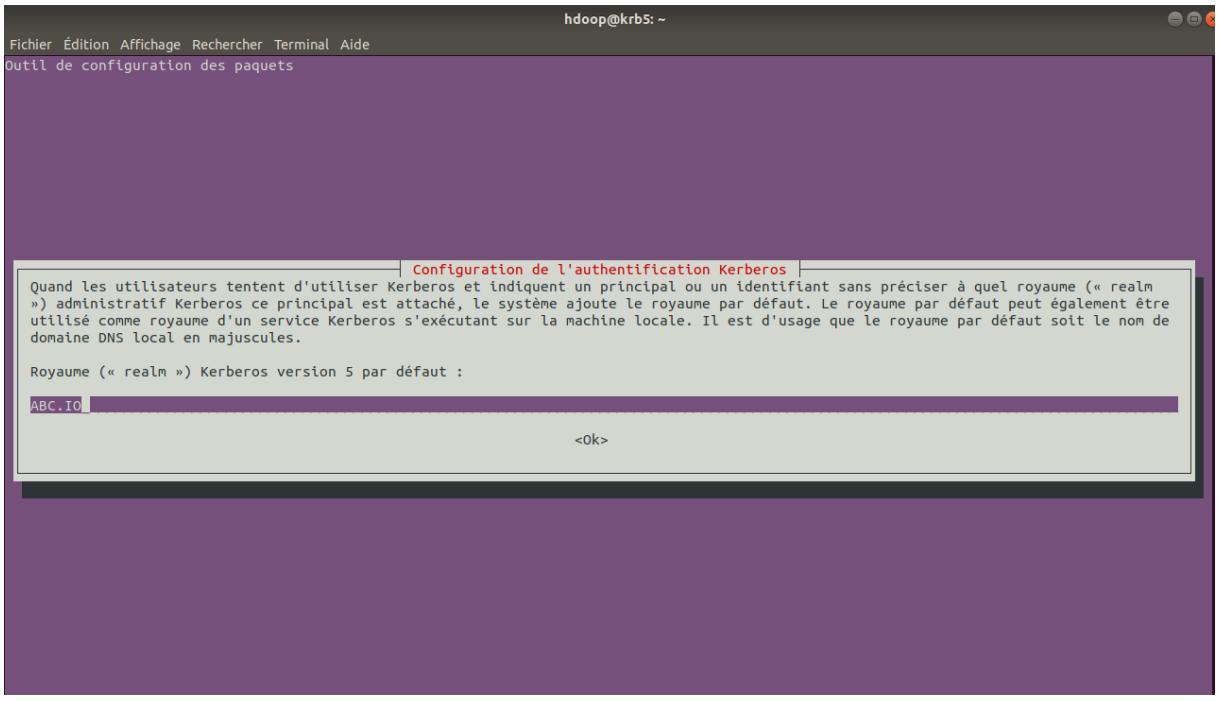
- **Installation serveur Kerberos :**

Avant de commencer l'installation de serveur **kerberos**, nous devons d'abord synchroniser l'heure de toutes les machine à l'aide du même serveur **NTP** (Network Time Protocol), car **kerberos** est un protocole sensible au temps. Ainsi, si l'heure système locale entre une machine cliente et le serveur diffère de plus de cinq minutes (par défaut), le poste de travail ne pourra pas s'authentifier.

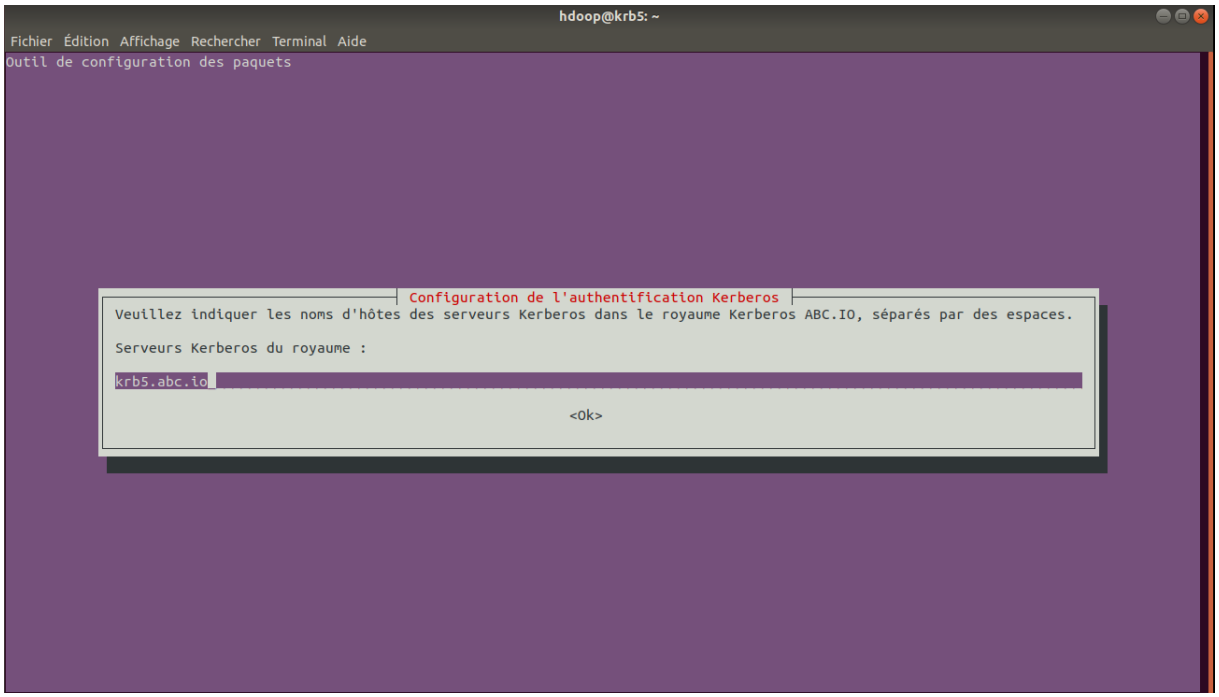
Nous commençons à installer les paquets **krb5-kdc, krb5-config** et **krb5-admin-server** sur le nœud maître à l'aide de la commande en ligne ci-dessous:

```
hdoop@krb5:~$ sudo apt-get install krb5-kdc krb5-config krb5-admin-server -y
```

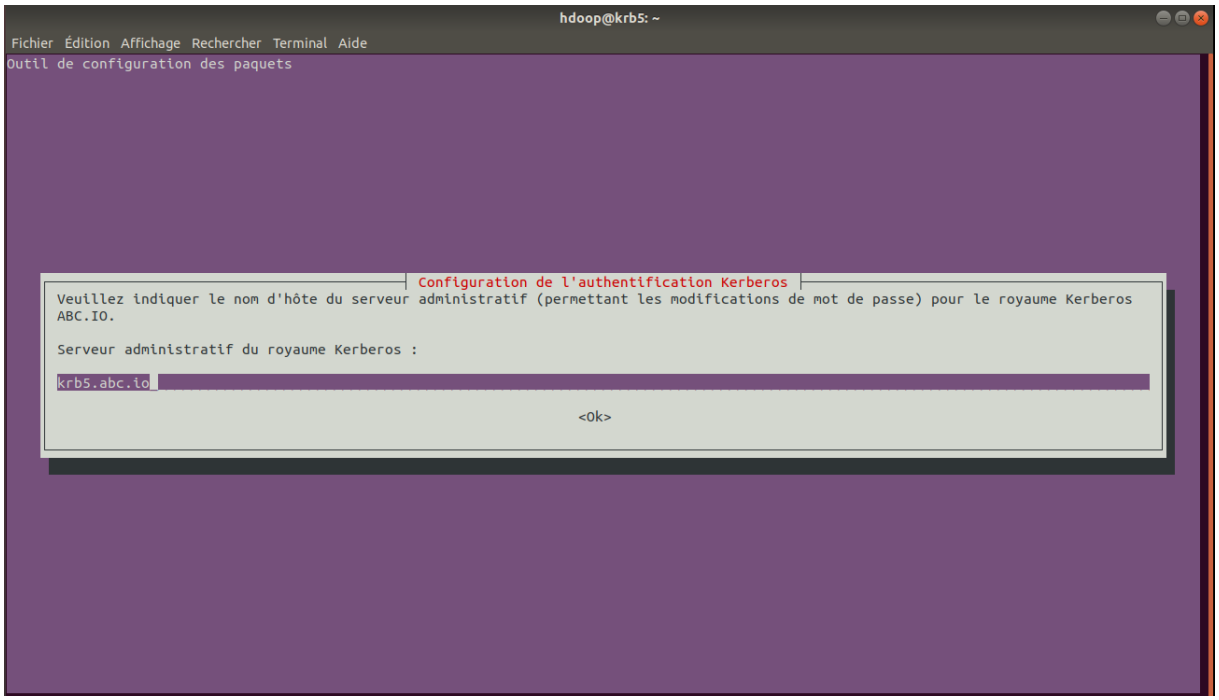
Dans ce qui suit, nous montrons la configuration du fichier **/etc/krb5.conf** qui est utilisé par les bibliothèques **kerberos5**. Tout d'abord, nous fournissons le **Realm Kerberos « ABC.IO »** par défaut comme **realm** de notre domaine :



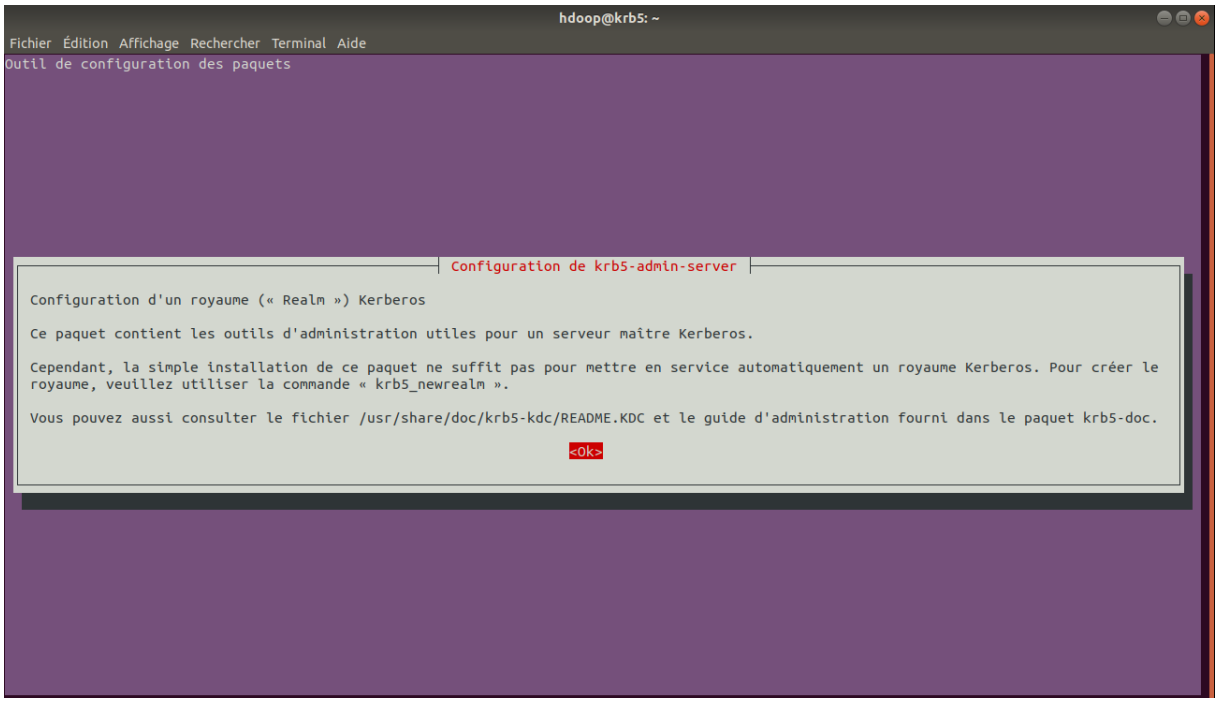
Dans l'étape qui suit, nous fournissons le nom d'hôte de serveur Kerberos, cependant nous allons fournir le nom d'hôte de la machine maître sur laquelle est installé le serveur kerberos (**AS,TGS**), qui est dans notre cas le hôte **« krb5.abc.io »**, la même machine sur laquelle nous avons installé **Hadoop** en mode master.



Ensuite, il nous a demandé de fournir le nom d'hôte du serveur d'administration, nous saisissons le nom d'hôte de la machine maître.



Mise en place d'un royaume de kerberos :



Ensuite, nous allons créer un nouveau royaume kerberos avec l'utilitaire **krb5_newrealm** avec la commande en ligne ci-dessous :

```
hdoop@krb5:~$ sudo krb5_newrealm
```

Il nous demandera un mot de passe principal de base de données, qui sera utilisé pour chiffrer la base de données locale.

- **La création des différents principaux Kerberos :**

Tout d'abord, nous créons le principal d'administration pour le serveur KDC kerberos appelé «**root**».

Un principal kerberos représente une identité unique dans un système kerberos à laquelle kerberos peut attribuer des tickets pour accéder aux services compatibles kerberos.

La commande qui permet d'ajouter un nouveau principal est **« addprinc »** avec une demande de saisir deux fois un mot de passe.

```
hdoop@krb5:~$ sudo kadmin.local
Authenticating as principal root/admin@ABC.IO with password.
kadmin.local:  addprinc root/admin
WARNING: no policy specified for root/admin@ABC.IO; defaulting to no policy
Enter password for principal "root/admin@ABC.IO":
Re-enter password for principal "root/admin@ABC.IO":
Principal "root/admin@ABC.IO" created.
kadmin.local:
```

Nous allons passer à la création d'un principal pour chaque instance du service **Hadoop**.

- **Création d'un principal pour hdfs utilisé pour le NameNode, les DataNodes et Secondary NameNode.**

```
kadmin.local:  addprinc -randkey hdfs/krb5.abc.io@ABC.IO
WARNING: no policy specified for hdfs/krb5.abc.io@ABC.IO; defaulting to no policy
Principal "hdfs/krb5.abc.io@ABC.IO" created.
```

L'option **-randkey** : définit la clé d'un principal sur une valeur aléatoire.

- **Création d'un principal de mapred utilisé pour le serveur d'historique Job MapReduce.**

```
kadmin.local:  addprinc -randkey mapred/krb5.abc.io@ABC.IO
WARNING: no policy specified for mapred/krb5.abc.io@ABC.IO; defaulting to no policy
Principal "mapred/krb5.abc.io@ABC.IO" created.
```

- **Création d'un principal yarn utilisé pour le NodeManager et ResourceManager.**

```
kadmin.local:  addprinc -randkey yarn/krb5.abc.io@ABC.IO
WARNING: no policy specified for yarn/krb5.abc.io@ABC.IO; defaulting to no policy
Principal "yarn/krb5.abc.io@ABC.IO" created.
```

- **Création du principal HTTP.**

```
kadmin.local:  addprinc -randkey HTTP/krb5.abc.io@ABC.IO
WARNING: no policy specified for HTTP/krb5.abc.io@ABC.IO; defaulting to no policy
Principal "HTTP/krb5.abc.io@ABC.IO" created.
```

Maintenant que nous avons réussie à créer nos nouveaux principaux, nous utilisons la commande **« list_principals »** qui permet d'afficher tous les principaux créer afin de confirmer nos nouvelles création.

```
hdoop@krb5:~$ sudo kadmin.local
[sudo] password for hdoop:
Authenticating as principal root/admin@ABC.IO with password.
kadmin.local:  list_principals
HTTP/client1.abc.io@ABC.IO
HTTP/client2.abc.io@ABC.IO
HTTP/krb5.abc.io@ABC.IO
K/M@ABC.IO
hdfs/client1.abc.io@ABC.IO
hdfs/client2.abc.io@ABC.IO
hdfs/krb5.abc.io@ABC.IO
hdoop/client2.abc.io@ABC.IO
kadmin/admin@ABC.IO
kadmin/changepw@ABC.IO
kadmin/krb5.abc.io@ABC.IO
kiprop/krb5.abc.io@ABC.IO
krbtgt/ABC.IO@ABC.IO
mapred/client1.abc.io@ABC.IO
mapred/client2.abc.io@ABC.IO
mapred/krb5.abc.io@ABC.IO
root/admin@ABC.IO
yarn/client1.abc.io@ABC.IO
yarn/client2.abc.io@ABC.IO
yarn/krb5.abc.io@ABC.IO
```

- **La génération des fichiers Keytab :**

Un **Keytab** (abréviation de key table) stocke les clés à long terme pour un ou plusieurs principaux. Les **keytabs** sont représentés par des fichiers dans un format standard. Les **keytabs** sont les plus souvent utilisés pour permettre aux applications serveur d'accepter les authentifications des clients, mais ils peuvent également être utilisés pour obtenir les informations d'identification initiales des applications client.

Dans ce qui suit nous allons créer un fichier **keytab** pour chaque service de **Hadoop**.

- **Création de fichier keytab hdfs utilisé pour le NameNode, les DataNodes et Secondary NameNode. Il contient le principal hdfs et le principal HTTP.**

```
hdoop@krb5:~$ sudo kadmin.local
Authenticating as principal root/admin@ABC.IO with password.
kadmin.local:  xst -k hdfs.keytab hdfs/krb5.abc.io HTTP/krb5.abc.io
Entry for principal hdfs/krb5.abc.io with kvno 2, encryption type aes256-cts-hmac-sha1-96 added to keytab WRFILE:hdfs.keytab.
Entry for principal hdfs/krb5.abc.io with kvno 2, encryption type aes128-cts-hmac-sha1-96 added to keytab WRFILE:hdfs.keytab.
Entry for principal HTTP/krb5.abc.io with kvno 2, encryption type aes256-cts-hmac-sha1-96 added to keytab WRFILE:hdfs.keytab.
Entry for principal HTTP/krb5.abc.io with kvno 2, encryption type aes128-cts-hmac-sha1-96 added to keytab WRFILE:hdfs.keytab.
```

-k : Utilise un **Keytab** pour déchiffrer la réponse **KDC** au lieu de demander un mot de passe.

- **Création du fichier keytab mapred utilisé pour le serveur d'historique Job MapReduce (dans le mode YARN). Il a un principal HTTP et un principal mapred.**

```
kadmin.local:  xst -k mapred.keytab mapred/krb5.abc.io HTTP/krb5.abc.io
Entry for principal mapred/krb5.abc.io with kvno 2, encryption type aes256-cts-hmac-sha1-96 added to keytab WRFILE:mapred.keytab.
Entry for principal mapred/krb5.abc.io with kvno 2, encryption type aes128-cts-hmac-sha1-96 added to keytab WRFILE:mapred.keytab.
Entry for principal HTTP/krb5.abc.io with kvno 3, encryption type aes256-cts-hmac-sha1-96 added to keytab WRFILE:mapred.keytab.
Entry for principal HTTP/krb5.abc.io with kvno 3, encryption type aes128-cts-hmac-sha1-96 added to keytab WRFILE:mapred.keytab.
```

- **Création du fichier keytab yarn utilisé pour le NodeManger et le ResourceManager. Il aura le principal yarn et le principal HTTP.**

```
kadmin.local:  xst -k yarn.keytab yarn/krb5.abc.io HTTP/krb5.abc.io
Entry for principal yarn/krb5.abc.io with kvno 2, encryption type aes256-cts-hmac-sha1-96 added to keytab WRFILE:yarn.keytab.
Entry for principal yarn/krb5.abc.io with kvno 2, encryption type aes128-cts-hmac-sha1-96 added to keytab WRFILE:yarn.keytab.
Entry for principal HTTP/krb5.abc.io with kvno 4, encryption type aes256-cts-hmac-sha1-96 added to keytab WRFILE:yarn.keytab.
Entry for principal HTTP/krb5.abc.io with kvno 4, encryption type aes128-cts-hmac-sha1-96 added to keytab WRFILE:yarn.keytab.
```

Nous utilisons la commande **« klist »** avec les options **-e -k -t** pour afficher un **keytab** :

```
hdoop@krb5:~$ sudo klist -e -k -t hdfs.keytab
[sudo] Mot de passe de hdoop :
Keytab name: FILE:hdfs.keytab
KVNO Timestamp           Principal
---- ------------------- ------------------------------------------------------
   2 09/12/2022 16:14:11 hdfs/krb5.abc.io@ABC.IO (aes256-cts-hmac-sha1-96)
   2 09/12/2022 16:14:11 hdfs/krb5.abc.io@ABC.IO (aes128-cts-hmac-sha1-96)
   2 09/12/2022 16:14:11 HTTP/krb5.abc.io@ABC.IO (aes256-cts-hmac-sha1-96)
   2 09/12/2022 16:14:11 HTTP/krb5.abc.io@ABC.IO (aes128-cts-hmac-sha1-96)
```

L'option **-e** : Affiche les types de cryptage de la clé de session et du ticket pour chaque clé dans le fichier **keytab**.

L'option **-k** : Liste les clés contenues dans un fichier **keytab**.

L'option **-t** : Affiche la saisie de l'heure pour chaque entrée de clé dans le fichier **keytab**.

Nous déplaçons tous les fichiers **keytab** vers le répertoire **/etc/hadoop/conf/**:

```
hdoop@krb5:~$ sudo mv hdfs.keytab mapred.keytab yarn.keytab /etc/hadoop/conf
```

Nous utilisons la commande «**chown**» pour changer le propriétaire ainsi que le groupe propriétaire de tous les fichiers keytabs, dans ce cas «**chown hdoop:hadoop**», l'utilisateur hdoop sera le nouveau propriétaire et le groupe hadoop sera le nouveau groupe propriétaire des fichiers keytabs.

```
hdoop@krb5:~$ sudo chown hdoop:hadoop /etc/hadoop/conf/hdfs.keytab
```

```
hdoop@krb5:~$ sudo chown hdoop:hadoop /etc/hadoop/conf/mapred.keytab
```

```
hdoop@krb5:~$ sudo chown hdoop:hadoop /etc/hadoop/conf/yarn.keytab
```

La commande «**chmod**» ci-dessous permet d'affecter les autorisations de lecture au propriétaire de tous les fichier keytab.

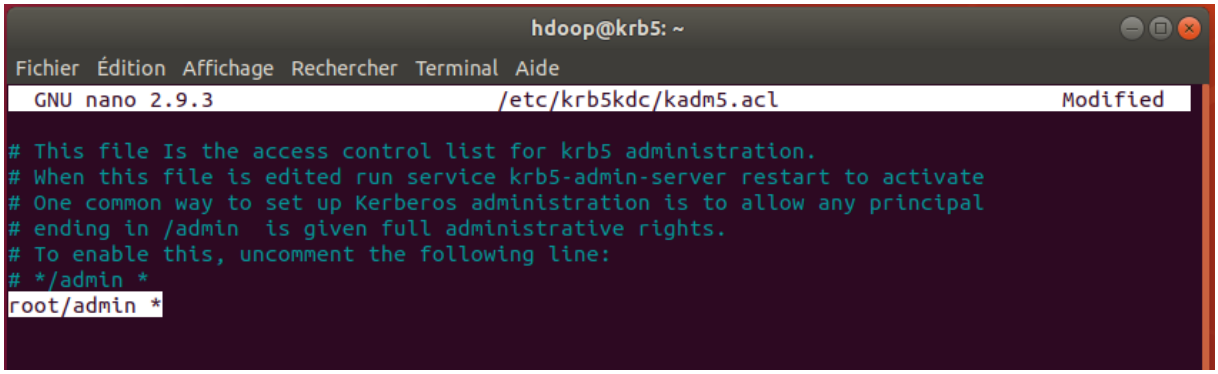
```
hdoop@krb5:~$ sudo chmod 400 /etc/hadoop/conf/*.keytab
```

Kerberos utilise un fichier de liste de contrôle d'accès (**ACL**) pour gérer les droits d'accès à la base de données Kerberos.

Nous éditons le fichier **/etc/krb5kdc/kadm5.acl** pour ajouter le principe de l'utilisateur administrateur au contrôle d'accès avec la commande ci-dessous :

```
hdoop@krb5:~$ sudo nano /etc/krb5kdc/kadm5.acl
```

La ligne ajouter **«root/admin *»** signifie que le super utilisateur **root** avec une instance d'administrateur dispose de tous les privilèges d'administration, à l'exception de l'extraction des clés.

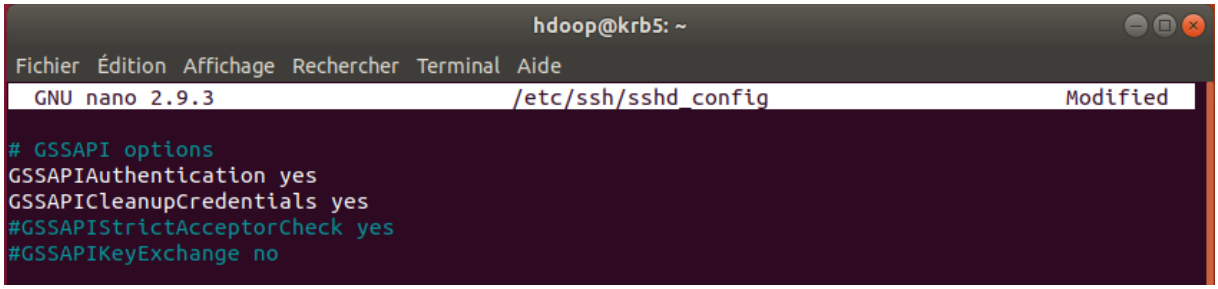


Ensuite nous redémarrons les services **krb5-admin-server** pour valider les modifications à l'aide de la commande qui suit :

```
hdoop@krb5:~$ sudo systemctl restart krb5-admin-server.service
```

Nous allons maintenant configurer le fichier **/etc/ssh/sshd_config**, nous modifions ainsi les paramètres suivantes :

```
hdoop@krb5:~$ sudo nano /etc/ssh/sshd_config
```



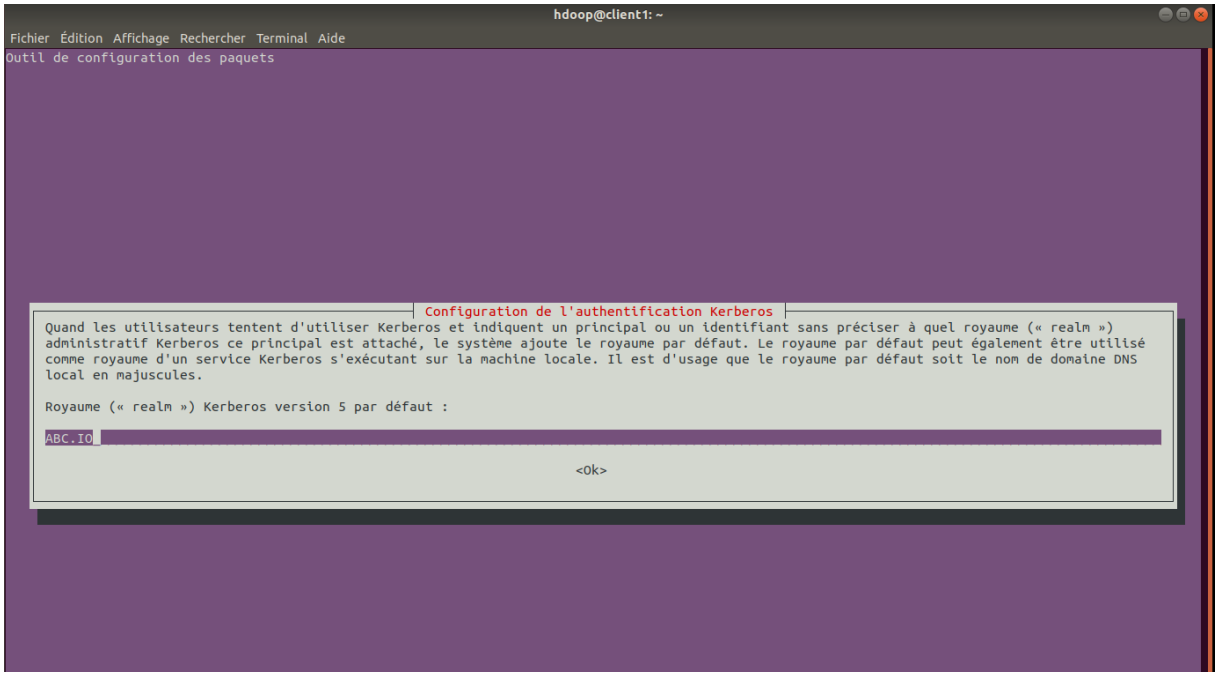

```
hdoop@krb5:~$ systemctl restart sshd
```

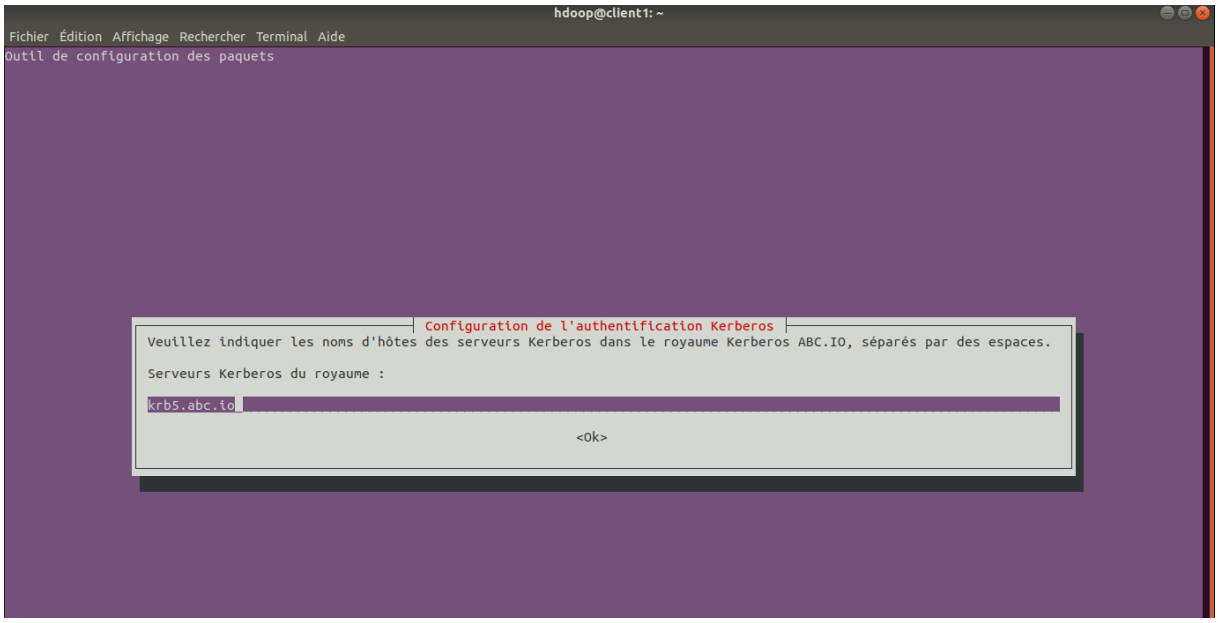
- **Installation client kerberos:**

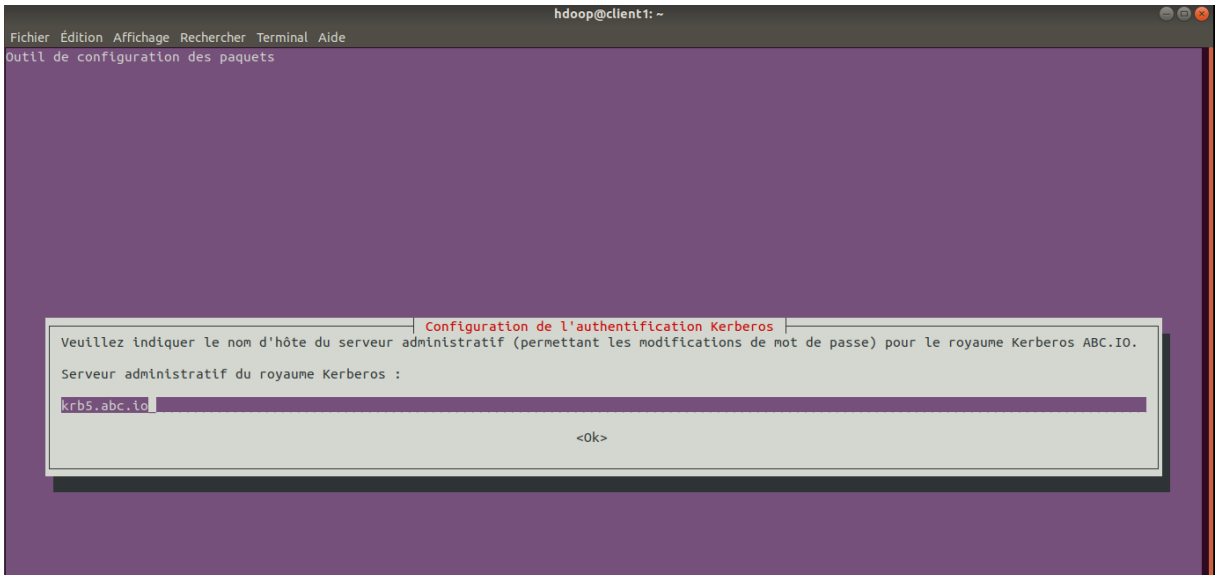
Les paquets **krb5-user** et **libpam-krb5** sont nécessaires pour pouvoir s'authentifier sur un domaine kerberos. Nous allons aussi ajouter un autres paquets **libpam-ccreds** qui n'est pas obligatoire mais nous facilite la tâche.

```
hdoop@client1:~$ sudo apt-get install -y krb5-user libpam-ccreds libpam-krb5
```

Nous allons fournir les informations de client kerberos comme celle d'installation de serveur kerberos suivantes:







- **La création des différents principaux Kerberos :**

Dans tous les nœuds clients, nous allons créer un principal pour chaque instance du service Hadoop (HDFS, MapReduce, YARN).

- **Création d'un principal pour hdfs.**

```
hdoop@client1:~$ sudo kadmin
[sudo] Mot de passe de hdoop :
Authenticating as principal root/admin@ABC.IO with password.
Password for root/admin@ABC.IO:
kadmin:  addprinc -randkey hdfs/client1.abc.io@ABC.IO
WARNING: no policy specified for hdfs/client1.abc.io@ABC.IO; defaulting to no policy
Principal "hdfs/client1.abc.io@ABC.IO" created.
```

- **Création d'un principal de mapred.**

```
kadmin:  addprinc -randkey mapred/client1.abc.io@ABC.IO
WARNING: no policy specified for mapred/client1.abc.io@ABC.IO; defaulting to no policy
Principal "mapred/client1.abc.io@ABC.IO" created.
```

- **Création d'un principal yarn.**

```
kadmin:  addprinc -randkey yarn/client1.abc.io@ABC.IO
WARNING: no policy specified for yarn/client1.abc.io@ABC.IO; defaulting to no policy
Principal "yarn/client1.abc.io@ABC.IO" created.
```

- **Création du principal HTTP.**

```
kadmin:  addprinc -randkey HTTP/client1.abc.io@ABC.IO
WARNING: no policy specified for HTTP/client1.abc.io@ABC.IO; defaulting to no policy
Principal "HTTP/client1.abc.io@ABC.IO" created.
```

➢ **La génération des fichiers Keytab :**

Dans tous les nœuds clients nous allons créer un fichier **keytab** pour chaque service de **Hadoop**.

- **Création de fichier keytab hdfs.**

```
kadmin:  xst -norandkey -k hdfs.keytab hdfs/client1.abc.io HTTP/client1.abc.io
Entry for principal hdfs/client1.abc.io with kvno 5, encryption type aes256-cts-hmac-sha1-96 added to keytab WRFILE:hdfs.keytab.
Entry for principal hdfs/client1.abc.io with kvno 5, encryption type aes128-cts-hmac-sha1-96 added to keytab WRFILE:hdfs.keytab.
Entry for principal HTTP/client1.abc.io with kvno 3, encryption type aes256-cts-hmac-sha1-96 added to keytab WRFILE:hdfs.keytab.
Entry for principal HTTP/client1.abc.io with kvno 3, encryption type aes128-cts-hmac-sha1-96 added to keytab WRFILE:hdfs.keytab.
```

-norandkey : Cette option est uniquement disponible dans **kadmin.local**. Les clés et leurs numéros de version restent inchangés et ne pas randomiser les clés.

- **Création du fichier keytab mapred.**

```
kadmin:  xst -norandkey -k mapred.keytab mapred/client1.abc.io HTTP/client1.abc.io
Entry for principal mapred/client1.abc.io with kvno 1, encryption type aes256-cts-hmac-sha1-96 added to keytab WRFILE:mapred.keytab.
Entry for principal mapred/client1.abc.io with kvno 1, encryption type aes128-cts-hmac-sha1-96 added to keytab WRFILE:mapred.keytab.
Entry for principal HTTP/client1.abc.io with kvno 3, encryption type aes256-cts-hmac-sha1-96 added to keytab WRFILE:mapred.keytab.
Entry for principal HTTP/client1.abc.io with kvno 3, encryption type aes128-cts-hmac-sha1-96 added to keytab WRFILE:mapred.keytab.
```

- **Création du fichier keytab yarn.**

```
kadmin:  xst -norandkey -k yarn.keytab yarn/client1.abc.io HTTP/client1.abc.io
Entry for principal yarn/client1.abc.io with kvno 1, encryption type aes256-cts-hmac-sha1-96 added to keytab WRFILE:yarn.keytab.
Entry for principal yarn/client1.abc.io with kvno 1, encryption type aes128-cts-hmac-sha1-96 added to keytab WRFILE:yarn.keytab.
Entry for principal HTTP/client1.abc.io with kvno 3, encryption type aes256-cts-hmac-sha1-96 added to keytab WRFILE:yarn.keytab.
Entry for principal HTTP/client1.abc.io with kvno 3, encryption type aes128-cts-hmac-sha1-96 added to keytab WRFILE:yarn.keytab.
```

Nous utilisons la commande **« klist »** avec l'option **-e -k -t** pour afficher un **keytab** :

```
hdoop@client1:~$ sudo klist -e -k -t hdfs.keytab
[sudo] Mot de passe de hdoop :
Keytab name: FILE:hdfs.keytab
KVNO Timestamp           Principal
---- ------------------- ------------------------------------------------------
   5 09/12/2022 22:52:58 hdfs/client1.abc.io@ABC.IO (aes256-cts-hmac-sha1-96)
   5 09/12/2022 22:52:58 hdfs/client1.abc.io@ABC.IO (aes128-cts-hmac-sha1-96)
   3 09/12/2022 22:52:58 HTTP/client1.abc.io@ABC.IO (aes256-cts-hmac-sha1-96)
   3 09/12/2022 22:52:58 HTTP/client1.abc.io@ABC.IO (aes128-cts-hmac-sha1-96)
```

Nous déplaçons tous les fichiers **keytab** vers le répertoire **/etc/hadoop/conf/**:

```
hdoop@client1:~$ sudo mv hdfs.keytab mapred.keytab yarn.keytab /etc/hadoop/conf
```

Nous utilisons la commande «**chown**» pour changer le propriétaire ainsi que le groupe de propriétaire de tous les fichiers keytab.

```
hdoop@client1:~$ sudo chown hdoop:root /etc/hadoop/conf/hdfs.keytab
hdoop@client1:~$ sudo chown hdoop:root /etc/hadoop/conf/mapred.keytab
hdoop@client1:~$ sudo chown hdoop:root /etc/hadoop/conf/yarn.keytab
```

Nous utilisons la commande «**chmod**» afin de donner les droits de permissions d'accès en lecture pour le propriétaire de tous les fichiers keytab.

```
hdoop@client1:~$ sudo chmod 400 /etc/hadoop/conf/*.keytab
```

Annexes D

Configuration de Hadoop avec Kerberos :

Tous d'abord nous allons configurer le fichier **/etc/ssh/sshd_config**, nous modifions ainsi les paramètres suivantes :

```
hdoop@client1:~$ sudo nano /etc/ssh/sshd_config
```

```
hdoop@client1: ~
Fichier  Édition  Affichage  Rechercher  Terminal  Aide
  GNU nano 2.9.3                  /etc/ssh/sshd_config                  Modified

# Kerberos options
KerberosAuthentication yes
KerberosOrLocalPasswd yes
KerberosTicketCleanup yes
KerberosGetAFSToken no
```

- **Configuration des fichiers core-site.xml, hdfs-site.xml et yarn-site.xml :**

Dans ce que suit, nous allons ajouter les propriétés qui permettent au cluster Hadoop d'utiliser l'authentification kerberos.

- **core-site.xml :**

Dans ce fichier nous allons définir la valeur de la propriété **hadoop.security.authentication** sur **« kerberos »** (indiquant que l'authentification kerberos serait utilisée), et **hadoop.security.authorization** sur **« true »** pour indiquer que nous avons activé l'autorisation au niveau de service **RPC** (Remote Procedure Call).

```
hdoop@krb5:~$ sudo nano /etc/hadoop/etc/hadoop/core-site.xml
[sudo] password for hdoop:
```

```
hdoop@krb5: ~
Fichier  Édition  Affichage  Rechercher  Terminal  Aide
  GNU nano 2.9.3                /etc/hadoop/etc/hadoop/core-site.xml                Modified

<!-- kerbeors authentication configuration -->

<property>
<name>hadoop.security.authentication</name>
<value>kerberos</value>
</property>

<property>
<name>hadoop.security.authorization</name>
<value>true</value>
</property>

</configuration>
```

❖ **hdfs-site.xml :**

Dans le fichier **hdfs-site.xml** nous allons spécifier les propriétés représentés ci-dessous :

- La propriété **dfs.web.authentication.kerberos.principal :** indique le nom de principal pour **HTTP**.
- La propriété **dfs.block.access.token.enable :** avec sa valeur qui égale à true, signifie que les jetons d'accès aux HDFS pour des opérations sécurisées sont activés.

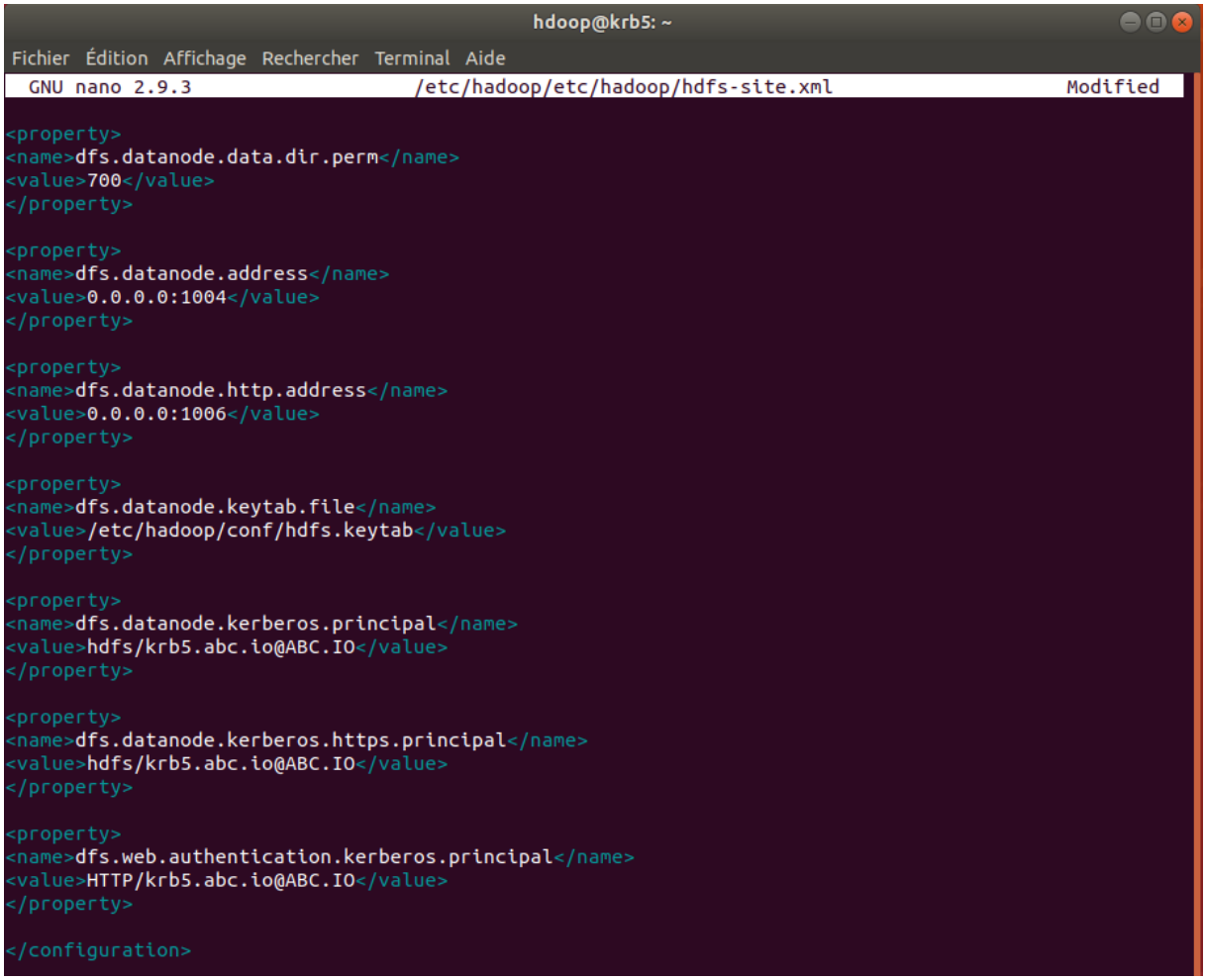
Dans les autres propriétés nous allons définir pour chaque instance de service Hadoop son principal kerberos et l'emplacement de son fichier keytab.

```
hdoop@krb5:~$ sudo nano /etc/hadoop/etc/hadoop/hdfs-site.xml
```

```
hdoop@krb5: ~
Fichier  Édition  Affichage  Rechercher  Terminal  Aide
 GNU nano 2.9.3                /etc/hadoop/etc/hadoop/hdfs-site.xml                Modified

               <!-- configuration kerberos -->

<property>
<name>dfs.block.access.token.enable</name>
<value>true</value>
</property>

<property>
<name>dfs.namenode.keytab.file</name>
<value>/etc/hadoop/conf/hdfs.keytab</value>
</property>

<property>
<name>dfs.namenode.kerberos.principal</name>
<value>hdfs/krb5.abc.io@ABC.IO</value>
</property>

<property>
<name>dfs.namenode.kerberos.internal.spnego.principal</name>
<value>HTTP/krb5.abc.io@ABC.IO</value>
</property>

<property>
<name>dfs.secondary.namenode.keytab.file</name>
<value>/etc/hadoop/conf/hdfs.keytab</value>
</property>

<property>
<name>dfs.secondary.namenode.kerberos.principal</name>
<value>hdfs/krb5.abc.io@ABC.IO</value>
</property>

<property>
<name>dfs.secondary.namenode.kerberos.internal.spnego.principal</name>
<value>HTTP/krb5.abc.io@ABC.IO</value>
</property>
```

- La propriété **dfs.namenode.keytab.file :** spécifie l'emplacement du fichier keytab hdfs.keytab qui est utilisé par le NameNode
- La propriété **dfs.namenode.kerberos.principal :** indique le nom du principal pour le NameNode.

- La propriété **dfs.namenode.kerberos.internal.spnego.principal :** indique le principal de serveur utilisé par le NameNode pour l'authentification SPNEGO de l'interface utilisateur Web lorsque la sécurité Kerberos est activée.
- La propriété **dfs.secondary.namenode.kerberos.internal.spnego.principal :** indique le principal de serveur utilisé pour le SecondaryNameNode pour l'authentification SPNEGO de l'interface utilisateur Web lorsque la sécurité Kerberos est activée.



- La propriété **dfs.datanode.data.dir.perm** : indique les autorisations qui doivent être définies sur les répertoires dfs.data.dir.
- La propriété **dfs.datanode.address** : indique l'adresse et le port du serveur DataNode pour le transfert de données.
- La propriété **dfs.datanode.http.address** : indique l'adresse et le port du serveur http du DataNode.
- La propriété **dfs.datanode.kerberos.https.principal** : spécifie le nom du principal kerberos HTTPS pour le DataNode.

❖ **yarn-site.xml :**

pour la configuration de fichier **yarn-site.xml** nous définissons pour chaque instance de service **hadoop** (**NodeManager**, **ResourceManager**) son principal kerberos et l'emplacement de son fichier keytab (la même chose pour le fichier **hdfs-site.xml**).

```
hdoop@krb5:~$ sudo nano /etc/hadoop/etc/hadoop/yarn-site.xml
```

```
hdoop@krb5: ~
Fichier  Édition  Affichage  Rechercher  Terminal  Aide
  GNU nano 2.9.3                /etc/hadoop/etc/hadoop/yarn-site.xml                Modified

<!-- kerberos configuration -->

<property>
<name>yarn.resourcemanager.keytab</name>
<value>/etc/hadoop/conf/yarn.keytab</value>
</property>

<property>
<name>yarn.resourcemanager.principal</name>
<value>yarn/krb5.abc.io@ABC.IO</value>
</property>

<name>yarn.nodemanager.keytab</name>
<value>/etc/hadoop/conf/yarn.keytab</value>
</property>

<property>
<name>yarn.nodemanager.principal</name>
<value>yarn/krb5.abc.io@ABC.IO</value>
</property>

<property>
<name>yarn.nodemanager.container-executor.class</name>
<value>org.apache.hadoop.yarn.server.nodemanager.LinuxContainerExecutor</value>
</property>

<property>
<name>yarn.nodemanager.linux-container-executor.group</name>
<value>yarn</value>
</property>

</configuration>
```

- La propriété **yarn.resourcemanager.keytab** : spécifie le keytab pour le ResourceManager.
- La propriété **yarn.resourcemanager.principal** : spécifie le principal kerberos pour le ResourceManager.
- La propriété **yarn.nodemanager.container-executor.class** : indique la classe qui exécutera (lancera) les conteneurs.
- La propriété **yarn.nodemanager.linux-container-executor.group** : spécifie un groupe spécial avec des autorisations exécutables pour l'exécuteur du conteneur, dont l'utilisateur NodeManager est le membre du groupe.

- **Obtention Du Ticket TGT :**

Nous tapons la commande **« klist »** pour afficher les tickets, nous recevrons le message suivant:

```
hdoop@krb5:~$ klist
klist: No credentials cache found (filename: /tmp/krb5cc_1001)
```

La commande **« klist »** renvois aucun ticket, cela signifie que nous avons aucun utilisateur (principal) authentifié.

Maintenant nous tapons la commande **« kinit »** suivi de l'emplacement du fichier keytab et de nom d'un principal, cela permet d'obtenir un ticket initial d'attribution de ticket TGT pour le principal:

```
hdoop@krb5:~$ kinit -t /etc/hadoop/conf/hdfs.keytab hdfs/krb5.abc.io
keytab specified, forcing -k
hdoop@krb5:~$
```

-t : spécifier l'emplacement d'un fichier keytab, pour utiliser la clé existante dans ce fichier, au lieu de mot de passe.

Nous exécutons une autre foie la commande **« klist »** qui affiche les informations sur le ticket **TGT**:

```
hdoop@krb5:~$ klist
Ticket cache: FILE:/tmp/krb5cc_1001
Default principal: hdfs/krb5.abc.io@ABC.IO

Valid starting       Expires              Service principal
09/15/2022 10:16:57  09/15/2022 20:16:57  krbtgt/ABC.IO@ABC.IO
        renew until 09/16/2022 10:16:43
```

Le **Ticket cache** signifie l'emplacement du fichier de ticket. Dans notre cas l'emplacement de ticket est **/tmp/krb5cc_1001**. Par contre **Default principal** nous renseigne sur le principale par default qui contient le ticket, dans notre cas c'est **hdfs/krb5.abc.io@ABC.IO**.

Les champs **Valid starting** et **Expires** décrivent la période de validité du ticket et le **Service principal** décrit chaque ticket. Le **TGT** a un premier composant qui est **krbtgt** et un second composant qui est le nom de domaine **ABC.IO.**

La commande **« kdestroy »** permet de détruire les tickets d'autorisation Kerberos actifs de l'utilisateur en supprimant le cache des informations d'identification qui les contient:

```
hdoop@krb5:~$ klist
Ticket cache: FILE:/tmp/krb5cc_1001
Default principal: hdfs/krb5.abc.io@ABC.IO

Valid starting       Expires              Service principal
09/15/2022 10:16:57  09/15/2022 20:16:57  krbtgt/ABC.IO@ABC.IO
        renew until 09/16/2022 10:16:43
hdoop@krb5:~$ kdestroy
hdoop@krb5:~$ klist
klist: No credentials cache found (filename: /tmp/krb5cc_1001)
```

Résumé :

Les Big Data désignent des données massives qui sont complexe et diversifiés dans leur structure et leurs formats. Ces données massives ont confronté des difficultés de stockage, d'analyse et de traitement.

L'écosystème Hadoop est la technologie la plus répondu pour ces problématiques, elle permet le traitement et le stockage des données volumineuses.

Dans ce travail nous parlerons des technologies de haute performance du Big Data et plus précisément le Framework Hadoop. Dans la partie applicative de notre projet nous estimons le temps de traitement des deux protocoles d'authentification (Kerberos basé sur la cryptographie symétrique et asymétrique).

Mots clés : Big Data, Hadoop, kerberos, KDC, HDFS.

Abstract:

Big Data refers to massive data that is complex and diverse in its structure and formats. This massive data has faced challenges in storage, analysis and processing.

The Hadoop ecosystem is the most popular technology for these issues, it allows the processing and storage of large data.

In this work we will talk about high performance Big Data technologies and more specifically the Hadoop Framework. In the application part of our project we estimate the processing time of the two authentication protocols (Kerberos based on symmetric and asymmetric cryptography).

Key words : Big Data, Hadoop, kerberos, KDC, HDFS.