

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



Université A/Mira de Béjaïa  
Faculté des Sciences Exactes  
Département d'Informatique

# Mémoire du Projet de Fin d'Études

En vue de l'obtention du diplôme de master Professionnel en informatique

*Spécialité : Administration et Sécurité des Réseaux*

## Thème

---

Mise en place d'une architecture big data pour la  
gestion et fouille de fichiers logs

---

Réalisé par :

AIT LOUNIS DINA AMARI FAIZA

Encadré par :

Mr. Amroun KAMAL  
Mr. ELSAKAAN Nadim

Membres de jury :

Président : Mme. TIGHIDET SORAYA  
Examineur : Mme. Khaled Hayette

2021/2022

# *Remerciements*

*C'est avec l'aide de DIEU tout puissant qui nous a donné l'opportunité de mener à bien ce travail et sans que ce mémoire n'aurait jamais vu le jour.*

Nous tenons à remercier chaleureusement en premier lieu notre encadrants Mr EL-SAKAAN Nadim et Mr AMROUN Kamal d'avoir accepté d'encadrer notre projet de fin d'étude, pour leur orientations, leur précieux conseils et encouragements qui nous ont permis de mener à bien ce travail.

Nous tenons à exprimer également notre gratitude aux membres de jury pour avoir accepté de juger ce modeste travail.

Un merci particulier à tout le personnel et à tous les travailleurs de la société Bati-compos, qui ont été toujours à notre service.

Avec les mots d'appréciation et de respect, nous remercions tous nos enseignants et les membres du département d'Informatique de l'université ABDERRAHMANE MIRA Béjaïa pour nous avoir appris la valeur de l'autodidactie.

En préambule à ce mémoire, nous souhaitons exprimer nos grande gratitude à nos familles, qui ont su nous supporter et encourager tout au long de notre vie, ainsi que pour leur aide inestimable, leur patience et leur soutien indéfectible.

Enfin, nous remercions tous ceux et celles qui, de Près ou de loin, ont contribué à l'aboutissement de ce mémoire.

# *Dédicaces*

## **Je dédie ce modeste travail :**

A mes chers parents, pour tous leurs sacrifices, leur amour et leur soutien moral ;

A mes soeurs et frère, pour leurs encouragements permanents ;

A toute ma famille qui m'a doté d'une éducation digne ;

A mes ami(e)s et collègues, et tous qui m'ont aidé ;

A ma binôme Dina et sa famille.

*AMARI Faiza*

# *Dédicaces*

## **Je dédie ce modeste travail :**

A mes chers parents, pour tous leurs sacrifices, leur amour et leur soutien durant ces  
années d'études.  
A mon mari Mohand pour son aide moral ,sa patience et sa compréhension tout au long  
de cette année d'étude.  
A ma soeur Liza et mon frère Salas, pour leurs encouragements permanents.  
A ma fille Nelia .  
A ma belle famille .  
A mes ami(e)s et collègues, et tous ceux qui m'ont aidé .  
A ma binôme Faiza et sa famille.

*AIT LOUNIS Dina*



# *Résumé*

*La transformation digitale place au coeur de son action la mise en valeur des données. Cela génère une accélération de l'évolution du système d'information ce qui propulse les entreprises dans l'ère du big data et a cause de manque d'outils, limitations technologiques et la rareté des compétences clés nécessaires ces données ne sont plus exploitées correctement. D'où L'objet de cette étude qui consiste à apporter un éclairage nouveau s'agissant d'une mise en place d'une approche en employant les outils du Big Data afin de traiter, stocker et restituer les données logs. La plateforme Hadoop et le framework MapReduce sont suffisamment bon pour réaliser les actions précédemment citées.*

**Mots clés :** *Hadoop, elastic-search, kibana, java, Linux Fedora, fichiers logs, systèmes d'exploitations, émulation réseaux (GNS3), virtualisation (vmware workstations 16)*

# *Abstract*

*Data is on the heart of the digital transformation. This generates an acceleration of the information system evolution that propels companies into the age of big data and due to lack of tools, technological limitations and the scarcity of the key competences needed, these data are no longer being properly exploited. Hence The purpose of this study, which is to shed new light on the implementation of an approach using Big Data tools to process, store and return log data. The Hadoop platform and the MapReduce framework are good enough to accomplish the above mentioned actions.*

**key words :** *Hadoop, elastic-search, kibana, java, Linux Fedora, fichiers logs, systèmes d'exploitations, émulation réseaux (GNS3), virtualisation (vmware workstations 16)*

# Table des matières

<b>Table des matières</b>	<b>i</b>
<b>Table des figures</b>	<b>iii</b>
<b>Liste des abréviations</b>	<b>vi</b>
<b>Introduction générale</b>	<b>1</b>
<b>1 Les systèmes d'informations au cœur des métiers</b>	<b>3</b>
1.1 Parc informatique et ses composants . . . . .	3
1.1.1 Parc matériel . . . . .	3
1.1.2 Parc Logiciel . . . . .	4
1.1.3 Les fichiers Logs . . . . .	4
1.2 Les applications de gestion orientées métiers . . . . .	5
1.2.1 L'urbanisation des S.I . . . . .	5
1.2.2 Les services WEB . . . . .	6
1.2.3 ERP . . . . .	6
1.2.4 CRM . . . . .	7
1.2.5 GMAO . . . . .	8
1.2.6 TMS . . . . .	8
1.3 Le mouvement NoSQL au sein de l'entreprise . . . . .	8
1.3.1 Qu'est-ce que le NoSQL? . . . . .	8
1.3.2 Les environnements NoSQL . . . . .	9
1.3.3 Docker et kubernetes . . . . .	12
1.3.4 Hadoop . . . . .	13
1.3.5 ElasticSearch . . . . .	16
<b>2 Introduction au cas d'étude</b>	<b>17</b>
2.1 Présentation de la société d'accueil . . . . .	17
2.1.1 Les produits fabriqués . . . . .	17
2.2 Organigramme de BATICOMPOS . . . . .	18
2.2.1 Cellule contrôle de gestion . . . . .	20
2.2.2 Service contrôle qualité . . . . .	20
2.2.3 Service HSE . . . . .	20
2.2.4 Cellule informatique . . . . .	20
2.2.5 Direction technico-commerciale et marketing . . . . .	20

2.2.6	Direction approvisionnements . . . . .	21
2.2.7	Direction technique . . . . .	21
2.2.8	Direction des ressources humaines . . . . .	21
2.2.9	Direction de la comptabilité et des finances . . . . .	21
2.3	Caractéristique des équipements réseau de l'entreprise . . . . .	22
2.3.1	Serveur HP DL380 G7 Proliant . . . . .	22
2.3.2	Modem . . . . .	22
2.3.3	Onduleur . . . . .	22
2.3.4	La passerelle . . . . .	23
2.3.5	L'armoire de brassage . . . . .	23
2.3.6	Les prises . . . . .	25
2.3.7	Les câbles . . . . .	26
2.4	Cas d'étude : Baticompos SPA . . . . .	26
2.4.1	La problématique . . . . .	26
2.4.2	La solution proposée . . . . .	26
<b>3</b>	<b>Mise en place de la solution</b>	<b>28</b>
3.1	Mise en place de la solution . . . . .	28
3.1.1	Installation et configuration de fedora server sur vmware worksta- tion . . . . .	28
3.1.2	Installation de Docker sur Fedora . . . . .	38
3.1.3	Installation de kubernetes sur Fedora . . . . .	40
3.1.4	Installation d'Elastic Search sur Fedora . . . . .	41
3.1.5	Installation de Kibana sur Fedora . . . . .	44
3.1.6	Installation de Hadoop . . . . .	44
<b>4</b>	<b>Déploiement et exploitation de la solution</b>	<b>51</b>
4.1	La simulation de la topologie . . . . .	51
4.1.1	Topologie du réseau existant . . . . .	51
4.1.2	simulation de la topologie sous GNS3 . . . . .	53
4.2	Mise en oeuvre de la solution . . . . .	68
4.2.1	Installation du serveur FTP . . . . .	68
4.2.2	Importation du fichier de données . . . . .	71
4.2.3	Insertion des données dans HDFS . . . . .	72
4.3	Programme MapReduce . . . . .	73
4.3.1	Code Source : . . . . .	73
	<b>Conclusion générale et perspectives</b>	<b>79</b>
	<b>Bibliographie</b>	<b>80</b>
	<b>Glossaire</b>	<b>82</b>

# Table des figures

1.1	Les fonctionnalités d'un ERP . . . . .	6
1.2	Les taches d'un CRM . . . . .	7
1.3	Fonctionnalités de NoSQL . . . . .	9
1.4	Key-values Stores . . . . .	9
1.5	Document Databases . . . . .	10
1.6	Column Family Stores . . . . .	11
1.7	Graph Databases . . . . .	11
1.8	Écosystème Hadoop [5] . . . . .	14
1.9	l'architecture maitre/esclave [5] . . . . .	14
1.10	la place MapReduce dans l'architecture d'Hadoop [23] . . . . .	15
1.11	L'architecture globale de MapReduce[5] . . . . .	16
2.1	Organigramme de la filiale BATICOMPOS [2] . . . . .	19
2.2	Serveur HP DL380 G7 Proliant [2] . . . . .	22
2.3	Modem ADSL [2] . . . . .	22
2.4	Onduleurs MGE [2] . . . . .	23
2.5	L'armoire de brassage [2] . . . . .	23
2.6	Les jarretières ST/SC [2] . . . . .	24
2.7	Switch Cisco [2] . . . . .	24
2.8	Tiroir optique [2] . . . . .	24
2.9	Convertisseur FIBRIDJ [2] . . . . .	25
2.10	Patch panel [2] . . . . .	25
2.11	Les prises RJ45 [2] . . . . .	25
3.1	Création d'une nouvelle machine virtuelle . . . . .	28
3.2	Choix de type de configuration . . . . .	29
3.3	Choix des fonctionnalités matérielles . . . . .	29
3.4	Choix d'installation de système d'exploitation . . . . .	30
3.5	Sélection de système d'exploitation . . . . .	30
3.6	Nom de la machine virtuelle . . . . .	31
3.7	Configuration des processeurs (CPU) . . . . .	31
3.8	configuration de la taille de la mémoire vive (RAM) . . . . .	32
3.9	configuration du type du réseau . . . . .	32
3.10	configuration de la taille du disque . . . . .	33
3.11	Choix du support d'installation . . . . .	33
3.12	Démarrage de la machine virtuelle "fedora server" . . . . .	34

3.13	Vérification du support d'installation . . . . .	34
3.14	Configuration des rubriques . . . . .	35
3.15	La sélection de disque d'installation . . . . .	35
3.16	Configuration de l'heure et date . . . . .	36
3.17	Saisie du mot de passe d'administrateur . . . . .	36
3.18	Progression de l'installation . . . . .	37
3.19	Démarrage de la machine virtuelle . . . . .	37
3.20	connexion après le démarrage de la machine virtuelle sur navigateur web .	38
3.21	désinstallation des anciennes versions . . . . .	38
3.22	Configuration du référentiel . . . . .	38
3.23	Mise en place de dépôt Docker stable . . . . .	39
3.24	Installation du moteur Docker et ses conteneurs . . . . .	39
3.25	Démarrage de service Docker . . . . .	39
3.26	Lancement de l'image Docker Hello-world . . . . .	40
3.27	Ajout d'un référentiel . . . . .	40
3.28	L'installation des packages . . . . .	41
3.29	L'installation de l'environnement Kubernetes . . . . .	41
3.30	Installation de package OpenJDK . . . . .	42
3.31	Ajout du référentiel Elasticsearch . . . . .	42
3.32	Importation de la clé GPG . . . . .	42
3.33	Mise en place d'Elasticsearch . . . . .	43
3.34	Activation d'Elasticsearch . . . . .	43
3.35	Test sur Elasticsearch . . . . .	44
3.36	Installation de kibana . . . . .	44
3.37	Changement d'utilisateur . . . . .	45
3.38	Configuration basée sur la clé SSH . . . . .	45
3.39	Installation de Hadoop . . . . .	46
3.40	Modification effectuée dans le fichier bashrc . . . . .	46
3.41	Modification de l'environnement Hadoop . . . . .	46
3.42	Modification de la variable . . . . .	47
3.43	Modification réalisées au fichier core-site.xml . . . . .	47
3.44	Modification réalisées au fichier hdfs-site.xml . . . . .	48
3.45	Modification réalisées au fichier yarn-site.xml . . . . .	48
3.46	Modification réalisées au fichier mapred-site.xml . . . . .	49
3.47	La liste résultante des processus Java en cours d'exécution . . . . .	49
4.1	Topologie du réseau existant . . . . .	52
4.2	La topologie du réseau local Baticompos sous GNS3 . . . . .	53
4.3	Importation de l'équipement fortigate . . . . .	54
4.4	choix d'option . . . . .	55
4.5	installation de l'appliance depuis GNS3 . . . . .	55
4.6	Choix de l'équipement . . . . .	56
4.7	Installation de l'appliance sur GNS3 . . . . .	56
4.8	QEMU binary . . . . .	57
4.9	L'affichage des fichiers requis . . . . .	57
4.10	La sélection de la version souhaitée . . . . .	58
4.11	La sélection de l'image de disque virtuel . . . . .	58
4.12	Installation finie . . . . .	59

---

4.13	L'ajout d'une interface pour le réseau virtuel . . . . .	60
4.14	La sélection de l'interface . . . . .	61
4.15	Configuration de la carte réseau . . . . .	61
4.16	Configuration de mot de passe du pare-feu Fortigate . . . . .	62
4.17	Configuration de l'IP du port 1 . . . . .	62
4.18	Vérification de la configuration sur le port1 . . . . .	63
4.19	Accès a l'interface graphique web du pare-feu . . . . .	63
4.20	Création des VLANs . . . . .	64
4.21	Configuration de la liaison entre le switch et pare-feu . . . . .	64
4.22	Mise en place des interfaces . . . . .	65
4.23	Vérification de la configuration des VLANs . . . . .	65
4.24	Ajout de deux interfaces Client et Server . . . . .	65
4.25	Configuration de Vlan 20 . . . . .	66
4.26	Configuration de Vlan 30 . . . . .	66
4.27	Vérification . . . . .	67
4.28	Configuration de la politique du vlan20 au vlan30 . . . . .	67
4.29	Configuration de la politique du vlan30 au vlan20 . . . . .	68
4.30	Installation de vsftpd . . . . .	68
4.31	Démarrage du service vsftpd . . . . .	68
4.32	Changement de répertoire . . . . .	69
4.33	Accès au fichier vsftpd.conf . . . . .	69
4.34	Configuration du fichier vsftpd.conf . . . . .	69
4.35	Réduction des droits sur le fichier de configuration . . . . .	70
4.36	Changement des configuration du boolean . . . . .	70
4.37	Accès au fichier user list . . . . .	70
4.38	Ajout de l'utilisateur hadoop . . . . .	71
4.39	Redémarrage des services vsftpd . . . . .	71
4.40	Transfert du fichier de données . . . . .	72
4.41	Déplacement du fichier de données . . . . .	72
4.42	Création d'un répertoire d'entrée . . . . .	73
4.43	Stockage de fichier de données . . . . .	73
4.44	Vérification du fichier stocké . . . . .	73
4.45	Compilation du programme MapReduce . . . . .	76
4.46	Création d'un fichier jar . . . . .	76
4.47	Exécution du programme MapReduce . . . . .	77
4.48	Le contenu du répertoire de sortie . . . . .	77
4.49	Résultats du programme MapReduce . . . . .	78

# Liste des abréviations

ADSL	asymmetric digital subscriber line
CLI	Command-line interpreter
CMS	Content Management System
CPU	Central Processing Unit
CRM	Customer Relationship Management
ERP	Entreprise Resource Planning
GMAO	Gestion de maintenance assistée par ordinateur
FTP	File Transfer Protocol
GNS	Graphical Network Simulator
HDFS	Hadoop Distributed File System
HSE	Hygiène Sécurité Environnement
HTTP	Hypertext Transfer Protocol Secure
IDS	Intrusion detection System
IOS	Internetwork Operating Systems
IP	Internet Protocol
ISO	International Organization for Standardization
JAR	Java ARchive
JDK	Java Development Kit
KPI	Key Performance Indicator
LAN	Local Area Network
NAS	Network Attached Storage
NAT	Network Address Translation
NIST	National Institute of Standards and Technology
RAM	Random Access Memory
SAN	Storage Area Network
SGBD	système de gestion de base de données
SI	Système d'Information
SSH	Secure Shell
TMS	Transport Management System
USB	Universal Serial Bus
VLAN	Virtual LAN
VM	Virtual Machine
VPN	Virtual Private Network
VSFTPD	Very Secure FTP Daemon
WIFI	Wireless Fidelity
XML	eXtensible Markup Language

# *Introduction générale*

*Stephen Marsland a dit un jour  
" si les données avaient une masse, la terre serait un trou noir "*

Aujourd'hui nous assistons à une explosion du volume des données produites par la forte utilisation de l'informatique ainsi que l'évolution de la technologie qui connaît notamment un changement d'échelle majeur par la taille et la complexité des données qu'elle manipule. Cette perplexité dans la gestion de ces grands volumes de données a donné naissance au terme Big Data. Une grande partie de l'immense quantité de ces données n'est rien d'autre que des données de journal connues sous le champ de "Fichiers logs" générées par la machine.

la gestion des logs est une étape indispensable, elle fournit un aperçu instantané de l'état général des systèmes matériels et logiciels, améliore également l'intelligence économique en fournissant des informations en temps réel sur l'efficacité des applications et des serveurs.. Or que l'utilisation fréquente de multiservices et de microservices, entraîne une augmentation de fichiers journaux ce qui pose de nombreux problèmes de stockage et de traitement en termes de temps et donc de nouvelles formes de menaces peuvent mettre en péril la sécurité de l'ensemble des systèmes d'information.

C'est donc dans cette optique que se situe le sujet de ce projet, il consistera à mettre en place une architecture big data qui permettra de gérer un grand volume de fichiers logs d'une façon plus optimisée et plus rapide en montrant que lorsque le nombre et la variété des formats sont correctement gérés par les solutions Big Data, la stabilité et la sécurité de notre systèmes est facilement plus réalisable, tout en évaluant en temps réel l'impact des modifications et des mises à jour.

Pour parvenir à la réalisation effective de ce projet, il doit être réalisé en deux parties ; théorique et pratique qui seront préparées sur quatre chapitres :

- le premier chapitre "**Les systèmes d'informations au cœur des métiers**" : sera diviser en trois parties dans lesquelles nous présenterons le parc informatique et ses composants dans la première partie, puis dans la deuxième nous étaliserons les différentes applications de gestion orientées métiers, et au final, nous décrirons le mouvement NoSQL et les technologies Big data ...



- le deuxième chapitre "**Introduction du cas d'étude**", sera consacrer a étudier l'organisme d'accueil en premier lieu puis exposer la problématique et la méthodologie de conception suivie en deuxième lieu.
  
- le troisième chapitre "**Mise en place de la solution**"; durant cette étape nous illustrons l'installation des différents logiciels utilisés.
  
- le quatrième chapitre "**Déploiement et exploitation de la solution**", quant à lui, sera réservé à l'émulation de l'infrastructure réseau, Celui-ci, passera en revue l'environnement de travail et les résultats obtenus.

A la fin, une conclusion générale du mémoire sera nécessaire pour récapituler notre travail et évaluer les résultats obtenus, ainsi que quelques perspectives pour l'améliorer.

# Les systèmes d'informations au cœur des métiers

Le système d'information (SI) est au cœur des organisations. Il représente un instrument essentiel pour améliorer leur stratégie et ce ci grâce aux ressources matérielles, logiciels et humaines. Il se révèle donc d'une utilité indispensable pour les acteurs de l'organisation et pour ses différents métiers.

Dans ce chapitre notre objectif est d'introduire les principes des systèmes d'information, pour cela nous allons présenter le parc informatique et ses différents composants en premier, puis les applications de gestion orienté métiers et a la fin Le mouvement NoSQL au sein de l'entreprise.[11]

## 1.1 Parc informatique et ses composants

Afin d'assurer le bon fonctionnement de système d'information dans une entreprise il faut avoir une bonne gestion et suivie des ressources matérielles et logicielles. Ces derniers consistent d'une part à lister et localiser les équipements de l'entreprise, d'autre part à effectuer des tâches de maintenance et d'assistance aux utilisateurs. Cet assemblage de ressources matérielles et logicielles d'une entreprise donnent naissance a ce que on appelle **le parc informatique**.

Ce dernier est placé sous la main de la DSI (direction du SI ) ou des administrateurs réseaux, un responsable de parc informatique est chargé du bon fonctionnement des postes de travail (fixes et mobiles) et doit garantir la disponibilité permanente aux utilisateurs. Il supervise et assure également la maintenance du matériel et l'adapte aux évolutions technologiques ainsi qu'aux besoins de l'entreprise.[7]

### 1.1.1 Parc matériel

Le parc informatique a tendance à être plus large selon plusieurs critères mais souvent il est composé de :

- (a) Le réseau physique et les équipements réseau  
Qui dit réseau dit internet et les équipements qui nous relient sont par exemple les Bornes Wifi (un point d'accès Wifi)et Les Modems etc....

Et qui dit réseau dit forcément sécurité, et pour cela on utilise les firewalls, les routeurs, les IDS (systèmes de détection d'intrusion) et pleins d'autres équipements.

- (b) Les postes de travail fixes ou portables, unités centrales et leurs accessoires ;
- (c) Les différents périphériques telle que les imprimantes, scanner, tablettes graphiques, caméra et micro de visioconférence, disques durs externes etc... ;
- (d) Les appareils mobiles, smartphones et tablettes ;
- (e) Les serveurs de stockage réseaux ; On diffère deux types de stockage réseaux : [8]
  - i. **NAS : Network Attached Storage** (Serveur de stockage en réseau) désigne un périphérique de stockage (généralement un ou plusieurs disques durs) relié à un réseau par un protocole de communication tel que TCP/IP ;
  - ii. **SAN : Storage Area Network** est un réseau spécialisé permettant de partager de l'espace de stockage à une librairie de sauvegarde et à des serveurs.
- (f) Le Cloud Computing

Le Cloud signifie « nuage » et Computing « informatique », est donc l'informatique en nuage pour une traduction uni latérale anglais français. Selon NIST (National Institute of Standards and Technology), *le Cloud Computing est l'accès via un réseau de télécommunication, à la demande et en libre-service, à des ressources informatiques partagées configurables* [14].

## 1.1.2 Parc Logiciel

Dans ce parc nous trouvons :

- Les éditeurs de texte (Word, Excel, ...);
- Progiciel de Gestion Intégré (entreprise ressource planning);
- Les logiciels conçus pour les réseaux et même des logiciels de base de données (MySQL, SGBD, oracle, ...).
- Les systèmes d'exploitation (Windows 7, 8, 10, XP);

## 1.1.3 Les fichiers Logs

Aussi appelé les messages journaux ,Il s'agit des fichiers qui contiennent les différentes informations liées à l'utilisation d'un serveur, d'une application, d'un logiciel ou d'un système informatique. Un fichier log peut contenir certaines données confidentielles sur l'utilisateur telle que l'adresse IP, la configuration des systèmes et la liste des processus . Nous utilisons les fichiers logs afin de connaître les erreurs qui se produisent sur notre ordinateur ou savoir plus sur le fonctionnement de notre système d'exploitation ainsi identifier tous les processus déroulant en arrière-plan de nos actions.

Ils peuvent être classés dans les catégories générales suivantes : [3] [4]

- **Informatif** : Ce type de message est conçu pour informer les utilisateurs et les administrateurs qu'un événement s'est produit.

- **Débogage** : Les messages de débogage sont généralement générés par les systèmes logiciels pour aider les développeurs à dépanner et à identifier les problèmes d'exécution avec le code d'application.
- **Avertissement** : Les messages d'avertissement concernent des situations dans lesquelles des éléments peuvent être manquants ou requis par le système, mais leur absence n'affecte pas le fonctionnement du système.
- **Erreur** : Ces messages sont utilisés pour relayer les erreurs qui se produisent à différents niveaux dans un système informatique ; de nombreux messages d'erreur ne donnent qu'un point de départ pour expliquer pourquoi ils se sont produits. Souvent, une enquête plus approfondie est nécessaire pour trouver la cause première de l'erreur.
- **Alerte** : une alerte indique qu'un événement intéressant s'est produit.

Un message journal contient généralement les informations suivantes :

**Timestamp** : indique l'heure à laquelle le message de journal a été généré.

**Source** : est le système qui a généré le message de journal.

**Données** : Il n'existe pas de format standard pour représenter les données dans les messages de journal. Certaines des données les plus courantes pouvant être trouvées dans les messages de journal sont les adresses IP source et de destination, les ports source et de destination, les noms d'utilisateur, les noms de programme, etc.

## 1.2 Les applications de gestion orientées métiers

Une application métier est une application informatique qui permet de gérer l'activité d'une entreprise, elle permet également d'informatiser et d'automatiser les processus de gestion.

Une entreprise peut développer ses applications personnalisées, ou acheter des applications standards prêtes à être utilisées. Les applications choisies dépendent du besoin de l'entreprise, des différents problèmes qu'ils doivent régler ainsi que d'autres facteurs comme la structure de l'entreprise etc. . .

### 1.2.1 L'urbanisation des S.I

En premier terme, urbaniser ; c'est organiser la transformation progressive et continue du système d'information en le simplifiant, en optimisant sa valeur et en le rendant plus réactif et flexible vis-à-vis des évolutions stratégiques de l'entreprise.

L'urbanisme définit des règles et des démarches en identifiant des processus en accord avec les différentes informations sur le système d'information de l'entreprise (services, applications, bases de données, etc.).

En outre, les bénéfices d'urbanisations sont si nombreux, nous distinguons quelques uns :

- Une meilleure prise de décisions sur l'évolution (risques, valeur apportée. . .) ;
- Une réduction de complexité et renforcement de la cohérence ;
- Une définition claire des rôles et responsabilités des différents acteurs.[17]

## 1.2.2 Les services WEB

Avec l'essor d'Internet, il devient de plus en plus intéressant pour les entreprises d'utiliser un support électronique pour leurs transactions commerciales. Le commerce électronique ne se limite maintenant plus à l'échange de messages. En effet, les entreprises veulent désormais permettre à leurs partenaires d'affaires d'accéder directement aux fonctions de leur système d'information.

Une nouvelle notion a été introduite, puisqu'elles veulent de plus que les différents logiciels de gestion (ERP, CRM...) communiquent entre eux, on l'appelle l'intégration entre différents systèmes d'information. Les Web services sont une solution à cette approche d'intégration. En se basant sur XML et en utilisant Internet comme un canal de communication, ils permettent en théorie de faire communiquer entre elles les entreprises.[20]

## 1.2.3 ERP

Les ERP(Enterprise Resource Planning) en anglais ou progiciels de gestion intégrés (PGI) , sont des progiciels qui couvrent plusieurs fonctions de l'entreprise et partagent une base de données garantissant l'unicité des informations.

Ils visent à répondre aux problèmes posés par des systèmes d'information présentant des redondances, incohérences ou lourdeurs de communication en vue des différentes applications utilisées dans les différents secteurs d'une entreprise.

Un ERP (PGI) permet la gestion comptable et financière, le contrôle de gestion, la gestion des ressources humaines, de production, chaîne logistique et la gestion des achats et des stocks, etc ...



FIGURE 1.1 – Les fonctionnalités d'un ERP

Il garantit l'unicité des informations grâce à la centralisation des données dans une base unique accessible a tous les modules applicatifs, ce qui implique une facilité de détection d'erreurs (dysfonctionnement) et un renforcement de sécurité et une optimisation des processus de gestion. Aussi ,il permet une meilleurs coordination des services vue qu'une seule modification sur un module provoque la mise a jour en temps réel des autres modules liés et une minimisation des coûts (formation et maintenance).

Il existe différents ERP sur le marché actuel comme : ODOO,SAP,SAGE FRP 1000,zoho ,Oracle , NetSuite et autres , sachant que il y a des ERP gratuit et d'autres payant et cher et chaque entreprise choisies son ERP selon les caractéristiques voulue.[17]

## 1.2.4 CRM

Customer Relationship Management ou gestion de la relation client est une plateforme logicielle centralisée qui permet l'identification, la compréhension et un meilleur service offert aux clients, tout en développant le relationnel avec chacun d'entre eux en vue d'améliorer leurs satisfactions et d'optimiser les gains .

En tant qu'outil orienté client, la solution CRM s'adresse avant tout aux Services Marketing, Services de Vente et au Services Après-vente.[20]

Nous citons ci-dessous quelques avantages de CRM :

- Augmenter la satisfaction du client et reconquérir les clients inactifs ;
- Réduction des coûts d'acquisition des nouveaux clients et nouvelles ventes ;
- Une réduction des problèmes clients ;
- Des décisions marketing plus avisées ;
- Redistribuer les moyens vers les clients les plus rentables ;
- Acquérir de nouveaux clients ;
- Automatiser les campagnes marketing, marketing ciblé ;
- Amélioration des processus opérationnels ;
- Avantage concurrentiel.



FIGURE 1.2 – Les tâches d'un CRM

### 1.2.5 GMAO

D'après Messieurs Marc Gabriel et Yves Pimor (Maintenance assistée par ordinateur) ;

*« Un système informatique de management et de la maintenance est un progiciel organisé autour d'une base de données permettant de programmer et de suivre sous 3 aspects (technique, budgétaire, organisationnel), toutes les activités d'un service de maintenance et les objets de cette activité (services, lignes ateliers, machines, équipements, sous-ensembles, pièces, etc.) à partir de terminaux disséminés dans les bureaux techniques, ateliers, magasins et bureaux d'approvisionnement. »* ‘

Autrement dit, GMAO est un logiciel de maintenance permettant la gestion des aspects de la maintenance préventive et corrective ainsi que la fiabilité des équipements.

Cet ancien système a été remplacé par le logiciel **COSWIN**. Celui-ci est la solution intuitive permettant une prise en main rapide et une exploitation efficace. Il assure la sécurisation des accès et des transactions ainsi que la traçabilité des travaux et améliore les prestations de services internes. [16] [11]

### 1.2.6 TMS

Le TMS (Transport Management System) est un système d'information permettant aux industriels et prestataires logistiques d'optimiser la gestion du transport. [6]

La plupart des spécialistes définissent le TMS comme une solution logicielle afin de traiter tous les mouvements de produits finis ou semi-finis externes.[6]

Cet outil informatique offre la possibilité aux différents acteurs de planifier et optimiser le transport en déterminant le mode (terrestre, aérien, ferroviaire ou par voie d'eau) à utiliser en fonction de diverses paramètres tels que la distance ou la desserte. [6]

D'autres problématiques sont prises en compte comme le moyen de transport utilisé (en propre ou affrété), le type de produits (colis, palette, vrac), les conditions de transport (urgence, température, groupage) ou encore le recours au multimodale. [6]

Les TMS proposés par les éditeurs peuvent posséder des fonctionnalités distinctes (comparateur de tarif, outil de calcul de réseau, optimisateurs de tournées,...) mais ont pour la majorité trois fonctions principales :

- l'analyse et l'aide à la décision ;
- le suivi opérationnel des opérations et tournées de transport ;
- le report des différentes opérations.

## 1.3 Le mouvement NoSQL au sein de l'entreprise

### 1.3.1 Qu'est-ce que le NoSQL ?

NoSQL est la combinaison de deux mots ; No et SQL. En fait, le « No » est un acronyme qui signifie « Not only », c'est-à-dire en français, « non seulement » c'est donc une manière de dire qu'il y a autre chose que les bases de données relationnelles.

Le terme NoSQL référence une catégorie de systèmes de gestion de bases de données (SGBD) distribués, conçus pour la plupart dans le but de traiter des jeux de données volumineux dans des délais acceptables pour l'utilisateur,

Aujourd'hui, les bases de données NoSQL sont devenues incontournables, avec de plus en plus d'entreprises qui se tournent vers ces solutions. Selon le rapport Forrester Wave

for Big Data NoSQL, Q3 2016, « le NoSQL n'est plus une option, c'est une nécessité pour les applications de nouvelle génération » [9]

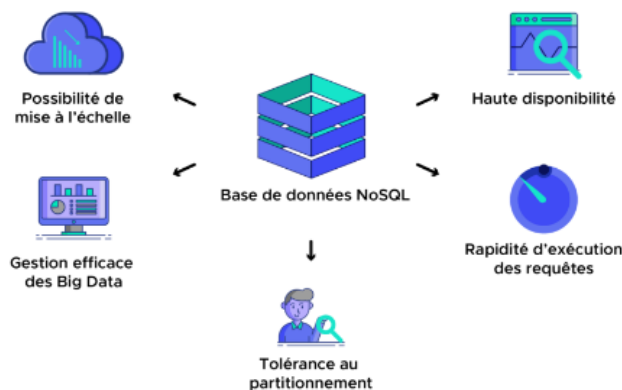


FIGURE 1.3 – Fonctionnalités de NoSQL

### 1.3.2 Les environnements NoSQL

Dans la mouvance NoSQL, il existe une diversité d'approches classées en quatre catégories. Chacune de ces catégories a un attribut unique et des limites spécifiques.

#### (a) Paradigme clé/valeur "Key-values Stores"

le principe de ce système est simple : chaque objet (valeur) est identifié par une clé unique. Celle-ci représente la seule manière de solliciter l'objet. Ce type de base de données est le plus basique. Il permet au développeur de stocker plus facilement des données sans schéma. En guise d'exemples, on peut citer Redis ou Dynamo. [15]

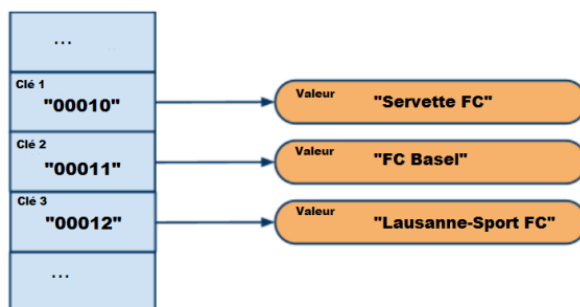


FIGURE 1.4 – Key-values Stores

- (b) **Bases documentaires "Document Databases"** Les systèmes de type documentaire sont composés de collections de documents. La représentation en document est une sorte d'extension du concept clé/valeur. La valeur est représentée sous forme de document contenant des données. Des champs et des valeurs associées composent le document. Ces valeurs associées peuvent soit être de type simple (integer, string, date, ...), soit de type composé, c'est-à-dire de plusieurs couples clé/valeur. [15]



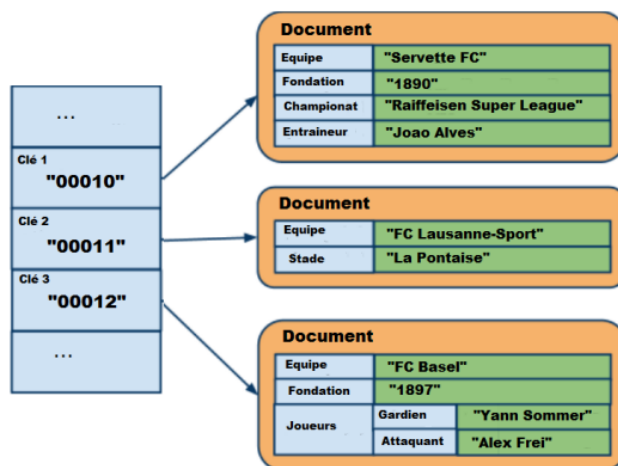


FIGURE 1.5 – Document Databases

Ce type de base de données offre donc une flexibilité accrue. Il est principalement utilisé pour les systèmes CMS, les plateformes de blogging, ou les applications de e-commerce. En revanche, il ne convient pas pour les transactions complexes nécessitant des opérations ou des requêtes multiples sur des structures agrégées variables. Les exemples les plus connus dans cette catégorie sont Amazon SimpleDB, CouchDB, MongoDB..etc [15]

(c) **Bases orientées colonnes "Column Family Stores"**

Repose sur des colonnes. Elles sont basées sur le modèle BigTable de Google. Chaque colonne est traitée séparément, et les valeurs sont stockées de façon contiguë. [15]

Comme il existe différents types de produits orientés colonnes, on trouve donc quelques variations selon le produit considéré. Le concept de base est décrit comme suit : [19]

- **Column** : c'est l'entité de base qui représente un champ de données. Toutes les colonnes sont définies par un couple clé/valeur ;
- **Super column** : c'est une colonne qui contient d'autres colonnes ;
- **Column family** : elle est considérée comme un conteneur de plusieurs colonnes ou super-colonnes ;
- **Row** : c'est l'identifiant unique de chaque ligne de colonne ;
- **Value** : c'est le contenu de la colonne. Cette valeur peut très bien être une colonne elle-même.

En guise d'exemples, on peut citer HBase, Cassandra ou Hypertable.

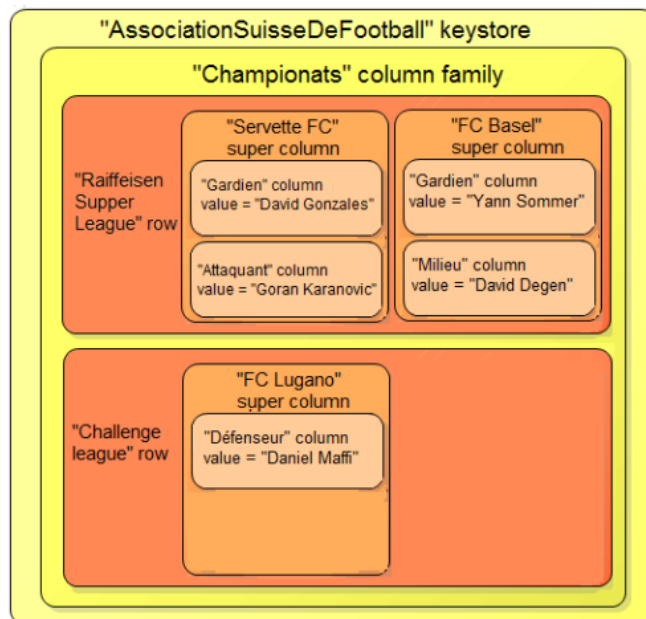


FIGURE 1.6 – Column Family Stores

(d) **Bases orientées graphe "Graph Databases"**

Sont les moins connues de la mouvance NoSQL. Ces bases permettent la modélisation, le stockage ainsi que le traitement de données complexes reliées par des relations. Ce modèle est composé d'un :[19]

- **Moteur de stockage pour les objets** : c'est une base documentaire où chaque entité de cette base est nommé nœud ;
- **Mécanisme qui décrit les arcs** : c'est les relations entre les objets, elles contiennent des propriétés de type simple (integer, string, date..).

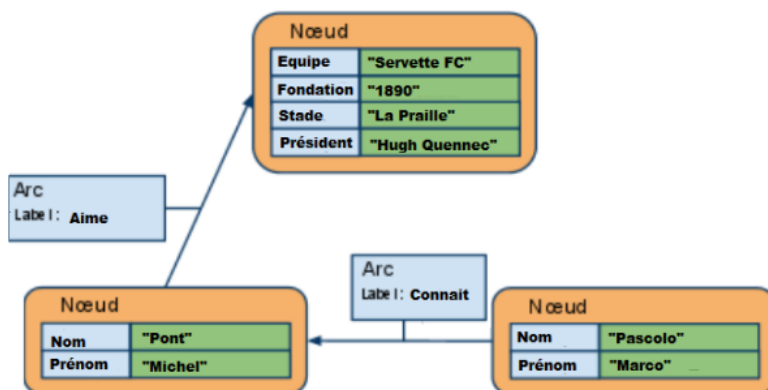
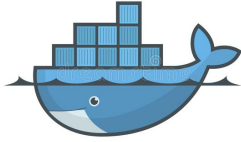


FIGURE 1.7 – Graph Databases

Parmi les exemples les plus populaires, on peut citer Neo4J, Infinite Graph, OrientDB et FlockDB.

### 1.3.3 Docker et kubernetes

#### — Docker



Est un open source qui automatise le déploiement des applications dans des conteneurs. Il a été écrit par l'équipe de Docker, Inc. Docker a pour mission de fournir : [22]

- (a) Une façon simple et légère de modéliser la réalité ;
- (b) Une séparation logique des tâches ;
- (c) l'architecture orientée services ;

Docker suit l'architecture *client-serveur*, qui comprend les trois principaux composants que sont Docker Client , Docker Host et Docker Registry ;[5]

1. **Client Docker** : Comme son nom l'indique, il s'agit d'un client sur le front-end qui est utilisé pour donner des commandes à l'hôte Docker ; ces commandes peuvent être de tout type ;
2. **Docker Host** : est utilisé pour stocker les images créées dans le registre.
3. **Docker Registry** : gère les images Docker. Il existe deux types de registres dans Docker ; Registre public est également appelé hub Docker et Registre privé,est utilisé pour partager des images au sein de l'entreprise.

## — kubernetes



Est un système open source, initialement annoncée par Google en 2014.

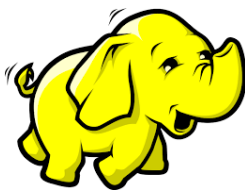
Kubernetes fournit un environnement d'exécution pour le déploiement, l'exécution, gestion et orchestration des conteneurs dans les clusters ou clusters d'hôtes ; [10] Il est constitué de quatre composants : [5]

1. **Pod** : s'agit de la plus petite unité d'une application Kubernetes, composé d'un ou plusieurs conteneurs. Un pod peut contenir plusieurs conteneurs étroitement couplés (cas d'utilisation avancé) ou un conteneur unique (cas d'utilisation plus courant) ;
2. **ReplicaSets** : il est le responsable du démarrage ou l'arrêt d'un pod, il permet également la vérification régulière de l'état de ce dernier ;
3. **Deployments** : est fichier de configuration. Il inclut des déclarations pour les variables d'environnement, les images de conteneurs, les volumes de données. Il contient également des informations relatives au ReplicaSets.
4. **PetSets** : représente un groupe de pods avec état qui nécessitent une notion d'identité plus forte.

## — Différence entre Docker et Kubernetes :

Kubernetes fonctionne en complément de Docker. Alors que Docker permet de gérer le cycle de vie des conteneurs, Kubernetes apporte l'orchestration et la gestion de clusters de conteneurs.

### 1.3.4 Hadoop



Créé par Doug Cutting et fait partie des projets de la fondation logicielle Apache depuis 2009.

Apache Hadoop est un Framework distribué, libre et open source, écrit en Java destiné à faciliter la création d'applications distribuées sur des systèmes en cluster dotés de milliers de noeuds impliquant des centaines de téraoctets de données. [5] [24]

La figure suivante illustre l'architecture globale de l'écosystème Hadoop et la place qu'occupent les différents frameworks ;

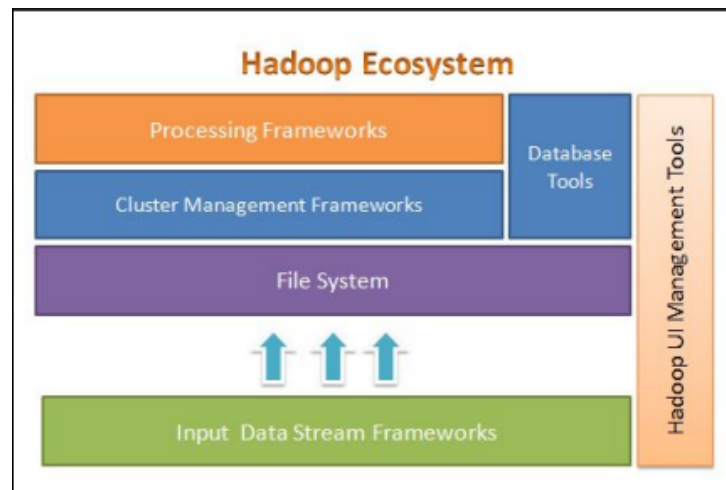


FIGURE 1.8 – Écosystème Hadoop [5]

Le noyau d'Hadoop est constitué de deux parties de stockage : [21]

- (a) **HDFS (Hadoop Distributed File System)** : un système de fichier virtuel distribué. Il s'agit de l'un des composants basiques du framework Hadoop Apache.

Il permet d'augmenter les possibilités de gestion de données du cluster HDFS Hadoop et donc de traiter le Big Data efficacement. Le système est capable de gérer des milliers de nœuds sans intervention d'un opérateur.

Il s'adosse à une architecture maître / esclave autrement dit, HDFS possède deux composants principaux qui fonctionnent essentiellement selon le concept de maître et d'esclave, le premier est NameNode "nœud maître", contient tous les noms et blocs des fichiers. Le deuxième est DataNodes "nœud esclave", Stocke et restitue les blocs de données.

En Effet, NameNode est responsable de la répartition des données sur différents DataNodes en fonction de la configuration définie. Il contient et gère également les métadonnées (les informations relatives aux fichiers stockés dans un système) des blocs de données disponibles sur les différents DataNodes. [5] [13]

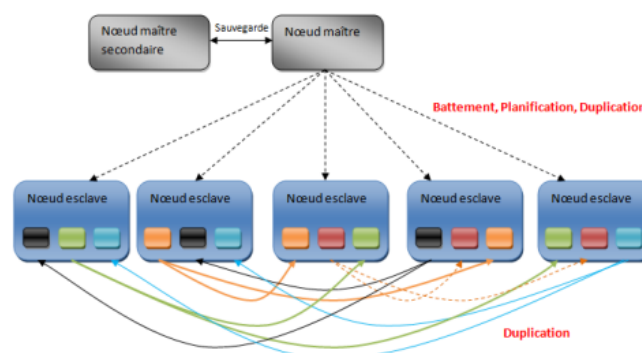


FIGURE 1.9 – l'architecture maître/esclave [5]

- (b) **MapReduce** : est un modèle de programmation disponible dans les environnements Hadoop, créé par Google pour le traitement de très grandes quantités de données.

La figure suivante illustre la place MapReduce dans l'architecture d'Hadoop ; [23]

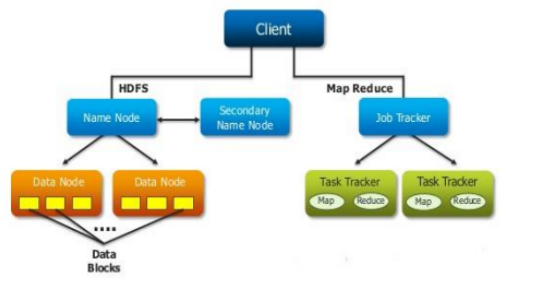


FIGURE 1.10 – la place MapReduce dans l'architecture d'Hadoop [23]

Au cœur de MapReduce se trouvent deux fonctions, Map et Reduce, qui sont séquencées l'une après l'autre. La fonction Map transforme les entrées du disque en paires "key,value", les traite et génère un autre ensemble de paires "key,value" intermédiaires en sortie. La fonction Reduce transforme également les entrées en paires "key,value" et génère une des paires "key,value" en sortie.[23] [18]

MapReduce est composée de deux processus essentiels sont « JobTracker » et « TaskTracker » ; JobTracker est un maître qui crée et exécute le travail. JobTracker qui peut s'exécuter sur le NameNode alloue le travail aux tasktrackers. Il suit la disponibilité des ressources et la gestion du cycle de vie des tâches, suit sa progression et sa tolérance aux pannes...etc. [23] [18]

Quant à TaskTracker, exécute les tâches et signale l'état de la tâche à JobTracker. TaskTracker s'exécute sur DataNodes. Il a pour fonction de suivre les commandes du suivi des travaux et de mettre à jour périodiquement le suivi des travaux avec son état d'avancement. [23] [18]

La figure suivante illustre l'architecture globale incluant le JobTracker et le TaskTracker ; [5]

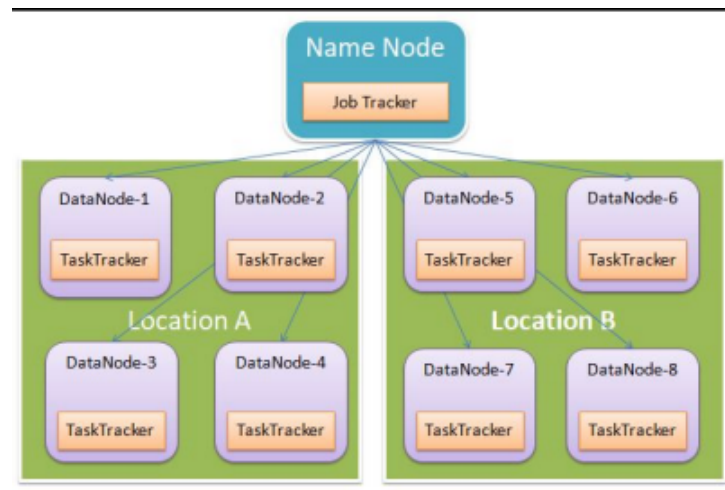


FIGURE 1.11 – L'architecture globale de MapReduce[5]

### 1.3.5 Elasticsearch



Il a été créé par Shay Banon, fondateur de Compass Project (en) en 2004. Elasticsearch est un système open source très puissant construit sur Apache Lucene, hautement évolutif et distribué.

Il est spécialement conçu pour fonctionner efficacement et rapidement avec des systèmes.

l'un de ses principaux cas d'utilisation est l'analyse d'énormes quantités de données en temps quasi réel ce qui signifie que nous pouvons analyser des milliards d'enregistrements en quelques secondes.[12]

### Conclusion

Dans ce chapitre, nous avons introduit et défini les logs tout en donnant un aperçu sur les informations qu'ils peuvent contenir ainsi que les types. Nous avons présenté également le parc informatique et les applications liées à la gestion orientées métiers et en fin du chapitre nous avons entamé la définition du NoSQL et vu les différentes catégories du mouvement ainsi que les différents frameworks qui permettent le traitement distribué de grands ensembles de données.

Dans le prochain chapitre, nous allons mettre en évidence une étude de la société d'accueil

## Introduction au cas d'étude

Le présent projet "Mise en place d'une architecture big data pour la gestion et fouille de fichiers logs" est choisi comme projet de fin d'étude au titre de l'année universitaire 2021/2022, qui a été réalisé au sein de **BATICOMPOS**, société de composants industrialisés, que nous allons présenter dans ce chapitre, en donnant un aperçu sur son architecture, puis nous allons exposer la problématique qui nous a motivé à choisir le sujet en question.

### 2.1 Présentation de la société d'accueil

BATICOMPOS, société de composants industrialisés par abréviation « BATICOMPOS Spa », Est une société à capitaux privés (SPA). Elle est issue de la filialisation du groupe industriel BATIMETAL, et a été acquise par le groupe CEVITAL le 16 décembre 2007. [1]

Elle s'étend sur une superficie de 61 200 m<sup>2</sup>, dont 31 204,70 m<sup>2</sup> de bâtis, l'usine est implantée à 50 Km à l'est de la ville de Bouira, au nœud ferroviaire de Béni Mançour. Son siège social est à Beni Mansour, commune de Boudjellil, Daira de Tazmalt, Wilaya de Bejaia. [1]

BATICOMPOS est le premier fabricant en Algérie du panneau sandwich. Son métier principal est la fabrication et commercialisation des éléments de la construction métallique de meilleure qualité. [1]

#### 2.1.1 Les produits fabriqués

- Panneau sandwich [1]
  - Panneau sandwich de couverture : TL75 ;
  - Panneau sandwich de bardage : LL35, LL 40 ;
  - Panneau sandwich pour usage frigorifique : LL60, LL80, LL100.
- Tôle nervurée [1]
  - Tôle nervurée pour bardage : TR35 ;
  - Tôle nervurée pour couverture : TR66 ;
  - Plancher collaborant : PC55.



- Pièce façonnée [1]
  - pièces d'assemblage et d'esthétique.
- Faux plafond : composé de [1]
  - Lames de sous plafond (LSP et PF) ;
  - Joint de remplissage : JR ;
  - Joint de bord : JB ;
  - Rail de suspension : RS.

BATICOMPOS dispose également d'un atelier spécialisé dans la préfabrication légère et montage, dont : [1]

- Bungalows ;
- Chalets ;
- Cabines sahariennes ;
- Chambres froides ;
- Shelters pour télécommunication.

## 2.2 Organigramme de BATICOMPOS

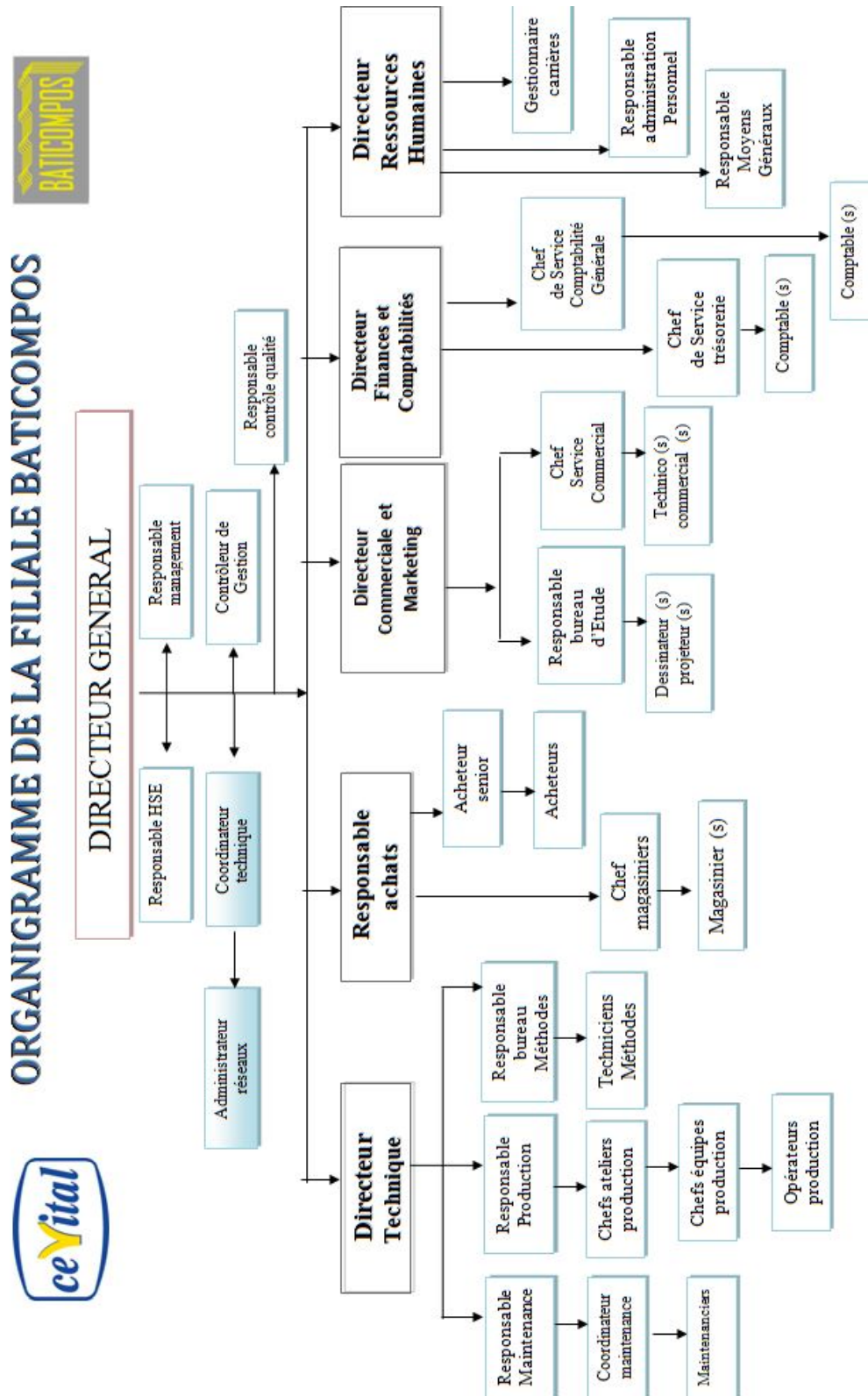


FIGURE 2.1 – Organigramme de la filiale BATICOMPOS [2]

Placés sous l'autorité du directeur général, les différentes structures de Baticompos, se présentent comme suit :[2]

### 2.2.1 Cellule contrôle de gestion

Rattachée directement à la DG, elle a pour tâches :

- Animation, coordination des activités de planification et contrôle de gestion dont l'élaboration du budget et le suivi de réalisation des objectifs.
- Contribution à la définition/L'exécution/L'évaluation de plans, de politiques et projets assignés à l'entreprise, assistant de gestion du Directeur Général, correspondant interne en termes d'information de gestion économique.
- Recherche de l'utilisation rationnelle et optimale des ressources de l'entreprise.
- Planifie et consolide les objectifs et bilans annuels, semestriels, trimestriels et mensuels de l'entreprise, aide à l'élaboration des programmes arrêtés y compris ceux liés à la fonction qualité et à la fonction environnementale.
- Établit le tableau de bord de l'entreprise, collecte les renseignements auprès des autres structures et analyse les résultats de gestion de chaque structure en préconisant des redressements pour se conformer au planning initial.
- Comparer en permanence (contrôle budgétaire) les résultats réels et les prévisions chiffrées figurant aux budgets afin de rechercher la (ou les) cause (s) d'écarts, informer les différents niveaux hiérarchiques, prendre les mesures correctives éventuellement nécessaires et apprécier l'activité des responsables budgétaires.
- Audit des différentes fonctions pour éviter tout dysfonctionnement de l'entreprise et présente les tableaux de bords devant le conseil de coordination.

### 2.2.2 Service contrôle qualité

A pour mission de veiller à la qualité des produits fabriqués et la conformité des matières premières achetées.

### 2.2.3 Service HSE

Son rôle est d'assurer la sécurité des personnes et des biens en milieu professionnel.

### 2.2.4 Cellule informatique

Elle est chargée d'assurer le fonctionnement et la cohérence du système informatique ainsi que la sauvegarde des données et d'assister les structures en matière d'informatique.

### 2.2.5 Direction technico-commerciale et marketing

qui est chargée de :

- L'identification des attentes et besoins des clients et de leur traduction en exigence
- Conseiller les clients dans la définition des exigences qualité relatives aux produits ;
- L'écoute client et du marché ;
- Le marketing et la promotion des produits de l'entreprise ;
- La commercialisation des produits de l'entreprise (placement des commandes, facturation, recouvrement, . . .).

### 2.2.6 Direction approvisionnements

Qui a pour missions de :

- Mettre des matières premières à la disposition de la production, les pièces de rechange à la disposition de ma maintenance et de l'achat des fournitures nécessaires au bon fonctionnement de l'entreprise ;
- Exprimer les besoins d'achat ;
- Consulter, évaluer et sélectionner les fournisseurs ;
- Réceptionner les matières et fournitures achetées
- Assurer la gestion des stocks et l'administration des commandes.

### 2.2.7 Direction technique

cette direction comporte à la fois deux sous structures qui sont la production et la maintenance.

#### (a) Production

Elle a pour missions principales :

- Planifier et réaliser les produits en conformité avec les commandes et les exigences des clients exprimées auprès de la direction commerciale ;
- Mettre en œuvre le processus de production dans les meilleures conditions de maîtrise des procédés et des équipements ;
- Améliorer les techniques et standards de production.

#### (b) Maintenance

Elle a pour missions :

- L'élaboration et la mise en œuvre d'un planning de maintenance préventive ;
- L'élaboration et la tenue à jour de la documentation technique (fiches techniques, fiches historiques, . . .) ;
- La maintenance curative (en temps de panne ou d'arrêt).

### 2.2.8 Direction des ressources humaines

Qui a pour missions :

- Gérer et développer les ressources humaines ;
- Gérer le patrimoine et les moyens matériels de l'entreprise ;
- Veiller au respect de la réglementation relative aux relations de travail, au respect du règlement intérieur et à la mise en œuvre de la convention de l'entreprise ;
- Mettre en œuvre un processus de recrutement, d'évaluation des compétences et de la formation.

### 2.2.9 Direction de la comptabilité et des finances

Elle a pour missions :

- L'établissement des plans de financement et le suivi de leur exécution ;
- La gestion de la trésorerie ;
- La tenue de la comptabilité et l'établissement des bilans et de comptes de résultats ;
- L'établissement et la maîtrise des coûts de réalisations des produits.

Cette organisation est complétée par des fiches de postes pour l'ensemble des postes de travail qui compte l'entreprise

## 2.3 Caractéristique des équipements réseau de l'entreprise

Le réseau de BATICOMPOS est un réseau local qui relie toutes les unités de l'entreprise entre elles ; il comprend les équipements suivants : [2]

### 2.3.1 Serveur HP DL380 G7 Proliant

Ce produit assure un service sans faille dans la mesure où il est doté d'un processeur Intel Xeon quad core 3.6 Ghz, d'une mémoire ram ddr de grande capacité, et d'un disque de stockage compatible système sas, sata ou SFF.



FIGURE 2.2 – Serveur HP DL380 G7 Proliant [2]

### 2.3.2 Modem

Un modem est un boîtier qui se branche à l'ordinateur et qui permet de se connecter à internet, il sert à moduler et à démoduler les informations pour pouvoir les transporter (entre internet et l'ordinateur). L'entreprise BATICOMPOS utilise un modem ADSL de marque TZE et un modem WIFI.



FIGURE 2.3 – Modem ADSL [2]

### 2.3.3 Onduleur

Un onduleur est un dispositif permettant de protéger le matériel électronique contre les aléas électronique. Il s'agit ainsi d'un boîtier placé en interface entre le réseau électronique (branché sur le secteur) et les matériels à protéger .L'onduleur permet de basculer sur une batterie de secours pendant quelques minutes en cas de problèmes électrique. Le réseau de BATICOMPOS utilise des onduleurs MGE évolution 800 VA



FIGURE 2.4 – Onduleurs MGE [2]

### 2.3.4 La passerelle

La passerelle est un ordinateur qui se trouve dans la salle machine de l'entreprise. Elle possède deux cartes réseaux, l'une est pour le réseau local et l'autre permet d'effectuer le branchement avec le réseau mondial, autrement dit elle permet de brancher le réseau local de l'entreprise au réseau internet.

### 2.3.5 L'armoire de brassage

L'entreprise possède des armoires de brassage de types 32U, une dans la salle machine et 6 autres de petit format se trouvent dans les unités suivants : gestion des stocks, les approvisionnement, ateliers panneau sandwich, la production mole, le commerciale est une dernière au magasin de pièce de recharge.



FIGURE 2.5 – L'armoire de brassage [2]

Il contient les équipements suivants :

(a) **Les jarretières ST/SC des ports pour la fibre optique**

Utilisés pour réaliser une connexion au niveau du panneau de brassage. Ces jarretières ont été conçus pour :

- i. Câblage informatique et réseau locaux (LAN)
- ii. Transmission de données et contrôle industriel.



FIGURE 2.6 – Les jarretières ST/SC [2]

(b) **Les Switch Cisco**

Ayant les caractéristiques suivantes

- i. 24 ports haute vitesse optimisés pour le cœur du réseau ou pour prendre en charge des applications gourmandes en bande passante
- ii. Qualité de service améliorée assurant une expérience réseau cohérente et prenant en charge des applications réseaux ,telles que la voix ,la vidéo et le stockage des données, sécurité renforcée protégeant le trafic réseau pour que les utilisateurs non autorisés ne puissent pas pénétrer.



FIGURE 2.7 – Switch Cisco [2]

(c) **Tiroir optique**

Il comporte un certain nombre de connecteurs fibres qui sont reliés aux équipements actifs



FIGURE 2.8 – Tiroir optique [2]

(d) **Les convertisseurs**

Les convertisseurs utilisés dans l'entreprise sont de type FIBRIDJ. Leur rôle est de convertir le signal lumineux en signale analogique.



FIGURE 2.9 – Convertisseur FIBRIDJ [2]

- (e) **Tableau de brassage** Appelé aussi panneau ou « patch panel » en anglais. Il est le point où se concentrent tous les câbles de chaque prise murale de l'entreprise.

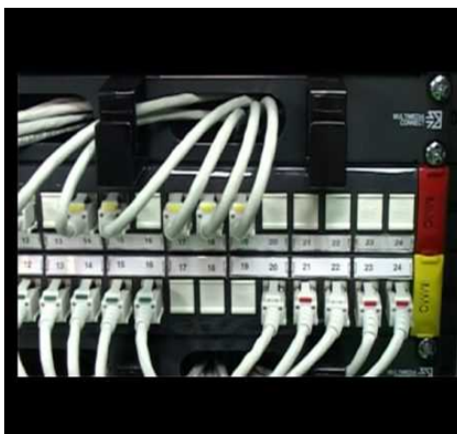


FIGURE 2.10 – Patch panel [2]

### 2.3.6 Les prises

Les prises utilisées dans l'installation de réseau de BATICOMPOS sont de types RJ45. Ils comportent huit broches de connexion électrique.

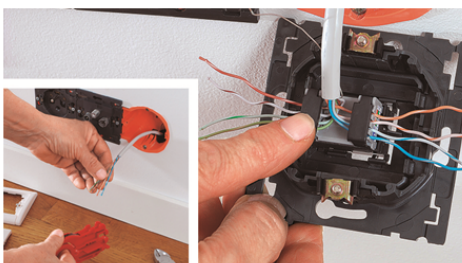


FIGURE 2.11 – Les prises RJ45 [2]



### 2.3.7 Les câbles

Pour relier les diverses entités de réseau de BATICOMPOS deux types de câbles sont utilisés :

- (a) La paire torsadée
- (b) La fibre optique

## 2.4 Cas d'étude : Baticompos SPA

### 2.4.1 La problématique

Depuis la révolution numérique, la quantité de données logs générées et stockées dans ou en dehors d'un système d'information ne cesse d'augmenter. Le développement et l'accès à ces données a conduit à l'apparition du terme Big Data.

Cependant, il est difficile de continuer à utiliser les outils traditionnels pour les exploiter et les manipuler de façon performante a cause des limitations technologiques et la rareté des compétences clé nécessaires a ce type d'opérations.

D'où l'idée du présent projet est de mettre en place une approche exploitant les outils de big data en vue de traiter, stocker et restituer ces données sous forme de KPI.

L'approche répondra a une triple problématique :

1. Un Volume de données important à traiter ;
2. Une grande Variété d'informations.
3. L'analyse des données en temps réel ;

### 2.4.2 La solution proposée

L'objectif de notre étude est de proposer un modèle parallèle et distribué ce qui nécessite d'utiliser des plateformes et outils dédiées à la gestion de ces données parmi lesquels la plateforme Hadoop, le choix de ce dernier se justifie principalement par le fait qu'il permet d'effectuer des traitements et des calculs complexes sur des données très volumineux. Il est composé de deux composants essentiels à savoir : MapReduce qui est un nouveau paradigme de programmation sur lequel sont effectués les calculs parallèles et distribués de grandes masses de données et HDFS qui est un système de gestion de fichier distribué.

De ce fait, différents challenges apparaissent, pour les accomplir les étapes suivantes doivent être réalisées :

1. Récupération des fichiers logs ;
2. stockage des fichiers dans le HDFS ;
3. création d'un programme mapreduce pour réaliser les différentes opérations telle que la gestion et la fouille dans les données stockées dans le système de fichiers Hadoop.

## **Conclusion**

Ce chapitre, s'est concentré en premier lieu sur l'étude de l'établissement d'accueil. Nous sommes passé ensuite à l'exposition de la problématique et la solution proposée pour la résoudre.

Dans le prochain chapitre, nous allons mettre en place la solution.

## Mise en place de la solution

Dans ce chapitre, une présentation des étapes d'installation des différents environnements utilisés sera nécessaire afin de réaliser les objectifs de notre projet.

### 3.1 Mise en place de la solution

#### 3.1.1 Installation et configuration de fedora server sur vmware workstation

la figure ci-dessous représente le premier écran lorsque VMware Workstation 16 s'ouvre pour créer une nouvelle machine virtuelle.

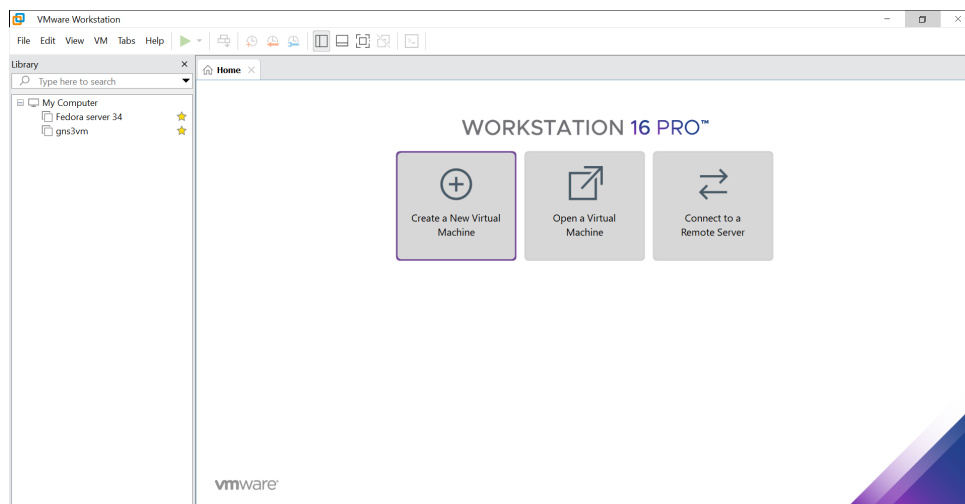


FIGURE 3.1 – Création d'une nouvelle machine virtuelle

Cette nouvelle fenêtre qui apparaît consiste à créer la machine virtuelle "Fedora server 34". Nous sélectionnons l'option "custom (advanced)" puis "Next".

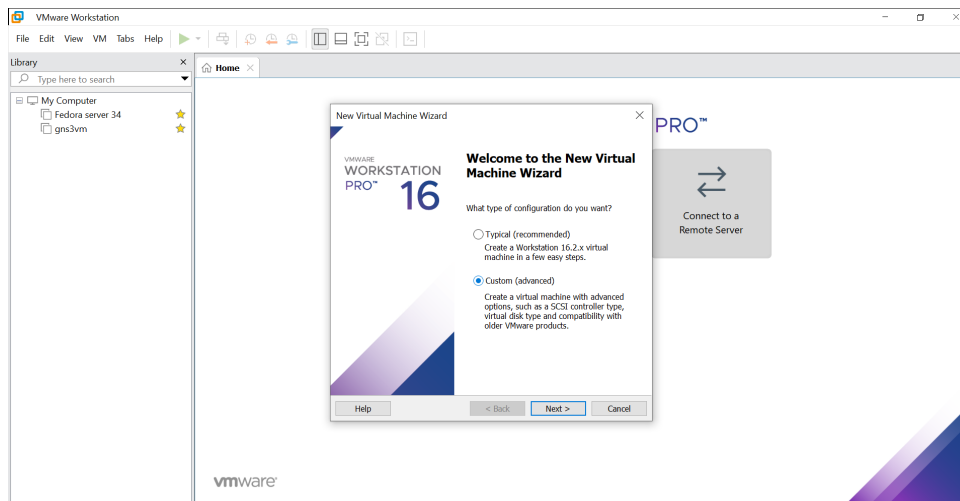


FIGURE 3.2 – Choix de type de configuration

Afin de poursuivre la procédure, nous cliquons sur le bouton "Next". Nous choisissons les fonctionnalités matérielles qui sont nécessaires pour notre machine virtuelle, dans notre cas, c'est l'option "Workstation 16,2x".

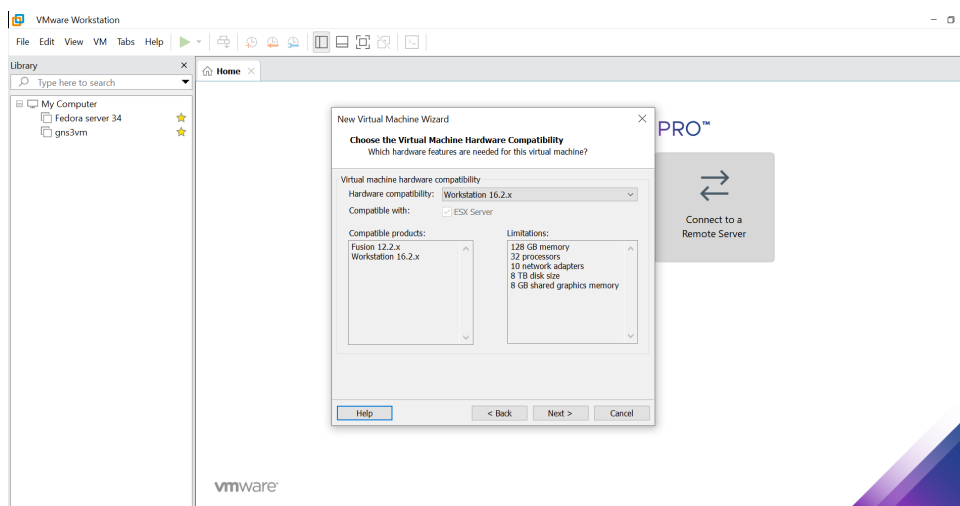


FIGURE 3.3 – Choix des fonctionnalités matérielles

Dans cette étape, nous devons sélectionner l'option de fichier d'image disque du programme d'installation (.iso). Dans notre cas, nous choisissons l'option "i will install the operating system later" puis "Next".

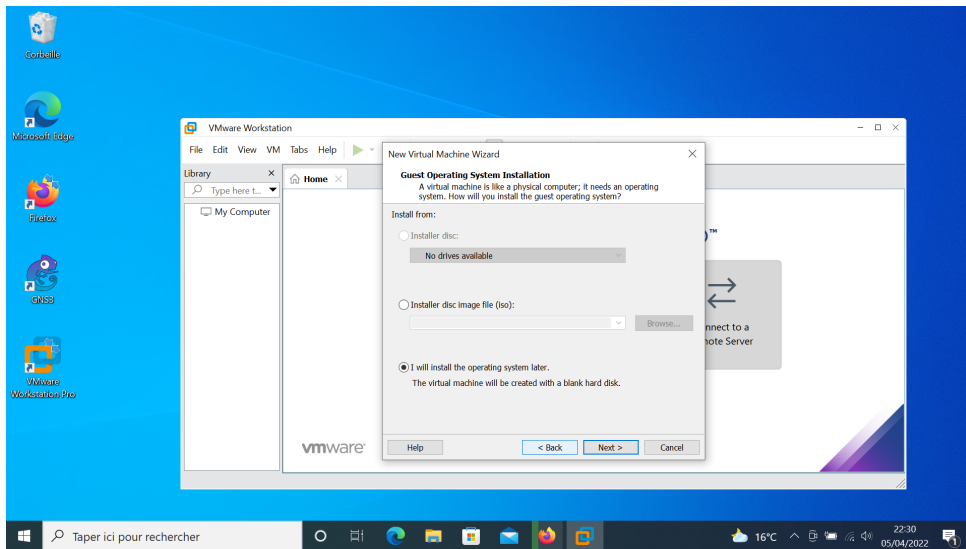


FIGURE 3.4 – Choix d'installation de système d'exploitation

Ensuite, nous sélectionnons "Linux" pour le système d'exploitation et "Fedora 64 bits" pour la version, comme la figure ci-dessous le montre.

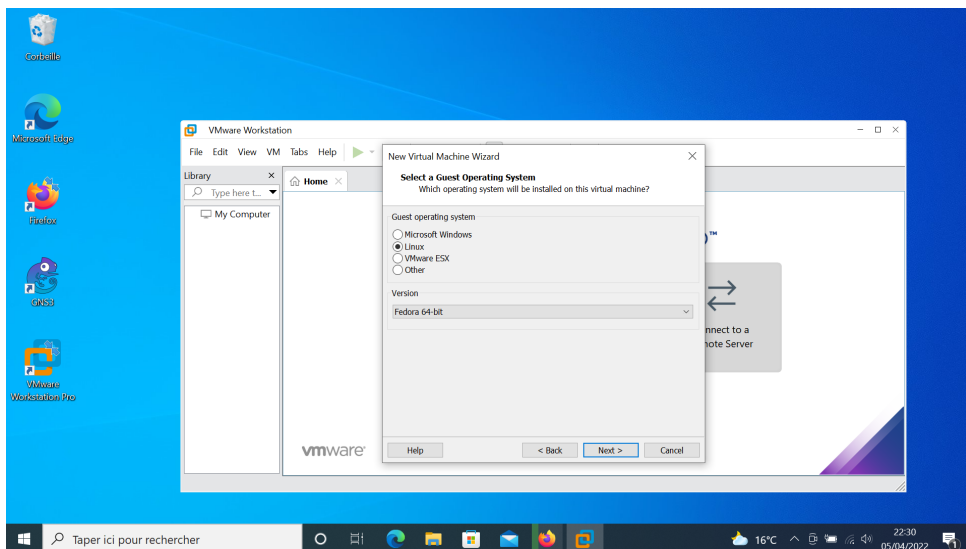


FIGURE 3.5 – Sélection de système d'exploitation

Nous définissons ensuite le nom et le chemin où la machine virtuelle Fedora server34 doit être stockée et cliquons à nouveau sur "Next".

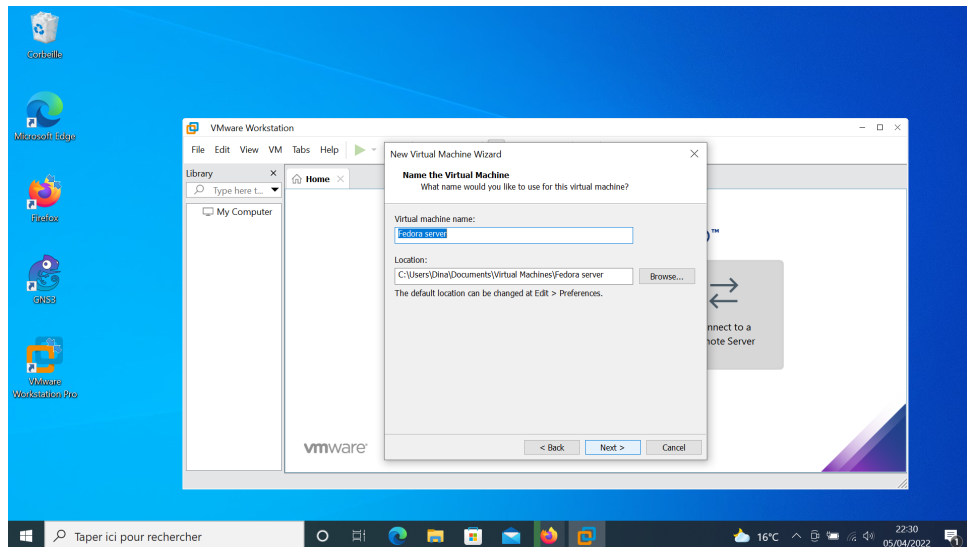


FIGURE 3.6 – Nom de la machine virtuelle

Nous configurons les processeurs (CPU) pour notre machine virtuelle et cliquons sur le bouton "Next" pour l'étape suivante.

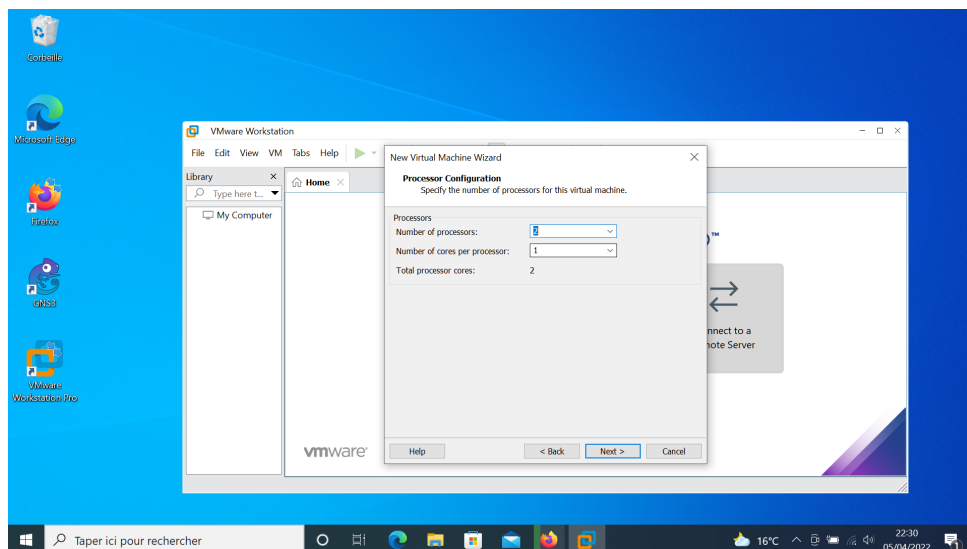


FIGURE 3.7 – Configuration des processeurs (CPU)

la fenêtre suivante nous permet de configurer la taille de la mémoire vive (RAM) de la machine virtuelle, nous lui attribuons 4GB.

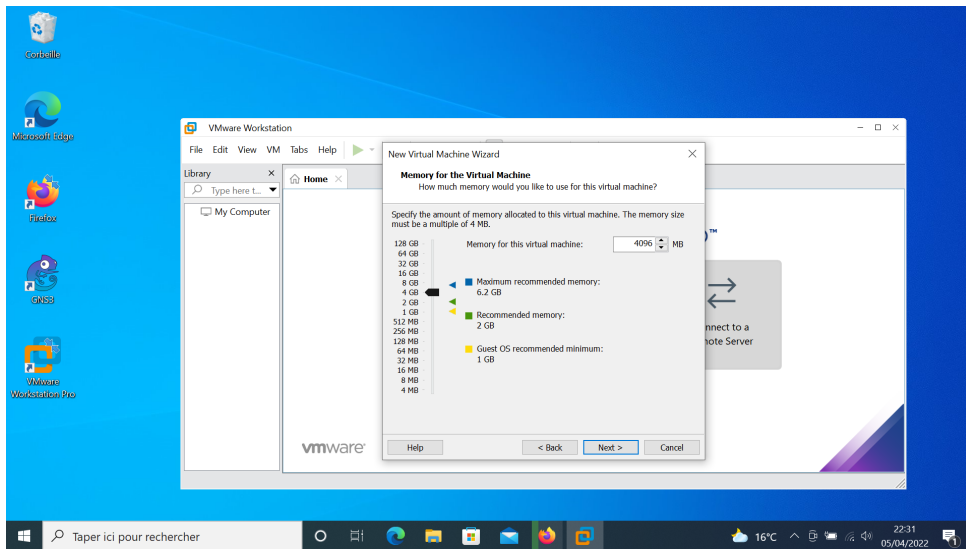


FIGURE 3.8 – configuration de la taille de la mémoire vive (RAM)

Maintenant, nous choisissons le type de réseau (NAT) pour connecter notre machine virtuelle a internet dès que la machine physique se connecte.

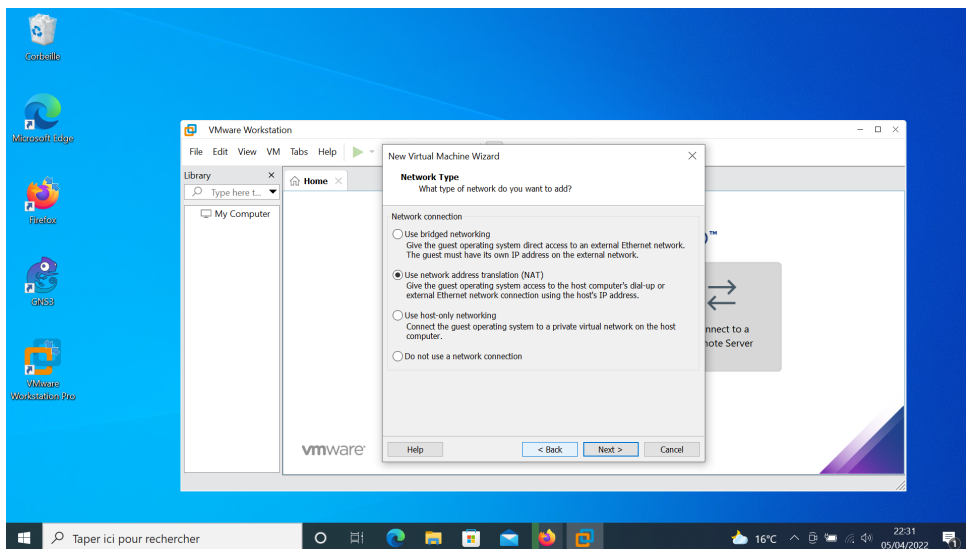


FIGURE 3.9 – configuration du type du réseau

Nous attribuons la taille au disque dur et le type de stockage

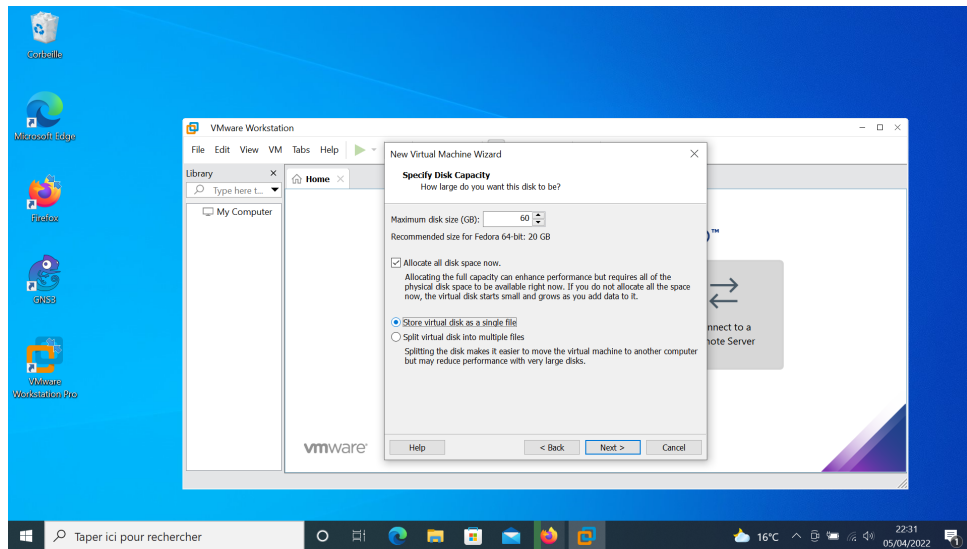


FIGURE 3.10 – configuration de la taille du disque

Une fois que ces étapes sont réalisées, nous passons à l'insertion de l'emplacement de notre image du serveur Fedora puis nous cliquons sur "OK" pour terminer la configuration.

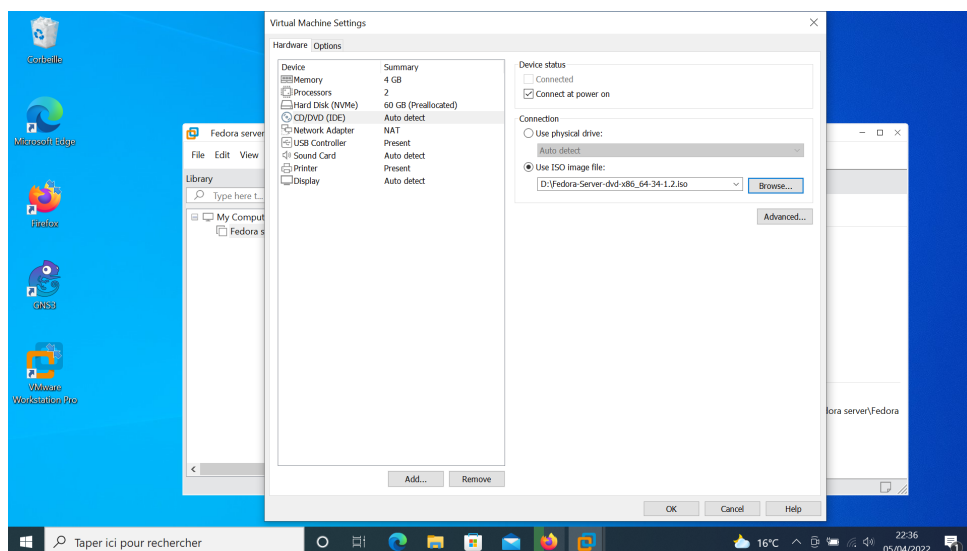


FIGURE 3.11 – Choix du support d'installation

La machine virtuelle est prête à démarrer le processus. L'écran ci-dessous représente le début d'une nouvelle machine virtuelle avec l'image ISO de Fedora server. Dans cette fenêtre, nous sélectionnons l'option "Install Fedora 34".



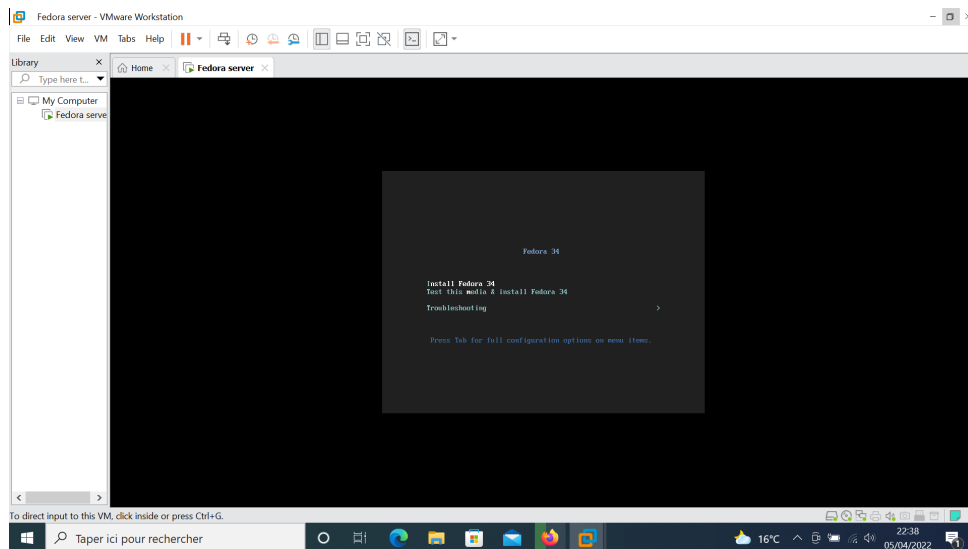


FIGURE 3.12 – Démarrage de la machine virtuelle "fedora server"

cette fenêtre nous affiche le processus d'installation qui consiste à vérifier la configuration système de base avant de l'installer localement

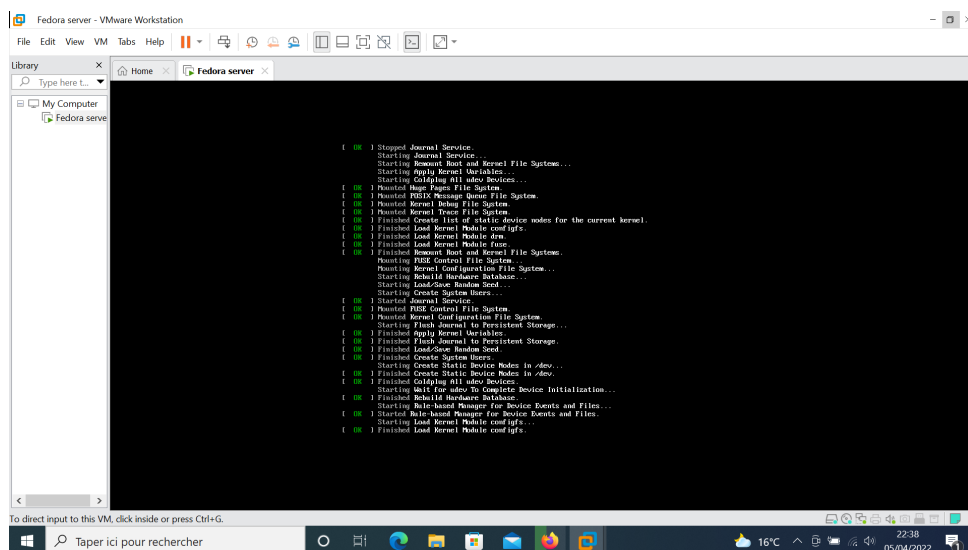


FIGURE 3.13 – Vérification du support d'installation

Une fois que la vérification est terminée, la fenêtre au-dessous apparaîtra. A partir de là, l'installation est parallèle, nous accédons aux rubriques " Destination de l'installation", "mot de passe administrateur" et "heure et date" pour faire les configurations nécessaires.

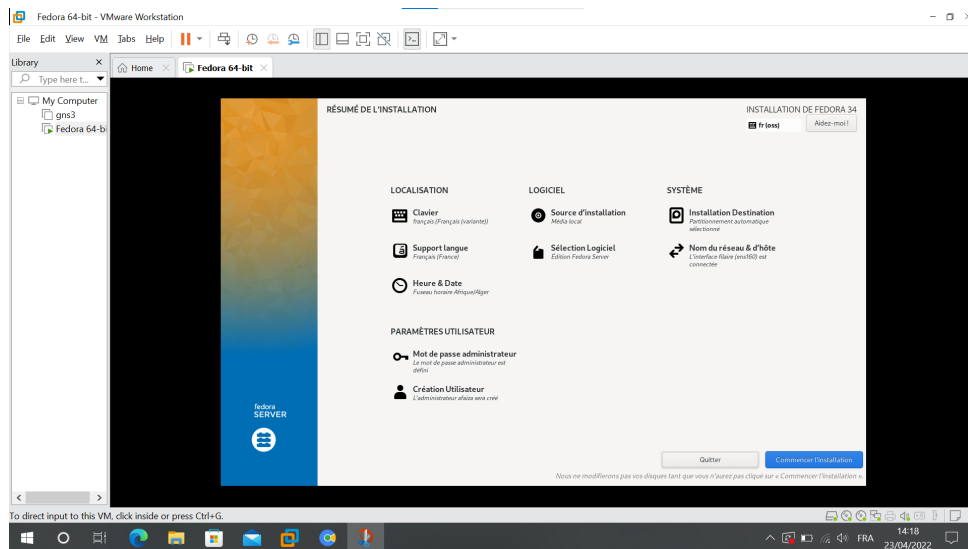


FIGURE 3.14 – Configuration des rubriques

En cliquant sur "Destination de l'installation", nous aboutissons sur la page qui nous permet de définir sur quel disque et partition Fedora sera installé comme la figure ci-dessous le montre.

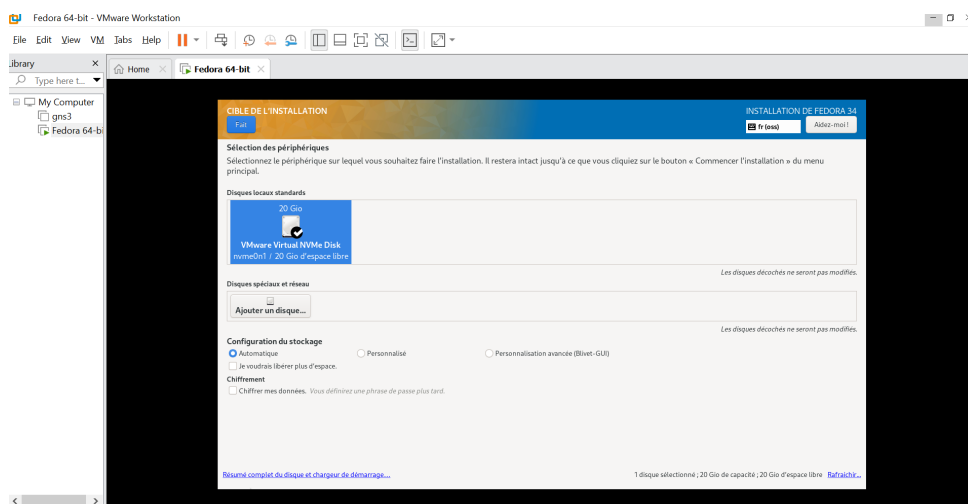


FIGURE 3.15 – La sélection de disque d'installation

Après avoir achevé le partitionnement et cliqué sur le bouton "Fait", nous passons à la configuration de "Heure et date". Cette étape consiste à définir le fuseau horaire, l'heure et la date par défaut.

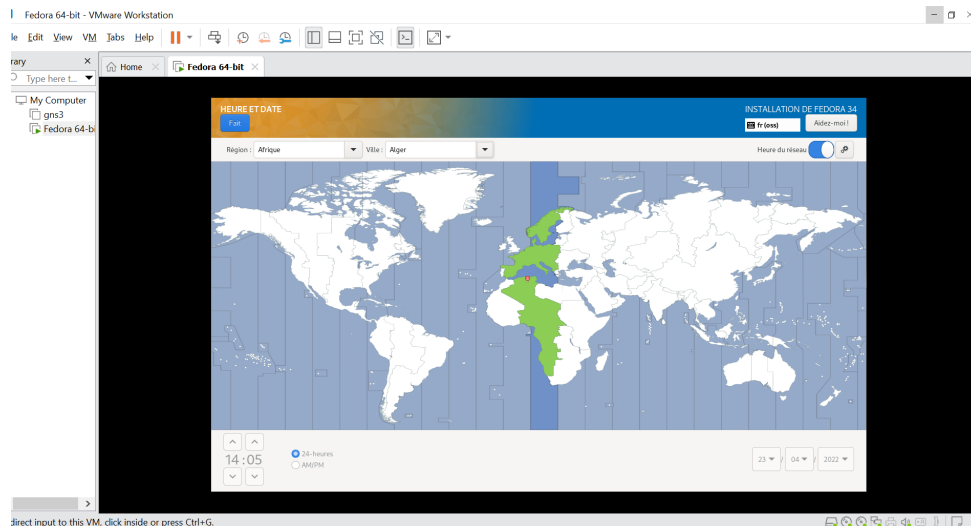


FIGURE 3.16 – Configuration de l’heure et date

Dans cette étape, nous saisissons un mot de passe pour l’administrateur.

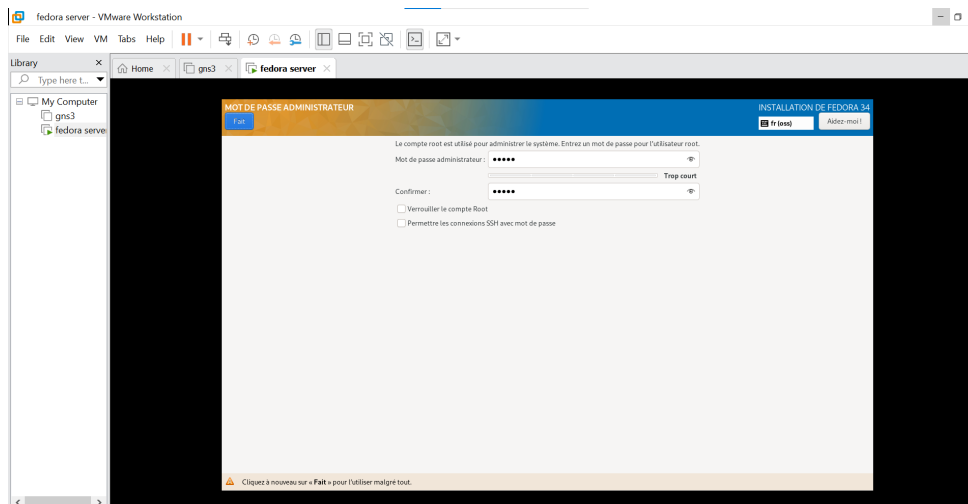


FIGURE 3.17 – Saisie du mot de passe d’administrateur

Toutes les rubriques sont complétées, nous pouvons maintenant lancer l’installation en cliquant sur le bouton ”Commencer l’installation”.

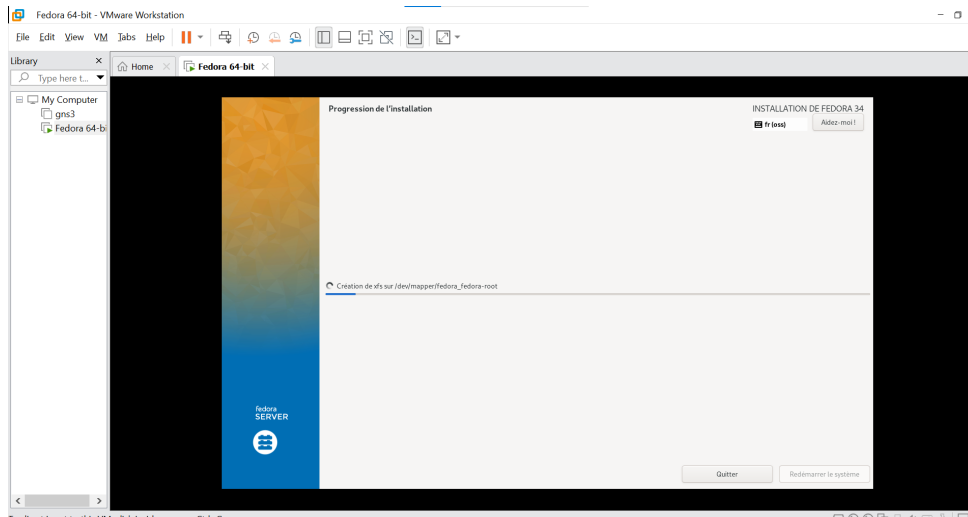


FIGURE 3.18 – Progression de l'installation

Une fois que l'installation est finie, nous cliquons sur "redémarrer le système".

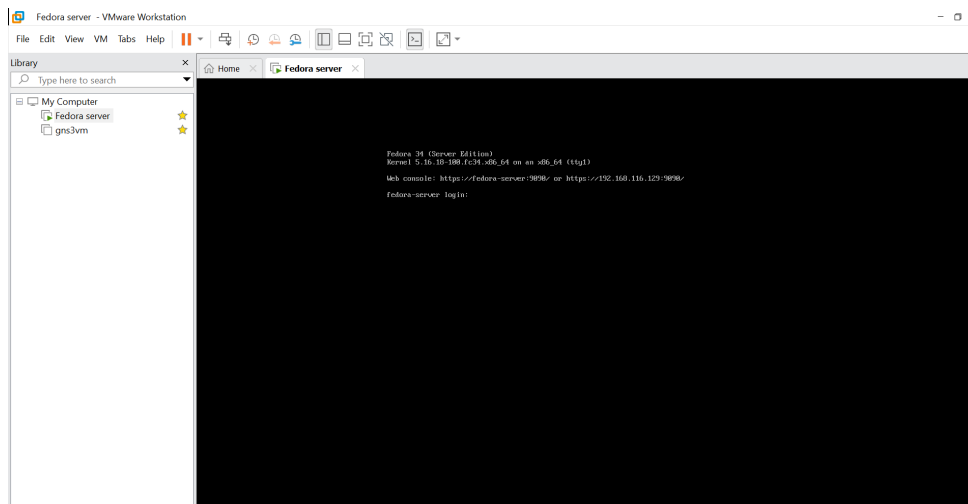


FIGURE 3.19 – Démarrage de la machine virtuelle

Après cela, nous nous connectons au serveur Fedora 34 via notre navigateur en copiant le web console affiché sur la figure ci-dessus. Nous obtenons l'écran de connexion suivant qui est la page d'accueil de fedora server 34. Pour accéder au système nous utilisons le nom d'utilisateur "root" et le mot de passe configurés.

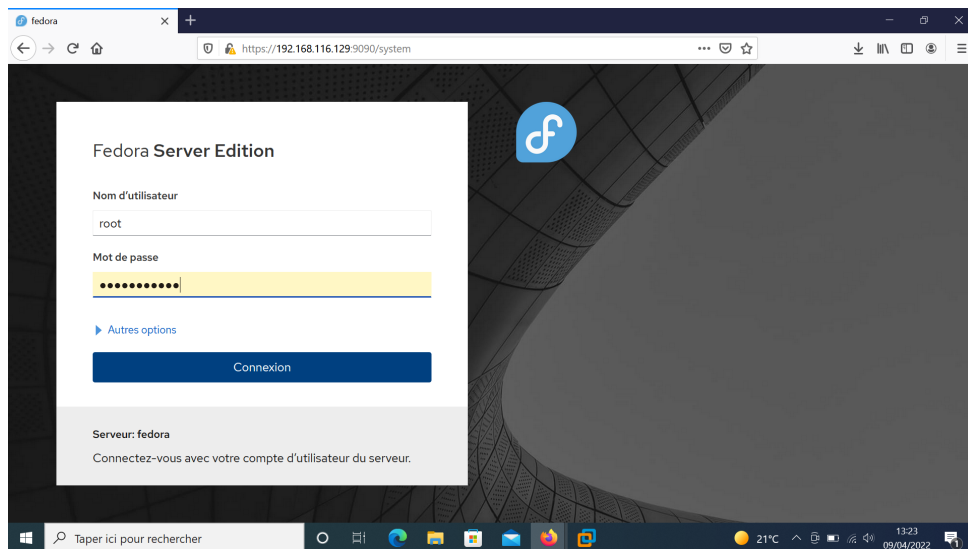


FIGURE 3.20 – connexion après le démarrage de la machine virtuelle sur navigateur web

### 3.1.2 Installation de Docker sur Fedora

- **Étape 1** : Avant de commencer l'installation de docker, nous désinstallons Les anciennes versions avec la commande dans la figure ci-dessous ;

```
[root@fedora ~]# sudo dnf remove docker
Aucune correspondance pour le paramètre: docker
Aucun paquet marqué pour suppression.
Dépendances résolues.
Rien à faire.
Terminé !
```

FIGURE 3.21 – désinstallation des anciennes versions

Comme c'est OK, nous passons à l'étape suivante.

- **Étape 2** : Nous configurons le référentiel Docker. Ensuite, nous installons et mettons à jour Docker à partir de ce référentiel.

```
[root@fedora ~]# sudo dnf -y install dnf-plugins-core
```

FIGURE 3.22 – Configuration du référentiel

Cette commande permet de mettre en place le dépôt Docker stable.

```
[root@fedora ~]# sudo dnf config-manager --add-repo https://download.docker.com/linux/fedora/docker-ce.repo
Ajout du dépôt depuis : https://download.docker.com/linux/fedora/docker-ce.repo
[root@fedora ~]#
```

FIGURE 3.23 – Mise en place de dépôt Docker stable

- **Etape 3** : Cette étape est dédiée à installer le moteur docker et ses conteneurs.

```
root@fedora:~
Font size - 16 + Apparence Noir Réinitialiser

[root@fedora ~]# sudo dnf install docker-ce docker-ce-cli containerd.io docker-compose-plugin
Docker CE Stable - x86_64 10 kB/s | 16 kB 00:01
Dernière vérification de l'expiration des métadonnées effectuée il y a 00:01 le lun. 09 mai 2022 15:17:36.
Dépendances résolues.
=====
Paquet Architecture Version Dépôt Taille
=====
Installation:
containerd.io x86_64 1.6.4-3.1.fc34 docker-ce-stable 32 M
docker-ce x86_64 3:20.10.15-3.fc34 docker-ce-stable 21 M
docker-ce-cli x86_64 1:20.10.15-3.fc34 docker-ce-stable 29 M
docker-compose-plugin x86_64 2.5.0-3.fc34 docker-ce-stable 6.8 M
Installation des dépendances:
container-selinux noarch 2:2.158.0-1.gite78ac4f.fc34 fedora 49 k
docker-ce-rootless-extras x86_64 20.10.15-3.fc34 docker-ce-stable 3.8 M
docker-scan-plugin x86_64 0.17.0-3.fc34 docker-ce-stable 3.7 M
fuse-overlayfs x86_64 1.7.1-2.fc34 updates 72 k
fuse3 x86_64 3.10.4-1.fc34 updates 54 k
fuse3-libs x86_64 3.10.4-1.fc34 updates 91 k
libgroup x86_64 0.42.2-4.fc34 fedora 66 k
libslirp x86_64 4.4.0-4.fc34 updates 68 k
slirp4netns x86_64 1.1.12-2.fc34 updates 55 k
```

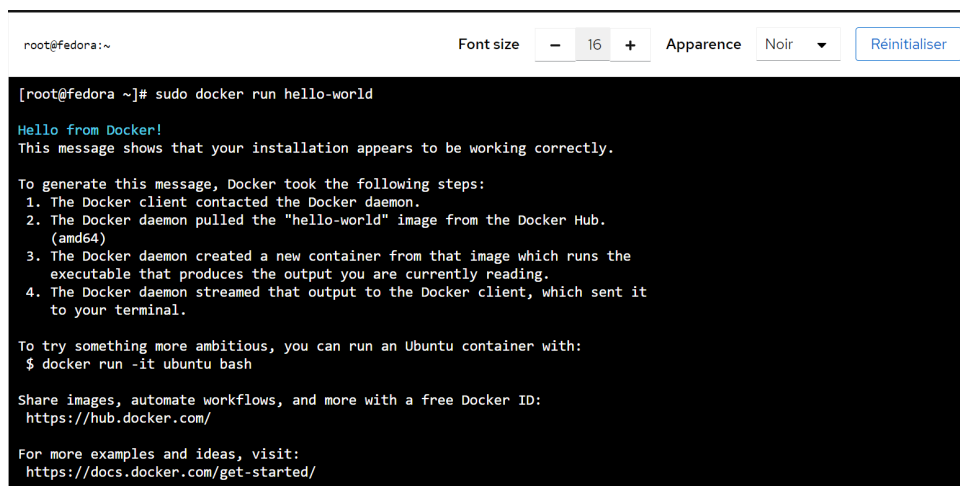
FIGURE 3.24 – Installation du moteur Docker et ses conteneurs

Une fois que le moteur est installé avec succès, nous exécutons les commandes suivantes pour démarrer le service Docker

```
[root@fedora ~]# sudo systemctl start docker
[root@fedora ~]#
```

FIGURE 3.25 – Démarrage de service Docker

- **Etape 4** : Cette étape consiste à démarrer Docker en utilisant la commande **sudo systemctl start docker** et vérifier qu'il est correctement installé en exécutant "hello-world".



```

root@fedora:~
Font size - 16 + Apparence Noir Réinitialiser

[root@fedora ~]# sudo docker run hello-world

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

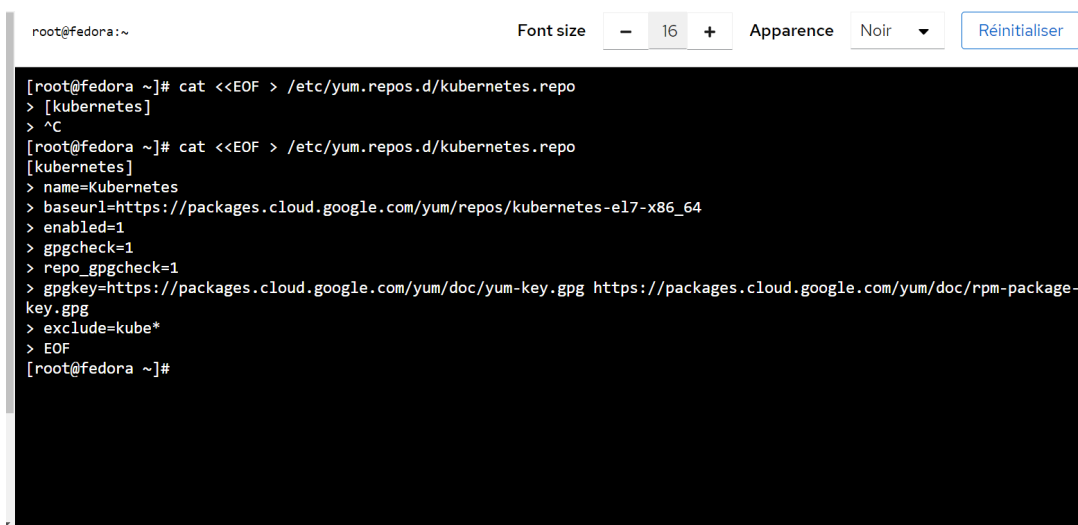
For more examples and ideas, visit:
https://docs.docker.com/get-started/

```

FIGURE 3.26 – Lancement de l’image Docker Hello-world

### 3.1.3 Installation de kubernetes sur Fedora

- **Étape 1** : Dans cette étape, nous créons et configurons un fichier de dépôt car il n’y a pas un existant sur Kubernetes



```

root@fedora:~
Font size - 16 + Apparence Noir Réinitialiser

[root@fedora ~]# cat <<EOF > /etc/yum.repos.d/kubernetes.repo
> [kubernetes]
> ^C
[root@fedora ~]# cat <<EOF > /etc/yum.repos.d/kubernetes.repo
[kubernetes]
> name=Kubernetes
> baseurl=https://packages.cloud.google.com/yum/repos/kubernetes-el7-x86_64
> enabled=1
> gpgcheck=1
> repo_gpgcheck=1
> gpgkey=https://packages.cloud.google.com/yum/doc/yum-key.gpg https://packages.cloud.google.com/yum/doc/rpm-package-key.gpg
> exclude=kube*
> EOF
[root@fedora ~]#

```

FIGURE 3.27 – Ajout d’un référentiel

Avant d’utiliser le kubeadm pour installer le kubernetes, nous devons installer trois packages dont nous avons besoin. le ”kubect!”, ”kubelet” et ”kubeadm”. Cela en utilisant la commande `sudo dnf install -f kubelet kubect! kubeadm –disableexcludes=kubernetes` .

Le ”–disableexcludes” permet de désactiver la liste d’exclusion de kubernetes.

```

Installé:
contrack-tools-1.4.5-7.fc34.x86_64          containernetworking-plugins-1.0.1-1.fc34.x86_64
cri-tools-1.23.0-0.x86_64                  kubeadm-1.24.0-0.x86_64
kubectl-1.24.0-0.x86_64                   kubelet-1.24.0-0.x86_64
libnetfilter_cthelper-1.0.0-19.fc34.x86_64 libnetfilter_cttimeout-1.0.0-17.fc34.x86_64
libnetfilter_queue-1.0.2-17.fc34.x86_64   socat-1.7.4.2-1.fc34.x86_64

Terminé !

```

FIGURE 3.28 – L’installation des packages

- **Étape 2** : Nous installons l’environnement Kubernetes

```
[root@fedora ~]# dnf install kubernetes etcd
```

FIGURE 3.29 – L’installation de l’environnement Kubernetes

### 3.1.4 Installation d’Elastic Search sur Fedora

- **Étape 1** : Faut savoir qu’Elastic Search depend de JAVA. Pour cela, nous devons installer le package OpenJDK en executant la commande affichée dans la figure.

```

root@fedora:~
Taille de police - 16 + Apparence Noir Réinitialiser

[root@fedora ~]# sudo dnf install java-11-openjdk
Dernière vérification de l’expiration des métadonnées effectuée il y a 3:36:03 le lun. 09 mai 2022 15:37:05.
Dépendances résolues.
=====
Paquet                Architecture  Version                Dépôt                Taille
=====
Installation:
java-11-openjdk       x86_64        1:11.0.15.0.10-1.fc34 updates              247 k
Installation des dépendances:
alsa-lib              x86_64        1.2.6.1-3.fc34        updates              497 k
copy-jdk-configs      noarch        4.0-1.fc34            updates              27 k
fontconfig            x86_64        2.13.94-5.fc34       updates              272 k
freetype              x86_64        2.10.4-3.fc34        fedora               394 k
giflib                x86_64        5.2.1-7.fc34         fedora               49 k
graphite2            x86_64        1.3.14-7.fc34        fedora               95 k
harfbuzz              x86_64        2.7.4-3.fc34         fedora               634 k
java-11-openjdk-headless x86_64        1:11.0.15.0.10-1.fc34 updates              38 M
javapackages-filestsystem noarch        5.3.0-15.fc34        fedora               11 k
langpacks-core-font-en noarch        3.0-14.fc34          fedora               10 k
lcms2                 x86_64        2.12-1.fc34          fedora               171 k
libX11                x86_64        1.7.2-3.fc34         updates              646 k
libX11-common         noarch        1.7.2-3.fc34         updates              152 k

```



```

Installé:
alsa-lib-1.2.6.1-3.fc34.x86_64
fontconfig-2.13.94-5.fc34.x86_64
giflib-5.2.1-7.fc34.x86_64
harfbuzz-2.7.4-3.fc34.x86_64
java-11-openjdk-headless-1:11.0.15.0.10-1.fc34.x86_64
langpacks-core-font-en-3.0-14.fc34.noarch
libX11-1.7.2-3.fc34.x86_64
libXau-1.0.9-6.fc34.x86_64
libXext-1.3.4-6.fc34.x86_64
libXrender-0.9.10-14.fc34.x86_64
libfontenc-1.1.3-15.fc34.x86_64
libxcb-1.13.1-7.fc34.x86_64
lua-5.4.2-2.fc34.x86_64
mkfontscale-1.2.1-2.fc34.x86_64
tzdata-java-2022a-1.fc34.noarch
xorg-x11-fonts-Type1-7.5-31.fc34.noarch
copy-jdk-configs-4.0-1.fc34.noarch
freetype-2.10.4-3.fc34.x86_64
graphite2-1.3.14-7.fc34.x86_64
java-11-openjdk-1:11.0.15.0.10-1.fc34.x86_64
javapackages-filesystem-5.3.0-15.fc34.noarch
lcms2-2.12-1.fc34.x86_64
libX11-common-1.7.2-3.fc34.noarch
libXcomposite-0.4.5-5.fc34.x86_64
libXi-1.7.10-6.fc34.x86_64
libXtst-1.2.3-14.fc34.x86_64
libjpeg-turbo-2.0.90-3.fc34.x86_64
lksctp-tools-1.0.18-9.fc34.x86_64
lua-posix-35.0-3.fc34.x86_64
ttmkfdir-3.0.9-63.fc34.x86_64
xml-common-0.6.3-56.fc34.noarch

Terminé !

```

FIGURE 3.30 – Installation de package OpenJDK

- **Étape 2** : Par défaut, Elasticsearch n'est pas disponible sur le référentiel de base Fedora 35. Nous ajoutons maintenant le référentiel Elasticsearch RPM à notre système Fedora comme suivant :

```

[root@fedora ~]# cat <<EOF | sudo tee /etc/yum.repos.d/elasticsearch.repo
> [Elasticsearch-7]
> name=Elasticsearch repository for 7.x packages
> baseurl=https://artifacts.elastic.co/packages/7.x/yum
> gpgcheck=1
> gpgkey=https://artifacts.elastic.co/GPG-KEY-elasticsearch
> enabled=1
> autorefresh=1
> type=rpm-md
> EOF
[Elasticsearch-7]
name=Elasticsearch repository for 7.x packages
baseurl=https://artifacts.elastic.co/packages/7.x/yum
gpgcheck=1
gpgkey=https://artifacts.elastic.co/GPG-KEY-elasticsearch
enabled=1
autorefresh=1
type=rpm-md
[root@fedora ~]#

```

FIGURE 3.31 – Ajout du référentiel Elasticsearch

Ensuite, nous importons la clé GPG.

```

[root@fedora ~]# sudo rpm --import https://artifacts.elastic.co/GPG-KEY-elasticsearch
[root@fedora ~]#

```

FIGURE 3.32 – Importation de la clé GPG

Une fois cela est fait, nous passons à l'installation de package Elasticsearch en exécutant la commande "sudo dnf install elasticsearch" et nous obtenons le résultat suivant :

```

root@fedora:~
Taille de police - 16 + Apparence Noir Réinitialiser

[root@fedora ~]# sudo dnf install elasticsearch
Elasticsearch repository for 7.x packages                    580 B/s | 1.3 kB    00:02
Dépendances résolues.
=====
Paquet                Architecture      Version           Dépôt              Taille
=====
Installation:
elasticsearch         x86_64            7.17.3-1          Elasticsearch-7     297 M

Résumé de la transaction
=====
Installer 1 Paquet

Taille totale des téléchargements : 297 M
Taille des paquets installés : 494 M
Voulez-vous continuer ? [o/N] : o
Téléchargement des paquets :
elasticsearch-7.17.3-x86_64.rpm                            458 kB/s | 297 MB   11:04
=====
Total                                                         458 kB/s | 297 MB   11:04
Test de la transaction
La vérification de la transaction a réussi.

```

FIGURE 3.33 – Mise en place d'Elasticsearch

Une fois que l'installation est terminée, nous activons Elasticsearch pour qu'il démarre automatiquement au démarrage du système en exécutant la commande "sudo systemctl enable elasticsearch".

```

root@fedora:~
Taille de police - 16 + Apparence Noir Réinitialiser

[root@fedora ~]# sudo systemctl enable elasticsearch
Synchronizing state of elasticsearch.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable elasticsearch
Failed to execute /usr/lib/systemd/systemd-sysv-install: Aucun fichier ou dossier de ce type
[root@fedora ~]# /usr/lib/systemd/systemd-sysv-install enable elasticsearch
bash: /usr/lib/systemd/systemd-sysv-install: Aucun fichier ou dossier de ce type
[root@fedora ~]# sudo systemctl status elasticsearch
● elasticsearch.service - Elasticsearch
   Loaded: loaded (/usr/lib/systemd/system/elasticsearch.service; disabled; vendor preset: disabled)
   Active: active (running) since Mon 2022-05-09 20:48:19 CET; 1min 57s ago
     Docs: https://www.elastic.co
   Main PID: 7903 (java)
    Tasks: 58 (limit: 2288)
   Memory: 1.1G
      CPU: 1min 36.327s
   CGroup: /system.slice/elasticsearch.service
           └─7903 /usr/share/elasticsearch/jdk/bin/java -Xshare:auto -Des.networkaddress.cache.ttl=60 -Des.network
             8091 /usr/share/elasticsearch/modules/x-pack-ml/platform/linux-x86_64/bin/controller

mai 09 20:46:53 fedora systemd[1]: Starting Elasticsearch...
mai 09 20:48:19 fedora systemd[1]: Started Elasticsearch.
lines 1-14/14 (END)

```

FIGURE 3.34 – Activation d'Elasticsearch

- **Étape 3 :** Nous testons Elasticsearch en exécutant la commande "curl http://127.0.0.1:9200".

9200 est le numéro de port sur lequel Elasticsearch écoute les API REST.

Nous obtenons le résultat comme ci-dessous.



```

root@fedora:~
~
[root@fedora ~]# curl http://127.0.0.1:9200
{
  "name" : "fedora",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "EmY07Ao7R1q-rxOKpDZmIlg",
  "version" : {
    "number" : "7.17.3",
    "build_flavor" : "default",
    "build_type" : "rpm",
    "build_hash" : "5ad023604c8d7416c9eb6c0eadb62b14e766caff",
    "build_date" : "2022-04-19T08:11:19.070913226Z",
    "build_snapshot" : false,
    "lucene_version" : "8.11.1",
    "minimum_wire_compatibility_version" : "6.8.0",
    "minimum_index_compatibility_version" : "6.0.0-beta1"
  },
  "tagline" : "You Know, for Search"
}
[root@fedora ~]#

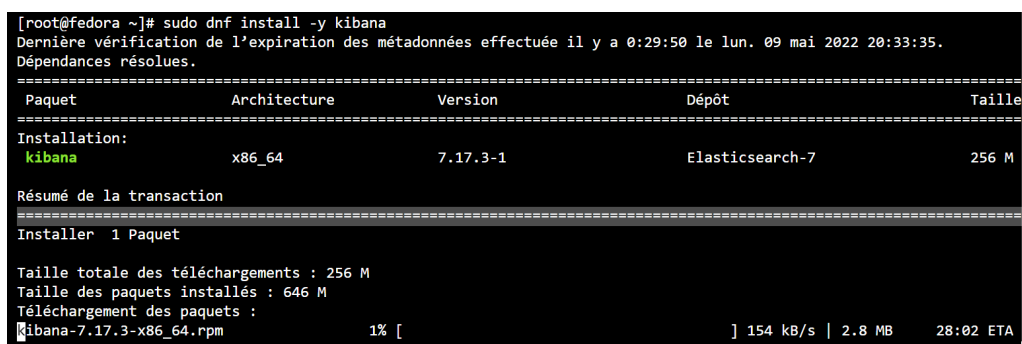
```

FIGURE 3.35 – Test sur Elasticsearch

### 3.1.5 Installation de Kibana sur Fedora

Selon quelques documentation officielles, nous ne devons installer Kibana qu’après avoir installé Elasticsearch, car l’installation dans cet ordre garantit que les composants dont chaque produit dépend sont correctement en place.

Comme nous avons ajouté le référentiel et installé la clé GPG, nous devons simplement installer Kibana avec la commande "sudo dnf install -y kibana".



```

[root@fedora ~]# sudo dnf install -y kibana
Dernière vérification de l'expiration des métadonnées effectuée il y a 0:29:50 le lun. 09 mai 2022 20:33:35.
Dépendances résolues.
=====
Paquet          Architecture  Version      Dépôt          Taille
=====
Installation:
kibana          x86_64        7.17.3-1     Elasticsearch-7 256 M
=====
Résumé de la transaction
=====
Installer 1 Paquet

Taille totale des téléchargements : 256 M
Taille des paquets installés : 646 M
Téléchargement des paquets :
kibana-7.17.3-x86_64.rpm          1% [          ] 154 kB/s | 2.8 MB 28:02 ETA

```

FIGURE 3.36 – Installation de kibana

### 3.1.6 Installation de Hadoop

Hadoop est écrit en java et comme nous avons installé le package OpenJDK précédemment, nous passons aux étapes suivantes directement.

- **Étape 1** : Pour des raison de sécurité, il est préférable de créer un utilisateur hadoop et le mot de passe en exécutant les commandes suivantes respectivement "sudo adduser hadoop" et "passwd hadoop"
- **Étape 2** : Nous configurons l’authentification SSH sans mot de passe pour le système local.

D’abord, nous changeons l’utilisateur en hadoop.

```
hadoop@fedora:~
[root@fedora ~]# su - hadoop
[hadoop@fedora ~]$
```

FIGURE 3.37 – Changement d'utilisateur

Ensuite, nous exécutons la commande "ssh-keygen -t rsa -P '' -f ~/.ssh/id\_rsa" pour générer des paires de clés publiques et privées et nous ajoutons les clés publiques générées de "id\_rsa.pub" à "authorized keys" puis définissons l'autorisation appropriée.

```
hadoop@fedora:~
[hadoop@fedora ~]$ ssh-keygen -t rsa -P '' -f ~/.ssh/id_rsa
Generating public/private rsa key pair.
Created directory '/home/hadoop/.ssh'.
Your identification has been saved in /home/hadoop/.ssh/id_rsa
Your public key has been saved in /home/hadoop/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:o04NOuyF02BEAiYHSlv7z7JvtB7BaTcf01DiysbTAs hadoop@fedora
The key's randomart image is:
+---[RSA 3072]-----+
|*o.o
|=o
|o . = .
|.ooo+.o
|.o.E .oS+
|= .o +
|oo+. .
|.o*=o..
|.*X*oo
+---[SHA256]-----+
[hadoop@fedora ~]$ cp -r ~/.ssh/id_rsa.pub ~/.ssh/authorized_keys
[hadoop@fedora ~]$ chmod 0600 ~/.ssh/authorized_keys
[hadoop@fedora ~]$
```

FIGURE 3.38 – Configuration basée sur la clé SSH

Pour vérifier l'authentification, nous effectuons "ssh localhost".

- **Étape 3** : Dans cette étape, nous passons à l'installation de Hadoop. Tout d'abord, à l'aide de la commande "wget https://downloads.apache.org/hadoop/common/hadoop-3.3.2/hadoop-3.3.2.tar.gz" nous installons la version 3.3.2 de hadoop.

```
[hadoop@fedora ~]$ wget https://downloads.apache.org/hadoop/common/hadoop-3.3.2/hadoop-3.3.2.tar.gz
--2022-04-26 08:55:45-- https://downloads.apache.org/hadoop/common/hadoop-3.3.2/hadoop-3.3.2.tar.gz
Résolution de downloads.apache.org (downloads.apache.org)... 135.181.214.104, 88.99.95.219, 2a01:4f8:10a:201a::2, ...
Connexion à downloads.apache.org (downloads.apache.org)[135.181.214.104]:443... connecté.
requête HTTP transmise, en attente de la réponse... 200 OK
Taille : 638660563 (609M) [application/x-gzip]
Sauvegarde en : « hadoop-3.3.2.tar.gz »

hadoop-3.3.2.tar.gz 0%[ ] 808,00K 47,1KB/s tps 2h 51m
```

```

Résolution de downloads.apache.org (downloads.apache.org)... 135.181.214.104, 88.99.95.219, 2a01:4f8:10a:201a::2, ...
Connexion à downloads.apache.org (downloads.apache.org)[135.181.214.104]:443... connecté.
requête HTTP transmise, en attente de la réponse... 200 OK
Taille : 638660563 (609M) [application/x-gzip]
Sauvegarde en : « hadoop-3.3.2.tar.gz »

hadoop-3.3.2.tar.gz          100%[=====] 609,07M  793KB/s  ds 23m 51s
2022-04-26 09:19:37 (436 KB/s) - « hadoop-3.3.2.tar.gz » sauvegardé [638660563/638660563]

```

FIGURE 3.39 – Installation de Hadoop

Une fois téléchargé, nous extrayons le fichier en affectuant "tar -xvzf hadoop-3.3.2.tar.gz". Ensuite, nous renommons répertoire extrait en hadoop, comme ce-ci "mv hadoop-3.3.0 hadoop" mv hadoop-3.3.0 hadoop".

- **Étape 4 :** Maintenant place a la configuration des variables d'environnement Hadoop et Java sur notre système en suivant les étapes suivantes
  - Premièrement, nous devons ajouter les variables affichées dans la figure ci-dessous dans le fichier "bashrc" en exécutant la commande "nano /.bashrc".

```

export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-11.0.10.0.9-0.fc34.x86_64
export HADOOP_HOME=/home/hadoop/hadoop
export HADOOP_INSTALL=$HADOOP_HOME
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_HDFS_HOME=$HADOOP_HOME
export YARN_HOME=$HADOOP_HOME
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export PATH=$PATH:$HADOOP_HOME/sbin:$HADOOP_HOME/bin
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib/native"

```

FIGURE 3.40 – Modification effectuée dans le fichier bashrc

- Deuxièmement, nous appliquons les modifications dans l'environnement d'exécution actuel avec la commande "source /.bashrc".
- Troisièmement, la commande montrée dans la figure ci-dessous, nous permet d'ouvrir l'environnement Hadoop.

```

[root@fedora ~]# nano $HADOOP_HOME/etc/hadoop/hadoop-env.sh

```

FIGURE 3.41 – Modification de l'environnement Hadoop

Une fois ouvert, nous ajoutons la ligne suivante pour modifier la variable JAVA Home :

- **Étape 5 :** Il est maintenant temps de configurer Hadoop.
  - Nous commençons par créer les répertoires **namenode** et **datanode** dans le

```

Generic settings for HADOOP
###

# Technically, the only required environment variable is JAVA_HOME.
# All others are optional. However, the defaults are probably not
# preferred. Many sites configure these options outside of Hadoop,
# such as in /etc/profile.d

# The java implementation to use. By default, this environment
# variable is REQUIRED on ALL platforms except OS X!
# export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-11.0.10.0.9-0.fc34.x86_64

# Location of Hadoop. By default, Hadoop will attempt to determine
# this location based upon its execution path.
# export HADOOP_HOME=

# Location of Hadoop's configuration information. i.e., where this
# file is living. If this is not defined, Hadoop will attempt to
# locate it based upon its execution path.

```

FIGURE 3.42 – Modification de la variable

répertoire d'accueil hadoop avec les deux commandes :

— `mkdir -p /hadoopdata/hdfs/namenode` Et

— `mkdir -p /hadoopdata/hdfs/datanode`

-Ensuite, la commande `vi core-site.xml` nous permet d'ouvrir le fichier **core-site.xml** dans un éditeur de texte pour pouvoir définir les propriétés principales HDFS et Hadoop en ajoutant les lignes suivantes :

```

<configuration>

  <property>
    <name>fs.defaultFS</name>
    <value>hdfs://localhost:9000</value>
  </property>

</configuration>

```

FIGURE 3.43 – Modification réalisées au fichier core-site.xml

La même chose pour :

— Fichier **hdfs-site.xml** en exécutant la commande `vi hdfs-site.xml`.

```
<configuration>

  <property>

    <name>dfs.replication</name>

    <value>1</value>

  </property>

</configuration>
```

FIGURE 3.44 – Modification réalisées au fichier hdfs-site.xml

La valeur par défaut **dfs.replication** dans la figure ci-dessus doit être remplacée par 1 pour correspondre à la configuration de nœud unique.

- Fichier **yarn-site.xml**, qui est utilisé pour définir les paramètres relatifs à YARN et cela avec "vi yarn-site.xml"

```
<configuration>
<property>

  <name>yarn.nodemanager.aux-services</name>

  <value>mapreduce_shuffle</value>

</property>

</configuration>
```

FIGURE 3.45 – Modification réalisées au fichier yarn-site.xml

- Fichier **mapred-site.xml** et définir les valeurs MapReduce en modifiant la valeur par défaut du nom de framework MapReduce en yarn en effectuant la commande "vi mapred-site.xml".

```
<configuration>
  <property>
    <name>mapreduce.framework.name</name>
    <value>yarn</value>
  </property>
</configuration>
```

FIGURE 3.46 – Modification réalisées au fichier mapred-site.xml

-Maintenant qu'on a fini avec les fichiers, nous devons formater le **namenode** avant de démarrer les services Hadoop pour la première fois en effectuant "hdfs namenode -format".

Afin de démarrer le NameNode et le DataNode nous exécutons le script suivant :

- ./start-dfs.sh

Une fois que le namenode, les datanodes et le SecondaryNamenode sont opérationnels, nous démarrons la ressource YARN et les nodemanagers en tapant :

- ./start-yarn.sh

Pour vérifier si tous les démons sont actifs et s'exécutent en tant que processus Java nous tapons la commande "jps" et nous obtenons le résultat suivant :

```
[root@fedora sbin]# jps
5635 Jps
5045 SecondaryNameNode
5468 NodeManager
4860 DataNode
4733 NameNode
3821 ResourceManager
```

FIGURE 3.47 – La liste résultante des processus Java en cours d'exécution



## **Conclusion**

Dans ce chapitre nous avons implémenté la mise en place de la solution. En premier lieu, nous avons élaboré les étapes nécessaires à l'installation et configuration de fedora server 34 sur vmware workstation. En second lieu, nous avons mis en exergue l'installation de chacun des outils et des environnements utilisés.

# Déploiement et exploitation de la solution

Pour visualiser l'efficacité de notre travail et mettre en évidence la configuration de notre topologie, nous avons utilisé le simulateur GNS3 version 2.2.32 qui est un logiciel libre permettant l'émulation ou la simulation de réseaux informatique. Il est utilisé également pour reproduire différents systèmes d'exploitation dans un environnement virtuel.

Dans ce chapitre nous allons présenter les outils utilisés et la procédure de configuration pour mettre en œuvre notre solution.

## 4.1 La simulation de la topologie

Notre cas d'étude s'est porté sur le réseau local au niveau de l'entreprise ou nous allons implémenter Fortigate.

### 4.1.1 Topologie du réseau existant

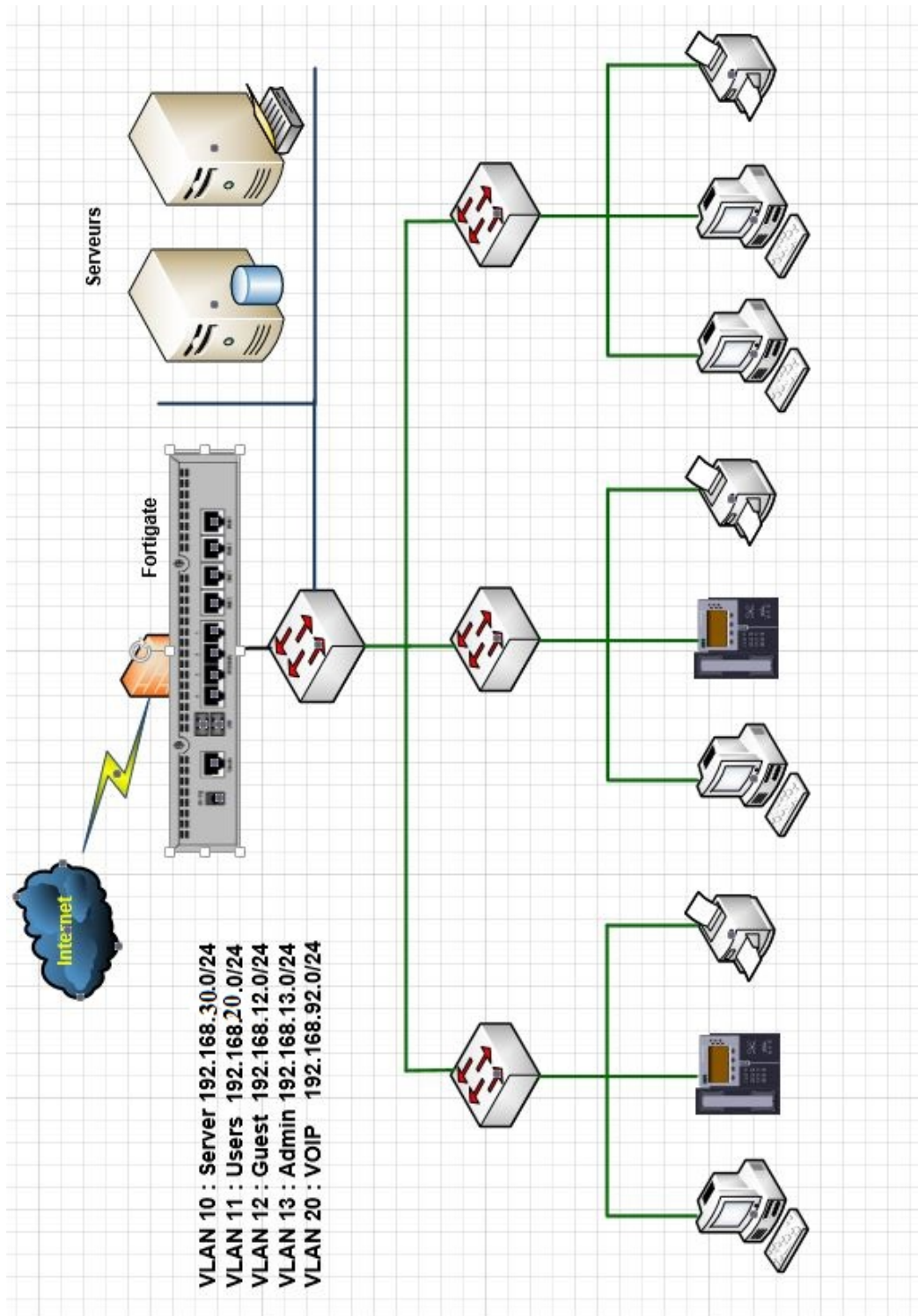


FIGURE 4.1 – Topologie du réseau existant

### 4.1.2 simulation de la topologie sous GNS3

Pour des raisons de manque de certains dispositifs, nous avons choisi de remplacer le switchs par un switch de niveau trois avec un IOS routeur pour pouvoir les configurer en y créant des VLANs. Nous avons obtenus la topologie suivante :

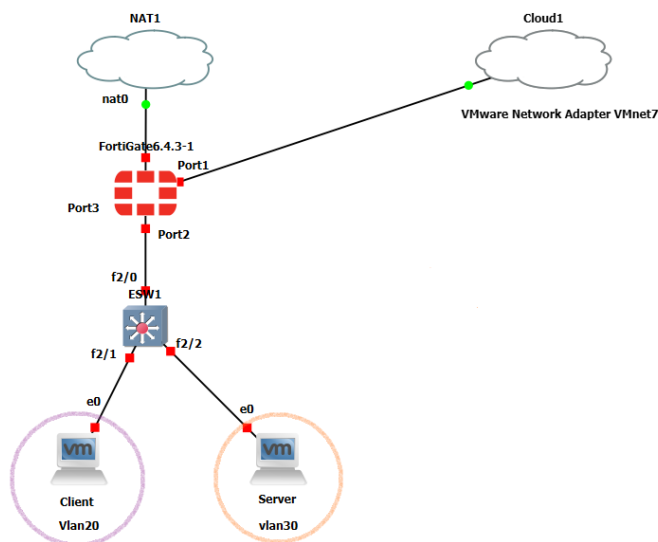


FIGURE 4.2 – La topologie du réseau local Baticompos sous GNS3

La topologie est composée de :

- Nat
- cloud pour connecter Fortigate a internet.
- Un switch pour configurer les VLANs
- Windows 7 qui est utilisé comme Client
- Windows server16, utilisé comme serveur

Pour réaliser cette topologie et pourvoir se connecter a l'interface web de Fortigate, nous suivons les étapes suivantes :

**(a) Installation et configuration de Fortigate**

Un serveur Fortinet Fortigate est un serveur de sécurité tout en un. Cette solution remplit au minimum les mêmes fonctionnalités qu'un serveur proxy et bien plus : pare-feu, antivirus, système de prévention d'intrusion, VPN, filtrage Web, antispam, etc.

Il permet donc de rassembler la plupart des fonctionnalités de sécurité dont une entreprise moderne a besoin en un seul point d'administration, redondant.

**— Installation de fortigate sur GNS3**

- i. Dans l'interface utilisateur GNS3, nous cliquons sur **file** puis sur **new template**.

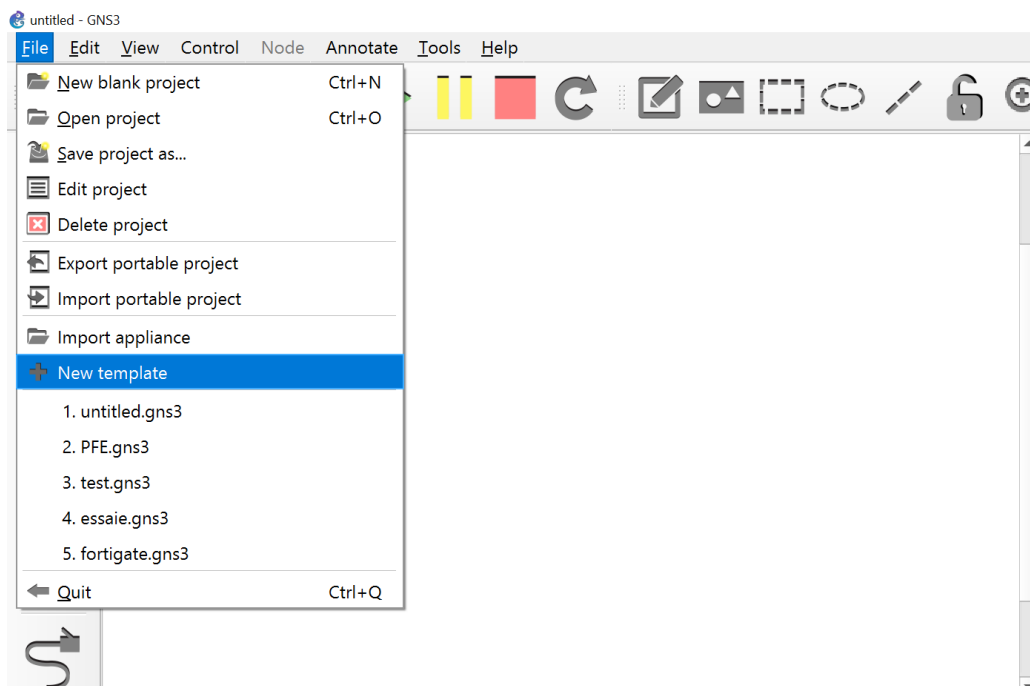


FIGURE 4.3 – Importation de l'équipement fortigate

- ii. La fenêtre **server** s'affiche, nous choisissons la première option pour installer l'équipement depuis le serveur GNS3.

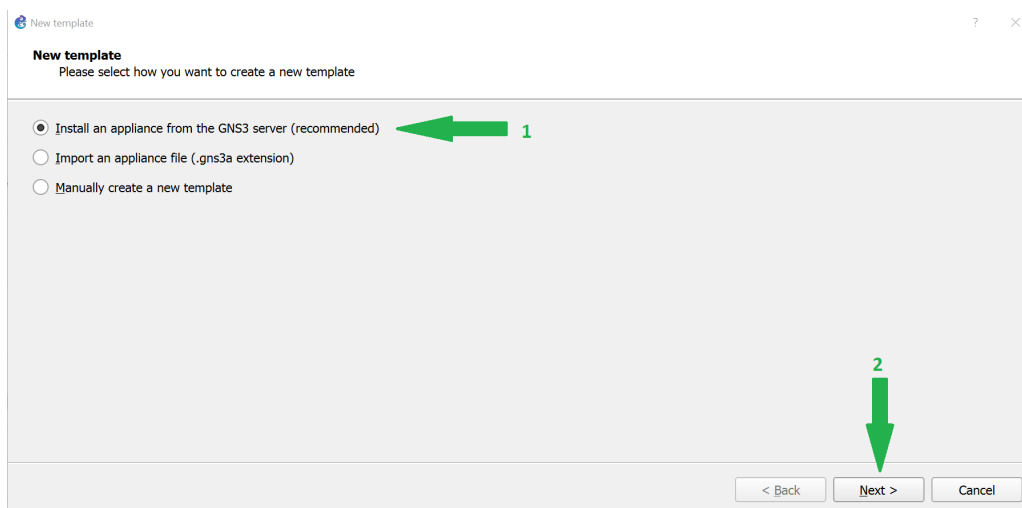


FIGURE 4.4 – choix d’option

- iii. Dans la nouvelle fenêtre qui s’affiche, nous choisissons **Firewalls** puis cliquons sur le boutons **Next**.

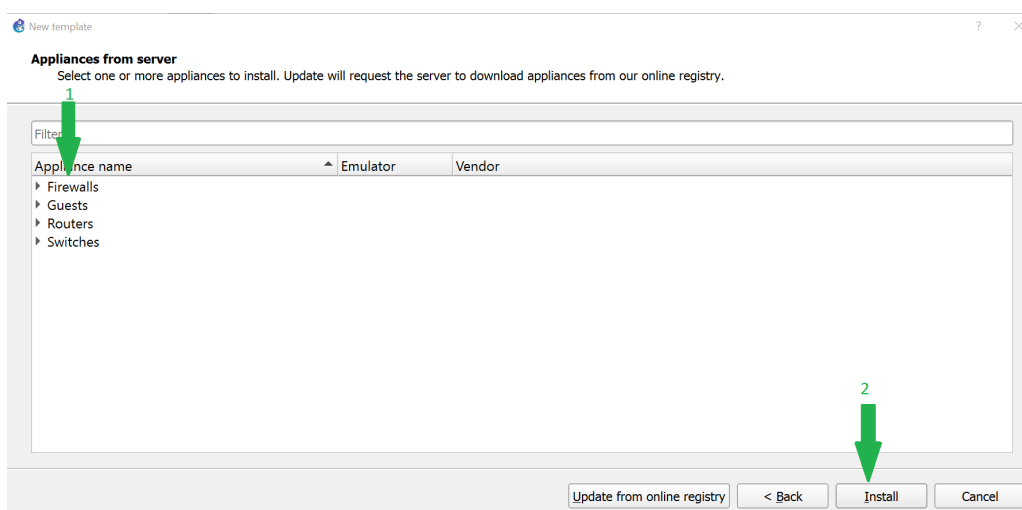


FIGURE 4.5 – installation de l’appliance depuis GNS3

- iv. Comme la figure ci-dessous le montre, nous choisissons l’équipement ”Fortigate”.

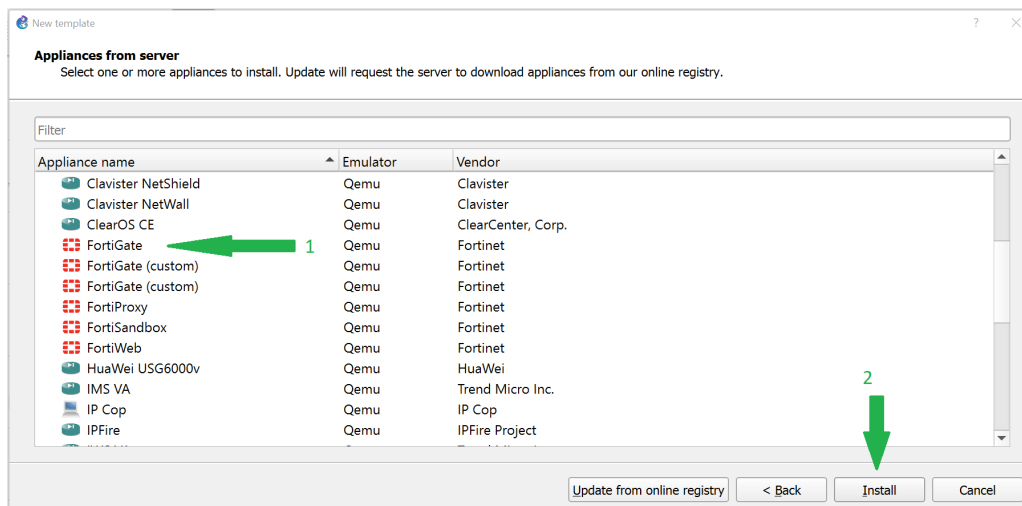


FIGURE 4.6 – Choix de l'équipement

- v. Une fois cliqué sur **Install**, l'écran suivant s'affiche, nous choisissons le deuxième type de serveur pour installer l'appliance.



FIGURE 4.7 – Installation de l'appliance sur GNS3

- vi. La fenêtre des paramètres QEMU s'affiche. La version 2.2.32 de la machine virtuelle GNS3 inclut les binaires QEMU 4.2.1 64 et 32 bits. cliquons sur **Next**.

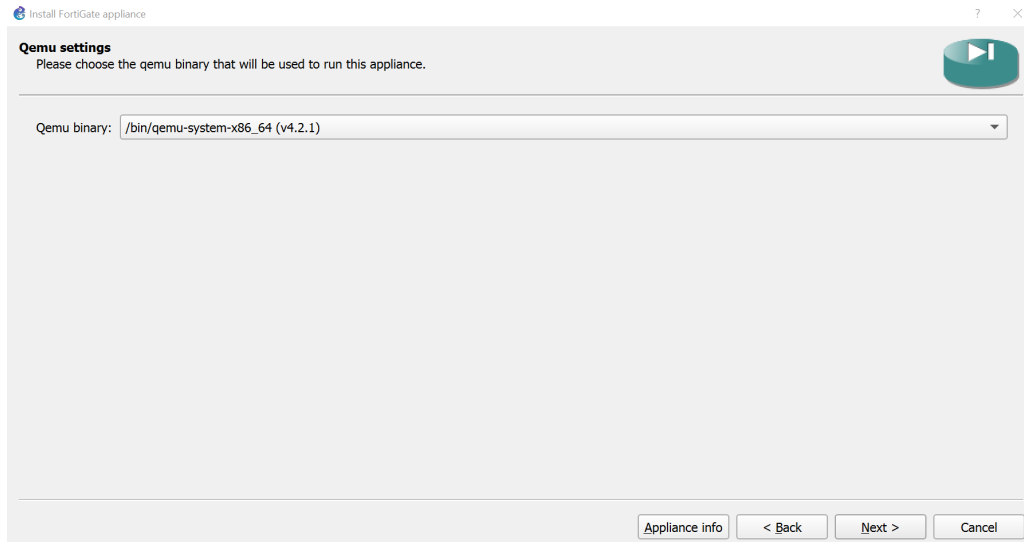


FIGURE 4.8 – QEMU binary

vii. La fenêtre Fichiers requis s’affiche :

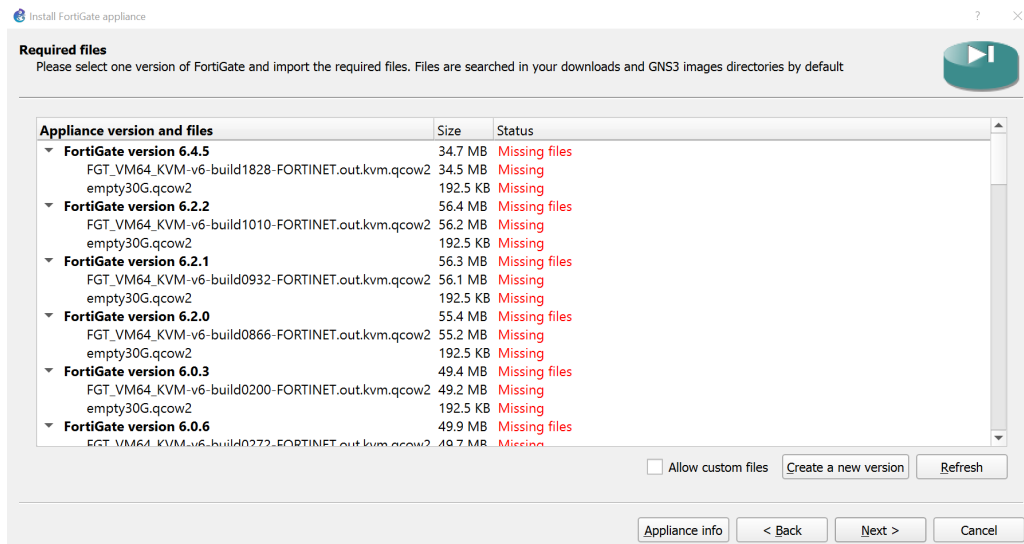


FIGURE 4.9 – L’affichage des fichiers requis



- viii. Nous sélectionnons maintenant la version que nous souhaitons installer, nous mettons en surbrillance son nom de fichier puis cliquons sur "import".

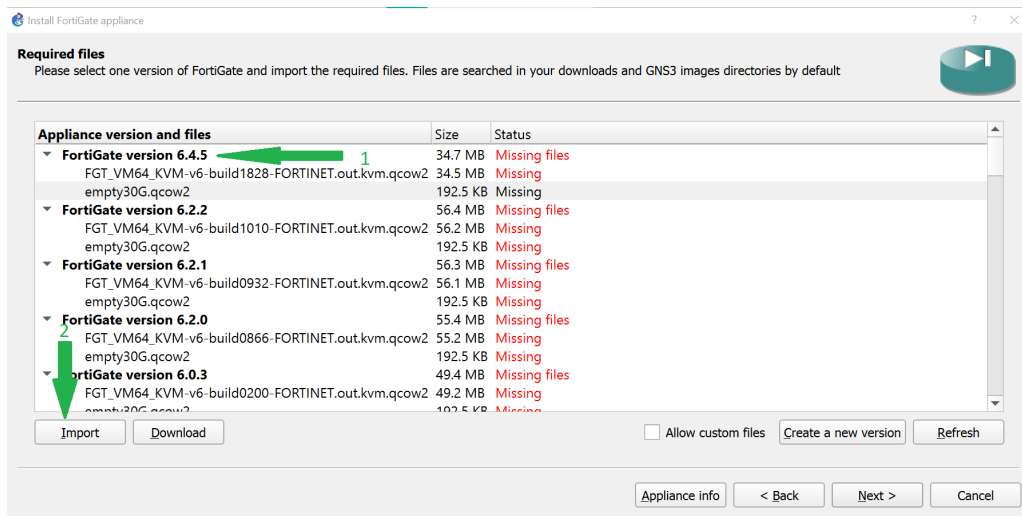


FIGURE 4.10 – La sélection de la version souhaitée

- ix. une fenêtre s'ouvre afin de sélectionner l'image de disque virtuel approprié pour le périphérique.

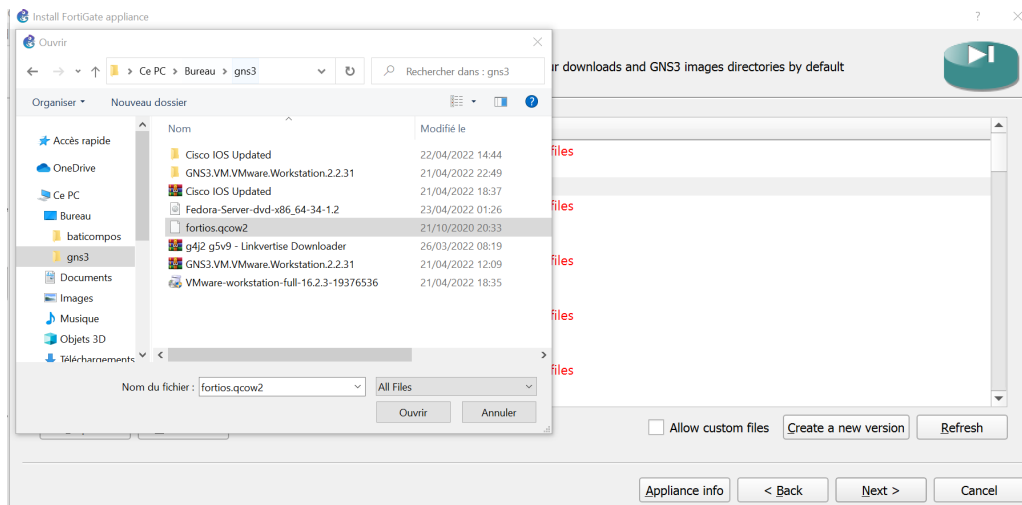


FIGURE 4.11 – La sélection de l'image de disque virtuel

- x. Une fois que c'est fait, nous sélectionnons le nom de la version de périphérique et l'installation est faite avec succès et l'appliance est ajoutée a GNS3.

Cliquons sur le menu de "Browse security devices" ou sur celui de "Browse all devices" pour afficher l'appliance nouvellement ajoutée.

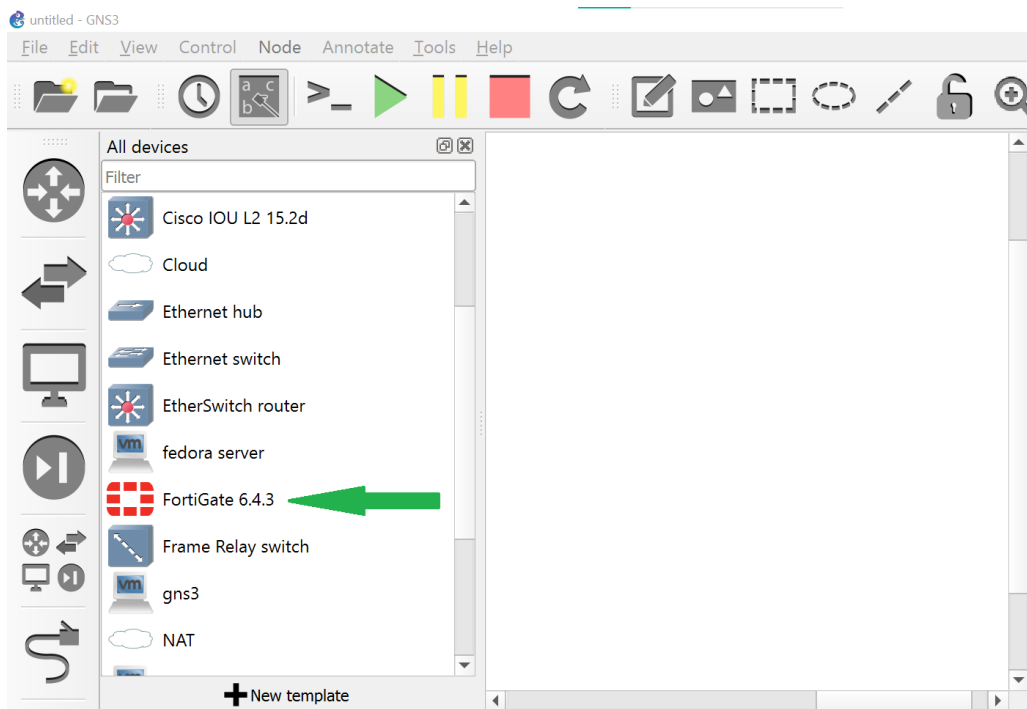


FIGURE 4.12 – Installation finie

Maintenant que nous avons fini avec l'installation, passons a la configuration de Fortigate.

#### — Configuration de Fortigate

- i. En premier lieu, nous allons configurer le cloud pour que nous puissions accéder a l'interface graphique web de Fortigate. Pour cela
  - Nous accédons a la fenêtre de configuration de cloud afin d'ajouter une interface pour le réseau virtuel.
  - Cliquons sur "show special Ethernet interfaces".

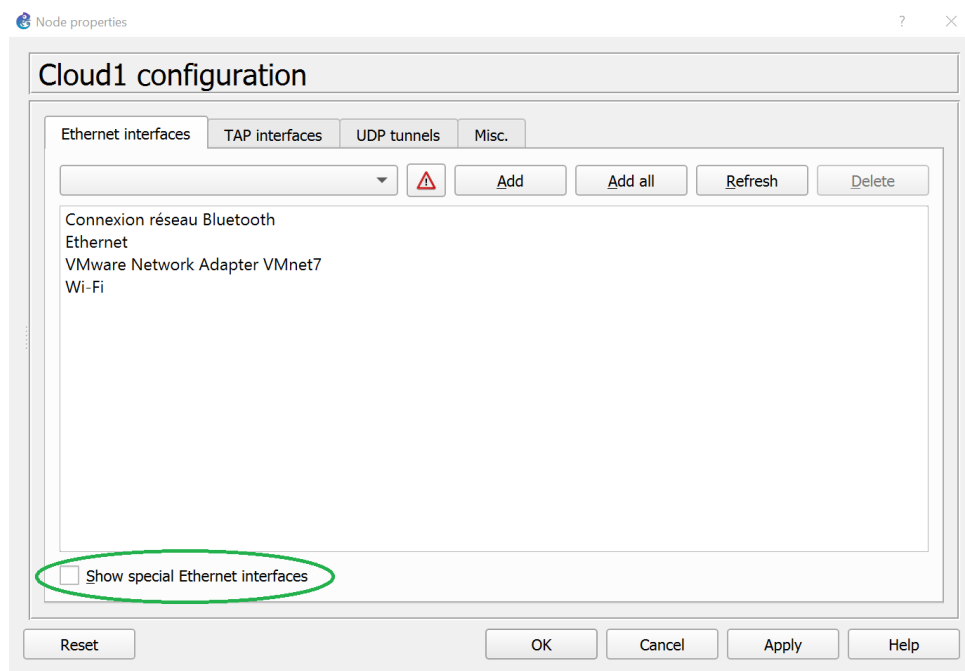


FIGURE 4.13 – L'ajout d'une interface pour le réseau virtuel

- Pour sélectionner une VMnet, il suffit de cliquer sur la flèche et une liste s'affiche puis sur le bouton "add" afin de l'ajouter aux interfaces et enfin sur le bouton "ok" pour confirmer et quitter la fenêtre. Dans notre cas, c'est la VMnet 7 que nous avons choisi.

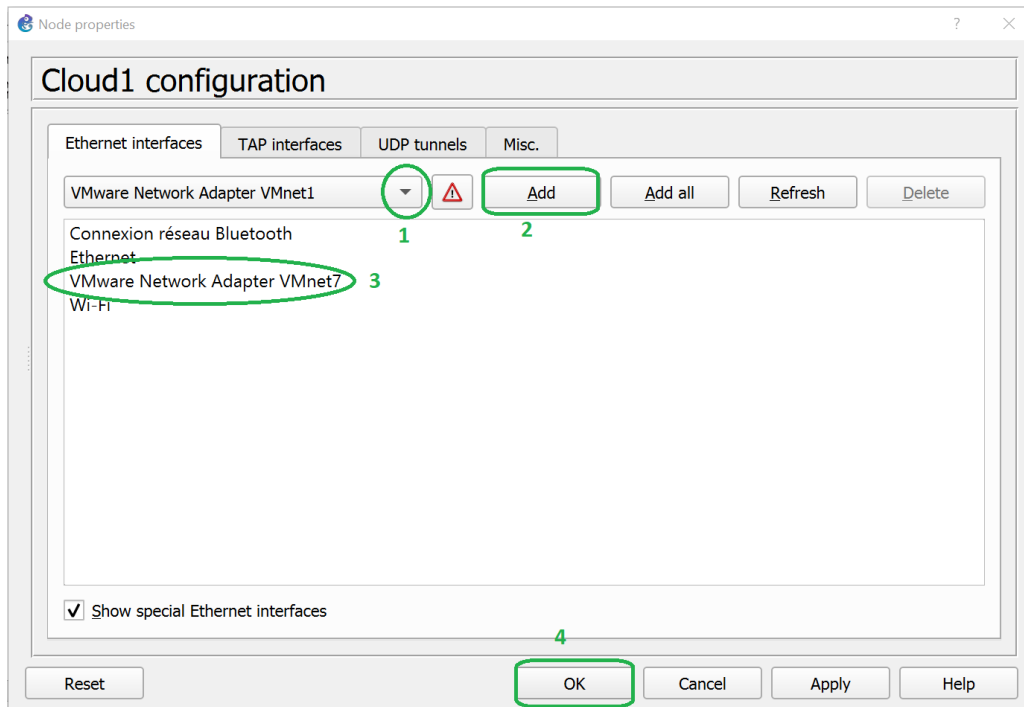


FIGURE 4.14 – La sélection de l’interface

- Pour que la connexion soit établie, nous devons également configurer la carte réseau "VMnet7" sur notre machine physique. Donc, nous entrons les paramètres de connexion réseau comme illustré dans la figure ci-dessous.

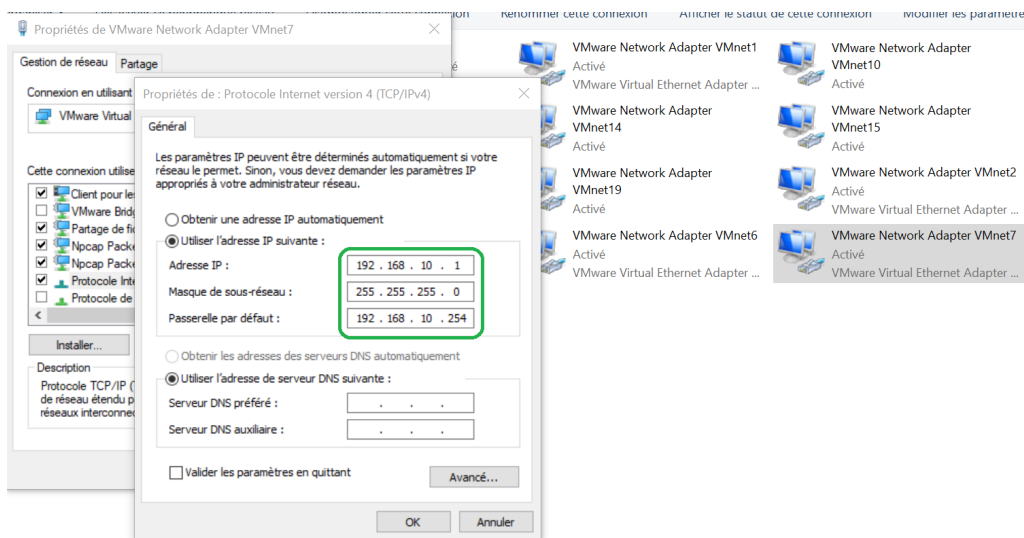


FIGURE 4.15 – Configuration de la carte réseau

Ici, la passerelle par défaut doit être l'adresse de Fortigate si non une erreur peut se produire.

- ii. Maintenant, au tour de Fortigate.
  - Nous devons le démarrer d'abord en faisant un clic droit dessus et cliquant sur "Start". Après quelques secondes, le pare-feu sera chargé et un accès CLI est possible pour la configuration, pour la première fois, cette dernière demande le nom d'utilisateur qui est "admin" par défaut ainsi que le mot de passe. Pour en configurer un nouveau, il suffit de taper sur la touche "Entrer" et une deuxième fois pour le confirmer.

```
FortiGate-VM64-KVM login: admin
Password:
You are forced to change your password. Please input a new password.
New Password:
Confirm Password:
Welcome!
```

FIGURE 4.16 – Configuration de mot de passe du pare-feu Fortigate

- Une fois que l'étape précédente est faite, nous passons à la configuration de l'IP du **port 1**. Par défaut, le pare-feu FortiGate aura un accès HTTPS activé à des fins de gestion sur le port1. Le port 1 est activé en mode dynamique donc nous devons le configurer en mode statique et donner l'adresse 192.168.10.254/24. Nous activons également l'accès HTTP, TELNET, SSH et PING.

```
FortiGate-VM64-KVM # config system interface
FortiGate-VM64-KVM (interface) # edit port1
FortiGate-VM64-KVM (port1) # set mode static

FortiGate-VM64-KVM (port1) # set ip 192.168.10.254/24
FortiGate-VM64-KVM (port1) # set allowaccess http telnet ssh ping
FortiGate-VM64-KVM (port1) # end
```

FIGURE 4.17 – Configuration de l'IP du port 1

- Avant de continuer, nous vérifions la configuration en tapant la commande ” **show system interface port1**”.

```
FortiGate-VM64-KVM # show sys int port1
config system interface
  edit "port1"
    set vdom "root"
    set ip 192.168.10.254 255.255.255.0
    set allowaccess ping ssh http telnet
    set type physical
    set snmp-index 1
  next
end
```

FIGURE 4.18 – Vérification de la configuration sur le port1

- Nous pouvons maintenant nous connecter à l’interface graphique web en tapant **http ://192.168.10.254** sur un navigateur quelconque.

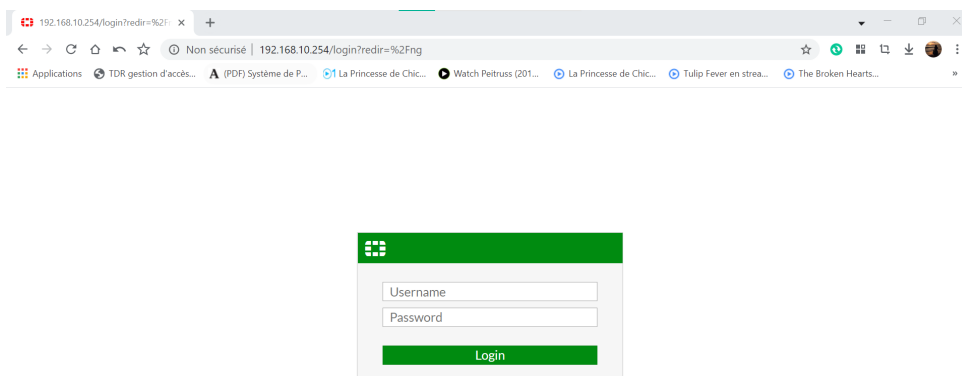


FIGURE 4.19 – Accès a l’interface graphique web du pare-feu

- (b) **Configuration des VLANs** La configuration des VLANs se fait au niveau du commutateur (Switch) et cela en passant par quelques étapes.
- Commençons par nous connecter a la console du switch puis créons les deux vlans **20** et **30** pour les machines **Client** et **Serveur** après avoir entré en mode **Configuration**.

```
ESW1#config t
Enter configuration commands, one per line. End with CNTL/Z.
ESW1(config)#vlan 20
ESW1(config-vlan)#name Client
ESW1(config-vlan)#exit
ESW1(config)#vlan 30
ESW1(config-vlan)#name Server
ESW1(config-vlan)#exit
ESW1(config)#
```

FIGURE 4.20 – Création des VLANs

- ii. Ensuite, nous définissons la liaison entre le switch et le pare-feu fortigate comme un **Trunk**.

```
ESW1(config)#interface fastEthernet2/0
ESW1(config-if)#switchport trunk encapsulation dot1q
ESW1(config-if)#switchport mode trunk
```

FIGURE 4.21 – Configuration de la liaison entre le switch et pare-feu

- iii. plaçons maintenant les interfaces dans les vlans et définissons le mode **access** pour chaque interface.

```
ESW1(config)#interface fastEthernet2/1
ESW1(config-if)#switchport mode access
ESW1(config-if)#switchport access vlan 20
ESW1(config-if)#
ESW1(config-if)#exit
```

```

ESW1(config)#interface fastethernet2/2
ESW1(config-if)#switchport mode access
ESW1(config-if)#switchport access Vlan 30
ESW1(config-if)#exit
ESW1(config)#

```

FIGURE 4.22 – Mise en place des interfaces

- iv. Enfin, Vérifions si tout est bien en place avec la commande **show vlan-switch**.

```

ESW1#show vlan-switch

```

VLAN	Name	Status	Ports
1	default	active	Fa1/0, Fa1/1, Fa1/2, Fa1/3 Fa1/4, Fa1/5, Fa1/6, Fa1/7 Fa1/8, Fa1/9, Fa1/10, Fa1/11 Fa1/12, Fa1/13, Fa1/14, Fa1/15 Fa2/3, Fa2/4, Fa2/5, Fa2/6 Fa2/7, Fa2/8, Fa2/9, Fa2/10 Fa2/11, Fa2/12, Fa2/13, Fa2/14 Fa2/15
20	CLIENT	active	Fa2/1
30	server	active	Fa2/2
1002	fddi-default	act/unsup	
1003	token-ring-default	act/unsup	
1004	fddinet-default	act/unsup	
1005	trnet-default	act/unsup	

FIGURE 4.23 – Vérification de la configuration des VLANs

Pour que la configuration des VLANs réussisse, nous devons les ajouter aussi dans l'interface graphique du pare-feu en suivant les étapes suivantes :

### Étapes d'ajout des VLANs dans l'interface graphique web du pare-feu

- d'abord, nous ajoutons deux nouvelles interfaces que nous nommons "Client" et "Server" pour Vlan20 et Vlan30 respectivement.

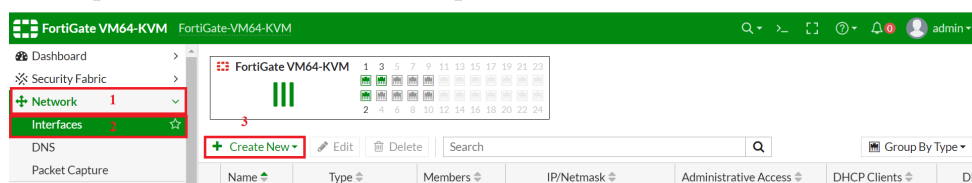


FIGURE 4.24 – Ajout de deux interfaces Client et Server

- Puis,
  - donnons un nom aux interfaces VLAN.
  - Choisissons l'interface physique sur laquelle attacher les VLANs, dans notre cas c'est **Port2**.
  - sélectionnons le type comme Vlan.



(d) Définissons l'adresse IP et le masque pour chaque Vlan.

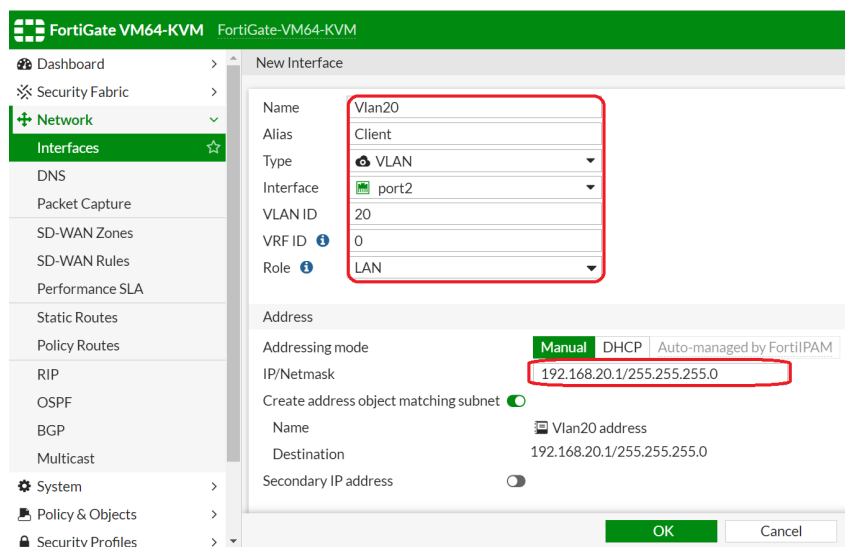


FIGURE 4.25 – Configuration de Vlan 20

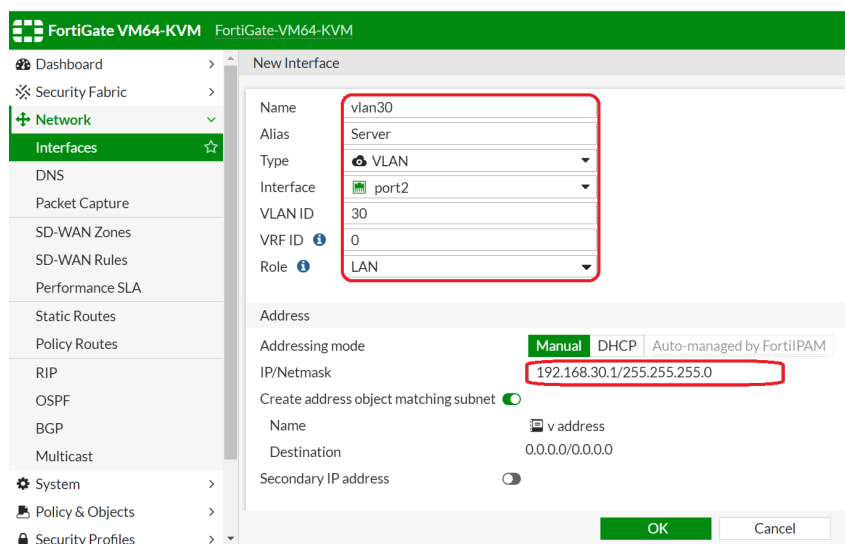


FIGURE 4.26 – Configuration de Vlan 30

3. Pour vérifier si elles ont été bien ajoutées, dans la liste des interfaces physiques, nous déployons le port2 et les VLANs s'afficheront.

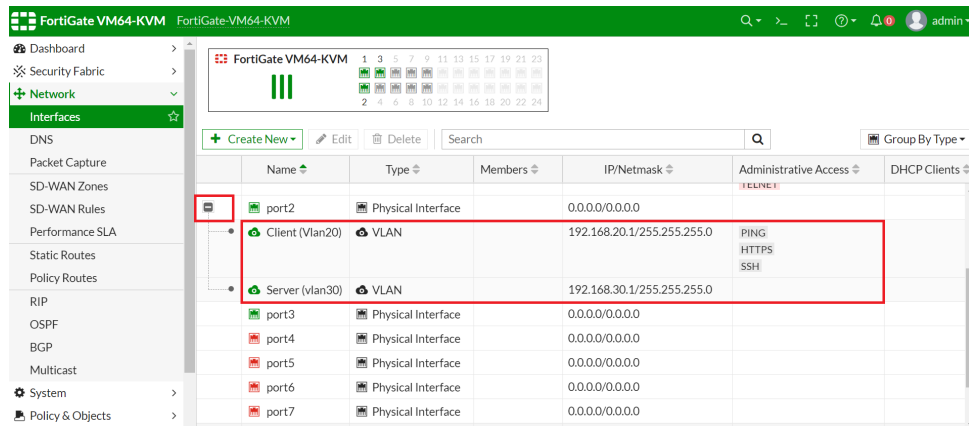


FIGURE 4.27 – Vérification

4. Il est important de noter que **tout trafic qui n'est pas explicitement autorisé par la politique de pare-feu est refusé**. Donc, il est nécessaire de configurer la communication de base pour autoriser la connexion entre les VLANs en ajoutant une politique.

Pour cela, dans le menu de gauche, nous déployons **Policy and Objects** puis sélectionnons **Firewall Policy** et paramétrons les paramètres représentés dans la figure ci-dessous.

Name: ClientToServer

Incoming Interface: Client (Vlan20)

Outgoing Interface: Server (Vlan30)

Source: Vlan20 address

Destination: Vlan30 address

Schedule: always

Service: ALL

Action:  ACCEPT  DENY

Inspection Mode:  Flow-based  Proxy-based

Firewall / Network Options

NAT:

Protocol Options: PROT default

FIGURE 4.28 – Configuration de la politique du vlan20 au vlan30

The screenshot shows a configuration window for a firewall policy. The 'Name' field is 'ServerToClient'. The 'Incoming Interface' is 'Server (Vlan30)' and the 'Outgoing Interface' is 'Client (Vlan20)'. The 'Source' is 'Vlan30 address' and the 'Destination' is 'Vlan20 address'. The 'Schedule' is set to 'always' and the 'Service' is 'ALL'. The 'Action' is 'ACCEPT' (indicated by a green checkmark) and 'DENY' (indicated by a red X). The 'Inspection Mode' is 'Flow-based'. Under 'Firewall / Network Options', 'NAT' is disabled and 'Protocol Options' is set to 'PROT default'.

FIGURE 4.29 – Configuration de la politique du vlan30 au vlan20

La section prochaine est réservée au traitement de données. Pour cela, nous décrirons d'abord les étapes du transfert du fichier vers le serveur fedora via le serveur ftp , puis les étapes de l'insertion du fichier de données dans le hdfs ,ensuite le programme mapreduce et enfin les étapes d'exécution de ce programme.

## 4.2 Mise en oeuvre de la solution

### 4.2.1 Installation du serveur FTP

Afin de pouvoir importer le fichier de données vers notre machine nous optons pour l'utilisation du serveur ftp et pour cela nous suivent les étapes d'installations suivante :

1. Installer vsftpd, le serveur FTP choisi par défaut par Fedora

```
[root@fedora ~]# dnf install vsftpd
```

FIGURE 4.30 – Installation de vsftpd

2. Démarrer le service vsftpd

```
[root@fedora ~]# systemctl start vsftpd.service
```

FIGURE 4.31 – Démarrage du service vsftpd

3. Déplacer vers le répertoire /etc/vsftpd/ ou se trouve le fichier de configuration de vsftpd qui se nomme vsftpd.conf.

```
[root@fedora ~]# cd /etc/vsftpd/
```

FIGURE 4.32 – Changement de répertoire

4. Accéder au fichier vsftpd.conf

```
[root@fedora vsftpd]# vi /etc/vsftpd/vsftpd.conf
```

FIGURE 4.33 – Accès au fichier vsftpd.conf

5. Effacer l'ancienne configuration pour la remplacer par la configuration suivante :

```
local_enable=YES
xferlog_enable=YES
connect_from_port_20=YES
pam_service_name=vsftpd
# Enable upload by local user.
write_enable=YES
pasv_enable=YES
pasv_addr_resolve=YES
# Enable read by anonymous user (without username and password).
anonymous_enable=YES
anon_root=/var/ftp
no_anon_password=YES
# Fichier de users
userlist_file=/etc/vsftpd/user_list
# Chargement de la liste userlist_file
userlist_enable=YES
# On refuse les utilisateurs de la liste
userlist_deny=NO

~
-- INSERTION --
```

FIGURE 4.34 – Configuration du fichier vsftpd.conf

6. Réduire les droits sur le fichier de configuration

```
[root@fedora vsftpd]# chmod 600 /etc/vsftpd/vsftpd.conf
```

FIGURE 4.35 – Réduction des droits sur le fichier de configuration

7. Changer les configurations du Boolean vue qu'on utilise un SELinux

```
[root@fedora vsftpd]# setsebool -P ftpd_full_access on
```

FIGURE 4.36 – Changement des configuration du boolean

8. Ajouter l'utilisateur hadoop

```
[root@fedora vsftpd]# cd /etc/vsftpd/  
[root@fedora vsftpd]# ls  
user_list vsftpd.conf vsftpd.conf.default vsftpd_conf_migrate.sh vsftpd.conf.old vsftpd_user_conf  
[root@fedora vsftpd]# vi user_list  
[root@fedora vsftpd]#
```

FIGURE 4.37 – Accès au fichier user list

```
root
bin
daemon
adm
lp
sync
shutdown
halt
mail
news
uucp
operator
games
nobody
hadoop
~
~
~
~
~
~
"user_list" 21L, 368B
```

FIGURE 4.38 – Ajout de l'utilisateur hadoop

9. Redémarrer les services après les configurations.

```
[root@fedora vsftpd]# sudo systemctl enable vsftpd
[root@fedora vsftpd]# sudo systemctl restart vsftpd
```

FIGURE 4.39 – Redémarrage des services vsftpd

### 4.2.2 Importation du fichier de données

Une fois que le serveur FTP est bien installé ainsi FileZilla sur la machine physique , nous transférons le fichier de données vers le serveur fedora en suivant les étapes suivantes :

1. Utiliser FileZilla pour transférer le fichier de données(datas.txt)

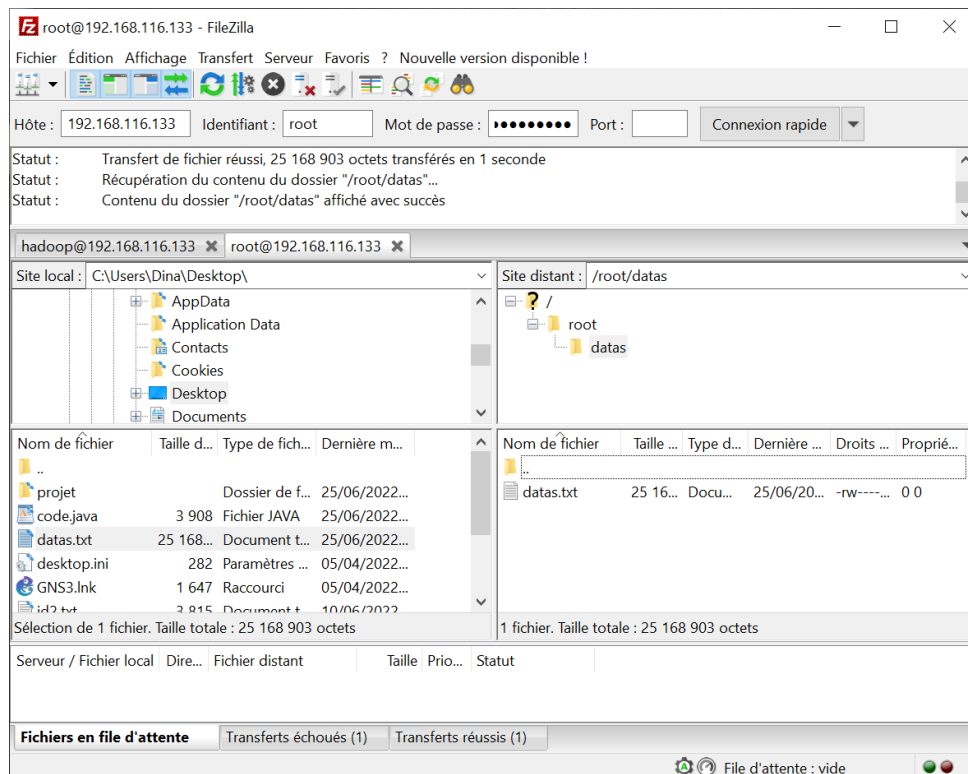


FIGURE 4.40 – Transfert du fichier de données

2. Déplacer le fichier "datas.txt" vers le répertoire /datas

```
[root@fedora ~]# ls
anaconda-ks.cfg  datas
[root@fedora ~]# cd datas
[root@fedora datas]# ls
datas.txt
[root@fedora datas]# mv datas.txt /datas
[root@fedora datas]#
```

FIGURE 4.41 – Déplacement du fichier de données

### 4.2.3 Insertion des données dans HDFS

HDFS signifie **Hadoop Distributed File System**, il s'agit d'un système de stockage et de traitement de fichiers adapté au Big Data puisqu'il permet la gestion de données volumineuses.

Pour pouvoir insérer le fichier dans ce système de fichiers Hadoop, nous suivons les étapes mentionnées ci-dessous :

1. Créer un répertoire d'entrée /datas

```
[root@fedora ~]# hadoop fs -mkdir /datas
```

FIGURE 4.42 – Création d'un répertoire d'entrée

2. Stocker le fichier de données dans le HDFS

```
[hadoop@fedora ~]$ hadoop fs -copyFromLocal /datas/datas.txt /datas
```

FIGURE 4.43 – Stockage de fichier de données

3. Vérifier si le fichier a été bien enregistré

```
[root@fedora hadoop]# hadoop fs -ls /datas
Found 1 items
-rw-r--r--  1 root supergroup  25168903 2022-06-01 13:52 /datas/datas.txt
[root@fedora hadoop]#
```

FIGURE 4.44 – Vérification du fichier stocké

## 4.3 Programme MapReduce

A partir de l'API Hadoop de MapReduce, nous allons écrire le code en langage java qui correspondra à :

**Opération Map :** nous utilisons les différentes variables hadoop (IntWritable, texte, LongWritable) selon les entrée/sortie à gérer, et nous créons une classe FDMapper qui hérite de la classe MapReduceBase qui implémente l'opération Mapper.

**Opération Reduce :** dans cette opération , seul le type IntWritable de Hadoop va être utiliser. ensuite nous écrivons la classe FDReducer qui implémente l'opération Reduce. Enfin, nous écrivons le code correspondant au FDDriver ainsi que la méthode main.

### 4.3.1 Code Source :

Nous allons examiner le code exécutable ci-dessous :

```
1 import java.io.IOException;
2 import java.util.Iterator;
3 import org.apache.hadoop.io.IntWritable;
4 import org.apache.hadoop.io.WritableComparable;
5 import java.io.DataInput;
6 import java.io.DataOutput;
7 import org.apache.hadoop.io.LongWritable;
8 import org.apache.hadoop.io.Text;
9 import org.apache.hadoop.mapred.MapReduceBase;
10 import org.apache.hadoop.mapred.Mapper;
11 import org.apache.hadoop.mapred.OutputCollector;
12 import org.apache.hadoop.mapred.Reporter;
13 import org.apache.hadoop.mapred.Reducer;
14 import org.apache.hadoop.conf.Configured;
15 import org.apache.hadoop.fs.Path;
```



```
16 import org.apache.hadoop.mapred.FileInputFormat;
17 import org.apache.hadoop.mapred.FileOutputFormat;
18 import org.apache.hadoop.mapred.JobClient;
19 import org.apache.hadoop.mapred.JobConf;
20 import org.apache.hadoop.util.Tool;
21 import org.apache.hadoop.util.ToolRunner;
22
23
24 public class id2 {
25     public static class KeyPairIdLevel implements WritableComparable <
        KeyPairIdLevel > {
26
27         private int id;
28         private String level;
29
30         public void setId(int id){
31             this.id = id;
32         }
33
34         public void setLevel(String level){
35             this.level = level;
36         }
37
38         public String toString(){
39             return "id=" + id + "\tniveau=" + level;
40         }
41
42         public void write(DataOutput out) throws IOException {
43             out.writeInt(id);
44             out.writeInt(level.length());
45             out.writeBytes(level);
46         }
47
48         public void readFields(DataInput in) throws IOException {
49             this.id = in.readInt();
50             int len = in.readInt();
51             byte[] blevel = new byte[len];
52             for (int i =0; i<len; i++){
53                 blevel[i] = in.readByte();
54             }
55             this.level = new String(blevel);
56         }
57         public static KeyPairIdLevel read(DataInput in) throws IOException {
58             KeyPairIdLevel w = new KeyPairIdLevel();
59             w.readFields(in);
60             return w;
61         }
62
63         public int compareTo(KeyPairIdLevel other) {
64             if (this.id < other.id)
65                 return -1;
66             else if (this.id > other.id)
67                 return 1;
68             else
69                 return this.level.compareTo(other.level);
70         }
71     }
72 }
```

```

73 public static class FDMapper extends MapReduceBase implements Mapper <
    LongWritable, Text, KeyPairIdLevel, IntWritable> {
74     @Override
75     public void map(LongWritable key, Text value, OutputCollector <
        KeyPairIdLevel, IntWritable> output, Reporter rep) throws IOException
        {
76         String data[] = value.toString().split("\t");
77         int id= -1;
78         KeyPairIdLevel kpIdLevel = new KeyPairIdLevel();
79         try{
80             id = Integer.parseInt(data[3]);
81             kpIdLevel.setId(id);
82             kpIdLevel.setLevel(data[0]);
83         }
84         catch(Exception e){}
85         if(id != -1){
86             output.collect(kpIdLevel, new IntWritable(1));
87         }
88     }
89 }
90
91 public static class FDReducer extends MapReduceBase implements Reducer
    <KeyPairIdLevel, IntWritable, Text, Text> {
92     @Override
93     public void reduce(KeyPairIdLevel key, Iterator <IntWritable>
        value, OutputCollector
94         <Text, Text> output, Reporter rep) throws IOException {
95         int sum = 0;
96         while (value.hasNext())
97         {
98             IntWritable i = value.next();
99             sum += i.get();
100         }
101         output.collect(new Text(key.toString()), new Text("
        nombre_de_repetitions=" + sum));
102     }
103 }
104
105 public static class FDDriver extends Configured implements Tool {
106     @Override
107     public int run(String[] args) throws Exception {
108         if (args.length < 2)
109         {
110             System.out.println("Please enter valid arguments");
111             return -1;
112         }
113         JobConf conf = new JobConf(FDDriver.class);
114         FileInputFormat.setInputPaths(conf, new Path(args[0]));
115         FileOutputFormat.setOutputPath(conf, new Path(args[1]));
116         conf.setMapperClass(FDMapper.class);
117         conf.setReducerClass(FDReducer.class);
118         conf.setMapOutputKeyClass(KeyPairIdLevel.class);
119         conf.setMapOutputValueClass(IntWritable.class);
120         conf.setOutputKeyClass(Text.class);
121         conf.setOutputValueClass(Text.class);
122         JobClient.runJob(conf);
123         return 0;
124     }

```

```

125 }
126
127 public static void main(String args[]) throws Exception {
128     int exitcode = ToolRunner.run(new FDDriver(), args);
129     System.out.println(exitcode);
130 }
131 }

```

### 1. Compilation du programme :

- (a) Compilation : les différents paths après `-classpath` correspondent au chemin vers les quatre fichiers jar suivant :

**hadoop-common-2.7.2.jar**  
**hadoop-mapreduce-client-core-2.7.2.jar**  
**common-cli-1.2.jar**  
**hadoop-mapreduce-client-core-2.7.2.jar**  
**common-cli-1.2.jar**

```

[root@fedora app]# javac -classpath /home/hadoop/hadoop/share/hadoop/common/hadoop-
common-2.7.2.jar:/home/hadoop/hadoop/share/hadoop/mapreduce/hadoop-mapreduce-client
-core-2.7.2.jar:/home/hadoop/hadoop/share/hadoop/common/lib/common-cli-1.2.jar:hadoop-
-mapreduce-client-core-2.7.2.jar:/home/hadoop/hadoop/share/hadoop/common/lib/common-cli-1.2.j
ar: /app/id2.java
[root@fedora app]#

```

FIGURE 4.45 – Compilation du programme MapReduce

- (b) Création d'un fichier jar :

```

[root@fedora app]# jar -cvf id2.jar *.class
manifeste ajouté
ajout : id2\FDDriver.class(entrée = 1600) (sortie = 842)(compression : 47 %)
ajout : id2\FDMapper.class(entrée = 1873) (sortie = 761)(compression : 59 %)
ajout : id2\FDReducer.class(entrée = 1614) (sortie = 663)(compression : 58 %)
ajout : id2.class(entrée = 718) (sortie = 469)(compression : 34 %)
[root@fedora app]#

```

FIGURE 4.46 – Création d'un fichier jar

### 2. Exécution du programme :

Le programme est exécuté comme suit :

```

[root@fedora app]# su hadoop
[hadoop@fedora app]$ hadoop jar /app/id2.jar id2 /datas/data2.txt /resultas
22/06/17 12:11:38 INFO Configuration.deprecation: session.id is deprecated. Instead, use dfs.metrics.session-id
22/06/17 12:11:38 INFO jvm.JvmMetrics: Initializing JVM Metrics with processName=JobTracker, sessionId=
22/06/17 12:11:38 INFO jvm.JvmMetrics: Cannot initialize JVM Metrics with processName=JobTracker, sessionId= - alrea
y initialized
22/06/17 12:11:38 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement th
Tool interface and execute your application with ToolRunner to remedy this.
22/06/17 12:11:38 INFO mapred.FileInputFormat: Total input paths to process : 1

```

```

File System Counters
  FILE: Number of bytes read=1226826
  FILE: Number of bytes written=2400392
  FILE: Number of read operations=0
  FILE: Number of large read operations=0
  FILE: Number of write operations=0
  HDFS: Number of bytes read=50337656
  HDFS: Number of bytes written=848
  HDFS: Number of read operations=13
  HDFS: Number of large read operations=0
  HDFS: Number of write operations=4
Map-Reduce Framework
  Map input records=211200
  Map output records=60968
  Map output bytes=487744
  Map output materialized bytes=609686
  Input split bytes=89
  Combine input records=0
  Combine output records=0
  Reduce input groups=119
  Reduce shuffle bytes=609686
  Reduce input records=60968
  Reduce output records=119
  Spilled Records=121936
  Shuffled Maps =1
  Failed Shuffles=0
  Merged Map outputs=1
  GC time elapsed (ms)=79
  Total committed heap usage (bytes)=450887680
Shuffle Errors
  BAD_ID=0
  CONNECTION=0
  IO_ERROR=0
  WRONG_LENGTH=0
  WRONG_MAP=0
  WRONG_REDUCE=0
File Input Format Counters
  Bytes Read=25168828
File Output Format Counters
  Bytes Written=848
0
[hadoop@fedora app]$

```

FIGURE 4.47 – Exécution du programme MapReduce

## 3. Obtention des résultats :

(a) Le contenu du répertoire de sortie :

```

[hadoop@fedora app]$ hadoop fs -ls /resultas
Found 2 items
-rw-r--r--  1 hadoop supergroup          0 2022-06-17 12:11 /resultas/_SUCCESS
-rw-r--r--  1 hadoop supergroup        848 2022-06-17 12:11 /resultas/part-00000
[hadoop@fedora app]$

```

FIGURE 4.48 – Le contenu du répertoire de sortie

(b) Examinons les résultats :

Le résultat affiche l’ID de l’évènement, son niveau ainsi que son nombre de répétition.

```
id=0    niveau=Information    nombre_de_repetitions=10
id=1    niveau=Erreur    nombre_de_repetitions=16
id=1    niveau=Information    nombre_de_repetitions=181
id=2    niveau=Information    nombre_de_repetitions=1
id=3    niveau=Information    nombre_de_repetitions=18
id=6    niveau=Information    nombre_de_repetitions=82
id=7    niveau=Erreur    nombre_de_repetitions=213
id=9    niveau=Erreur    nombre_de_repetitions=1
id=12   niveau=Information    nombre_de_repetitions=13
id=13   niveau=Information    nombre_de_repetitions=7
id=14   niveau=Information    nombre_de_repetitions=9
id=15   niveau=Information    nombre_de_repetitions=23
id=16   niveau=Information    nombre_de_repetitions=232
id=18   niveau=Information    nombre_de_repetitions=9
id=19   niveau=Information    nombre_de_repetitions=7
id=20   niveau=Erreur    nombre_de_repetitions=220
id=20   niveau=Information    nombre_de_repetitions=9
id=24   niveau=Avertissement    nombre_de_repetitions=3
id=25   niveau=Information    nombre_de_repetitions=9
id=26   niveau=Information    nombre_de_repetitions=2
id=27   niveau=Avertissement    nombre_de_repetitions=1
id=27   niveau=Information    nombre_de_repetitions=9
id=30   niveau=Erreur    nombre_de_repetitions=1
id=32   niveau=Information    nombre_de_repetitions=19
id=34   niveau=Information    nombre_de_repetitions=214
id=35   niveau=Avertissement    nombre_de_repetitions=1149
id=35   niveau=Information    nombre_de_repetitions=9
```

FIGURE 4.49 – Résultats du programme MapReduce

## Conclusion

Dans ce chapitre, nous avons mis en œuvre l'exploitation de la solution. Tout d'abord, nous avons émulé la topologie du réseau de l'entreprise sous GNS3. Ensuite nous avons élaboré les étapes nécessaires à l'insertion des données dans le HDFS. Après cela nous avons représenté le programme MapReduce que nous avons créé. Enfin, nous avons présenté les étapes nécessaires à la compilation et à l'exécution de notre programme afin d'obtenir les résultats voulus. ‘

# Conclusion générale et perspectives

Ce projet nous a été d'un apport indéniable en matière de connaissances et compétences acquises sur les technologies Big Data grâce à la mise en place d'une approche exploitant ses outils.

Afin de répondre aux problèmes liés à l'exploitation de données logs générées à cause des activités au sein d'un système d'informations digital, nous avons présenté un travail organisé en deux parties, théorique et pratique chacune comprend deux chapitres. L'aspect théorique a fait l'objet de la première partie, s'est consacré sur les systèmes d'information au cœur des métiers dans laquelle nous avons décrit en manière générale le système d'information et ses composants dans le premier chapitre, puis dans le deuxième nous avons passé à l'étude de l'organisme d'accueil ainsi que les différents équipements informatiques que l'entreprise dispose.

La deuxième partie s'est centrée sur l'aspect pratique dont nous avons implémenté la mise en place de la solution dans le premier chapitre en élaborant les étapes nécessaires à l'installation et configuration des outils et des environnements utilisés. Dans le second, nous avons déployé et exploité la solution en proposant une architecture à un modèle distribué et parallèle et définissant les fonctions appropriées.

À la suite des résultats obtenus, plusieurs perspectives sont envisageables pour notre modeste expérimentation. Parmi les améliorations possibles nous citons :

- Mettre en place un cluster elastic search pour rechercher les données ;
- visualiser les données en utilisant un cluster Kibana ;
- Automatiser le déploiement avec un cluster Kubernetes ;

# Bibliographie

- [1] <https://baticompos.dz/>.
- [2] Le document interne de baticompos spa.
- [3] Qu'est-ce qu'un fichier log et comment l'utiliser ?  
<https://www.1min30.com/dictionnaire-du-web/fichier-log-definition>.
- [4] Dr.Anton A, Chuvakin Kevin J, and Schmidt Chrisopher. *Logging and Log Management*. 2014.
- [5] Syed Muhammad Fahad Akhtar. *Big Data Architect's Handbook*. Packt Publishing Ltd, june 2018.
- [6] Borri Alain, Keller Yvan, and Vienney Alexandre. *Livre Blanc : Comment appréhender l'acquisition et le déploiement d'un logiciel TMS*. aout 2014.
- [7] Pierre Le Bel. *Gérer un parc informatique*. Pratikio.
- [8] Xavier Bouche and Henri Gillares-Callia. *SAN et NAS Les infrastructures des réseaux de stockage*. Dunod.
- [9] Rudi Bruchez. *Les bases de données NoSQL et le Big Data*. 2015.
- [10] Brendan Burns, Joe Beda, and Kelsey Hightower. *Kubernetes : Up and Running*. O'Reilly Media, Inc, 1 edition, 2017.
- [11] Sylvie Delattre, Viviane Deschamps, Élodie Couratier, Maïmouna Fossorier, Sophie Alexandre, and Pierre Lamblin. *LES MÉTIERS DES SYSTÈMES D'INFORMATION*. Apec.
- [12] Clinton Gormley and Zachary Tong. *Elasticsearch - The Definitive Guide*. January 23, 2015.
- [13] Jain.A and dubes. *R.C "Algorithms for Clustering data"*. 1988.
- [14] Frédéric Lau. *LES FONDAMENTAUX DU CLOUD COMPUTING*. Cigref.
- [15] Pirmin Lemberger, Marc Batty, Médéric Morel, and Jean-Luc Raffaëlli. *Big data et machine learning : manuel du data scientist*. 2015.
- [16] FRÉDÉRIC Marc. *Mettre en oeuvre une GMAO*. aout 2003.
- [17] Chantal Morley, Marie Bia-Figueiredo, and Yves Gillette. *PROCESSUS MÉTIERS ET SYSTÈMES D'INFORMATION Gouvernance,management, modélisation*. Dunod.
- [18] Benjamin Renaut. *Introduction à MapReduce*. 2014.
- [19] Pramod J. Sadalage and Martin Fowler. *NoSQL Distilled "A Brief Guide to the Emerging World of Polyglot Persistence"*. August 2012.

- [20] Ralph M. Stair and George W. Reynolds. *Principles of Information Systems*. Course Technology, Cengage Learning.
- [21] Pang-Ning Tan, Steinbach M, and Vipin K. “*introduction to data minig*”. Addison Wesley, May 2005.
- [22] James Turnbull. *The Docker Book*. March 11, 2017.
- [23] Rijsbergen C. V. *Information retrieval*. 1979.
- [24] Tom White. *Hadoop : The Definitive Guide*. O'Reilly Media, Inc. June 2009.



# Glossaire

**C** : CLI (COMMAND-LINE INTERPRETER) :

comme son nom l'indique est Un interpréteur de commandes permettant à l'utilisateur d'interagir avec un programme à l'aide de commandes sous forme de lignes de texte.

**F** : FTP (FILE TRANSFER PROTOCOL) :

Le Serveur FTP est un logiciel, qui, installée sur une machine A (communément appelée elle aussi Serveur), permet à une autre machine B (usuellement appelée Client) de lire, modifier ou supprimer les fichiers de la machine A. La machine B, pour accéder aux fichiers, doit être équipée d'un logiciel dit « client FTP » comme Filezilla. La machine A est identifiée par son adresse IP. Cette adresse va permettre au logiciel client installé sur la machine B d'identifier le serveur et de se connecter à lui au travers les ports 20 (données) et 21 (commandes).

**K** : KPI (KEY PERFORMANCE INDICATOR) :

signifie indicateur de performance clé, une mesure quantifiable de la performance dans le temps pour un objectif spécifique

**P** : PING (PACKET INTERNET GROPER) :  
est une commande informatique, permettant de vérifier si une machine distante répond et qu'elle est accessible par le réseau grâce à l'envoi de paquets, elle mesure également le temps mis pour recevoir une réponse.