

---

./img/



République Tunisienne  
Ministère de l'Enseignement Supérieur  
et de la Recherche Scientifique  
Université de Carthage  
Institut Supérieur de Sciences Appliquées et de Technologie  
de Mateur



## RAPPORT DE PROJET DE FIN D'ÉTUDES

Présenté en vue de l'obtention  
de la Licence Appliquée en Technologies de l'Information et de la Communication  
Spécialité : Télécommunications

Par

Zouaoui Nour El Houda & Harbaoui Roudayna

---

# Mise en place de VMware Tanzu sur un environnement virtualisé vsphere

---

Soutenue le 09 juin 2022 devant la commission de jury composé de :

Président :

Dr. Ben Salem Aicha

Rapporteur :

Dr. Ben Salah Tarek

Encadrant académique :

Dr. Hrizi Hafedh

Encadrant professionnel :

Mr. Ghezaiel Aymen

Réalisé au sein de 3S





---

J'autorise les étudiantes à faire le dépôt de son rapport de stage en vue d'une soutenance.

Encadrant professionnel, **Mr.GHEZAIEL AYMEN**

**Signature et cachet**

J'autorise les étudiantes à faire le dépôt de son rapport de stage en vue d'une soutenance.

Encadrant académique, **Mr.HRIZI HAFEDH**

**Signature**

# Dédicaces

Je dédie ce travail à :

## **A mon très cher père :**

De tous les pères, tu es le meilleur.

Tu as été et tu seras toujours un exemple pour moi, par tes qualités humaines, ta persévérance et perfectionnisme. Vous êtes et vous resterez pour moi ma référence, la lumière qui illumine mon chemin. Ce travail est le résultat de l'esprit de sacrifice dont vous avez fait preuve, de l'encouragement et le soutien que vous ne cessez de manifester, j'espère que vous y trouverez les fruits de votre semence et le témoignage de ma grande fierté de vous avoir comme père. Aucune dédicace ne saurait exprimer mes respects, ma reconnaissance et mon profond amour. Papa, tu me manques trop, repose en paix ..

## **A ma très chère mère :**

Aucune dédicace très chère maman, ne pourrait exprimer la profondeur des sentiments que j'éprouve pour vous, vos sacrifices innombrables et votre dévouement firent pour moi un encouragement. Vous m'avez aidé et soutenu pendant de nombreuses années avec à chaque fois une attention renouvelée. Puisse Dieu, tout puissant vous combler de santé, de bonheur et vous procurer une longue vie.

## **A mes amies : Meriem , Sabiha, Sara**

Je ne peux trouver les mots justes et sincères pour vous exprimer mon affection et mes pensées, vous êtes pour moi des sœurs et des amies sur qui je peux compter. En témoignage de l'amitié qui nous unit et des souvenirs de tous les moments que nous avons passés ensemble, je vous dédie ce travail et je vous souhaite une vie pleine de santé et de bonheur.

Nour El Houda Zouaoui

# Dédicaces

Je tiens à remercier chaleureusement, tous mes proches et tous ceux qui, de près ou de loin, m'ont apporté leurs sollicitudes pour accomplir ce Travail :

## **A mon cher père :**

Aucun dévouement ne peut manifester l'amour, l'estime, le dévouement et le respect que nous avons toujours en vous.

## **A ma chère mère :**

Qui est toujours à mes côtés malgré la distance. Elle est mon symbole de bonté par excellence, source de tendresse.

## **Ma cousine Asma**

Toujours présente avec la Privation et les sacrifices. Elle est toujours là avec son appui moral. Qui n'est pas cessé de me conseiller, encourager et soutenir tout au long de mes études. Que dieu la protège et l'offre la chance et le bonheur. .

## **A Mes chers amis**

Qui ont été toujours présents avec leurs sacrifices consentis...

Merci pour leurs amours et leurs encouragements.

Roudaina Harbaoui

# Remerciements

*L'équipe du projet de fin d'études aimerait remercier chaleureusement les personnes Suivantes qui ont joué un rôle clé dans la réalisation du projet de fin d'études.*

*Tout d'abord, nos vifs remerciements s'adressent à notre encadrant le Docteur **M.Hafedh hrizi**, le directeur d'Institut Supérieur des Sciences Appliquées et de la Technologie de Mateur, pour son encadrement, son soutien, ses suggestions pertinentes, ainsi que pour ses conseils instructifs durant toute la période de ce travail, pour son temps, sa patience, ses conseils, ses critiques les plus constructives et sa participation à l'avancement du projet et l'élaboration du rapport.*

*Nos remerciements les plus sincères et nos profond respect s'adressent aussi à **M.Aymen Ghzaïel**, notre encadrant à 3S, d'avoir fait preuve d'une grande disponibilité à notre égard, son aide précieuse et ses conseils pertinents ainsi que pour ses critiques constructives tout au long de ce projet.*

*Nous tenons à remercier également l'ingénieur **M. Ali Ben Rejeb** du département d'infrastructure, pour nous a aidé à mieux comprendre l'architecture et le fonctionnement de notre travail et pour nous avoir aiguillés au long du projet et qui nous accueillir et nous accordé beaucoup de temps et d'efforts dans le but de réussir notre projet.*

*Notre vif remerciement aux présidents et aux membres de jury pour l'honneur qu'ils nous font en acceptant de juger ce travail.*

# Table des matières

<b>Introduction générale</b>	<b>1</b>
<b>1 Présentation du cadre du projet</b>	<b>2</b>
1.1 Introduction . . . . .	3
1.2 Présentation de l'organisme d'accueil [1] . . . . .	3
1.2.1 Le groupe standard sharing software . . . . .	3
1.2.2 Identification . . . . .	4
1.2.3 Organigramme de la société . . . . .	4
1.2.4 Axes d'activités et de services . . . . .	5
1.3 Contexte du projet . . . . .	5
1.3.1 Problématique . . . . .	6
1.3.2 Etude de l'existant . . . . .	6
1.3.3 Solution proposée . . . . .	6
1.3.4 Identification des acteurs . . . . .	8
1.3.5 Découpage du projet . . . . .	9
1.3.6 Diagramme de GANTT . . . . .	9
1.4 Conclusion . . . . .	10
<b>2 Étude théorique</b>	<b>11</b>
2.1 Introduction . . . . .	12
2.2 Virtualisation . . . . .	12
2.2.1 Les Avantages de la Virtualisation . . . . .	13
2.2.2 les inconvénients de virtualisation . . . . .	13
2.2.3 Les hyperviseurs . . . . .	13
2.3 La conteneurisation . . . . .	17
2.3.1 Les conteneurs Docker . . . . .	18
2.4 Comparaison entre virtualisation et conteneurisation . . . . .	19
2.5 L'orchestration du conteneur . . . . .	20
2.5.1 Outils d'orchestration : Kubernetes . . . . .	20
2.6 Les microservices . . . . .	22
2.7 Tanzu . . . . .	23
2.7.1 fonctionnement du VMware Tanzu . . . . .	24



---

2.7.2	Un cluster Tanzu Kubernetes . . . . .	25
2.7.3	VMware Tanzu Kubernetes Grid (TKG) . . . . .	25
2.7.4	Le Cluster API . . . . .	25
2.7.5	Le cluster de gestion . . . . .	25
2.7.6	TKGS . . . . .	26
2.8	Conclusion . . . . .	26
<b>3</b>	<b>Mise en œuvre du projet</b>	<b>27</b>
3.1	Introduction . . . . .	28
3.2	Les choix techniques : . . . . .	28
3.2.1	Mise en réseaux de ESXI . . . . .	29
3.3	Déploiement de Tanzu sur VDS Networking . . . . .	34
3.3.1	Bibliothèque de contenu VMware . . . . .	34
3.3.2	Politique de stockage . . . . .	35
3.3.3	HA Proxy . . . . .	35
3.4	cluster superviseur . . . . .	35
3.5	Espace du nom . . . . .	36
3.6	Conclusion . . . . .	37
<b>4</b>	<b>Implémentation de la solution</b>	<b>38</b>
4.1	Introduction . . . . .	39
4.2	Environnement matériels . . . . .	39
4.3	Installation de VMware ESXI . . . . .	39
4.3.1	Les vSwitchs . . . . .	41
4.4	Installation de Vcenter . . . . .	43
4.4.1	Le schéma de déploiement . . . . .	44
4.5	Configuration du Virtuel Distributed Switch (VDS) . . . . .	45
4.6	Configuration de vmotion . . . . .	46
4.7	Bibliothèque de contenu VMware . . . . .	46
4.8	Configuration du politique de stockage . . . . .	47
4.9	Configuration du HA Proxy . . . . .	48
4.10	Configurer la gestion de la charge de travail . . . . .	51
4.11	Création du namespace . . . . .	52
4.11.1	Tanzu kubernetes cluster . . . . .	53
4.12	L'application : ACME-FITNESS . . . . .	54

---

4.12.1	Présentation . . . . .	54
4.12.2	Les étapes de configuration : . . . . .	54
4.13	Conclusion . . . . .	59
	<b>Conclusion générale</b>	<b>60</b>
	<b>Bibliographie</b>	<b>61</b>

# Table des figures

1.1	Logo L'entreprise 3S . . . . .	3
1.2	Organigramme de la société . . . . .	4
1.3	vSphere avec Tanzu[2] . . . . .	7
1.4	Pile d'applications actuelle . . . . .	8
1.5	Organigramme des tâches . . . . .	9
1.6	Représentation du diagramme de Gantt . . . . .	9
2.1	Le Principe de Virtualisation [4] . . . . .	12
2.2	Un hyperviseur de type1 . . . . .	14
2.3	Un hyperviseur de type 2 [5] . . . . .	15
2.4	KVM . . . . .	16
2.5	Xen . . . . .	16
2.6	Comparaison entre les différents types d'hyperviseurs[6] . . . . .	17
2.7	Architecture de la conteneurisation [7] . . . . .	18
2.8	Docker . . . . .	18
2.9	Comparaison entre les conteneurs et les machines virtuelles [2] . . . . .	20
2.10	Logo Kubernetes . . . . .	21
2.11	Elements du Kubernetes . . . . .	22
2.12	vmware tanzu . . . . .	23
2.13	vsphere with Tanzu[9] . . . . .	24
2.14	cluster de gestion . . . . .	25
3.1	Composants de Vsphere . . . . .	28
3.2	Architecture de Vmware ESXI . . . . .	29
3.3	Commutateur virtuel . . . . .	30
3.4	Topologie Vswitch . . . . .	30
3.5	commutateur Standard . . . . .	31
3.6	Commutateur distribuer . . . . .	31
3.7	Vmotion . . . . .	32
3.8	VMFS et ISCSI . . . . .	33
3.9	Vmware Vcenter . . . . .	34
3.10	HA : Hight Availability . . . . .	35

3.11	Architecture réseau du HA Proxy . . . . .	36
3.12	Cluster superviseur . . . . .	36
4.1	Installation de VMware ESXI . . . . .	40
4.2	interface de VMware ESXI . . . . .	40
4.3	Résultat du nslookup . . . . .	41
4.4	Configuration de iSCSI . . . . .	42
4.5	Représentation graphique du vSwitch0 . . . . .	43
4.6	Etape 2 avec succès . . . . .	44
4.7	Interface Vcenter . . . . .	44
4.8	Schéma de déploiement . . . . .	45
4.9	Ajout des ESXI . . . . .	45
4.10	Topologie de Vds . . . . .	46
4.11	Option Vmotion . . . . .	46
4.12	Adresse Vmotion pour ESXI . . . . .	47
4.13	Bibliothèque de contenu VMware . . . . .	47
4.14	Configuration du politique de stockage . . . . .	48
4.15	Réseau frontal . . . . .	49
4.16	Architecture du Réseau frontal . . . . .	50
4.17	Plage d'adresses IP d'équilibreur de charge . . . . .	50
4.18	Service anyip-routes.service n'est pas activé . . . . .	51
4.19	Service anyip-routes.service activé . . . . .	51
4.20	Déploiement des vms du cluster de superviseurs. . . . .	51
4.21	Résultat du configuration . . . . .	52
4.22	Résultat de la configuration . . . . .	52
4.23	Nœuds du plan de contrôle . . . . .	53
4.24	Politique de stockage . . . . .	53
4.25	Fichier Yaml de Tanzu kubernetes cluster . . . . .	53
4.26	Tanzu kubernetes cluster . . . . .	53
4.27	le kubeconfig du cluster de charge de travail . . . . .	54
4.28	Architecture d'application . . . . .	54
4.29	Déploiement de service de cart . . . . .	55
4.30	Cart activé . . . . .	55
4.31	Déploiement de service du catalog . . . . .	55

4.32	Catalog activé . . . . .	56
4.33	Déploiement de service de paiement . . . . .	56
4.34	Paie ment activé . . . . .	56
4.35	Déploiement de service de commande . . . . .	57
4.36	Service de commande activé . . . . .	57
4.37	Déploiement de service utilisateurs . . . . .	57
4.38	Service utilisateurs activé . . . . .	58
4.39	Déploiement de service frontal . . . . .	58
4.40	acme-fitness . . . . .	59

# Liste des tableaux

1.1	Indentification de 3S . . . . .	4
2.1	Tableau comparatif d’hyperviseurs Type 1 et Type2 : . . . . .	16
4.1	Configuration du DNS . . . . .	42
4.2	configuration du ISCSI . . . . .	42
4.3	Les interfaces réseaux . . . . .	45
4.4	les plages d’adresses des vlan . . . . .	49

# Liste des abréviations

- **OS**        =    **O**perating **S**ystem
- **vDS**       =    Virtuel Distributed Switch
- **vSS**       =    Virtuel Standars Switch
- **vmk**       =    VMkernel

# Introduction générale

Dans les entreprises modernes dotées des systèmes informatisés, la virtualisation a fourni une solution au problème d'utilisation des ressources, Bien qu'il soit moins connu, le domaine de la virtualisation a pris une place très importante. Les avancées technologiques ont eu une influence sur l'augmentation exponentielle de l'utilisation des outils informatiques, L'intérêt principal de cette stratégie pour les entreprises réside dans le fait qu'elles gagnent le temps et de l'argent. La société Vmware a apporté une nouvelle façon de concevoir la virtualisation avec un système plus léger et plus simple d'utilisation que les machines virtuelles : le « conteneur ». Les intérêts liés à son utilisation vont de la réduction des coûts informatiques à l'optimisation du développement d'applications qui devient plus en plus développé grâce à la service Tanzu. Le service dans lequel nous travaillons est basé sur un plateforme d'orchestration « Kubernetes » qui est composées de plusieurs micro services.

L'objectif de ce projet est d'emploi ces services modernes dans 3S. Notre rapport s'articule autour de quatres chapitres.

- chapitre I : intitulé « Présentation du cadre du projet » englobe la présentation de la société 3S et le contexte du projet.
- chapitre II : « Etude théorique » nous abordons les concepts de virtualisation, de conteneurisation et les dépendances du service TANZU.
- Chapitre III : « Mise en œuvre du projet » décrit les différentes notions théoriques impliquées.
- chapitre IV : Détaille la conception de la solution proposée et test.



---

# PRÉSENTATION DU CADRE DU PROJET

---

## Plan

1	Introduction . . . . .	3
2	Présentation de l'organisme d'accueil [1] . . . . .	3
3	Contexte du projet . . . . .	5
4	Conclusion . . . . .	10

## 1.1 Introduction

Dans ce chapitre, nous allons mettre le stage dans son cadre général. Nous commençons tout d'abord par la représentation de l'entreprise d'accueil puis nous allons présenter la problématique et la description de la solution proposée, aussi nous aboutissons ainsi le diagramme de Gantt.

## 1.2 Présentation de l'organisme d'accueil [1]

### 1.2.1 Le groupe standard sharing software

Le groupe Standard Sharing software (3S) est une idée, un projet, toujours en cours car il embrasse l'avenir. C'est avant tout un intégrateur d'infrastructures et un prestataire de services à haute valeur technique, privilégiant l'innovation et l'anticipation, et apportant des solutions de bout en bout dans deux secteurs d'activités. Le Groupe 3S a d'abord constitué un réseau d'entreprise dans le domaine de l'informatique, puis un réseau unique dans le domaine de la maîtrise de l'énergie et des énergies renouvelables. En 1997, 3S a créé la filiale GlobalNet pour devenir l'un des principaux fournisseurs de services Internet. La société de 25 ans devient le premier partenaire tunisien de Cisco Systems. Selon plusieurs observateurs, ce partenariat concerne une période durant laquelle 3S parvient à s'imposer comme un acteur majeur en matière de réseaux informatiques. 3S possède d'autres filiales œuvrant dans le domaine de l'informatique à l'instar Technical Training (formation technologique), Get Wireless (étude de réseaux mobile), NX Tunisie (déploiement d'infrastructure GSM), SmartUp (développement de logiciel de transaction), High Tech (technologie de codes-barres et radio-identification), Pro2C (centre d'appel multi canal), Cyberesa (plateforme de réservation en ligne) et Africa Solar (spécialisée dans les études, les solutions photovoltaïques et l'éclairage LED). La figure suivante présente le logo de Standard Sharing Software (3S) .



**FIGURE 1.1 :** Logo L'entreprise 3S

### 1.2.2 Identification

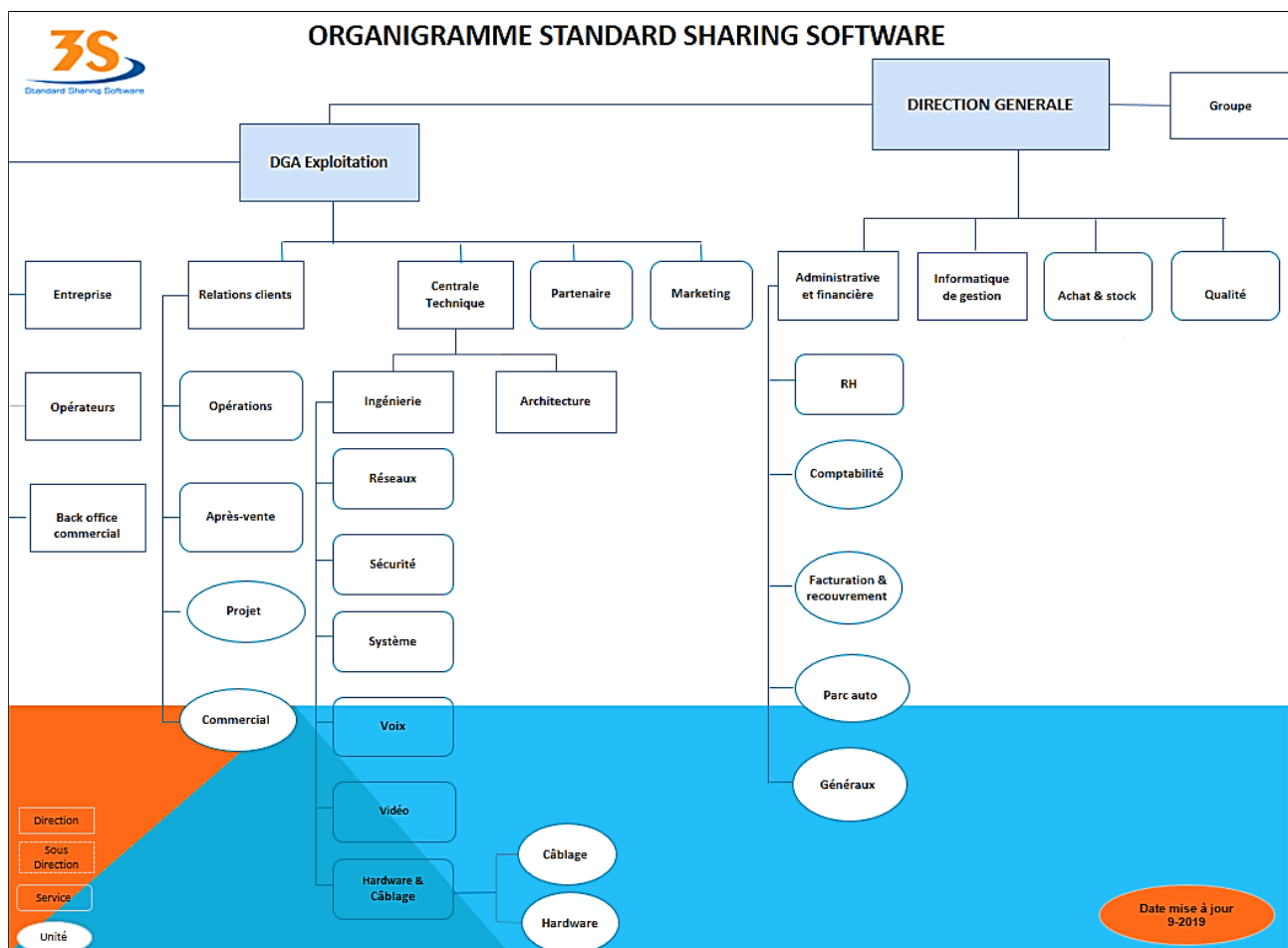
Le tableau 1 représente l'identification du groupe 3S telle que l'adresse, le site web, et le numéro du téléphone.

**TABLEAU 1.1** : Identification de 3S

Adresse du siège :	Immeuble 3S Lotissement Ennassim Montplaisir, Tunis 1073
Site web :	www.3S.com.tn
Téléphone :	71 111 100

### 1.2.3 Organigramme de la société

Comme illustré par la figure 2, 3S est composée de deux grandes unités : la DGA exploitation et la direction générale. La première est divisée en 7 sous directions, 10 services et 4 unités. La deuxième à son tour est composée de 2 sous directions, 3 services et 4 unités. Chaque partie prenante de l'organisation remplit un rôle incontestablement important.



**FIGURE 1.2** : Organigramme de la société

### 1.2.4 Axes d'activités et de services

Le groupe 3S maîtrise l'intégralité du cycle de vie du projet, des infrastructures réseau jusqu'aux applications, pour ambition de concrétiser les objectifs business et d'optimiser les performances de l'entreprise, tels que :

- Architecture, intégration et déploiement IT :

Le groupe 3S vous accompagne tout au long du cycle de vie de vos projets : Analyser vos besoins, apporter les recommandations, selon les tendances technologiques, concevoir, intégrer et déployer les solutions et les architectures, en respectant les bonnes pratiques et les méthodologies éprouvées.

- Consulting et stratégie IT :

Des experts vous aide à vous réorganiser aujourd'hui pour réussir demain. Des centaines de projets IT couronnés de succès sont à notre actif. Nous vous offrons conseil, accompagnement et meilleure stratégie IT, le tout converge vers votre objectif professionnel.

- Support et maintenance :

3S dédie une structure à la gestion de la relation client, en vue d'une meilleure expérience Client. Sa mission est d'assurer le support continu et la maintenance préventive et curative.

- Outsourcing et mission en régie :

Les ingénieurs dotés de certifications de haut niveau, vous libèrent des charges IT et vous permettent de vous focaliser davantage sur votre cœur de métier.

- Audit et mesures des performances :

Les expertise pointue nous habilite à auditer des SI complexes, et à vous fournir des KPI, ainsi que des mesures de performance pour un fonctionnement optimal de votre environnement digital

- Formation et Transfert de compétences :

Le savoir-faire des experts vous permet de réussir vos projets et d'améliorer votre ROI. Des formations certifiantes dans l'IT et la gouvernance sont assurées par notre centre de formation Advancia Training.

## 1.3 Contexte du projet

Cette partie comprend la présentation du cadre du projet, qui consiste à la réalisation d'une étude approfondie du système existant pour mettre en évidence les problèmes et proposer de nouvelles solutions pour remédier aux anomalies constatées lors de l'étude de l'existant.

### 1.3.1 Problématique

Au cours des dernières années, l'utilisation des technologies de virtualisation et de la conteneurisation a considérablement augmenté. Ceci est dû principalement aux avantages en termes d'efficacité d'utilisation des ressources, de robustesse et avoir un excellent niveau de service client. Malgré cette popularité, les applications n'offrent pas suffisamment d'agilité et de flexibilité. Il faut parfois des mois pour ajouter une fonctionnalité. D'autre part, Les systèmes distribués actuels sont constitués de plusieurs microservices deviennent de plus en plus complexes à maintenir, à gérer et à faire évoluer, c'est pourquoi Les opérations sur la pile d'applications actuelle peuvent être délicate et présentent dans ce cas plusieurs inconvénients au niveau de visibilité sur :

l'ensemble de la pile qui exécute des centaines d'applications.

l'environnement virtuel et la manière dont ils consomment les ressources.

La résolution des problèmes internes.

### 1.3.2 Etude de l'existant

Afin de dégager les objectifs de la solution cible, la mise en place d'un projet doit être précédée par une étude de l'existant. Cette étude nous permet d'optimiser les systèmes virtualisés basée sur VMware ESXI, nous avons mené des recherches sur les systèmes de la virtualisation utilisable par les entreprises. Cette recherche montre que pratiquement toutes les entreprises ont besoin d'une infrastructure virtuelle, capable d'atteindre son plein potentiel et de fournir un niveau de service exceptionnel, d'ailleurs, le taux d'utilisation des serveurs qui exécutent un système d'exploitation et une pile d'applications directement sur le matériel est généralement inférieur à 15 %. Depuis quelques années, la virtualisation entraine des problèmes de performance et de flexibilité c'est pourquoi les entreprises adoptent les technologies « Conteneurs et Kubernetes » qui permet d'accélérer leurs niveau de service, moderniser leurs applications, simplifier et automatiser la gestion des clusters de microservices d'applications conteneurisées, Mais cette notion simple cache un monde de complexité. Notre société souffre de cette complexité c'est pourquoi notre projet de fin d'étude nous amène à résoudre ce problème à travers la nouvelle technologie :La mise en place de VMware Tanzu qui permet aux entreprises de moderniser leurs applications et l'infrastructure sur laquelle elles fonctionnent.

### 1.3.3 Solution proposée

Notre objectif consiste à l'intégration du Kubernetes dans vSphere avec Tanzu, ce qui permet aux développeurs de continuer à utiliser les mêmes outils et interfaces standard qu'ils utilisaient pour créer des applications modernes. Les administrateurs vSphere en bénéficient également car ils

peuvent aider à gérer l'infrastructure Kubernetes en utilisant les mêmes outils et compétences qu'ils ont développés autour de vSphere. Pour jeter un regard sur ces deux mondes, nous avons introduit une nouvelle construction vSphere appelée espace de nom, qui permet aux administrateurs vSphere de créer un ensemble logique de ressources, de permissions et de politiques qui permettent une approche centrée sur les applications. Nous utilisons vSphere avec Tanzu pour créer un plan de contrôle Kubernetes directement sur la couche de l'hyperviseur. Ce plan de contrôle Kubernetes active les acteurs suivants dans vSphere :

- En tant qu'ingénieur DevOps :
  - Nous pouvons exécuter des charges de travail composées de conteneurs Kubernetes sur la même plate-forme avec des pools de ressources partagés au sein d'un Espace de nom de superviseur.
  - Dans vSphere avec Tanzu, nous pouvons également déployer des machines virtuelles standard créer et gérer plusieurs clusters Kubernetes dans un espace de nom et gérer leur cycle de vie à l'aide du Service Tanzu Kubernetes Grid.
- En tant qu'administrateur vSphere, nous avons une visibilité total sur les espaces vSphere, les machines virtuelles et les clusters Tanzu Kubernetes s'exécutant dans différents espaces de noms, leur positionnement dans l'environnement et la manière dont ils consomment les ressources.

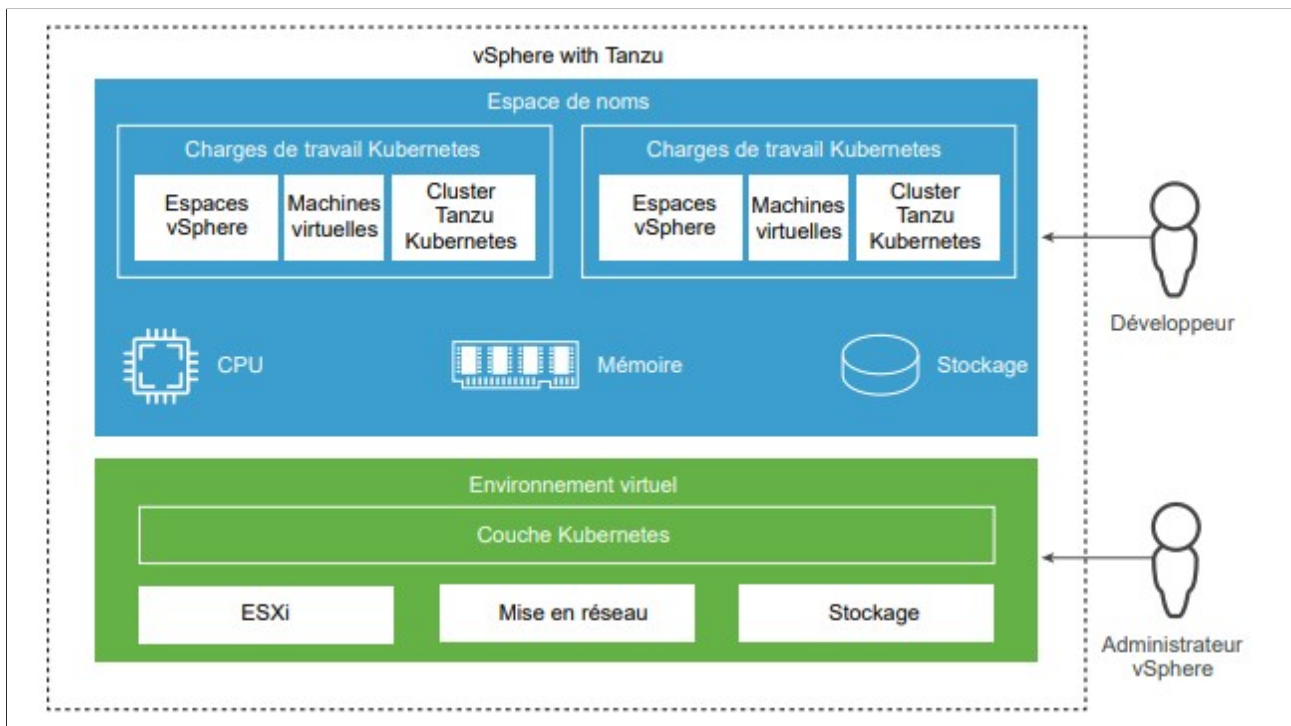


FIGURE 1.3 : vSphere avec Tanzu[2]

### 1.3.4 Identification des acteurs

Les acteurs sont des entités externes, telles que des humains ou des robots, qui interagissent avec le système. La même personne (ou robot) peut être plusieurs acteurs d'un système, c'est pourquoi les acteurs doivent d'abord être décrits par leurs rôles, qui décrivent les besoins et les capacités des acteurs. Ainsi, après avoir étudié les différentes interactions du système, nous avons distingué les acteurs suivants :

- Développeur d'applications : Il est chargé d'exécuter des espaces déployer et gérer des applications basées sur Kubernetes.
- Ingénieur DevOps : Il est chargé de contrôler l'infrastructure Kubernetes, avec des droits d'accées limités pour la gestion et la surveillance de l'environnement virtuel et de résoudre les problèmes liés aux ressources et autres problèmes.[3]
- Administrateur vSphere : Il est chargé de contrôler l'environnement virtuel sous-jacent, sur le placement des différents objets Kubernetes dans l'environnement virtuel et sur la manière dont ils consomment les ressources.

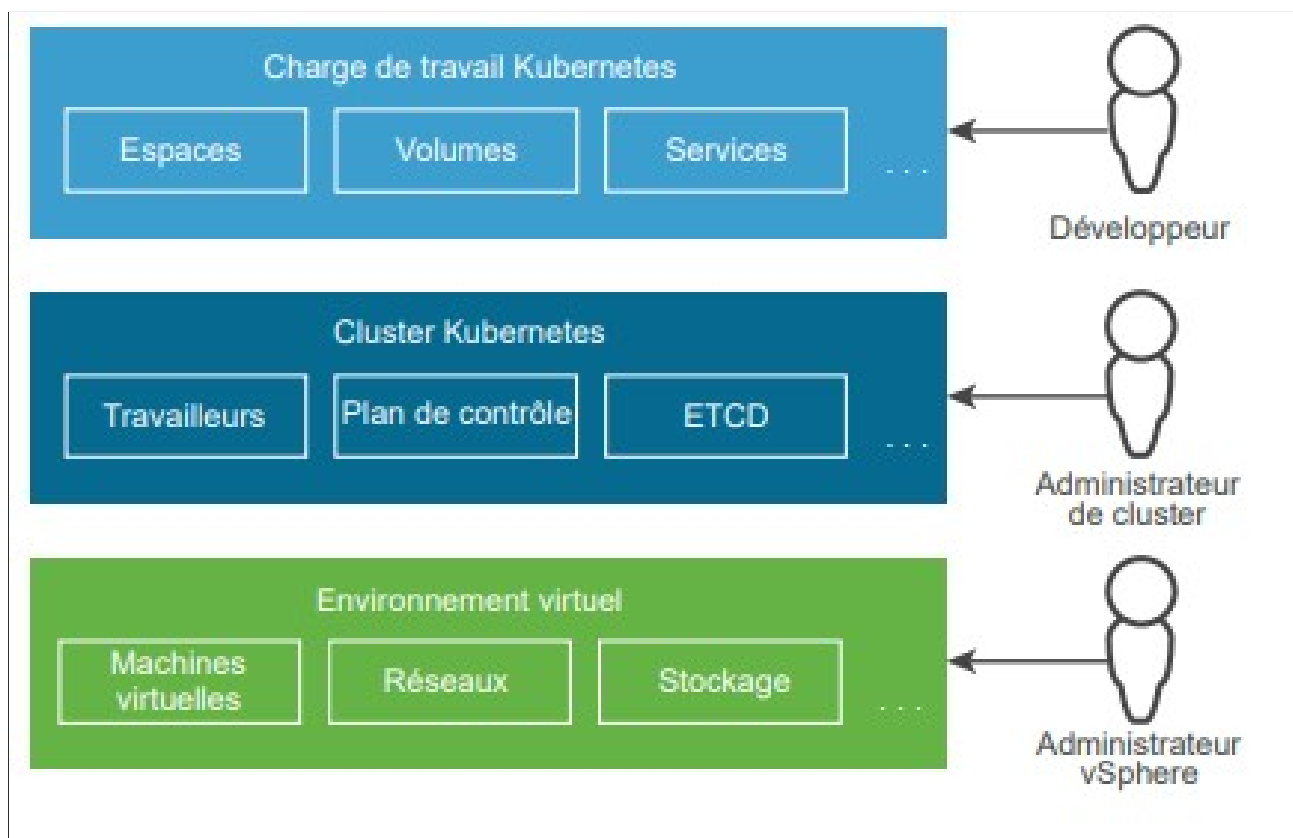


FIGURE 1.4 : Pile d'applications actuelle

### 1.3.5 Découpage du projet

Pour faciliter la conduite du projet un découpage en sous-ensembles maîtrisables est une étape essentielle donnant lieu à un bon aboutissement et à la réussite du projet. En effet, il permet de suivre efficacement le projet lors de son exécution. Le découpage d'un projet peut se faire selon plusieurs critères à savoir les Fonctionnalités, les spécificités techniques, les types de tâches etc.

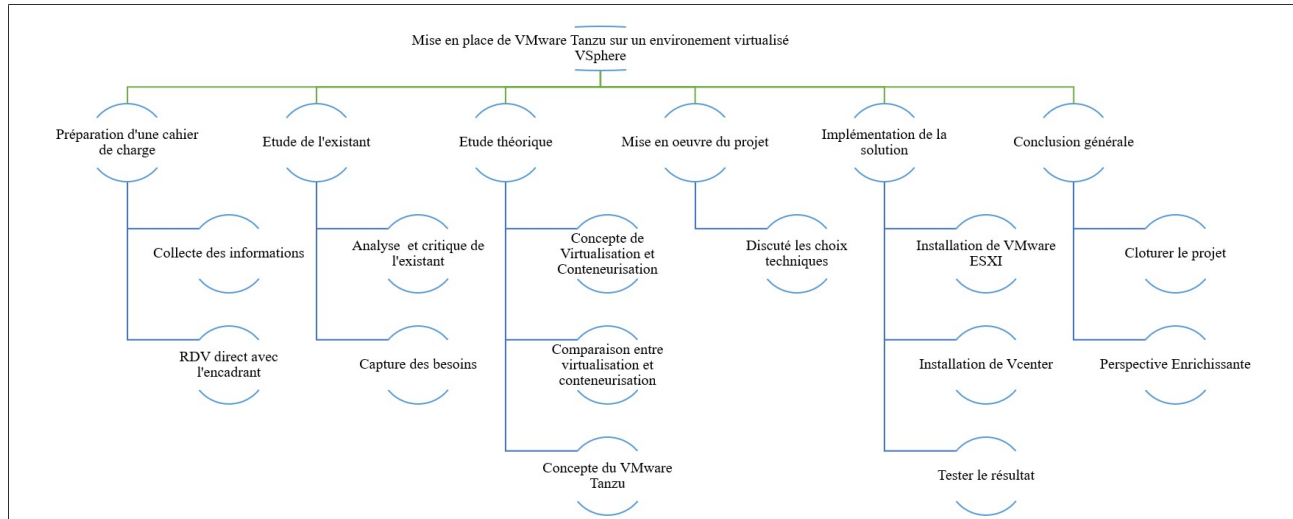


FIGURE 1.5 : Organigramme des tâches

### 1.3.6 Diagramme de GANTT

Dans cette section, nous présentons le planning de travail adoptés tout au long de la période de stage pour bien accomplir les objectifs attendus par ce projet ? Nous sommes arrivés à réaliser le diagramme de Gantt final à l'aide du logiciel gratuit Microsoft Project 2010 .

D'après cette étude, la durée minimale de réalisation du projet s'étale sur 71 jours.

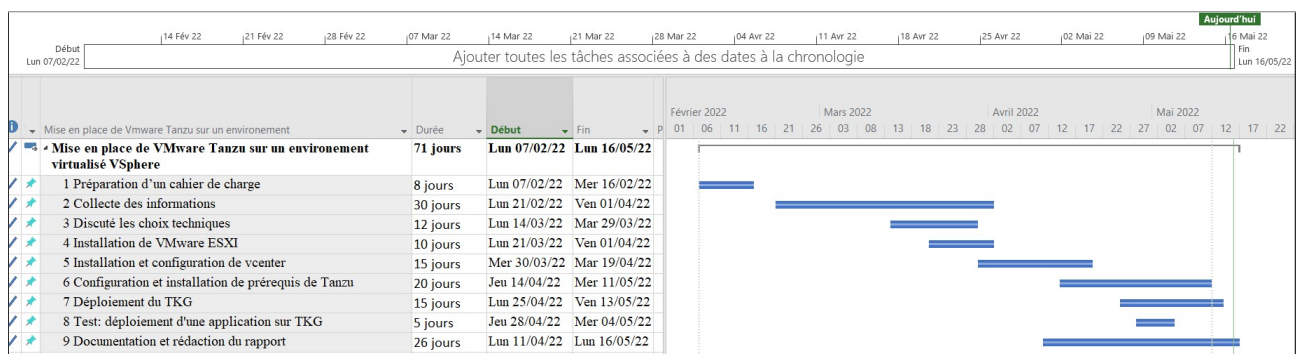


FIGURE 1.6 : Représentation du diagramme de Gantt



## 1.4 Conclusion

Au cours de ce chapitre, nous avons dégagé le contexte général du projet et ses besoins et nous avons fait une étude de l'existant. Dans le chapitre suivant, nous exposerons l'étude théorique qui décrit les différentes notions théoriques impliquées dans notre travail.

# ÉTUDE THÉORIQUE

---

## Plan

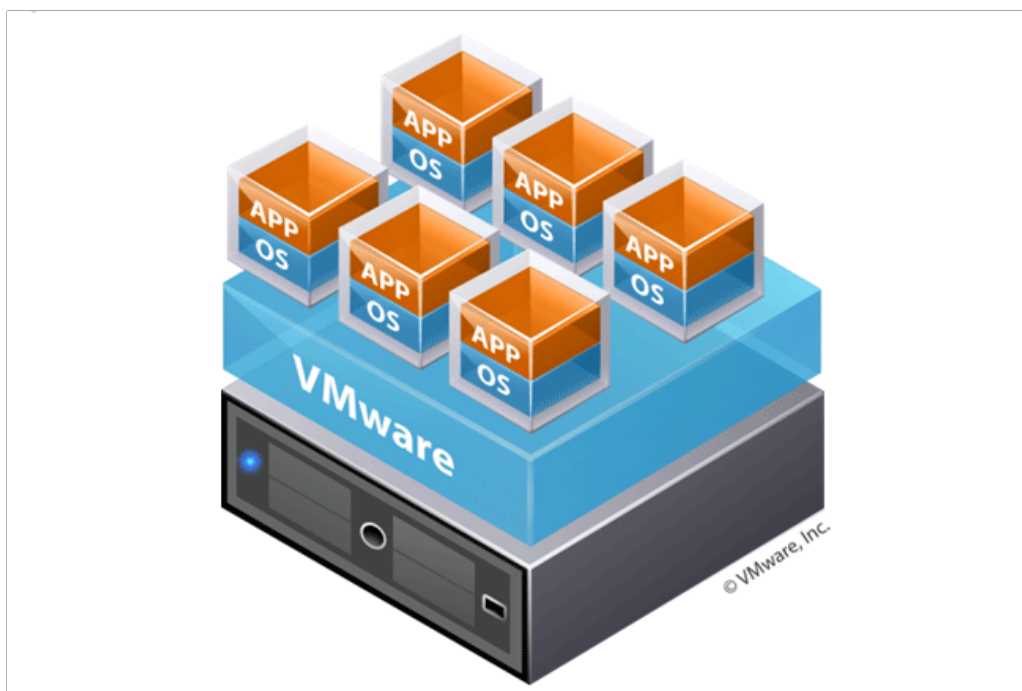
1	Introduction . . . . .	12
2	Virtualisation . . . . .	12
3	La conteneurisation . . . . .	17
4	Comparaison entre virtualisation et conteneurisation . . . . .	19
5	L'orchestration du conteneur . . . . .	20
6	Les microservices . . . . .	22
7	Tanzu . . . . .	23
8	Conclusion . . . . .	26

## 2.1 Introduction

Ce chapitre nous permet de présenter le projet qui a été soumis à notre étude, notamment pour ce qui concerne le déploiement du VMware Tanzu dans un environnement virtualisé. Nous présentons les notions fondamentales de la virtualisation et du Docker avec la technologie TANZU. Nous commençons par expliquer leurs principes, les différents services qu'ils fournissent, leurs modèles de déploiement ainsi que leurs avantages.

## 2.2 Virtualisation

Le matériel informatique x86 actuel est conçu pour exécuter un seul système d'exploitation et une seule application, ce qui laisse la plupart des machines sous-utilisées. La virtualisation vous permet d'exécuter plusieurs machines virtuelles sur une seule machine physique, chaque machine virtuelle partageant les ressources de cette machine physique dans plusieurs environnements. Différentes machines virtuelles peuvent exécuter différents systèmes d'exploitation et applications. La virtualisation consiste à placer une couche supplémentaire du logiciel, appelée hyperviseur, au-dessus de votre serveur physique. Un hyperviseur vous permet d'installer plusieurs systèmes d'exploitation et applications sur un seul serveur.



**FIGURE 2.1** : Le Principe de Virtualisation [4]

### 2.2.1 Les Avantages de la Virtualisation

La virtualisation est une solution technologique qui permet à notre entreprise d'améliorer la qualité, performance, investissement, mais aussi de simplifier et protéger son mode de fonctionnement. Parmi ses avantages, on peut citer :

- Réduction des coûts matériels :La virtualisation permet de réduire de 50 % le nombre de postes de travail et de serveurs physiques dans un parc informatique.
- Optimisation de la productivité :La virtualisation offre des possibilités d'optimiser la rentabilité des employés et de l'entreprise. Avec cette option, l'infrastructure informatique peut être gérée de manière moderne et innovante, permettant aux équipes techniques de passer moins de temps sur des tâches répétitives telles que les mises à jour à grande échelle, la surveillance et la maintenance.
- Une sécurité accrue :Dans une structure traditionnelle, si un malware infecte la messagerie, l'ensemble des applications sont vulnérables. Avec la virtualisation, les tâches d'un serveur physique sont isolées et les services sont cloisonnés.

### 2.2.2 les inconvénients de virtualisation

Comme toutes les solutions informatiques, la virtualisation a des limites.Parmis ces limites on peut citer :

- Une dégradation du performances.
- Une complexité accrue des analyse d'erreur.
- Une mise en oeuvre complexe.
- Une augmentation des risques de sécurité informatique

### 2.2.3 Les hyperviseurs

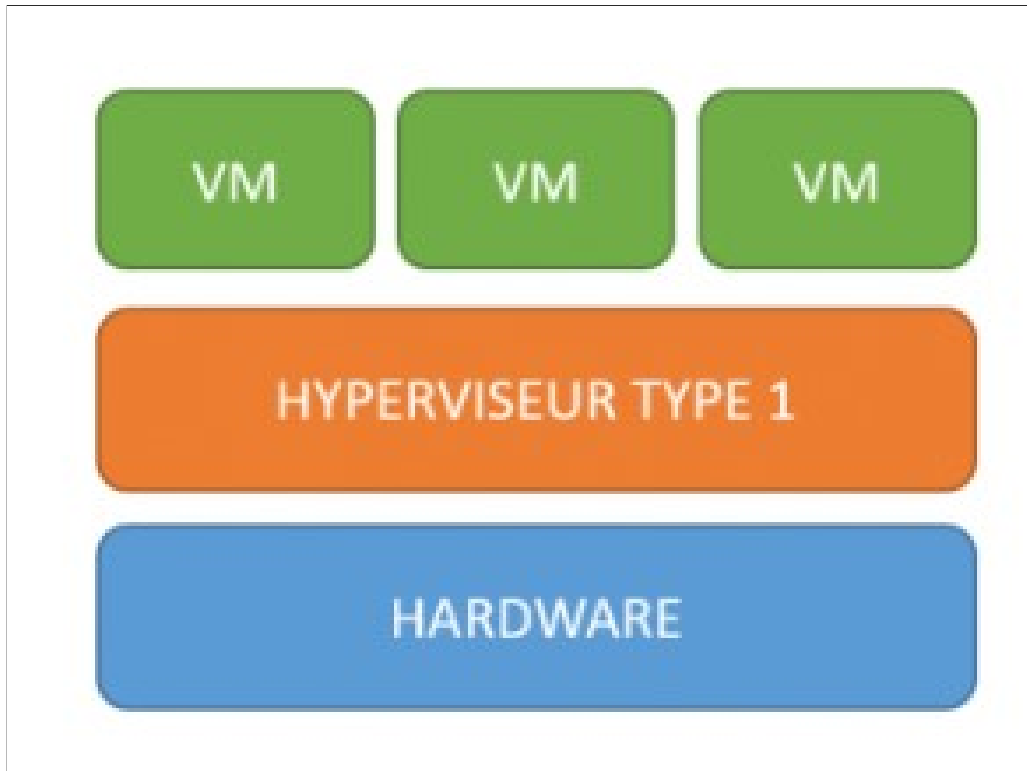
Un hyperviseur est une plate-forme de virtualisation qui permet à plusieurs systèmes d'exploitation de fonctionner simultanément sur la même machine physique.

Il existe deux types d'hyperviseurs :

Type 1 : Un hyperviseur bare metal est installé directement sur votre serveur physique et partitionné en plusieurs machines virtuelles. Toutes les machines virtuelles partagent les mêmes ressources physiques et peuvent s'exécuter simultanément.Exemples d'hyperviseurs de type 1 :

- VMware vSphere
- Citrix XenServer

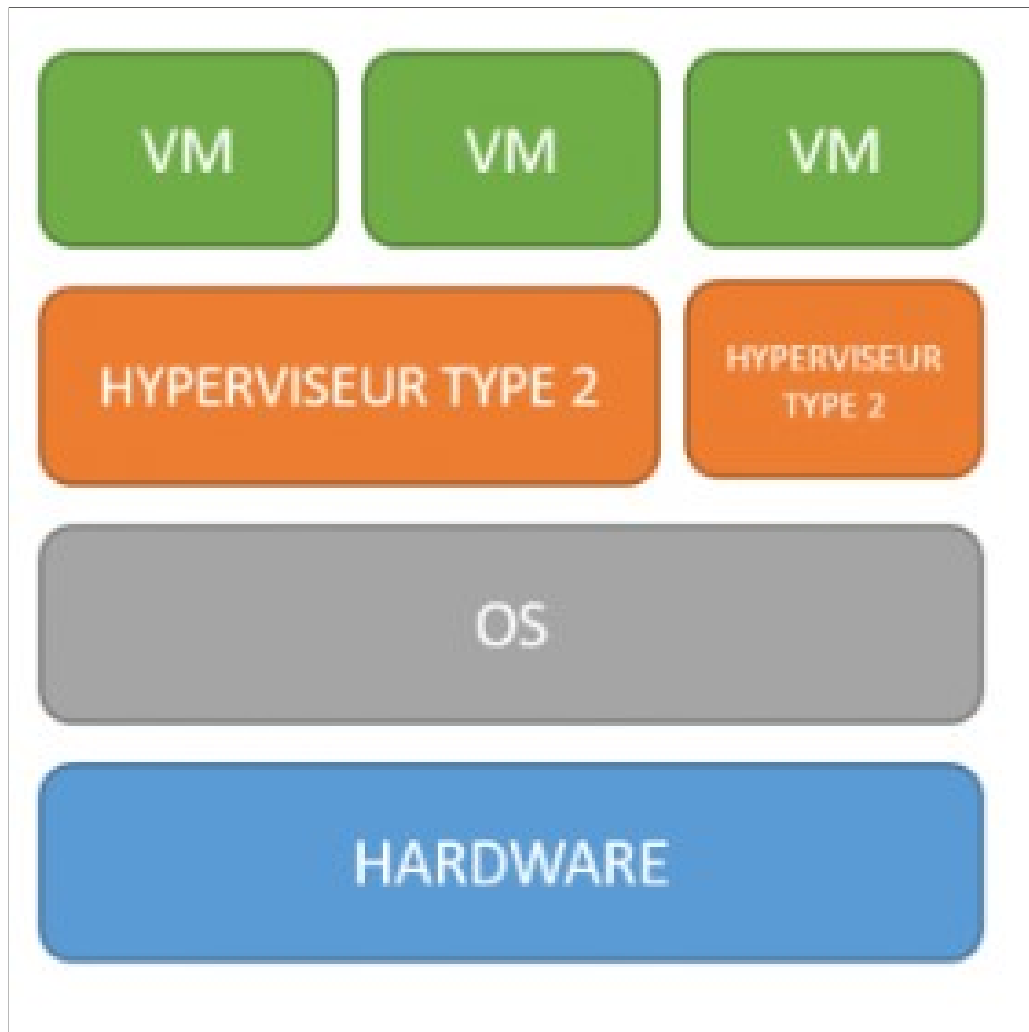
- Microsoft Hyper-V (intégré à Windows Server)
- KVM (Linux)



**FIGURE 2.2 :** Un hyperviseur de type1  
[5]

Type 2 : Un hyperviseur de type 2 est un logiciel qui s'exécute dans un autre système d'exploitation. Par conséquent, le système d'exploitation invité fonctionnera comme une troisième couche au-dessus du matériel. Exemples d'hyperviseurs de type 2 :

- Microsoft Virtual Desktop
- Virtual Box (Open Source)
- VMware Workstation .



**FIGURE 2.3 :** Un hyperviseur de type 2 [5]

#### 2.2.3.1 Les Avantages et les Inconvénients des Hyperviseurs

Le tableau suivant illustre Les Avantages et les Inconvénients des Hyperviseurs de type 1 et de type 2.

**TABLEAU 2.1** : Tableau comparatif d'hyperviseurs Type 1 et Type2 :

Hyperviseur	Type 1	Type 2
Avantages	Un maximum de ressources peut être alloué aux machines virtuelles car ce type d'hyperviseur est directement lié à la couche matérielle.	Il est possible d'exécuter plusieurs hyperviseurs en même temps car ceux-ci ne s'installent pas directement sur la couche matérielle
Inconvénients	exécuter d'un seul hyperviseur à la fois. Cette problématique n'est toutefois pas vraiment impactante puisque dans la grande majorité des cas, un seul et même hyperviseur est capable de gérer tous les applicatifs d'une entreprise.	Ce type d'hyperviseur ne peut pas fournir autant de ressources matérielles que les hyperviseurs de type 1 puisqu'ils sont installés sur un système d'exploitation, lui-même consommateur de ressources.

### 2.2.3.2 Les Différentes solutions d'hyperviseurs

Grâce aux avantages présentés dans le tableau précédent ,nous choisissons de travaillé sur un hyperviseur de type 1, et puisque il est le plus utilisable . Nous présenterons ici les différents hyperviseurs les plus couramment utilisées dans les entreprises, leur fonctionnement et leurs caractéristiques.

- KVM : un hyperviseur open source basé sur Linux. KVM s'exécute à l'intérieur de la machine Linux car un hyperviseur hébergé facilite le déploiement et la gestion.

**FIGURE 2.4** : KVM

- Citrix XenServe : Xen appartient à Citrix. Néanmoins, une version non payante est toujours disponible sous le nom de Xen Project.c'est Un hyperviseur open source fonctionne sous Linux.

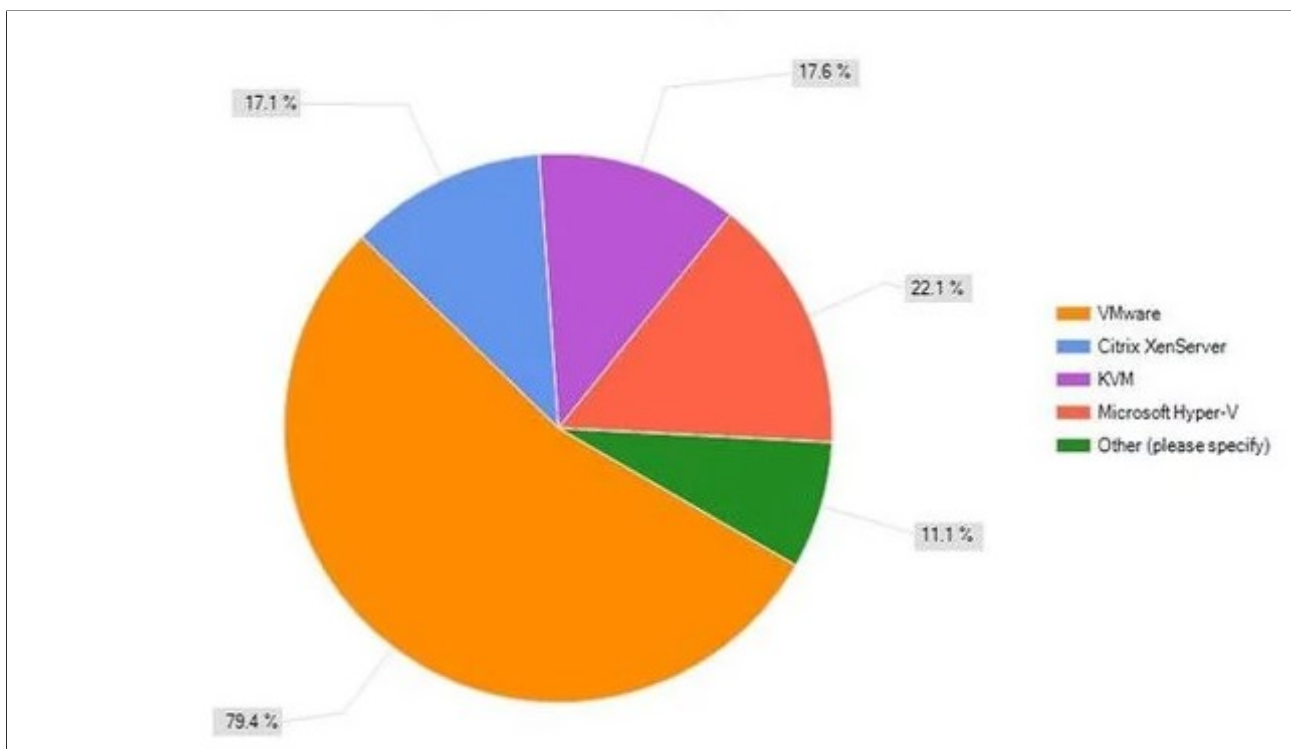
**FIGURE 2.5** : Xen

- HyperV

est un hyperviseur commercial fourni par Microsoft. Bien qu'il est excellent pour exécuter Windows. Etant un hyperviseur, il exécutera n'importe quel système d'exploitation pris en charge par la plate-forme matérielle.

- VMware vSphere : vSphere est un produit VMware. C'est la version 7.0 en 2020. VMware vSphere est une suite logicielle qui comprend des composants tels que ESXi, vCenter Server, vSphere Client, etc. Il fournit une plate-forme de virtualisation complète ainsi qu'un ensemble étendu de services applicatifs et d'infrastructure.

La figure ci-dessous montre les hyperviseurs les plus utilisables ainsi que les pourcentages d'applications. Cela montre que les administrateurs sondés emploient 79% VMware comme leur premier hyperviseur, Citrix, KVM et Hyper-V sont cités mais ils pèsent environ 20%.



**FIGURE 2.6 :** Comparaison entre les différents types d'hyperviseurs[6]

## 2.3 La conteneurisation

La conteneurisation est une méthode de virtualisation au niveau du système d'exploitation. Cette méthode repose sur la création de plusieurs instances isolées les uns des autres sur un noyau commun. Les conteneurs sont relativement légers : Ils ne contiennent que les bibliothèques et les outils nécessaires à l'exécution de l'application conteneurisée. Ceci nous amène à une flexibilité accrue et une utilisation efficace du stockage .[7]



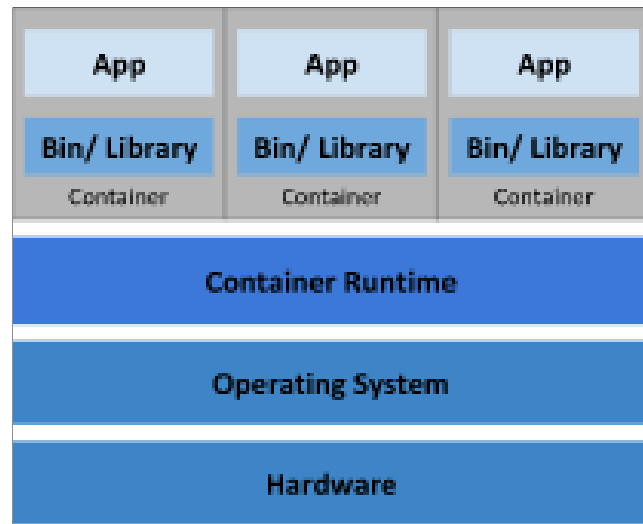


FIGURE 2.7 : Architecture de la conteneurisation [7]

### 2.3.1 Les conteneurs Docker

Docker est une solution de conteneurisation open source conçue pour améliorer le déploiement des applications sur notre infrastructure. Cette solution peut également fonctionner sur n'importe quel serveur Linux. Docker est utilisé pour créer et gérer des conteneurs, ce qui permet de simplifier et Mettre en œuvre un système distribué en permettant l'exécution de plusieurs applications. Il est naturellement adapté aux architectures basées sur les microservices.



FIGURE 2.8 : Docker

### **2.3.1.1 Les avantages de la solution docker**

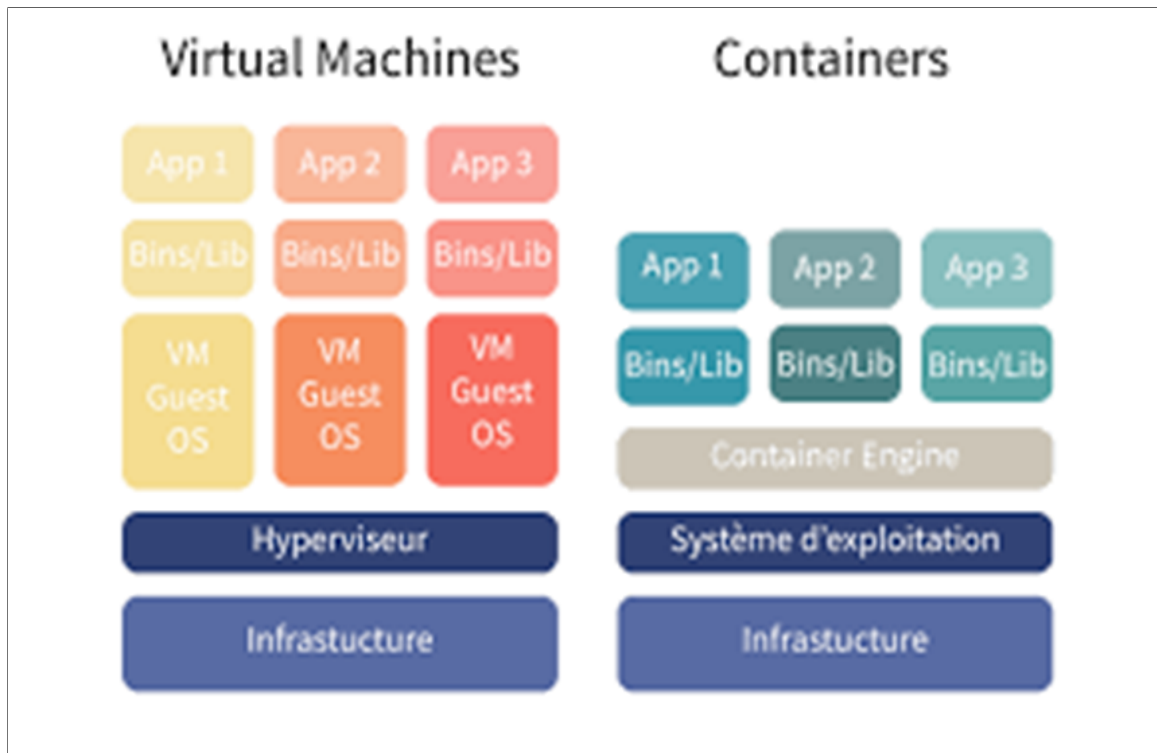
Nous avons plusieurs avantages à utiliser Docker dans l'espace informatique. . Les départements informatiques fournissent de plus en plus de services orientés Web, et les conteneurs peuvent être un moyen d'accélérer leur Les conteneurs pourraient être un facteur clé pour accélérer sa transition vers le cloud. :

- Légèreté des conteneurs.
- Rapidité et facilité de déploiement des applications.
- Portabilité et multicloud.

## **2.4 Comparaison entre virtualisation et conteneurisation**

La virtualisation crée de nouvelles machines virtuelles et assure l'allocation des ressources physiques entre elles à travers une couche logicielle appelée hyperviseur. Chaque machine virtuelle se comporte comme un serveur autonome, intégrant son propre système d'exploitation, sur lequel des applications seront déployées et exécutées. Contrairement à la virtualisation, les conteneurs se présentent sous la forme d'images qui peuvent regrouper des données, des bibliothèques ou des applications entières, Démarrer en tant que processus et utiliser le système hôte sans hyperviseur. Cependant, la principale différence entre eux est que chaque machine virtuelle s'exécute sur son propre système d'exploitation ,alors que les conteneurs partagent un seul système d'exploitation.

Le schéma suivant résume les différence entre un système de virtualisation traditionnelle et l'utilisation d'un conteneur :



**FIGURE 2.9 :** Comparaison entre les conteneurs et les machines virtuelles [2]

## 2.5 L'orchestration du conteneur

Il devient extrêmement difficile de gérer le cycle de vie du conteneur et sa gestion lorsque le nombre augmente dynamiquement avec la demande. L'orchestration de conteneurs résout le problème en automatisant la planification, le déploiement, l'évolutivité, l'équilibrage de charge, la disponibilité et la mise en réseau des conteneurs. C'est un processus automatique de gestion ou de planification du travail de conteneurs individuels pour les applications basées sur des micro services au sein de plusieurs clusters. Avec l'orchestration des conteneurs, l'équipe DevOps peut représenter l'état souhaité du cluster sous la forme d'une configuration. Un moteur d'orchestration de conteneurs applique la configuration souhaitée et automatise toutes les tâches de gestion.

### 2.5.1 Outils d'orchestration : Kubernetes

Kubernetes est un moteur d'orchestration qui fournit une automatisation en arrière-plan permettant aux entreprises d'utiliser des conteneurs, de même que les machines virtuelles avec vSphere et autres hyperviseurs. Il fournit également une couche de gestion et d'orchestration capable de gérer l'infrastructure de conteneurs sous-jacente afin que vos applications puissent s'adapter aux interruptions. En cas de défaillance d'un conteneur, il serait plus efficace d'avoir un processus automatique qui peut démarrer automatiquement un autre conteneur et reconfigurer l'application. Ce cas d'emploi particulier est celui de "pain et beurre" de Kubernetes.



FIGURE 2.10 : Logo Kubernetes

### 2.5.1.1 Les avantages du Kubernetes

- Il permet aux applications fonctionnant sur des conteneurs d'être hautement disponibles
- Il permet une élasticité facile de votre infrastructure
- Il vous permet de planifier en arrière-plan où les ressources sont les mieux adaptées pour fonctionner et sur quel(s) hôte(s) de conteneur.
- Il permet d'ajouter rapidement de nouveaux hôtes de conteneurs au cluster Kubernetes.

### 2.5.1.2 Les composant du kubernetes

L'architecture Kubernetes se compose de différents éléments. Associés, ceux-ci nous permet d'exploiter, de déployer et de mettre à jour les logiciels conteneurisés. Parmi ces composants, on peut citer :

**Nœuds** : Un ensemble de machines virtuelles utilisées pour exécuter des déploiements Kubernetes.

**Un nœud maître** : (appelé « Master ») chargé d'orchestrer le cluster (il ne contient pas d'autres services que Kubernetes).

**Des nœuds esclaves** : (appelés « minions » ou « nodes »), correspondant à des hôtes Docker.

**PODs** : Un regroupement d'un ou plusieurs conteneurs en tant qu'unité atomique.

Comme cette figure ci-dessous illustre :

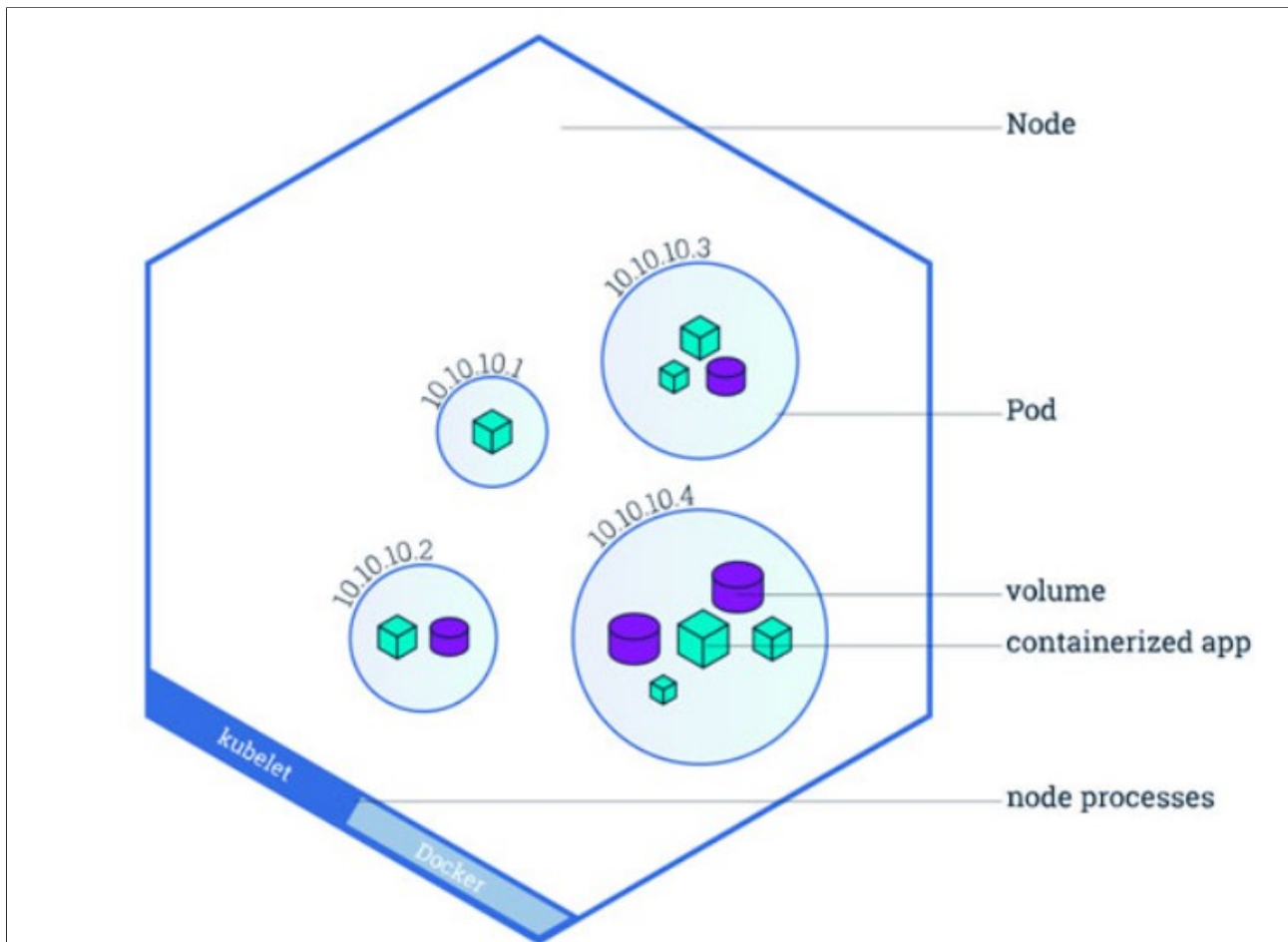


FIGURE 2.11 : Elements du Kubernetes

### 2.5.1.3 Modernisation des Application avec Kubernetes

Pour de nombreuses entreprises, reconstruire leurs applications pour utiliser des conteneurs offre une plateforme solide pour la modernisation des applications. Toutefois, les conteneurs présentent leurs propres complexités intrinsèques et ne disposent pas de la gestion et de l'orchestration nécessaires pour assurer la disponibilité et la résilience requises. Bien que Kubernetes résolve beaucoup de problèmes de gestion, il peut être difficile de configurer, maintenir et supporter. Les sociétés passant à des applications modernes fonctionnant sur des conteneurs peuvent trouver des outils, une configuration et une infrastructure très différentes de celles dont ils disposent dans le monde des machines virtuelles.

## 2.6 Les microservices

Les architectures de microservices ont été créées pour répondre au besoin d'agilité. Les entreprises doivent analyser les données, innover et lancer de nouveaux produits et services plus rapidement que leurs concurrents. Ils doivent être flexibles pour répondre aux besoins changeants de leurs clients. La migration vers une architecture de microservices permet d'atteindre ces objectifs :

### **Croissance de l'entreprise :**

lorsqu'une application sert plus de clients et traite plus de transactions, elle nécessite plus de capacité et de ressources .

**Capacité à absorber les pics** de charge sans risquer d'interruption de service .

Capacité à évoluer/fournir plus rapidement de **nouvelles fonctionnalités d'application** pour répondre à la demande

Les microservices divisent les applications en petits éléments indépendants, mais les administrateurs informatiques ont toujours besoin d'un moyen de les gérer. Grâce à Kubernetes, ils peuvent automatiquement gérer et faire évoluer les microservices conteneurisés.

## **2.7 Tanzu**

Lors du VMworld 2019 US, en août 2019, VMware a présenté une suite de produits qui aide les entreprises à relever les nombreux défis de la modernisation des applications : VMware Tanzu. VMware Tanzu est une collection de produits et services conçus pour moderniser les Applications et l'infrastructure avec un objectif commun : mieux mettre en production. Ce portefeuille simplifie les opérations multi-cloud tout en prenant en charge les développeurs ont un accès plus rapide aux bonnes ressources pour créer les meilleures applications. VMware Tanzu permet aux équipes de développement et d'exploitation d'exécuter et de gérer les applications résident en permanence sur une infrastructure évolutive.[8]

Parmi les meilleures solutions contenues dans VMware Tanzu on peut citer :

- Tanzu Kubernetes Grid - Le runtime Kubernetes d'entreprise intégré à vSphere : c'est un moteur d'exécution Kubernetes prêt pour l'entreprise qui simplifie les opérations dans une infrastructure virtualisée.



**FIGURE 2.12 :** vmware tanzu

Les entreprises préfèrent utiliser la plateforme TANZU pour les applications conteneurisées de tels

sorte que les chiffres d'affaires annuel enregistre une croissance» Pour les clients de VMware Tanzu, on distingue l'augmentation de performance :

400 %d'accélération des lancements.

60 %de réduction des coûts d'infrastructure.

En résumé, VMware Tanzu permet aux entreprises d'exécuter des conteneurs Kubernetes dans des environnements cloud et même sur site dans leurs environnements VMware vSphere. L'exécution native de VMware Tanzu dans vSphere offre aux clients de nombreux avantages. Premièrement, il facilite la configuration, la gestion et l'exploitation des conteneurs Kubernetes. Deuxièmement, il permet aux clients VMware vSphere existants qui utilisent vSphere comme hyperviseur sous-jacent dans leur environnement d'utiliser le même ensemble d'outils et la même interface de gestion pour gérer leur infrastructure de machines virtuelles existante et les applications modernes exécutées sur Kubernetes.

### 2.7.1 fonctionnement du VMware Tanzu

VMware Tanzu utilise Kubernetes pour modifier vSphere. Il combine les capacités de vSphere en intégrant Kubernetes dans le plan de contrôle vSphere. Les machines virtuelles et les conteneurs s'exécutent à l'aide de l'agrégation de pods vSphere. Les pods VMware vSphere natifs présentent de nombreux avantages, notamment la légèreté et la sécurité. VMware affirme qu'ils peuvent même être plus rapides que les conteneurs "bare metal" en raison de l'efficacité avec laquelle vSphere gère la planification du processeur. Les ressources de calcul, de stockage et de mise en réseau des conteneurs peuvent être gérées parallèlement aux machines virtuelles traditionnelles en intégrant Kubernetes dans le plan de contrôle vSphere. Cela présente des avantages considérables du point de vue de la gestion et des opérations. Cela signifie que les opérations informatiques peuvent gérer les objets de conteneur Kubernetes à partir du client vSphere.

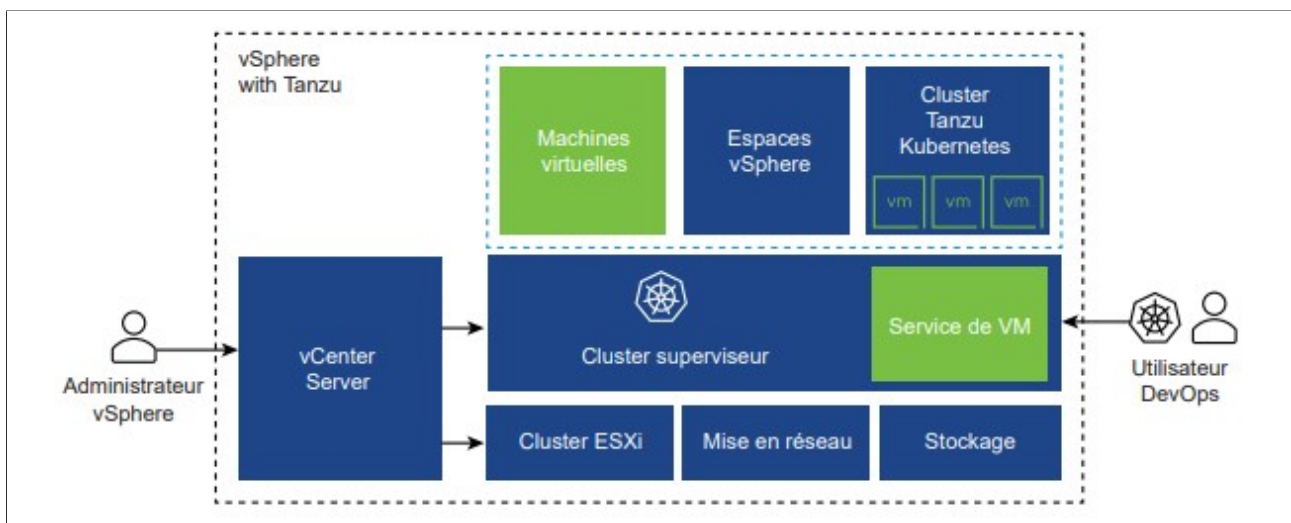


FIGURE 2.13 : vsphere with Tanzu[9]

### 2.7.2 Un cluster Tanzu Kubernetes

Un cluster Tanzu Kubernetes est une distribution complète de la plateforme d'orchestration de conteneur Kubernetes Open Source qui est développée, signée et prise en charge par VMware. Vous pouvez provisionner et utiliser des clusters Tanzu Kubernetes sur le Cluster superviseur à l'aide de Service Tanzu Kubernetes Grid.

### 2.7.3 VMware Tanzu Kubernetes Grid (TKG)

TKG est un ensemble de scripts d'automatisation écrits par VMware qui déploie des clusters Kubernetes. Une fois le cluster déployé, TKG met à jour le fichier de configuration Kubernetes afin que vous puissiez exécuter avec succès les commandes kubectl sur ces clusters .

### 2.7.4 Le Cluster API

TKG utilise le cluster API pour la gestion du cycle de vie. C'est un sous-projet de Kubernetes qui vise à fournir des API et des outils déclaratifs pour simplifier la configuration, la mise à niveau et l'exploitation de plusieurs clusters Kubernetes. De plus c'est un outils qui permet des déploiements de cluster cohérents et reproductibles dans une variété d'environnements d'infrastructure.

### 2.7.5 Le cluster de gestion

Pour créer un environnement de développement TKG sur VMware, il faut commencer par un cluster de gestion. Le cluster de gestion est responsable de la création et de la maintenance des clusters de charge de travail. Le cluster de charge de travail du développeur est intégré au cluster de gestion, et les développeurs à leur tour créent et exécutent leurs applications.

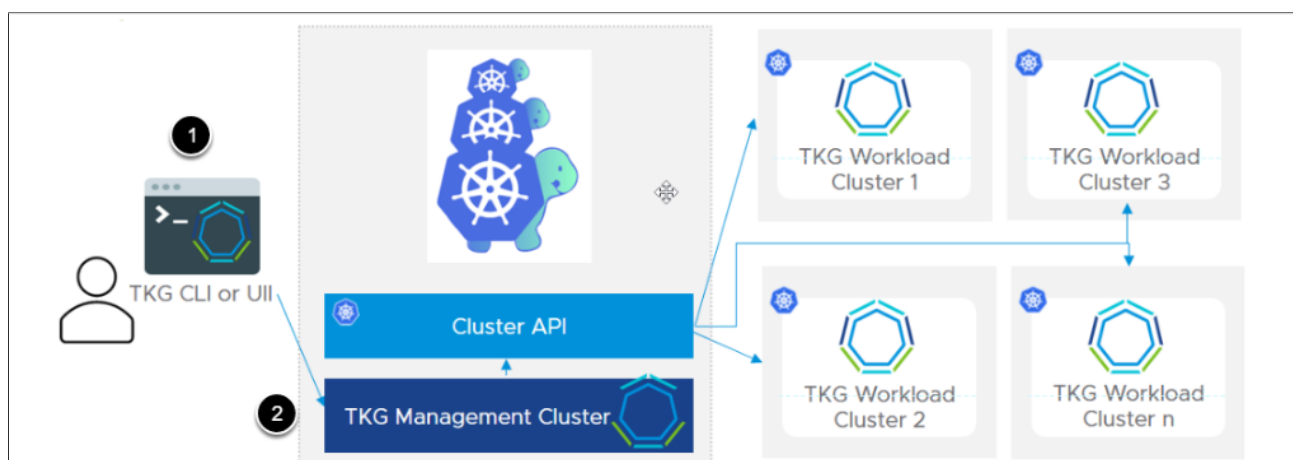


FIGURE 2.14 : cluster de gestion



### **2.7.6 TKGS**

Il présente le Tanzu Kubernetes Grid Service (TKGS), permettant aux clusters Tanzu Kubernetes d'être déployés et exécutés de manière native avec vSphere.

## **2.8 Conclusion**

Dans ce chapitre nous avons découvert les plus importantes notions de notre projet en traitons les différentes fonctionnalités de notre technologies. Dans le chapitre suivant, nous représenterons la mise en œuvre de notre projet.

---

# MISE EN ŒUVRE DU PROJET

---

## Plan

1	Introduction . . . . .	28
2	Les choix techniques : . . . . .	28
3	Déploiement de Tanzu sur VDS Networking . . . . .	34
4	cluster superviseur . . . . .	35
5	Espace du nom . . . . .	36
6	Conclusion . . . . .	37

### 3.1 Introduction

Après avoir bien compris le projet et ses différentes phases, nous passerons dans ce chapitre à la mise en œuvre d'où nous représentons l'environnement de travail, l'architecture de notre solution, et les différentes démarches de la réalisation.

### 3.2 Les choix techniques :

Notre choix s'est orienté vers la solution de virtualisation de VMware. Nous nous intéressons dans notre travail principalement à la partie virtualisation des services grâce à VMware vSphere avec TANZU. Cette solution est considérée comme première plate-forme de virtualisation de serveurs du marché.

- VMware Vsphere : VMware vSphere est une suite de composants logiciels de virtualisation, Ils comprennent ESXi, vCenter Server qui remplissent plusieurs fonctions différentes dans l'environnement vSphere. Elle permet aux utilisateurs d'exécuter leurs applications en toute sécurité et de répondre plus rapidement aux besoins de l'activité. La figure ci-dessous montre les composants de VMware Vsphère.

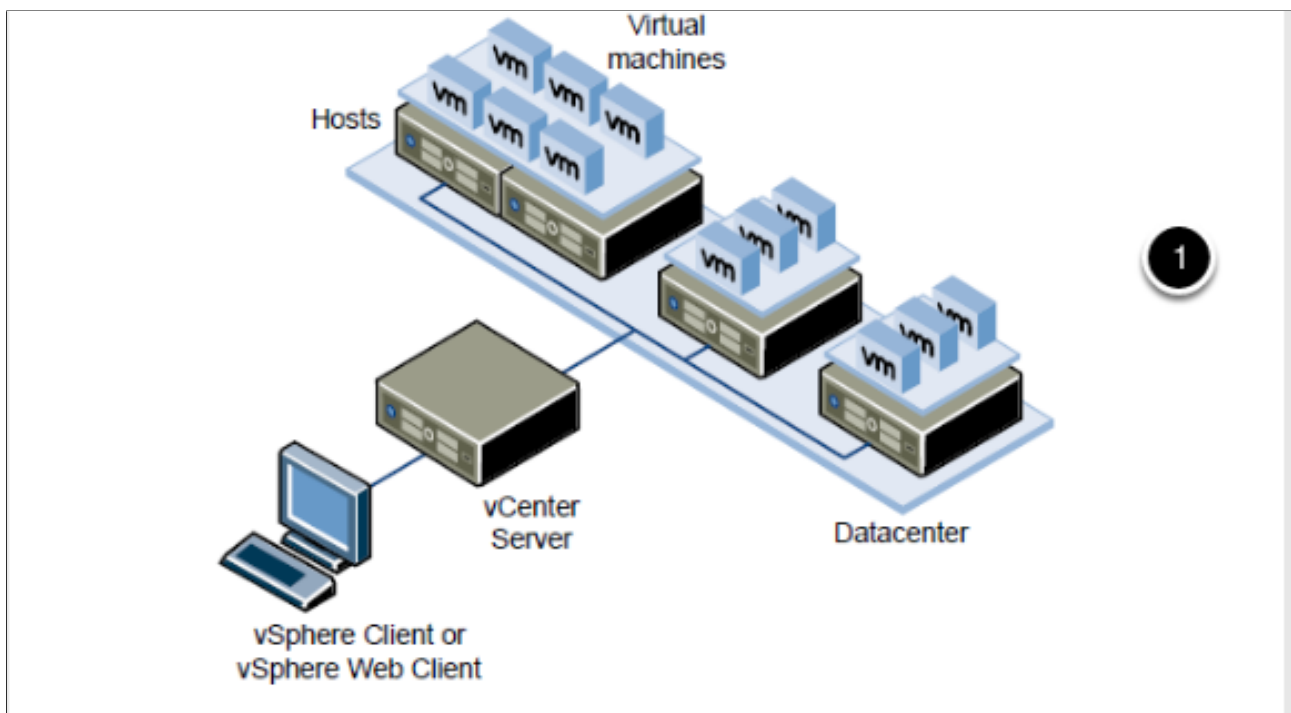
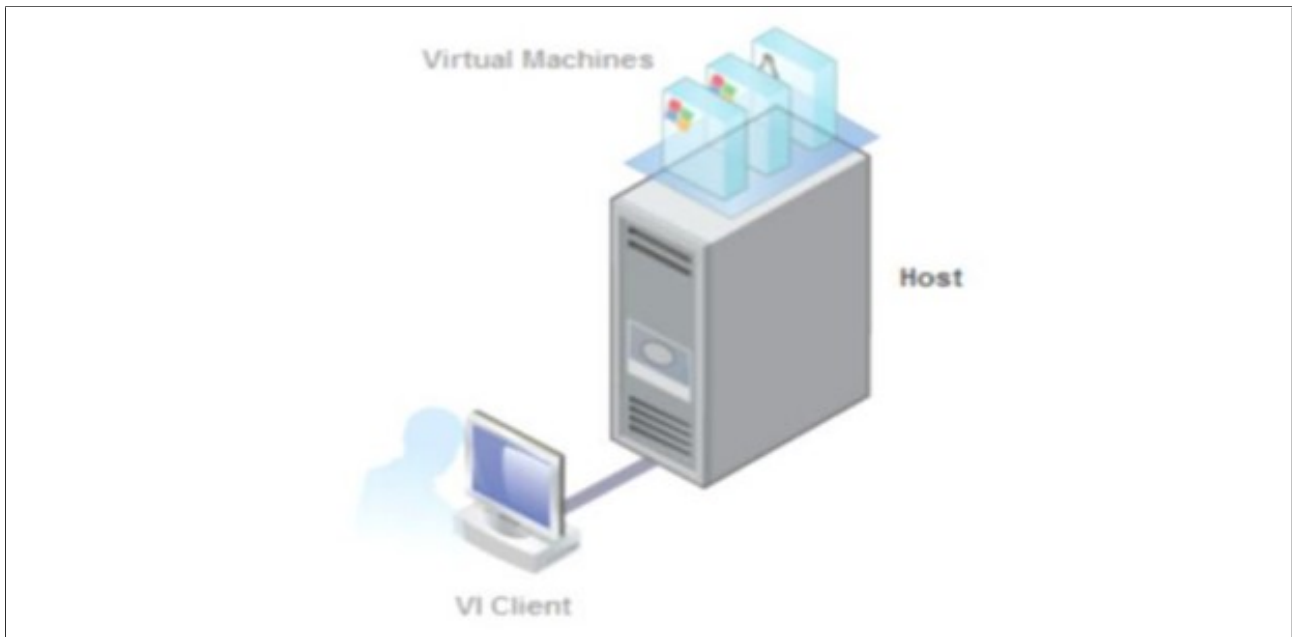


FIGURE 3.1 : Composants de Vsphere

- VMware ESXI : C'est un hyperviseur de type 1 « bare-metal » qui est le premier hyperviseur dédié du marché. Esxi s'installe directement sur le serveur physique pour gérer l'exécution des

serveurs logiques appelés machines virtuelles. Il permet d'allouer des ressources aux machines virtuelles selon leurs besoins.



**FIGURE 3.2 :** Architecture de VMware ESXi

### 3.2.1 Mise en réseaux de ESXi

- Commutateur virtuel (Vswitch) :

Un commutateur virtuel est un logiciel qui exécute les mêmes fonctions qu'un commutateur ordinaire, à l'exception de certaines fonctions avancées. Les commutateurs virtuels de VMware sont appelés vSwitches. Les vSwitches sont utilisés pour fournir une connectivité entre les machines virtuelles et pour connecter des réseaux virtuels et physiques. Le vSwitch utilise la carte réseau physique de l'hôte ESXi (également connue sous le nom vNIC ,virtuel Network Interface Controller) pour se connecter au réseau physique. Pour se connecter à un vSwitch, une machine virtuelle doit être associée à un adaptateur réseau virtuel (vNIC),

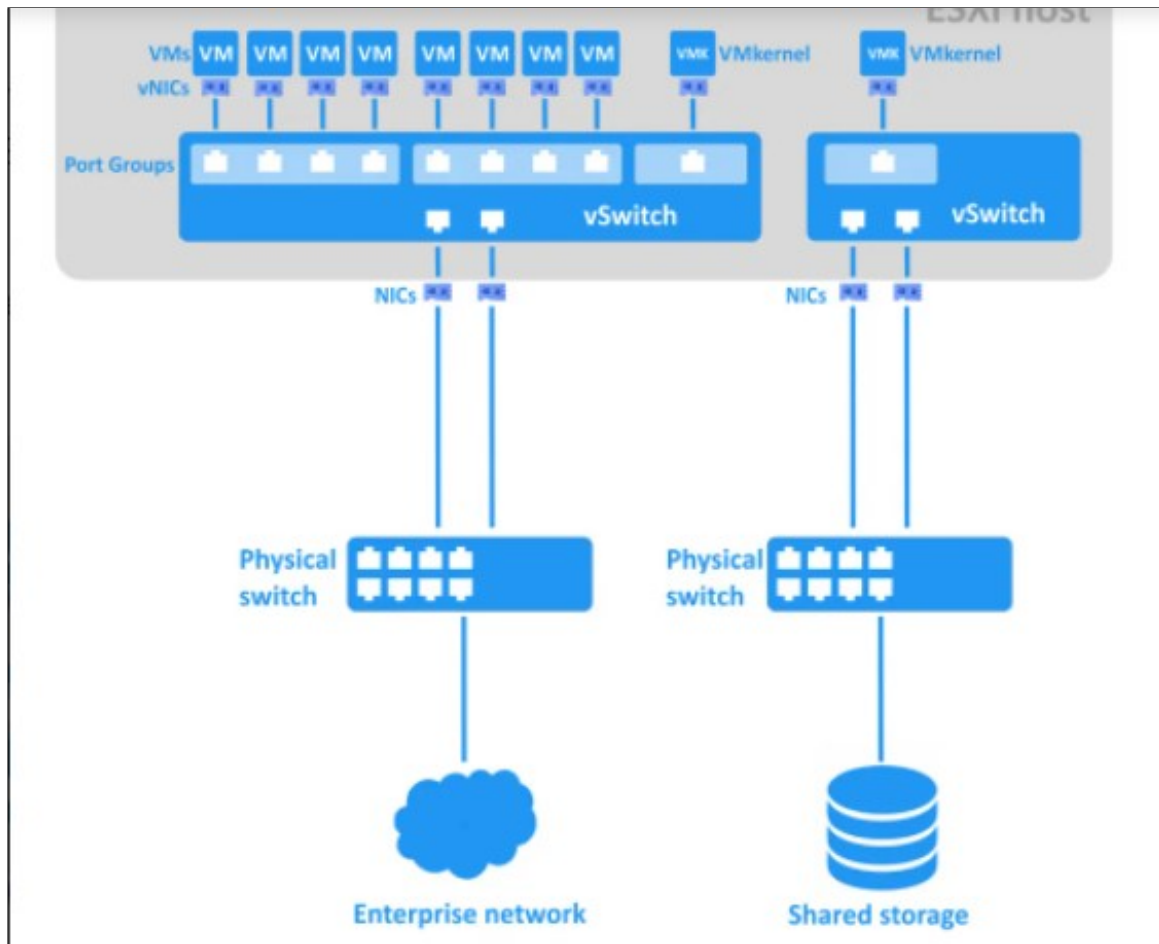


FIGURE 3.3 : Commutateur virtuel

Nous pouvons créer des réseaux segmentés sur un vSwitch existant en créant des groupes de ports pour différents groupes de machines virtuelles. Cette approche facilite la gestion de grands réseaux. Pour les réseaux de stockage ,gestion et vMotion, nous devons attacher un adaptateur VMkernel qui peut fournir une adresse IP différente pour chaque réseau.

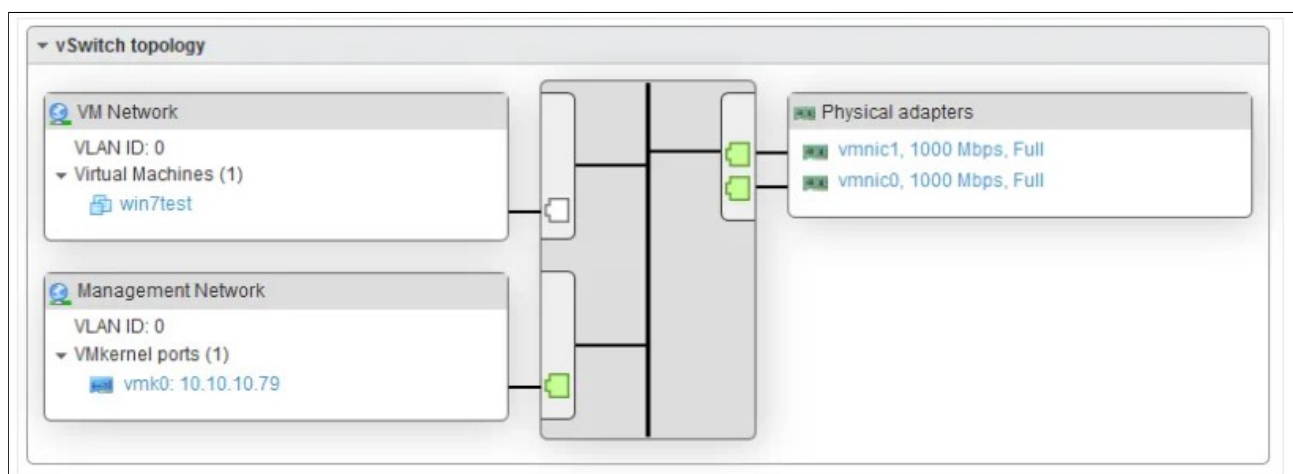


FIGURE 3.4 : Topologie Vswitch

- Les types de commutateur virtuels :

Les vSwitchs VMware peuvent être divisés en deux types : les commutateurs virtuels standard et les commutateurs virtuels distribués.

#### **Commutateur Standard :**

La configuration de chaque vSwitch réside sur un hôte ESXi spécifique. Les administrateurs informatique doivent maintenir manuellement la configuration de vSwitch sur tous les hôtes ESXi pour effectuer des opérations nécessaires.

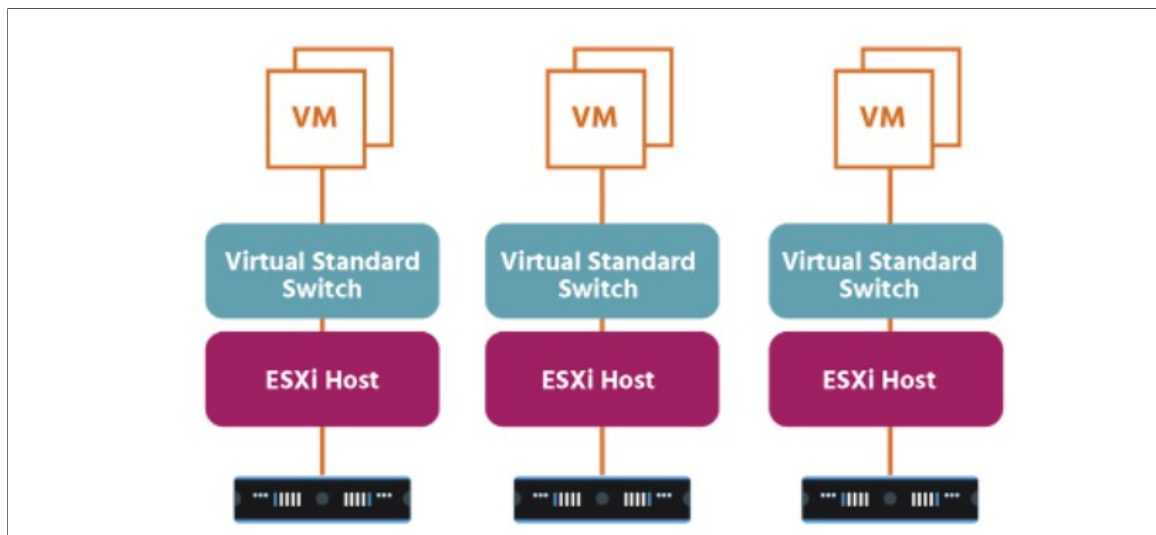


FIGURE 3.5 : commutateur Standard

#### **Commutateur distribuer :**

Le commutateur vSphere Distribuer permet à un seul commutateur virtuel de connecter et gérer la configuration réseau de plusieurs hôtes à la fois à partir d'un emplacement central.

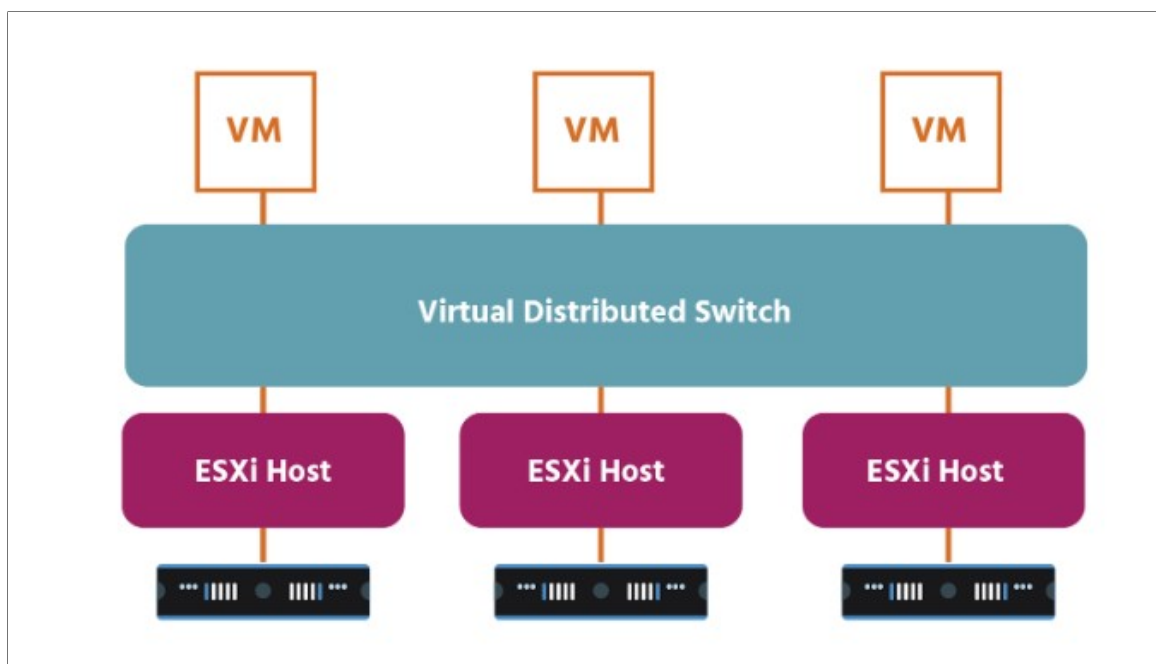


FIGURE 3.6 : Commutateur distribuer

- VMKernel :

Le port VMkernel est un adaptateur virtuel, c'est-à-dire un dispositif spécial avec lequel l'hôte vSphere communique avec le monde extérieur. Ce port permet de contrôler également plusieurs services spéciaux, tels que :

**vMotion** : un service qui permet de transférer une machine virtuelle lancée d'un serveur ESXI vers un autre sans aucune interruption, de manière que les utilisateurs puissent continuer à accéder aux systèmes dont ils ont besoin pour rester productifs. Cette manipulation est possible si la machine virtuelle se trouve sur un espace de stockage partagé et que les serveurs se trouvent dans le même cluster

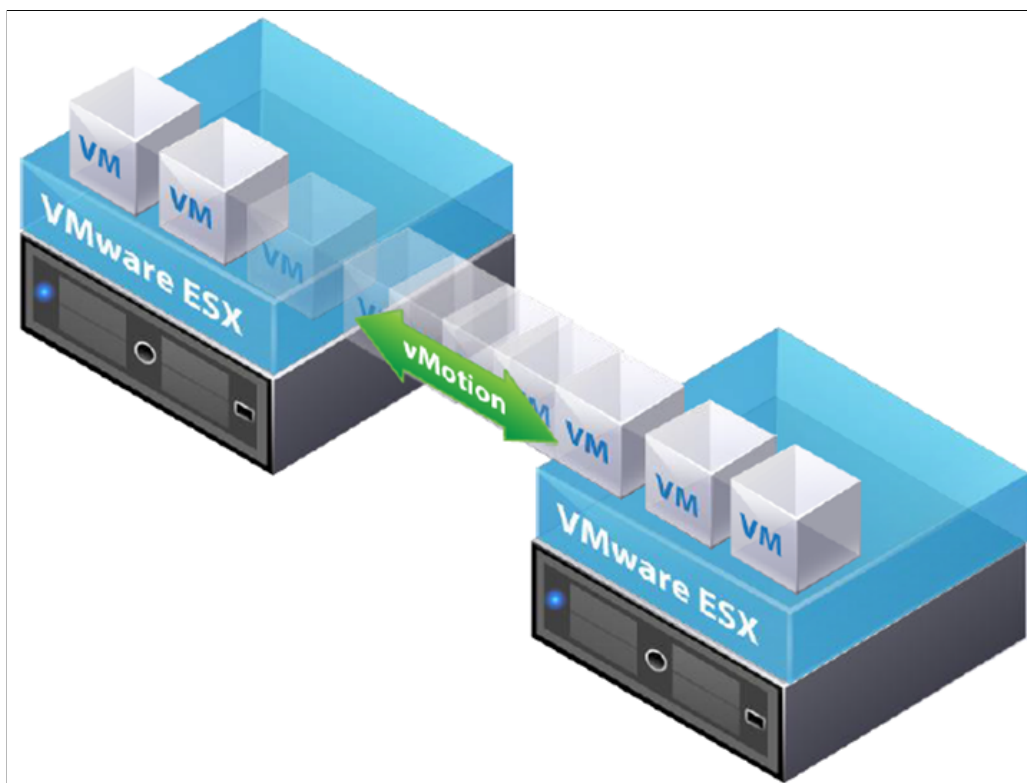
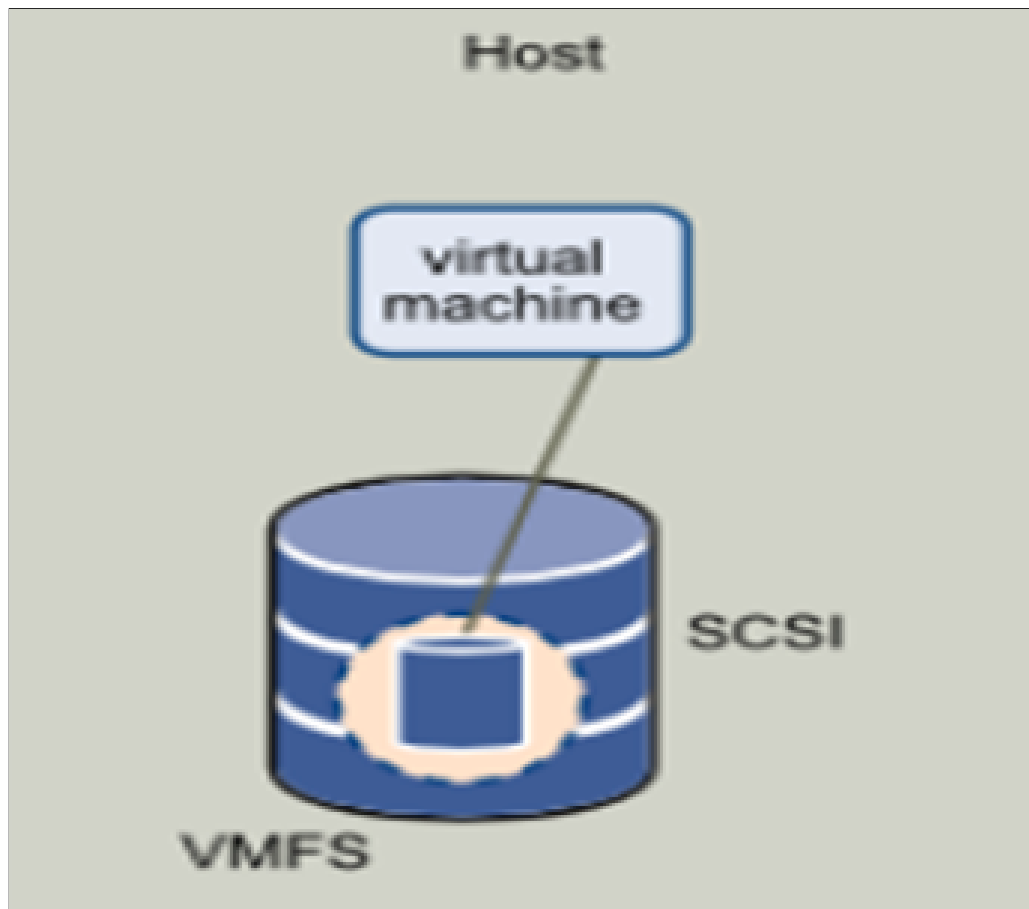


FIGURE 3.7 : Vmotion

**management** : il est nécessaire pour la configuration manuelle.

**VMFS** : C'est un système de fichiers que vSphere utilise pour gérer les ressources disponibles pour les machines virtuelles. Il est ensuite possible de créer des volumes sur ces Data Stores, afin de rendre l'espace disque disponible pour les machines virtuelles préconfigurés.



**FIGURE 3.8 : VMFS et ISCSI**

**ISCSI** : IL Stocke les données à distance en utilisant VM / pile norme communication IP TCP. L'hôte utilise un initiateur iSCSI logiciel dans le VMKernel pour se connecter au stockage. Avec ce type de connexion iSCSI, l'hôte ne requiert qu'un adaptateur réseau standard pour la connectivité réseau.

- VMware Vcenter :

C'est une plate-forme évolutive et extensible permettant de gérer pro activement la virtualisation et offrant une visibilité approfondie de l'infrastructure virtuelle et gérer leur environnement vSphere. IL Centralise la gestion de l'environnement virtuel complet, gère les hôtes ESXI, la machine virtuelle, la banque de données et le réseau virtuel. Il est également possible à travers VCenter Server de gérer L'utilisation des options suivantes (HA, VMotion, DRS, FT ). Dans notre travail, on va utiliser Vcenter client .C'est une application Windows qui permet l'accès à distance aux fonctionnalités de vCenter Server et aux machines virtuelles.



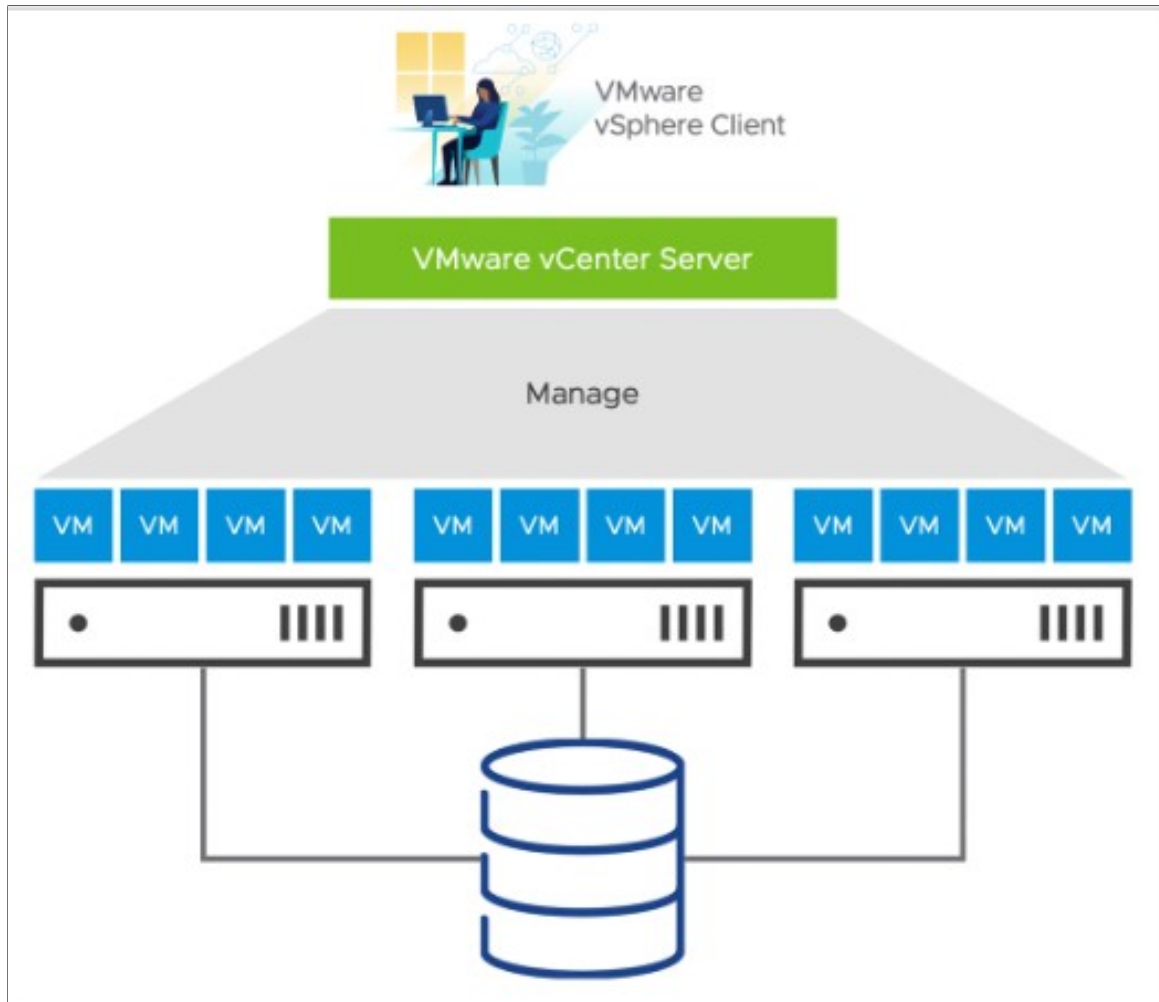


FIGURE 3.9 : VMware Vcenter

**DRS** : (Distributed Resource Scheduler) Il est disponible dans les clusters VMware, qui équilibre la charge des hôtes via le déplacement automatique (vMotion) des machines virtuelles. Il répartira les machines virtuelles sur différents hôtes du cluster en fonction de leur objectif et de leurs ressources.

**HA** : (High Availability) Transfère les données en cas de troubles vers un autre serveur dans un même cluster, il est moins coûteux et moins complexe à mettre en œuvre qu'une solution classique

### 3.3 Déploiement de Tanzu sur VDS Networking

#### 3.3.1 Bibliothèque de contenu VMware

Pour provisionner des clusters et des machines virtuelles Tanzu Kubernetes, nous avons besoin d'une bibliothèque de contenu créée dans vCenter Server qui gère le cluster vSphere sur lequel le cluster superviseur s'exécute.

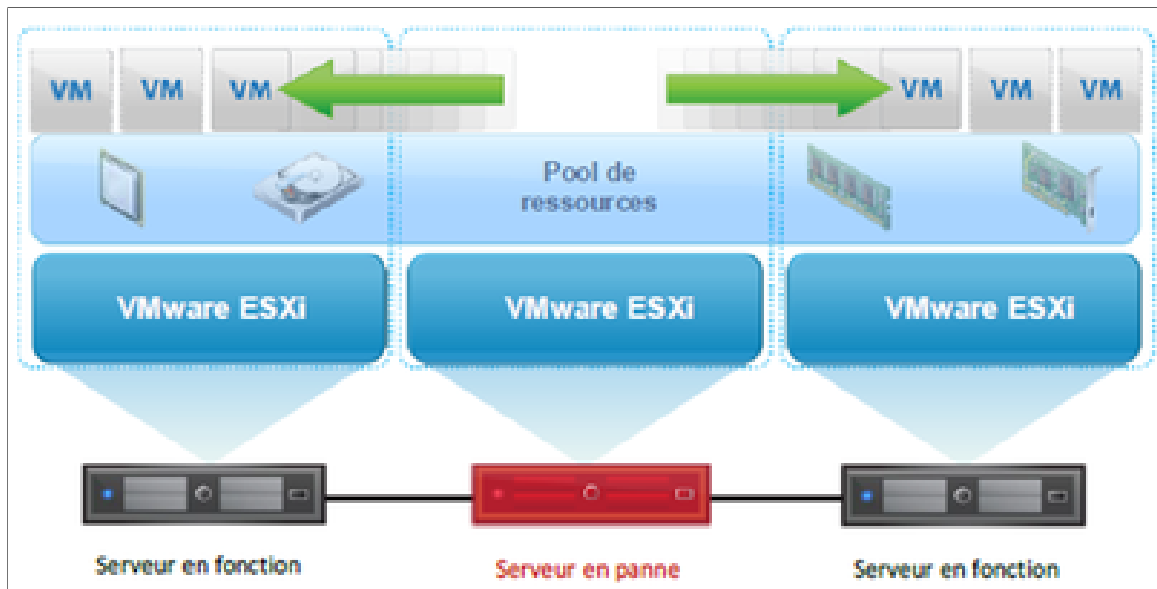


FIGURE 3.10 : HA : Hight Availability

### 3.3.2 Politique de stockage

Les politiques de stockage agissent comme les principaux canaux par lesquels les données sont incluses dans les opérations de protection et de récupération des données. Une politique de stockage est l'entité logique principale par laquelle un sous-client ou une instance est sauvegardé. Sa fonction principale est de faire correspondre les données de leur emplacement d'origine à un support physique.

### 3.3.3 HA Proxy

HAProxy est un équilibreur de charge logiciel et un contrôleur de livraison d'applications qui fournit des capacités d'équilibrage de charge pour vSphere avec Tanzu il possède 3 interface :

- Un sous-réseau sera destiné au réseau de **gestion**. C'est là que se trouveront vCenter, ESXi, le Supervisor Cluster
- L'autre sous-réseau sera utilisé pour le **Workload Networking**. C'est là que se trouveront nos clusters invités TKG.
- Le troisième sous-réseau sera utilisé pour le **Frontend Networking** qui sert de donner une plage d'IP virtuelle pour les clusters Kubernetes.

La figure ci-dessous présente l'architecture réseau du HA Proxy.

## 3.4 cluster superviseur

Lorsque Kubernetes est activé, le cluster vSphere devient le cluster superviseur. Les opérateurs informatiques activent le cluster Supervisor pour Kubernetes dans les clusters vSphere en collaboration avec un outil simple dans vsphere client. Il fournit le squelette Kubernetes sur lequel ont intégré des

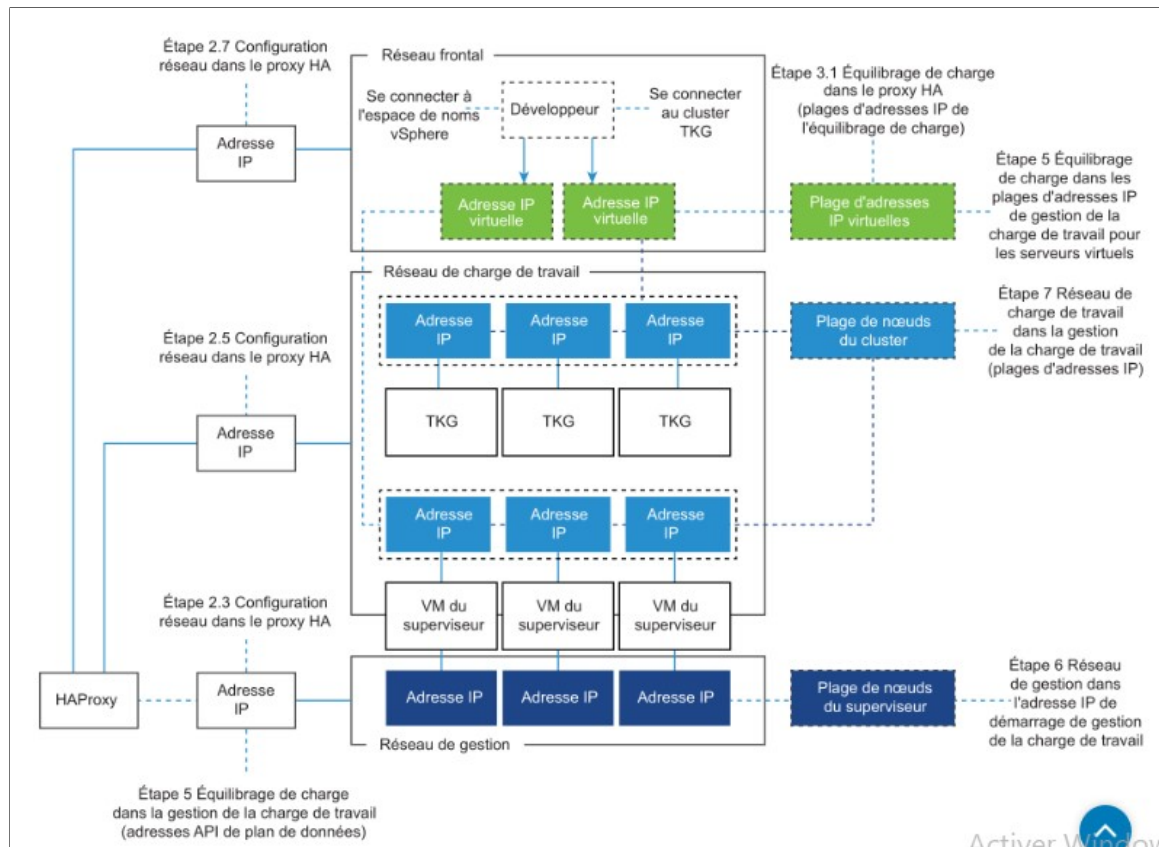


FIGURE 3.11 : Architecture réseau du HA Proxy

services qui peuvent être consommés à la fois par les opérateurs et les équipes DevOps. Trois nœuds de plan de contrôle sont créés et placés dans un cluster vSphere et les nœuds ESXi du cluster agissent en tant que nœuds de travail (fournisseurs de ressources) pour le cluster Kubernetes.

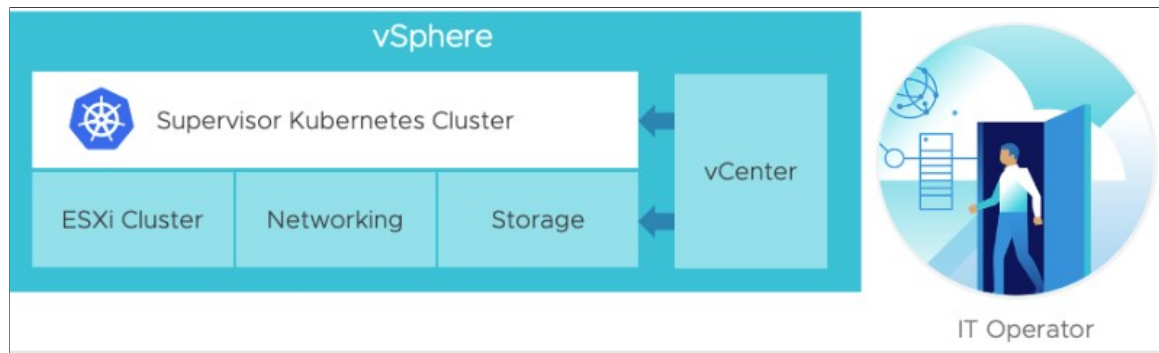


FIGURE 3.12 : Cluster superviseur

### 3.5 Espace du nom

Un espace de noms vSphere est un objet logique créé sur le cluster de superviseur vSphere Kubernetes, il prend en charge l'ensemble complet des fonctionnalités de la plateforme de gestion de la charge de travail. Cet objet suit et fournit un mécanisme pour éditer l'affectation des ressources

(Compute, Memory, Storage Network) et le contrôle d'accès aux ressources Kubernetes, comme les conteneurs ou les machines virtuelles

## 3.6 Conclusion

Dans ce chapitre une combinaison de différents choix techniques et logiciels a offert une vue complète de notre travail. Le chapitre suivant intitulé « Réalisation », nous présentons la partie réalisation en justifiant les choix technologiques, les étapes suivies pour les configurations, et les tests effectués pour la vérification du fonctionnement.

---

# IMPLÉMENTATION DE LA SOLUTION

---

## Plan

1	Introduction . . . . .	39
2	Environnement matériels . . . . .	39
3	Installation de VMware ESXI . . . . .	39
4	Installation de Vcenter . . . . .	43
5	Configuration du Virtuel Distributed Switch (VDS) . . . . .	45
6	Configuration de vmotion . . . . .	46
7	Bibliothèque de contenu VMware . . . . .	46
8	Configuration du politique de stockage . . . . .	47
9	Configuration du HA Proxy . . . . .	48
10	Configurer la gestion de la charge de travail . . . . .	51
11	Création du namespace . . . . .	52
12	L'application : ACME-FITNESS . . . . .	54
13	Conclusion . . . . .	59

## 4.1 Introduction

Après avoir affecté l'étude et la conception de notre travail, nous passons à la phase d'implémentation. Ce chapitre présente le résultat de travail effectué durant ce projet de fin d'études. Nous allons aussi présenter l'environnement matériel et les outils de travail utilisé. Nous clôturons ce chapitre par quelque capture d'écran démontant les fonctionnalités de notre travail.

## 4.2 Environnement matériels

L'équipement mis à notre disposition pour la réalisation de projet se compose par un ordinateur portable dont la configuration est la suivante :

Ordinateurs HP

RAM : 8,00 Go

Système d'exploitation : Windows 10

Type de système : 64 bits

Processeur : Intel(R) Core(TM) i3-3110M CPU @ 2.40GHz 2.40 GHz.

## 4.3 Installation de VMware ESXI

VMware ESXi est gratuit. Cependant, le logiciel est livré avec un mode d'essai de 60 jours. Il faut s'inscrire sur le site Web de VMware pour obtenir une clé de licence gratuite et quitter le mode d'évaluation. On crée deux ESXI : ESXI"3" et ESXI"4" .



FIGURE 4.1 : Installation de VMware ESXi

L'une des premières choses à faire est de configurer l'adresse IP pour la gestion de notre hyperviseur ESXi. Cette adresse IP est l'adresse IP de l'interface de Management. Celle qui nous permettra de nous connecter à l'interface d'administration d'ESXi via l'interface web.

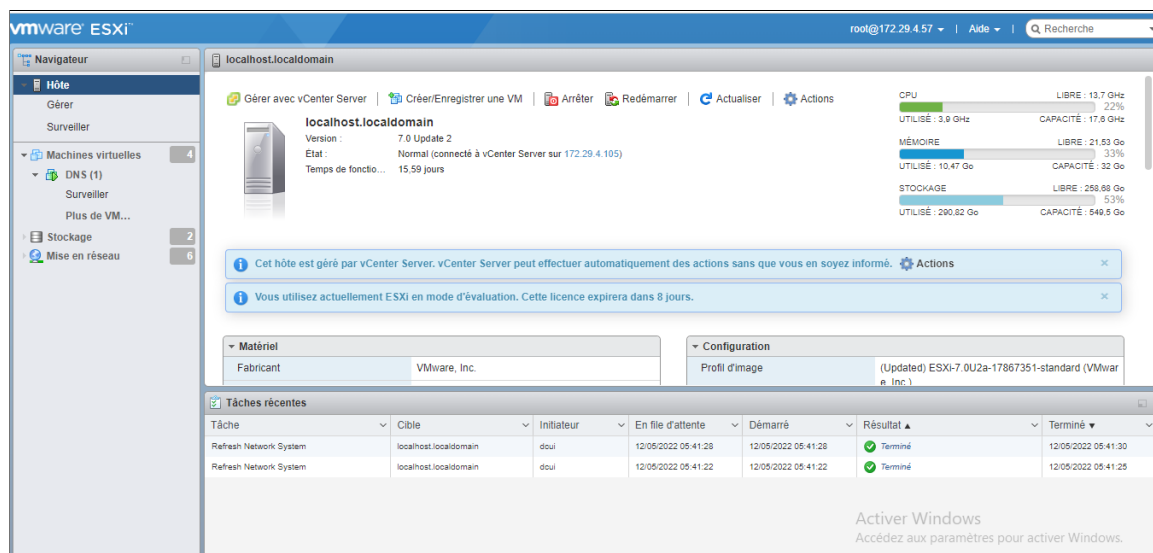


FIGURE 4.2 : interface de VMware ESXi

voici quelques informations sur nos Esxi

- Nom d'hôte : ESXi 7.0.2
- CPU :8

- disque dur :20 GB
- IP ESXI 3 :172.29.4.57
- IP ESXI 4 : 172.29.4.58
- Masque :255.255.255.0
- Des machines virtuelles sans OS

### 4.3.1 Les vSwitchs

Par défaut, le vSwitch0 est déjà créé. Il est connecté à notre réseau principal. notre commutateur standard (vSwitch0) comporte un seul port groupe "VM NETWORK" liée a une machine virtuelle :DNS et 3 port VM KERNEL comporte les services suivante : Management , ISCSI, Vmotion .

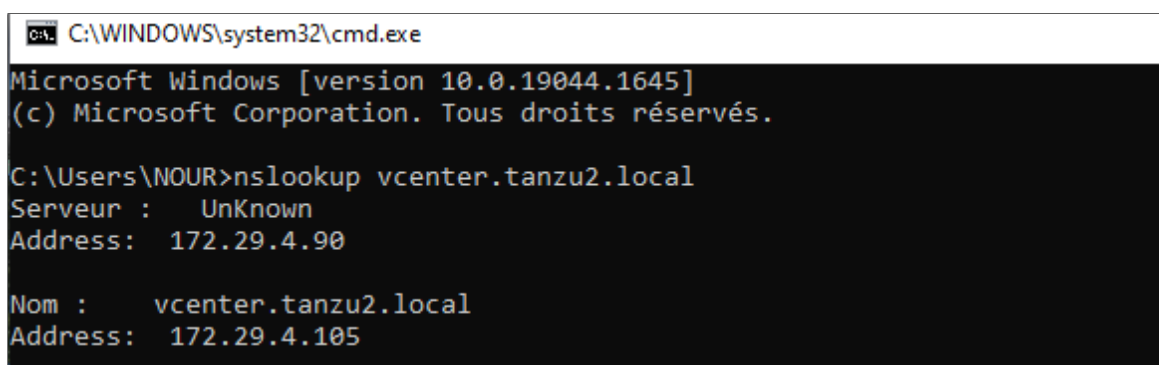
#### 4.3.1.1 Serveur DNS :

DNS(Domain Name System)est un système hiérarchique distribué permettant la résolution des noms de machines en adresses IP et inversement.

- configuration du DNS : on a créé une machine virtuelle avec le système Windows server pour la configuration DNS .

Avant de déployer vCenter Server Appliance avec une adresse IP statique, il faut assurer que celle-ci est enregistrée correctement dans le système de noms de domaine (DNS) interne On peut réaliser un test du serveur DNS a travers la commande suivante **nslookup**.

le résultat nous donne :



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [version 10.0.19044.1645]
(c) Microsoft Corporation. Tous droits réservés.

C:\Users\NOUR>nslookup vcenter.tanzu2.local
Serveur :      UnKnown
Address:  172.29.4.90

Nom :      vcenter.tanzu2.local
Address:  172.29.4.105
```

FIGURE 4.3 : Résultat du nslookup



voici un tableau récapitulatif des adresse IP :

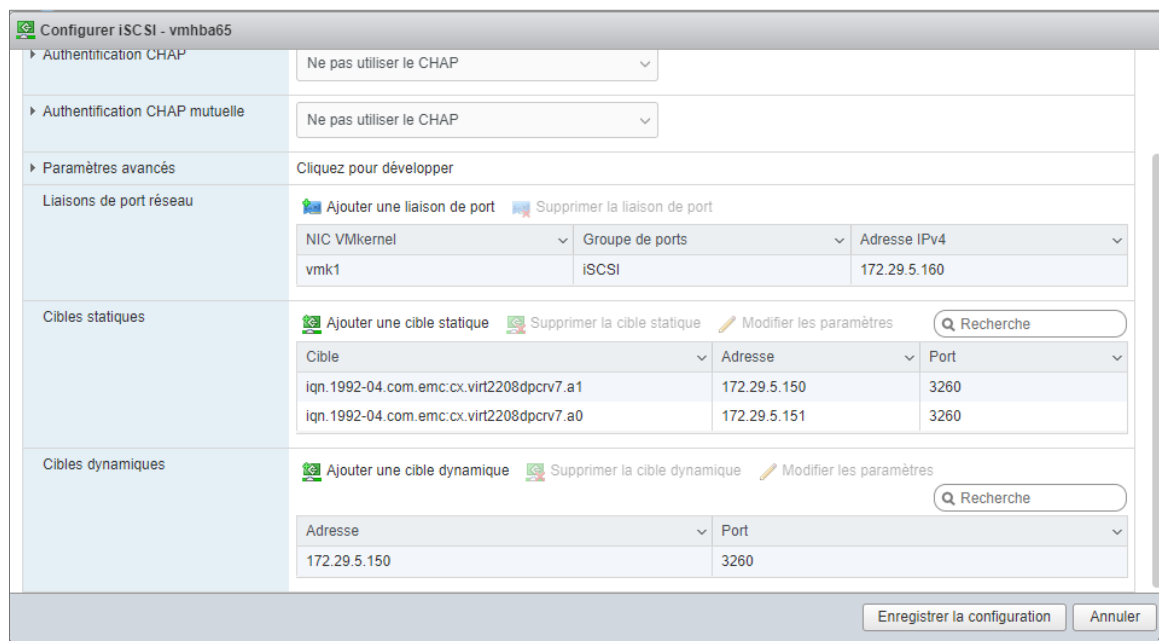
**TABLEAU 4.1** : Configuration du DNS

Domaine :	Tanzu2.local
Utilisateur :	root
Adresse IP DNS server :	172.29.4.90
Adresse IP de Vcenter :	172.29.4.105

- Configuration de iSCSI :

Dans VMware vSphere, nous pouvons utiliser des disques iSCSI comme magasins de données partagés pour les hôtes ESXi. Les hôtes ESXi utilisent le protocole TCP pour accéder au stockage iSCSI sur notre réseau local.

Tout d'abord, nous avons crée une interface réseau VMkernel distincte que l'hôte ESXi utilisera pour accéder au stockage iSCSI avec l'adresse IP : 172.29.5.160 En plus du port vmk, nous avons ajouter l'adresse IP statique suivante pour notre ISCSI : 172.29.5.150



**FIGURE 4.4** : Configuration de iSCSI

**TABLEAU 4.2** : configuration du ISCSI

VMkernel	172.29.5.160
ISCSI	172.29.5.150

Dans l'image ci-dessous, vous pouvez voir une représentation graphique du vSwitch0 :

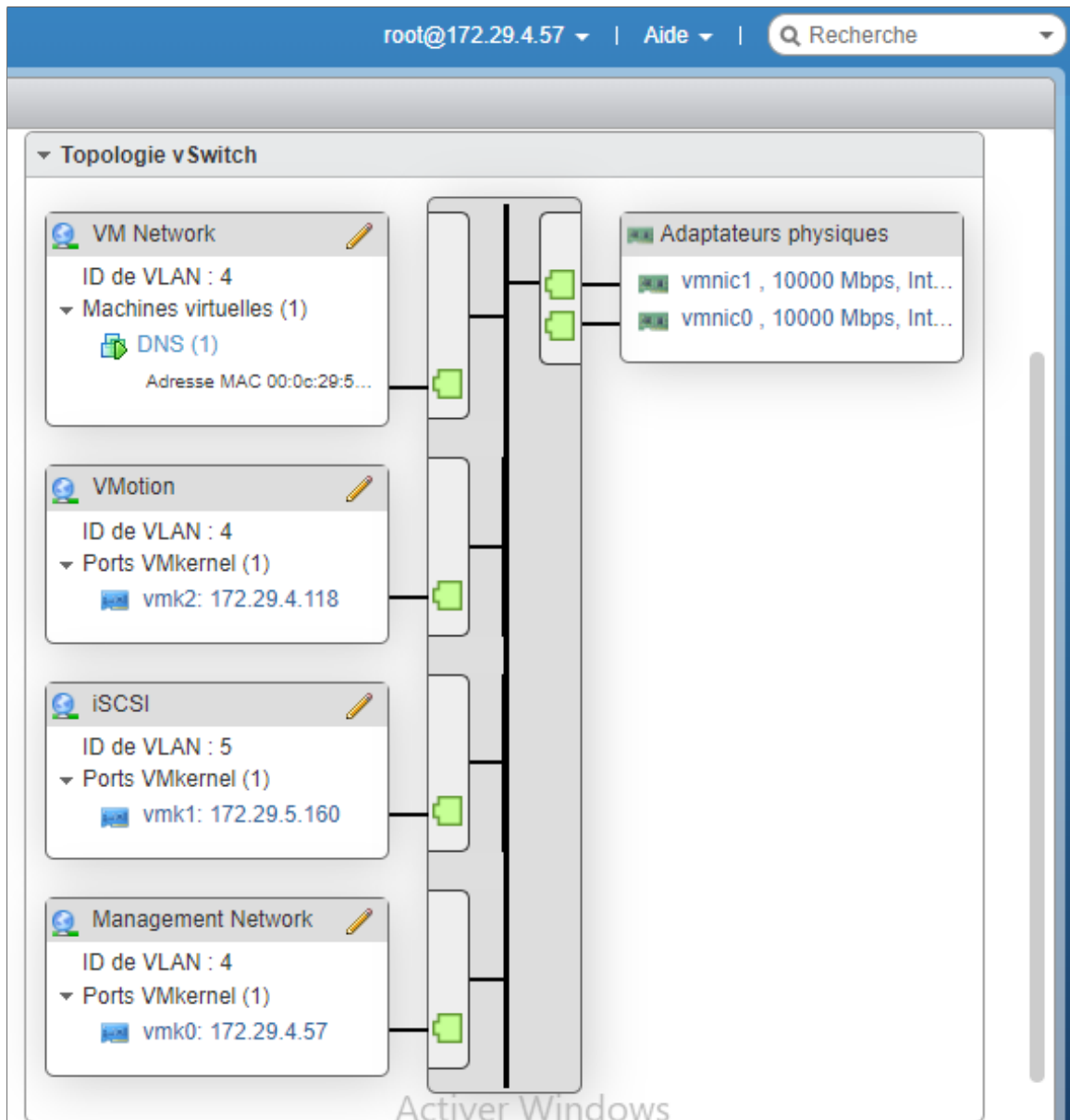


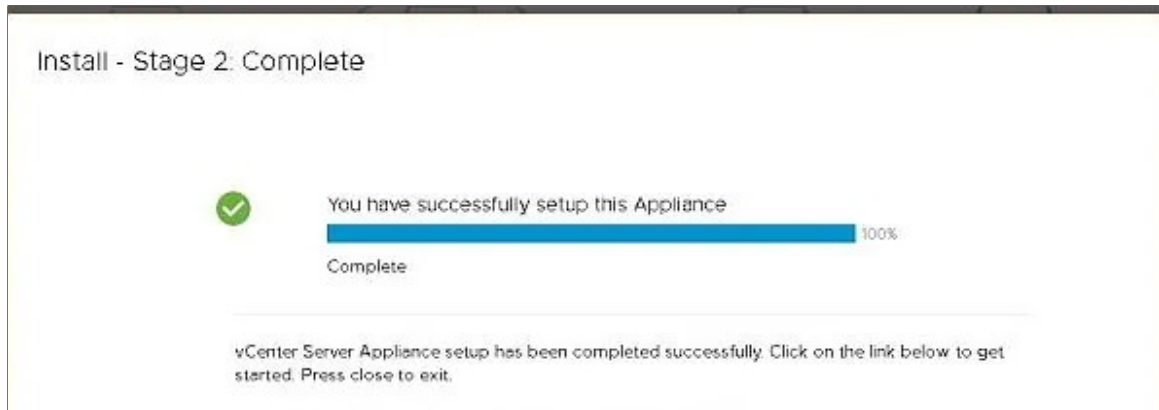
FIGURE 4.5 : Représentation graphique du vSwitch0

## 4.4 Installation de Vcenter

il existe deux méthodes pour déployer vCenter, mais dans notre cas on a utilisé la deuxième méthode, L'appliance vCenter Server (vCSA), c'est une seule machine virtuelle préconfigurée basée sur Linux et optimisée pour exécuter vCenter Server et les services associés. nous avons Téléchargé l'image ISO d'installation de vCenter Server Appliance sur le site Web de VMware :

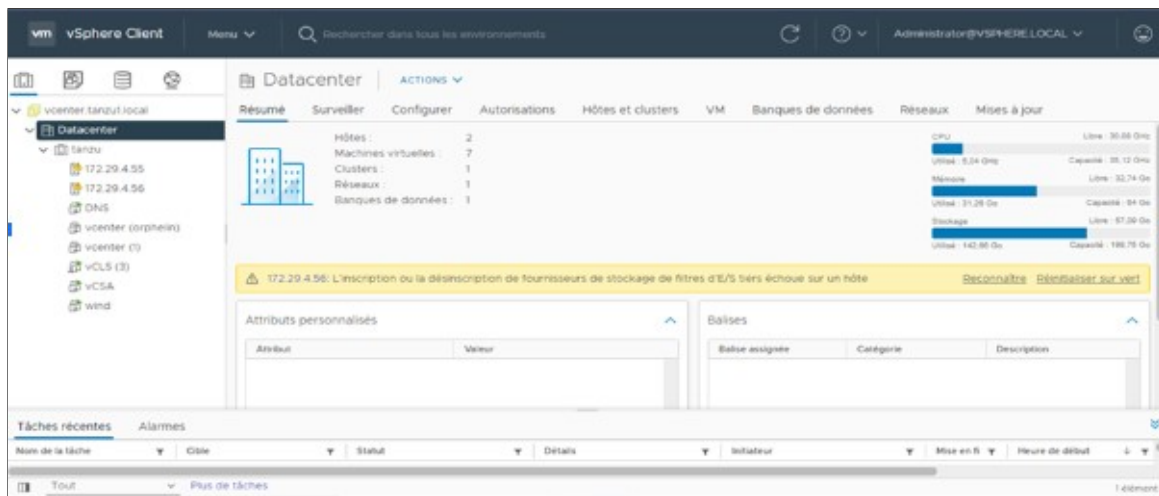
**VMware-VCSA-all-7.0.2-16189094.iso image d'installation de vCenter 7.**

On lance le programme d'installation, puis le processus d'installation s'effectue automatiquement par deux étapes distinctes. À la fin de la première étape, nous allons configurer l'appliance. Après un certain temps, la configuration sera terminée avec succès.



**FIGURE 4.6 :** Etape 2 avec succes

On accède a notre interface Vcenter :



**FIGURE 4.7 :** Interface Vcenter

#### 4.4.1 Le schéma de déploiement

Dans notre parcours, nous avons installé deux serveurs ESXi, déployer vCenter Server Appliance sur le premier hôte ESXi et utiliser le deuxième hôte ESXi pour exécuter d'autres VM. voici un schéma de déploiement pour notre configuration :

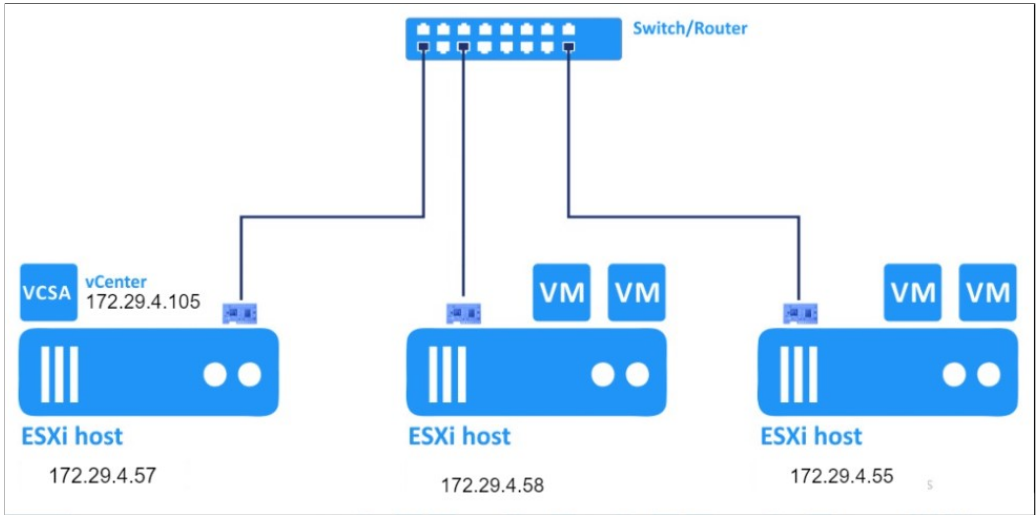


FIGURE 4.8 : Schéma de déploiement

### 4.5 Configuration du Virtuel Distributed Switch (VDS)

Après la création du cluter nommé "TANZU" et l'activation du HA et DRS , On ajoute les deux ESXI :172.29.4.57,172.29.4.58

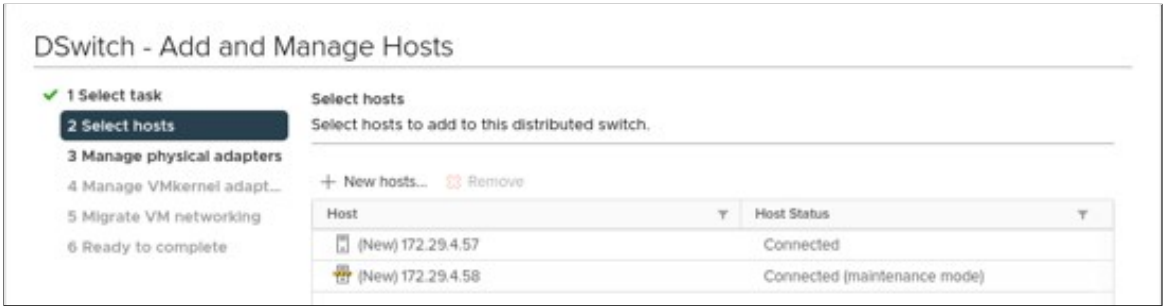


FIGURE 4.9 : Ajout des ESXI

Notre topologie comprenne 3 reseaux :Réseau de gestion,Réseau de charge de travail et Réseau frontal.

TABLEAU 4.3 : Les interfaces réseaux

Réseau	Objectif principal	Vlan
Gestion	Communiquer avec vCenter, HA Proxy	4
Gestion de charge de travail	Adresses IP pour les nœuds Kubernetes	5
frontal	Plage d'adresses IP virtuelles pour les clusters Kubernetes	6

Cette figure résume notre topologie :

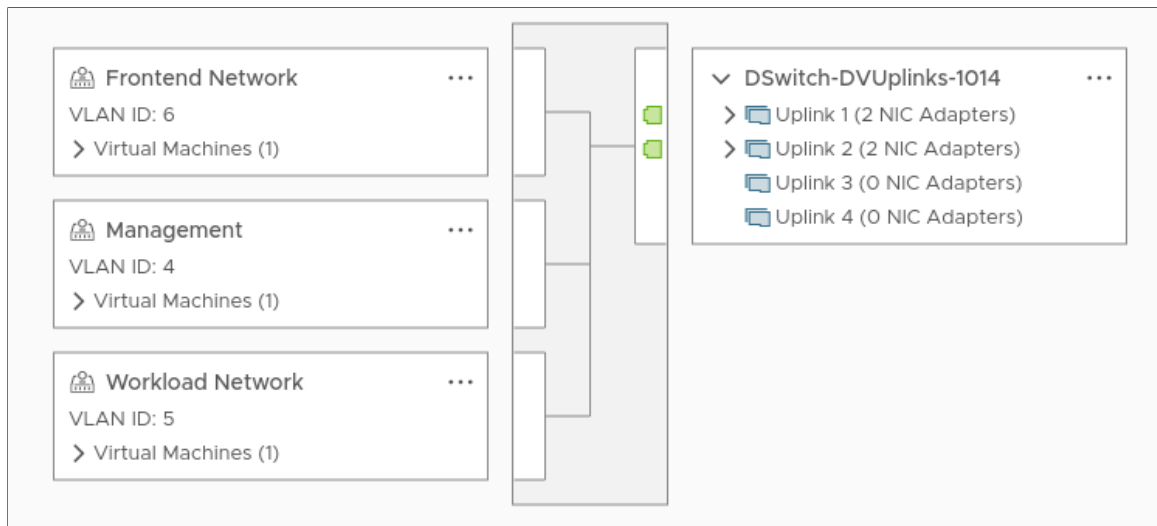


FIGURE 4.10 : Topologie de Vds

## 4.6 Configuration de vmotion

On souhaite utiliser vMotion dans notre environnement de virtualisation, c'est pourquoi il faut activer l'option de Vmotion.

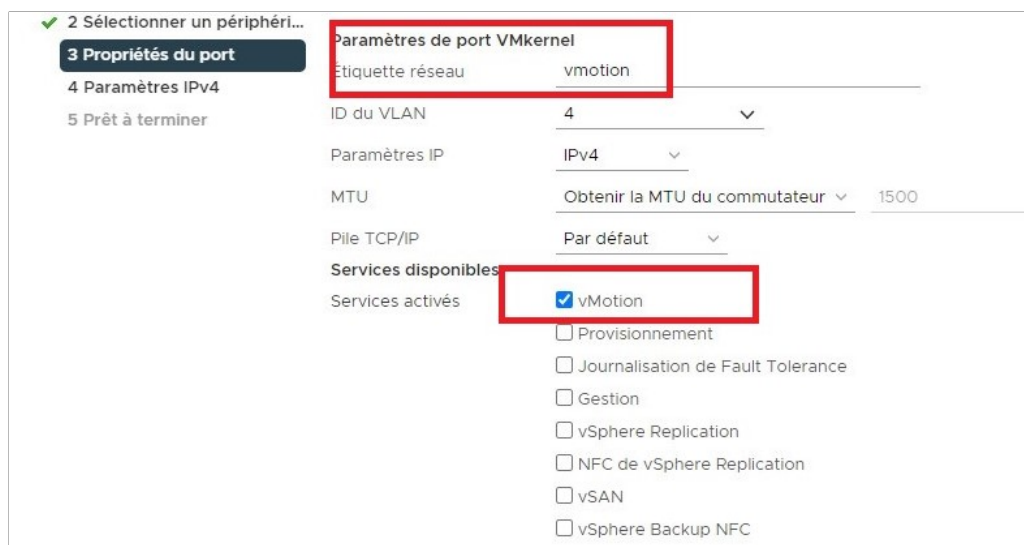


FIGURE 4.11 : Option Vmotion

Nous avons donné à chaque ESXI une adresse IP et le masque de sous-réseau de l'adaptateur.

## 4.7 Bibliothèque de contenu VMware

La dernière étape avant de créer notre HA Proxy est de créer notre bibliothèque de contenu à travers une simple configuration. Celle-ci nous permet à Tanzu de récupérer les images OVA dont il a besoin pour créer le cluster superviseur et le cluster Tanzu.

1 Sélectionner un type de c...  
 2 Sélectionner un périphéri...  
 3 Propriétés du port  
**4 Paramètres IPv4**  
 5 Prêt à terminer

Paramètres IPv4  
 Définissez les paramètres IPv4 de VMkernel.

☐ Obtenir automatiquement les paramètres IPv4  
☒ Utiliser des paramètres IPv4 statiques

Adresse IPv4: 172.29.4.116  
 Masque de sous-réseau: 255.255.255.0  
 Passerelle par défaut: ☒ Remplacer la passerelle par défaut pour l'adaptateur 172.29.4.254

FIGURE 4.12 : Adresse Vmotion pour ESXI

New Content Library

1 Name and location  
 2 Configure content library  
 3 Add storage  
**4 Ready to complete**

Ready to complete  
 Review content library settings.

Name: TKGS-CL  
 Notes:  
 vCenter Server: vcenter.tanzu2.local  
 Type: Local Content Library  
 Publishing: Disabled  
 Storage: DS1

CANCEL BACK FINISH

FIGURE 4.13 : Bibliothèque de contenu VMware

## 4.8 Configuration du politique de stockage

Il est intéressant de configurer une politique de stockage indépendante, c'est pourquoi nous devons sélectionner une politique de stockage qui définit les datastores compatibles. Nous créons une politique **Gold storage Policy** comme condition préalable qui sélectionne notre vsan Datastore. Sélectionnons le profil de stockage configuré pour chacun de ces composants.

The screenshot shows a web-based configuration wizard for creating a VM Storage Policy. The wizard is titled 'Create VM Storage Policy' and has a close button (X) in the top right corner. On the left, there is a vertical sidebar with five steps: 1 Name and description, 2 Policy structure, 3 Tag based placement, 4 Storage compatibility, and 5 Review and finish. Step 5 is currently selected and highlighted in dark blue. The main area of the wizard is titled 'Review and finish' and contains two sections: 'General' and 'Tag based placement'. The 'General' section has two rows: 'Name' with the value 'Gold Storage Policy' and 'Description' with the value 'vcenter.tanzu2.local'. The 'Tag based placement' section has one row: 'Tagged with: Storage for Kubernetes' with the value 'Gold'. At the bottom right of the wizard, there are three buttons: 'CANCEL' (blue), 'BACK' (blue), and 'FINISH' (green).

General	
Name	Gold Storage Policy
Description	vcenter.tanzu2.local

Tag based placement	
Tagged with: Storage for Kubernetes	Gold

FIGURE 4.14 : Configuration du politique de stockage

## 4.9 Configuration du HA Proxy

HA Proxy est disponible au format OVA est la meilleure dernière version peut être téléchargée depuis le dépôt Git de VMware. Après avoir téléchargé OVA de HA Proxy dans la bibliothèque de conteneurs, le processus de configuration commence. Sur la page de configuration, on choisit le type de déploiement comme "Réseau frontal" : pour séparer nos services Kubernetes Load balanced de notre réseau de charge de travail.

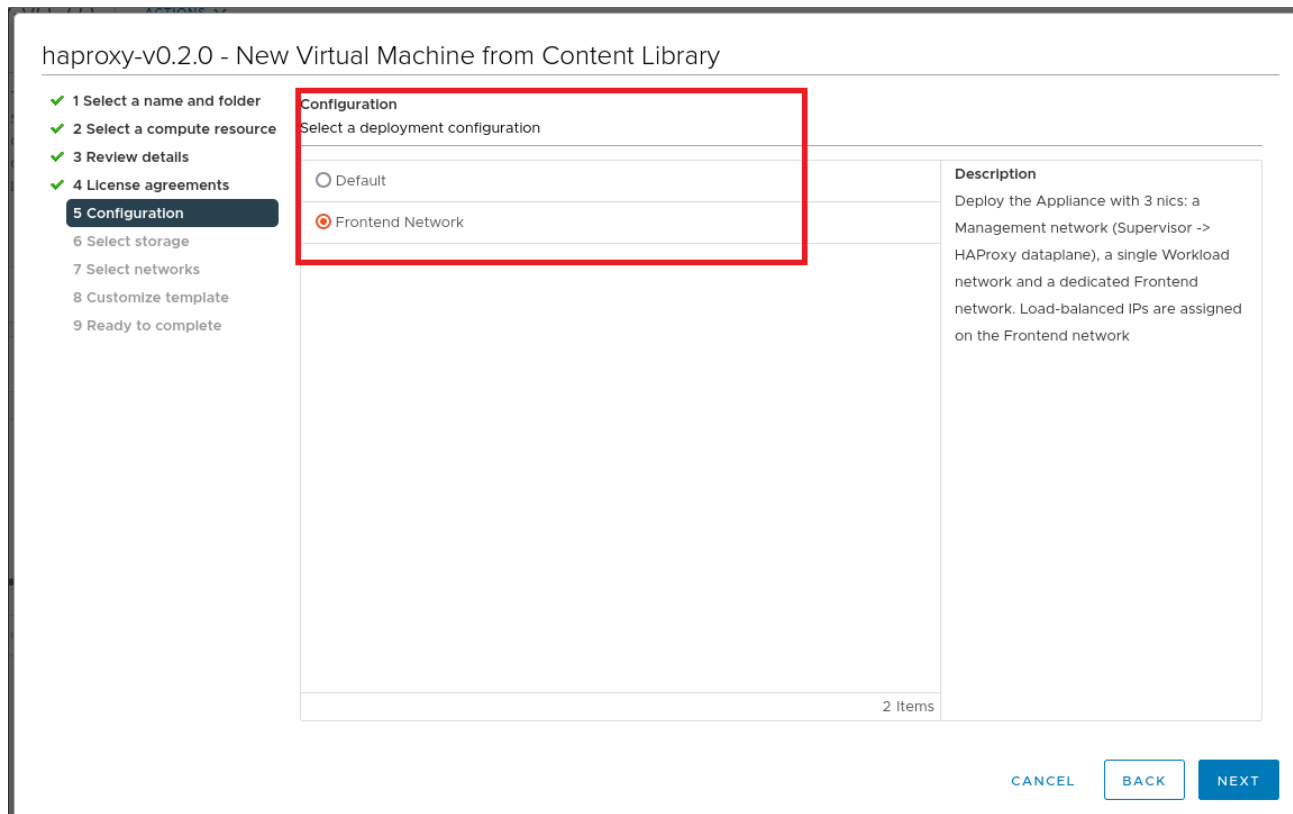


FIGURE 4.15 : Réseau frontal

On présente le tableau suivant avec une figure qui illustre les plage d'IP pour chacun VLAN :

TABLEAU 4.4 : les plages d'adresses des vlan

type de réseau	Vlan	Sous-réseau	passerelle
réseau de gestion	4	172.29.4.230/24	172.29.4.254
réseau de charge de travail	5	172.29.5.230/24	172.29.5.254
réseau frontal	6	172.29.5.230/24	172.29.6.254



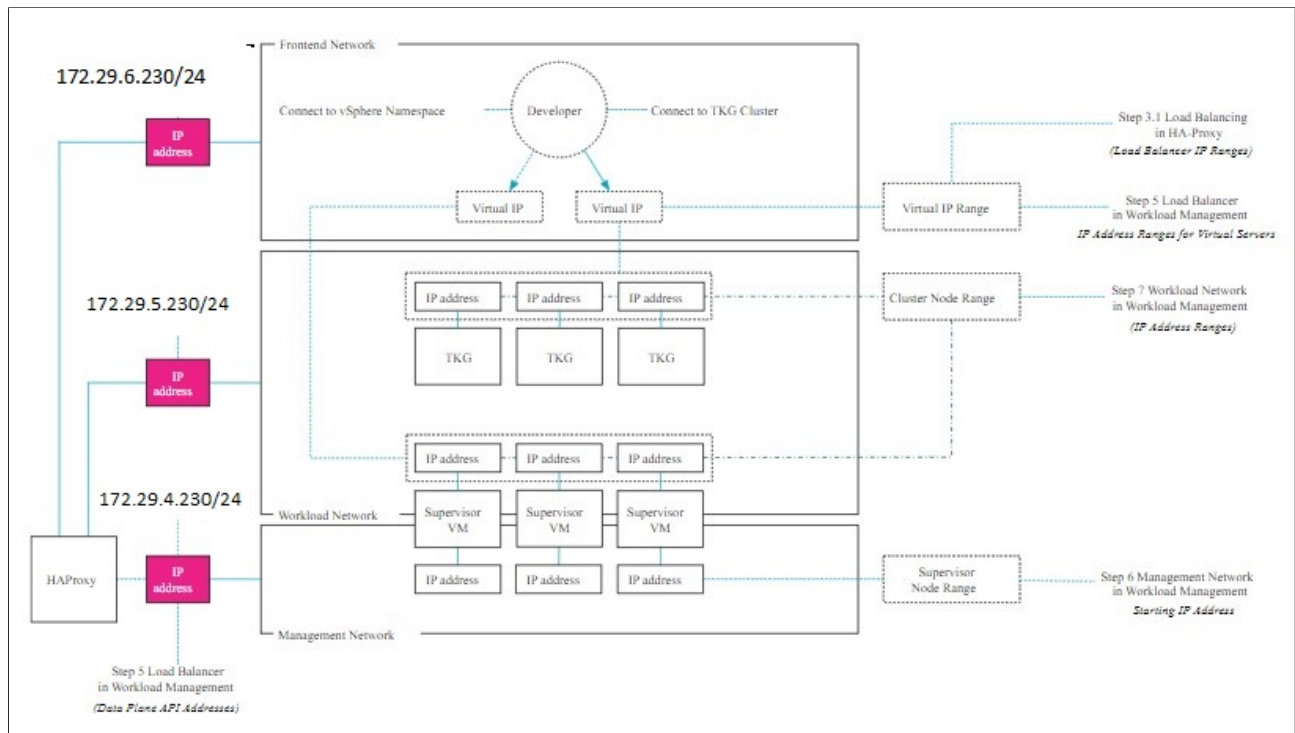


FIGURE 4.16 : Architecture du Réseau frontal

Dans cette partie , nous allons configurer une plage d'adresses IP d'équilibreur de charge, qui est utilisée pour fournir une IP virtuelle à un cluster Kubernetes. **plage d'adresse IP : 172.29.6.160/28**

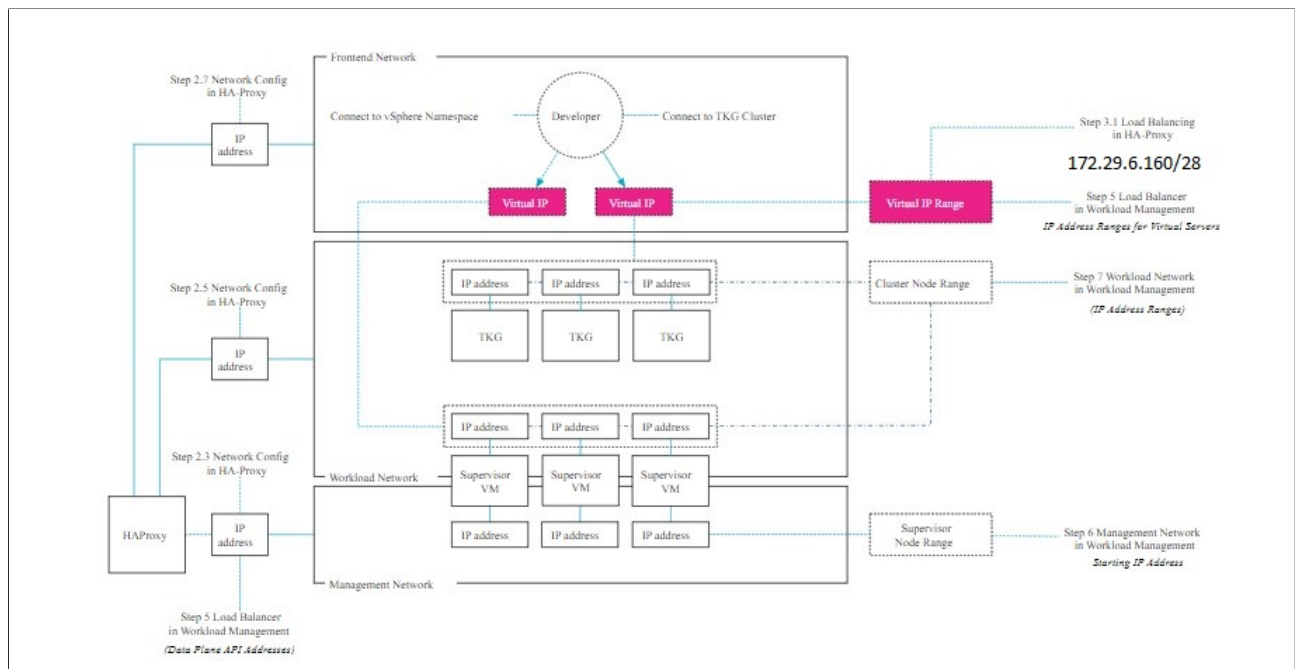


FIGURE 4.17 : Plage d'adresses IP d'équilibreur de charge

Nous avons fait une simple vérification de l'état de tous les services en exécutant `systemctl status`. On a remarqué un service qui ne fonctionne pas .

Le simple fait de redémarrer le service avec `systemctl restart anyip-routes.service`

```
root@haproxy [ / ]# systemctl list-units --state=failed

UNIT                                LOAD    ACTIVE SUB    DESCRIPTION
● anyip-routes.service loaded failed failed anyip-routes.service
```

FIGURE 4.18 : Service anyip-routes.service n'est pas activé

```
root@haproxy [ ~ ]# systemctl restart anyip-routes.service
root@haproxy [ ~ ]#
root@haproxy [ ~ ]# systemctl status anyip-routes.service
● anyip-routes.service
   Loaded: loaded (/etc/systemd/system/anyip-routes.service; enabled; vendor preset: enabled)
   Active: active (running) .
```

FIGURE 4.19 : Service anyip-routes.service activé

## 4.10 Configurer la gestion de la charge de travail

Au cours du processus d'activation, 3 VM Supervisor Control Plane ont été créées pour la haute disponibilité, qui agissent en tant que serveur API Kubernetes et hôtes etcd et fonctionnent comme plan de contrôle Kubernetes pour le cluster vSphere.

Pendant la phase de configuration, nous verrons le déploiement/la configuration automatisés des vms du cluster de superviseurs. chaque VM Supervisor Control possède 2 adresse IP :

Deploy OVF template	 SupervisorControlPlaneV...	<div><div></div></div>	9%	
Deploy OVF template	 SupervisorControlPlaneV...	<div><div></div></div>	10%	
Deploy OVF template	 SupervisorControlPlaneV...	<div><div></div></div>	10%	

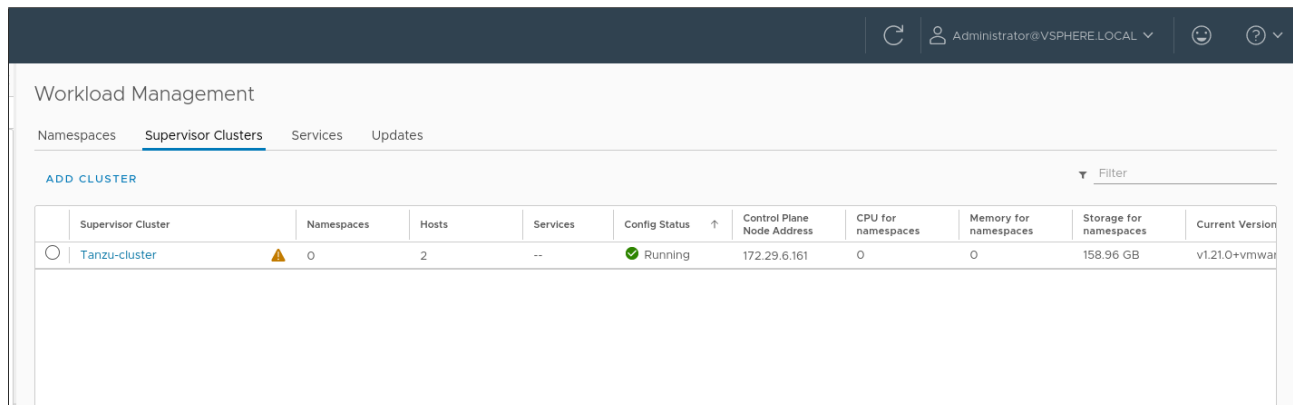
FIGURE 4.20 : Déploiement des vms du cluster de superviseurs.

\* La carte réseau 1 est le réseau de gestion appartient au VLAN 4 : prend en charge le trafic vers vCenter. Ce réseau est connecté à un groupe de ports vDS

\*L'adaptateur réseau 2 appartient au VLAN 5 : prend en charge le trafic vers l'API Kubernetes et la façon dont le cluster Supervisor communique avec les clusters TKG Workload.

Une fois la configuration du cluster terminée, l'état de la configuration sera indiqué comme étant en cours d'exécution et vous verrez le VIP du plan de contrôle auquel vous pouvez vous connecter pour télécharger les utilitaires permettant de provisionner/gérer les clusters invités TKG

Cette figure représente que la configuration du TANZU a été terminée avec succès



Workload Management										
Namespaces Supervisor Clusters Services Updates										
ADD CLUSTER Filter										
	Supervisor Cluster	Namespaces	Hosts	Services	Config Status ↑	Control Plane Node Address	CPU for namespaces	Memory for namespaces	Storage for namespaces	Current Version
<input type="radio"/>	Tanzu-cluster	0	2	--	Running	172.29.6.161	0	0	158.96 GB	v1.21.0+vmwar

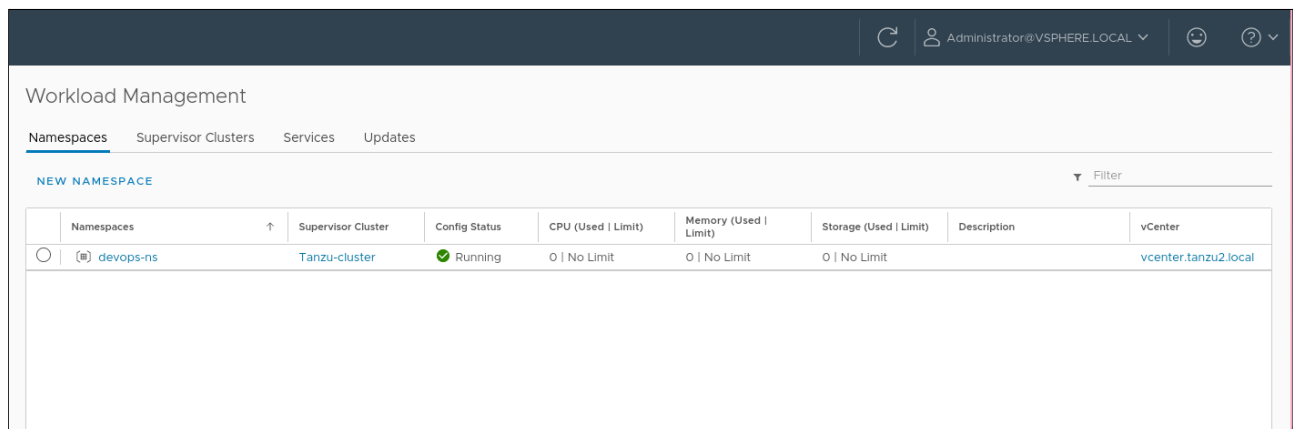
FIGURE 4.21 : Résultat du configuration

## 4.11 Création du namespace

Un espace de nom permet à un administrateur vSphere de contrôler les ressources qui sont disponibles pour un développeur afin de provisionner des clusters TKG. L'utilisation d'administrateurs vSphere d'espaces de noms empêche un développeur de consommer plus de ressources que celles qui lui sont attribuées.

Dans gestion de charge de travail, on commence la création du namespace en Sélectionnant le cluster où l'espace de noms sera créé et le réseau de charge de travail pour votre espace de noms.

Notre espaces de nom a pour nom : **Devops-ns**



Workload Management								
Namespaces Supervisor Clusters Services Updates								
NEW NAMESPACE Filter								
	Namespaces ↑	Supervisor Cluster	Config Status	CPU (Used   Limit)	Memory (Used   Limit)	Storage (Used   Limit)	Description	vCenter
<input type="radio"/>	devops-ns	Tanzu-cluster	Running	0   No Limit	0   No Limit	0   No Limit		vcenter.tanzu2.local

FIGURE 4.22 : Résultat de la configuration

Une fois l'espace de nom créé, nous devons lui attribuer des limites/quotas de ressources.

Nous avons téléchargé les outils pour le système d'exploitation Windows afin de pouvoir exécuter la commande kubectl depuis n'importe quel endroit du système d'exploitation.

Avant de commencer à déployer des clusters TKG, il est important de vérifier les détails importants du plan de contrôle déployé.

### Valider le plan de contrôle (Supervisor Cluster)

-Se connecter au contexte de l'espace de nom :

```
kubectl vsphere login --server 172.29.4.161 --insecure-skip-tls-verify
```

-Changement de contexte vers votre espace de nom :

**kubectl config use-context devops-ns** Obtenir des informations sur le plan de contrôle

Avant de déployer le cluster TKG, nous devons assurer que les nœuds du plan de contrôle sont dans un état Prêt et que la politique de stockage que nous avons spécifiée pendant l'activation de la charge de travail apparaît comme une classe de stockage.

```
C:\Users\NOUR>kubectl get nodes
NAME                                STATUS    ROLES    AGE     VERSION
421c2ca3ac4d32573639c7815110a135  Ready    control-plane,master  46h    v1.21.0+vmware.wcp.2
421c4991cc63ed36ca6fc915b4e1dceb  Ready    control-plane,master  2d     v1.21.0+vmware.wcp.2
421ccb7a7fe2d81b1b41434178edc1c0  Ready    control-plane,master  46h    v1.21.0+vmware.wcp.2
```

FIGURE 4.23 : Nœuds du plan de contrôle

```
C:\Users\NOUR>kubectl get storageclasses
NAME                                PROVISIONER    RECLAIMPOLICY    VOLUMEBINDINGMODE    ALLOWVOLUMEEXPANSION    AGE
gold-storage-policy                csi.vsphere.vmware.com  Delete           Immediate             true                     39h
```

FIGURE 4.24 : Politique de stockage

#### 4.11.1 Tanzu kubernetes cluster

Une fois le cluster de gestion en place, On peut démarrer le déploiementle du clusters de charge de travail. Le déploiement du cluster TKG se fait via un fichier au format YAML.Dans ce fichier on peut faire quelque modification :

FIGURE 4.25 : Fichier Yaml de Tanzu kubernetes cluster

Nous avons Exécuté la commande suivante pour déployer le cluster de charges de travail :

**tanzu cluster create tkg-workload --file .config/tanzu/tkg/clusterconfigs** Pour obtenir des informations sur le cluster, exécutez la commande **tanzu cluster get**, en spécifiant le nom du cluster.

FIGURE 4.26 : Tanzu kubernetes cluster

#### Définir le contexte Kubectl sur le cluster de charge de travail

VMware Tanzu ne définit pas automatiquement le contexte kubectl sur un cluster de charge de travail lorsque nous le créez. Il faut définir manuellement le contexte kubectl sur le cluster de charge de travail en utilisant la commande :

**kubectl config use-context.**

```
ali@maram-nour-roudayna:~/config/tanzu/tkg/clusterconfigs$ tanzu cluster kubeconfig get tkg-workload -n devops-ns --admin
Credentials of cluster 'tkg-workload' have been saved
You can now access the cluster by running 'kubectl config use-context tkg-workload-admin@tkg-workload'
```

FIGURE 4.27 : le kubeconfig du cluster de charge de travail

## 4.12 L'application : ACME-FITNESS

### 4.12.1 Présentation

Les clusters Kubernetes Tanzu sont des clusters Kubernetes prêts pour la production qui peuvent héberger des applications micro services complexes. L'application que nous allons déployer s'appelle acme-fitness. Il s'agit d'une application dont la logique métier a été décomposée en petites applications polyglottes. Elle est composée d'un service frontal, d'un service utilisateur, d'un service catalogue, d'un service panier, d'un service paiement et d'un service commande.

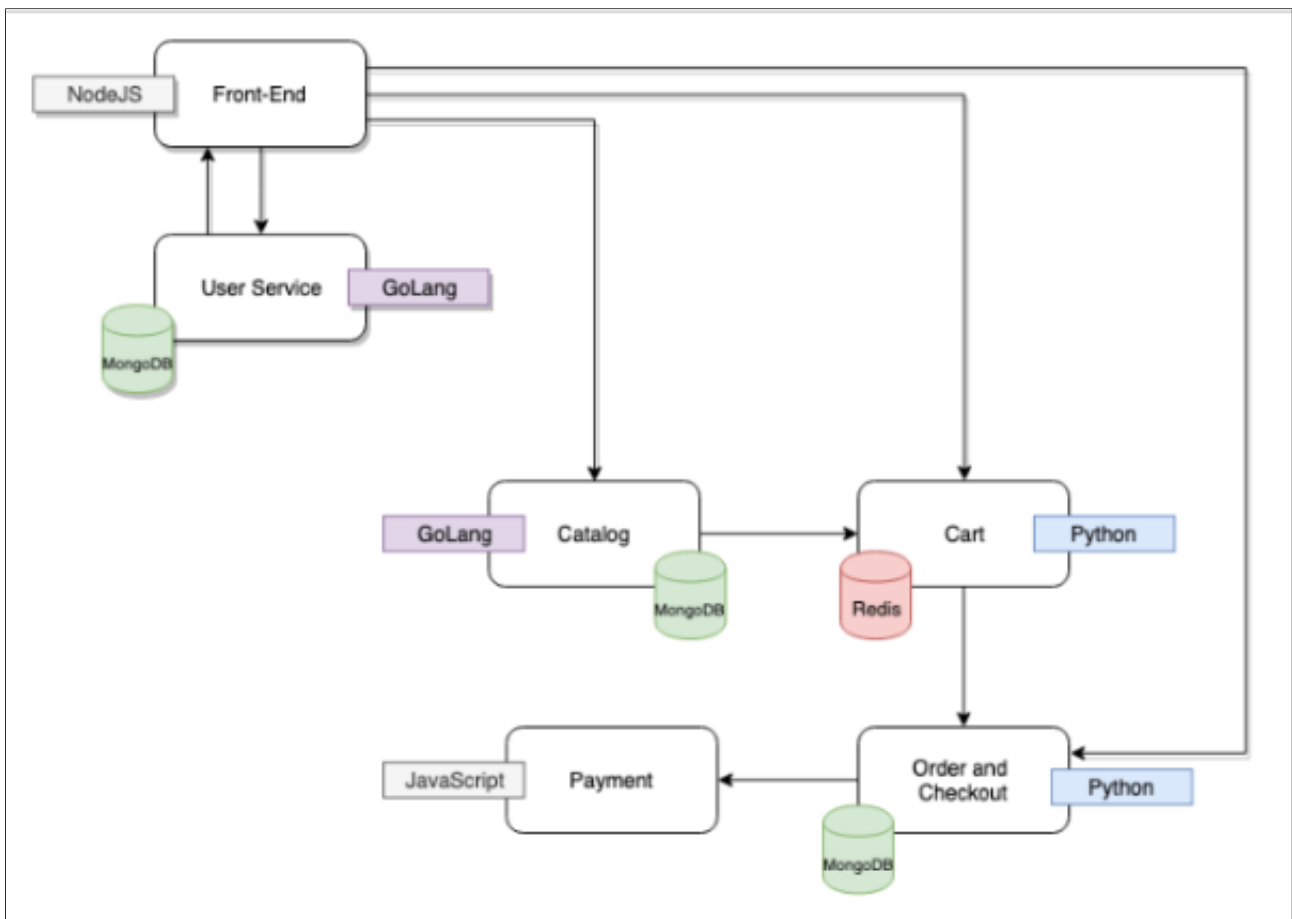


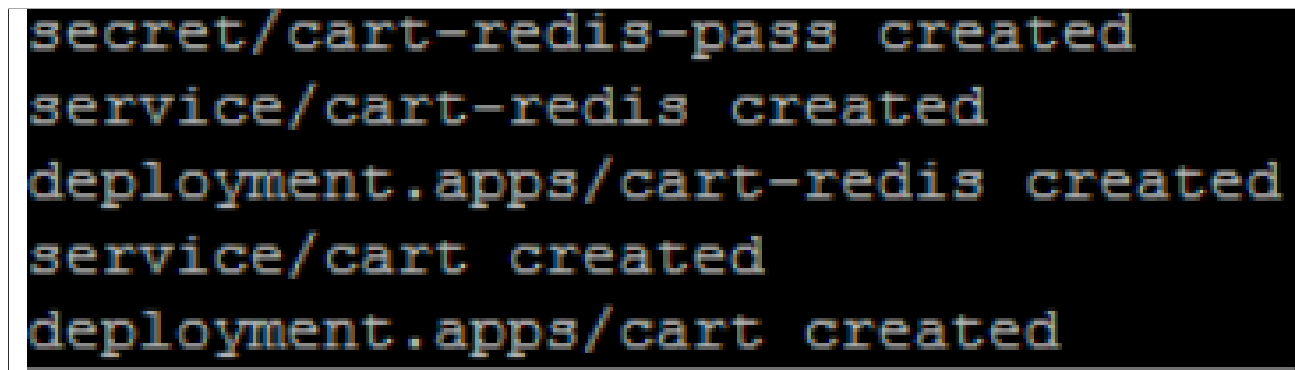
FIGURE 4.28 : Architecture d'application

### 4.12.2 Les étapes de configuration :

Naviguons dans le dossier labs /acme-fitness pour chaque service décrit ci-dessus : Les dossiers cart-service , catalog frontend-service order-service, payment-service et users-service contiennent les manifestes permettant de déployer l'application et les bases de données associées. Les services doivent

être initialement déployés dans un certain ordre. Pour regrouper toutes les ressources de l'application, nous allons déployer l'application dans un espace de noms Kubernetes. On tape la commande suivante dans Putty pour déployer le service Cart et sa base de données Redis :

```
kubectl apply -f cart-service.yaml
```

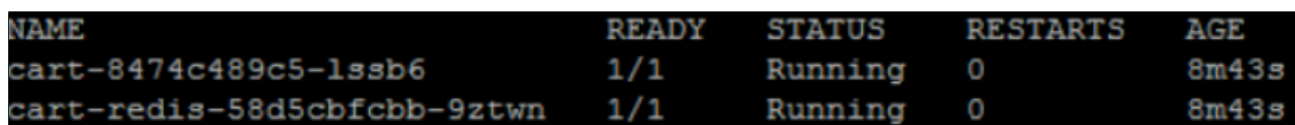


```
secret/cart-redis-pass created
service/cart-redis created
deployment.apps/cart-redis created
service/cart created
deployment.apps/cart created
```

FIGURE 4.29 : Déploiement de service de cart

la commande suivante dans Putty vérifie que le service Cart est en marche. nous devons attendre quelques minutes jusqu'à ce que tous les pods soient affichés comme étant en cours d'exécution :

```
kubectl get pod -n acme-fitness
```



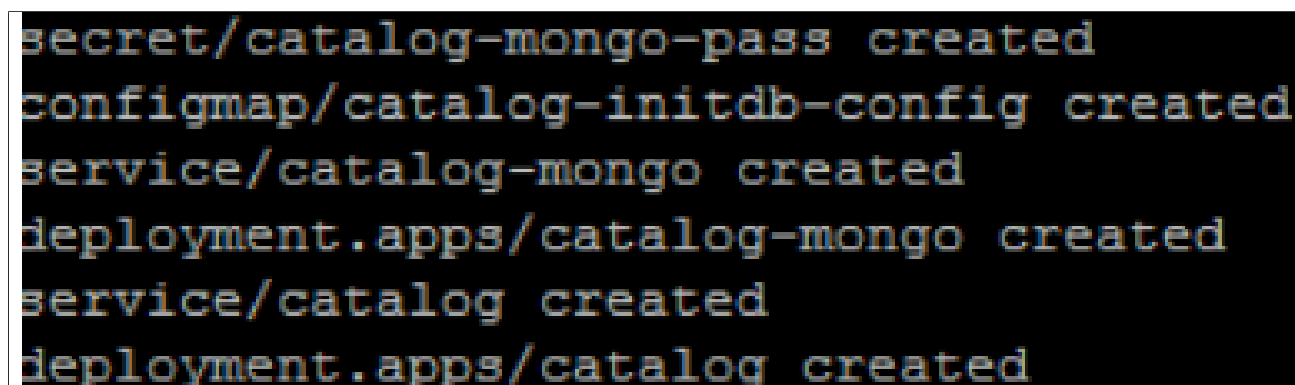
NAME	READY	STATUS	RESTARTS	AGE
cart-8474c489c5-ls3b6	1/1	Running	0	8m43s
cart-redis-58d5cbfcb-9ztwn	1/1	Running	0	8m43s

FIGURE 4.30 : Cart activé

#### 4.12.2.1 Service de catalogue

A partir de la commande suivante : `cd /labs/acme-fitness/catalog-service` , nous avons accédé dans le répertoire `/labs/acme-fitness/catalog-service` puis , nous avons tapé la commande suivante pour déployer le service de catalogue et sa base de données MongoDB :

```
kubectl apply -f catalog-service.yaml
```

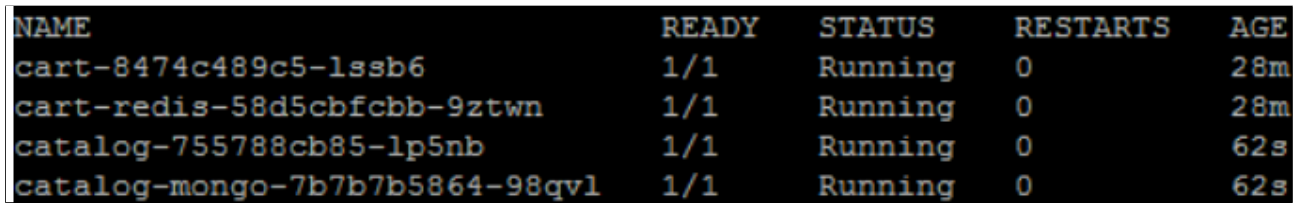


```
secret/catalog-mongo-pass created
configmap/catalog-initdb-config created
service/catalog-mongo created
deployment.apps/catalog-mongo created
service/catalog created
deployment.apps/catalog created
```

FIGURE 4.31 : Déploiement de service du catalog

La commande suivante dans Putty vérifie que les pods du service de catalogue sont opérationnels. Vous devrez peut-être attendre quelques minutes jusqu'à ce que tous les pods soient affichés comme étant en cours d'exécution :

**kubectl get pod -n acme-fitness**



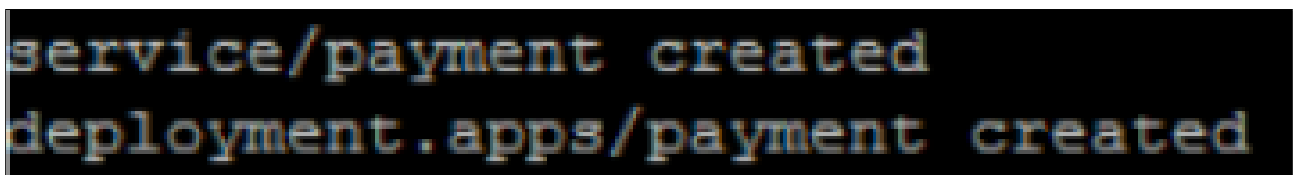
NAME	READY	STATUS	RESTARTS	AGE
cart-8474c489c5-1ssb6	1/1	Running	0	28m
cart-redis-58d5cbfcb-9ztwn	1/1	Running	0	28m
catalog-755788cb85-1p5nb	1/1	Running	0	62s
catalog-mongo-7b7b7b5864-98qvl	1/1	Running	0	62s

FIGURE 4.32 : Catalog activé

#### 4.12.2.2 Service de paiement

On Tape la commande suivante dans Putty : `cd /labs/acme-fitness/payment-service`, puis On tape la commande suivante dans Putty. Pour déployer le service de paiement. Nous avons tapé cette commande dans notre répertoire spécifique : `/labs/acme-fitness/payment-service`

**kubectl apply -f payment-service.yaml**

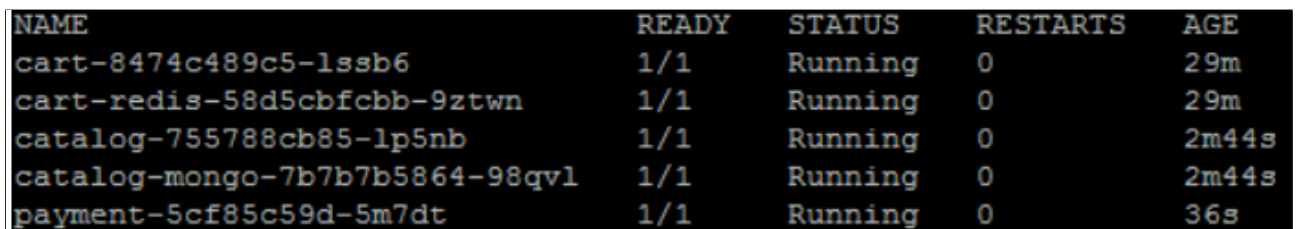


```
service/payment created
deployment.apps/payment created
```

FIGURE 4.33 : Déploiement de service de paiement

la commande suivante dans Putty exécute les pods du service de paiement sont en place et fonctionnent. nous devons peut-être attendre quelques minutes jusqu'à ce que tous les pods soient affichés comme étant en cours d'exécution :

**kubectl get pod -n acme-fitness**



NAME	READY	STATUS	RESTARTS	AGE
cart-8474c489c5-1ssb6	1/1	Running	0	29m
cart-redis-58d5cbfcb-9ztwn	1/1	Running	0	29m
catalog-755788cb85-1p5nb	1/1	Running	0	2m44s
catalog-mongo-7b7b7b5864-98qvl	1/1	Running	0	2m44s
payment-5cf85c59d-5m7dt	1/1	Running	0	36s

FIGURE 4.34 : Paiement activé

#### 4.12.2.3 Service de commande

On tape la commande suivante dans Putty : `cd /labs/acme-fitness/order-service` On tape la commande suivante dans Putty pour déployer le service de commande et sa base de données Postgres :



```
kubectl apply -f order-service.yaml
```

```
secret/order-postgres-pass created
service/order-postgres created
deployment.apps/order-postgres created
service/order created
deployment.apps/order created
```

FIGURE 4.35 : Déploiement de service de commande

On tape la commande suivante dans Putty pour vérifier que les pods du service d'ordre sont en marche.

```
kubectl get pod -n acme-fitness
```

NAME	READY	STATUS	RESTARTS	AGE
cart-8474c489c5-1ssb6	1/1	Running	0	31m
cart-redis-58d5cbfcbb-9ztwn	1/1	Running	0	31m
catalog-755788cb85-lp5nb	1/1	Running	0	4m21s
catalog-mongo-7b7b7b5864-98qvl	1/1	Running	0	4m21s
order-847d8f8749-pmk5g	0/1	ContainerCreating	0	23s
order-postgres-5ff8f9cf8d-b659m	1/1	Running	0	23s
payment-5cf85c59d-5m7dt	1/1	Running	0	2m13s

FIGURE 4.36 : Service de commande activé

#### 4.12.2.4 Service des utilisateurs

Nous avons accédé a notre repertoire specifique a travers ;

```
cd /labs/acme-fitness/users-service puis nous avons tapé kubectl apply -f users-service.yaml
```

tape pour déployer le Service Utilisateurs et ses bases de données Mongo et Redis :

```
secret/users-mongo-pass created
secret/users-redis-pass created
configmap/users-initdb-config created
service/users-mongo created
deployment.apps/users-mongo created
service/users-redis created
deployment.apps/users-redis created
service/users created
deployment.apps/users created
```

FIGURE 4.37 : Déploiement de service utilisateurs

Nous avons vérifié que les pods du service Utilisateurs sont opérationnels a travers cette commande :



```
kubectl get pod -n acme-fitness
```

NAME	READY	STATUS	RESTARTS	AGE
cart-8474c489c5-1ssb6	1/1	Running	0	32m
cart-redis-58d5cbfcb-9ztwn	1/1	Running	0	32m
catalog-755788cb85-lp5nb	1/1	Running	0	5m52s
catalog-mongo-7b7b7b5864-98qvl	1/1	Running	0	5m52s
order-847d8f8749-pmk5g	1/1	Running	0	114s
order-postgres-5ff8f9cf8d-b659m	1/1	Running	0	114s
payment-5cf85c59d-5m7dt	1/1	Running	0	3m44s
users-6b4cb56f9b-bnmlb	0/1	ContainerCreating	0	27s
users-mongo-69655c598f-b2jvr	0/1	ContainerCreating	0	27s
users-redis-6bc5ff47b4-xs2qd	1/1	Running	0	27s

FIGURE 4.38 : Service utilisateurs activé

#### 4.12.2.5 Service Frontal

Nous avons tapé la commande suivante dans Putty : `cd /labs/acme-fitness/frontend-service`  
Nous avons la commande suivante dans Putty pour déployer le service frontal. Ce service n'a pas de base de données associée :

```
kubectl apply -f frontend-service.yaml
```

```
service/frontend created
deployment.apps/frontend created
```

FIGURE 4.39 : Déploiement de service frontal

On tape la commande suivante dans Putty pour vérifier que les pods du service frontal sont opérationnels. Vous devrez peut-être attendre quelques minutes jusqu'à ce que tous les pods soient affichés comme étant en cours d'exécution : `kubectl get pod -n acme-fitness`

#### 4.12.2.6 Récupération du point de terminaison d'application

Maintenant que notre application est entièrement déployée, nous devons récupérer le point final de l'application. On tape la commande suivante dans Putty pour récupérer le point de terminaison de l'application. Vous verrez le point de terminaison de l'application listé sous frontend : `kubectl get service -n acme-fitness`

TESTER LE SERVICE CATALOGUE : Nous pouvons tester le service de catalogue en cliquant sur un produit et voilà l'interface de notre application :

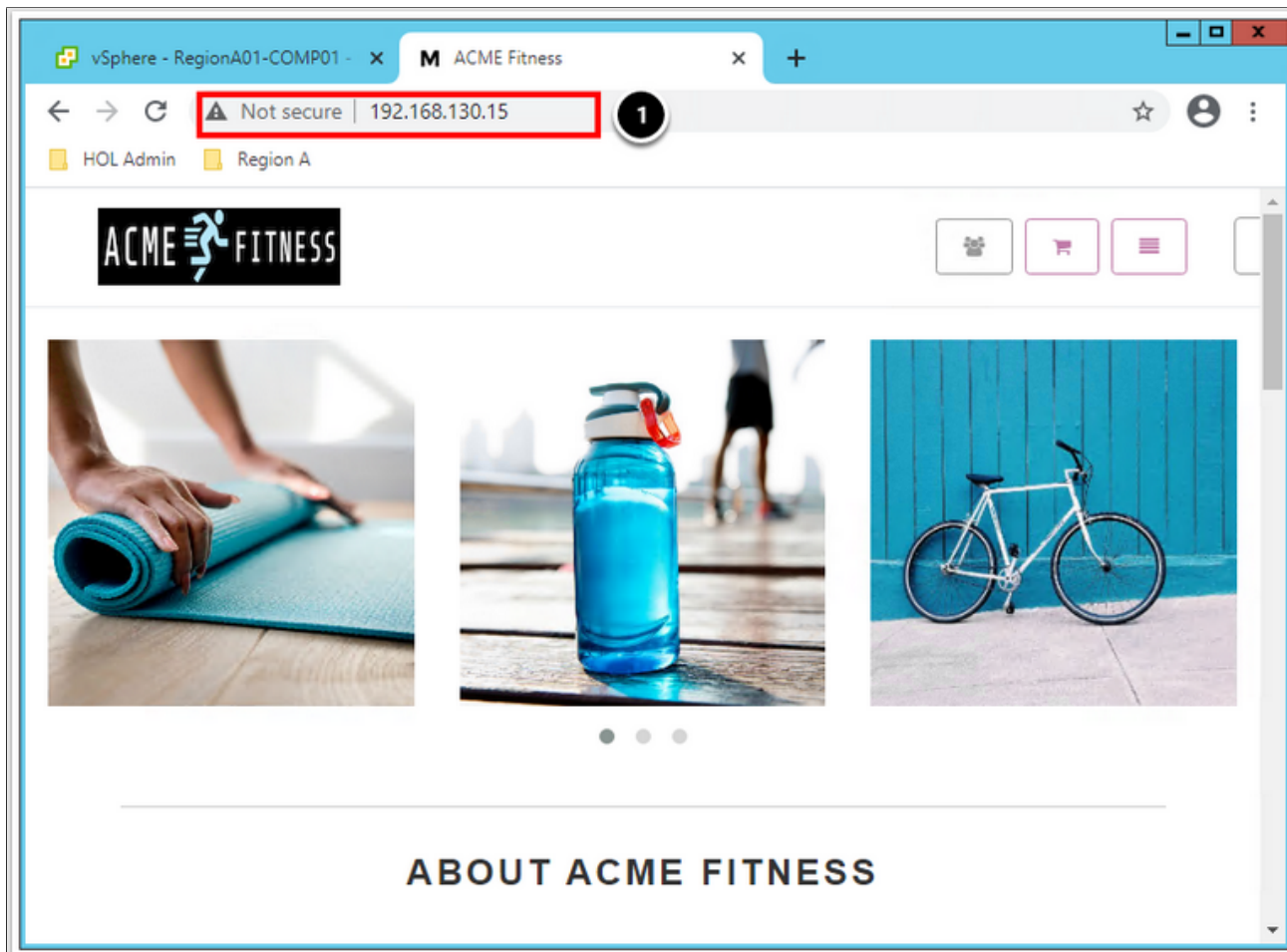


FIGURE 4.40 : acme-fitness

### 4.13 Conclusion

Dans ce chapitre nous avons décrit les environnements matériels et logiciels du projet, nous avons installé deux Esxi et Vcenter .Ensuite nous avons déployer le TKG en activant le cluster superviseur .Puis nous avons introduit une nouvelle construction vSphere appelée espace de nom, qui permet aux administrateurs vSphere de créer un ensemble logique de ressources, de permissions et de politiques qui permettent une approche centrée sur les applications. Enfin , nous avons déployé une application qui s'appelle acme-fitness sur TKG pour verifier le test.

# Conclusion générale

Notre travail a été réalisé dans le cadre d'un projet de fin d'étude lors d'un stage de trois mois au sein de 3S. Ce stage est l'occasion de se confronter au monde du travail et d'appréhender de près de la vie professionnelle. Les travaux de recherche menés dans ce travail portent principalement sur le service Tanzu accompagnant la plateforme d'orchestration Kubernetes. Nous avons ensuite recherché la solution vSphere Tanzu Kubernetes en précisant les technologies et les choix de virtualisation utilisés. Cela nous permet d'enrichir notre connaissance des technologies de virtualisation et des différentes solutions disponibles, notamment les concepts de conteneurisation, de micro services, de Vsphere et de Tanzu. L'obtention du résultat final de ce projet émerge sur plus qu'une étape : Dans La première étape, Nous avons eu l'occasion d'apprendre plus sur le domaine de la virtualisation, de la conteneurisation et vmware vSphere, nous tenons à mettre en exergue l'importance de cette solution. Dans La deuxième étape nous avons essayé d'installer un hyperviseur de type 1 vmware ESXI avec toutes les dépendances de notre solution, Cependant, nous avons été confrontés à des difficultés inattendue même pour les ingénieurs de l'entreprise et ceci nous oblige pour tenter de trouver les réponses à ces problèmes liés aux infrastructures physiques de la société, au problème au niveau de versions des logiciels installer et essentiellement au service du travail puisqu'il est une nouveau-né technologie dans le monde. Ceci a gêné l'aboutissement l'avancement du stage sans bloquer les résultats prévus. La dernière étape nous avons tester nos services configurés. Nous pouvons conclure que ce stage nous a exposé à de nouveaux concepts moderne utilisés dans le monde. D'autre part ce travail nous a permis d'approfondir nos connaissances. Finalement, ce projet nous a été bénéfique puisqu'il nous a permis d'appliquer les connaissances acquises au cours de nos formations et d'être privilégié de connaître un nouvel environnement. Cette solution peut être enrichie, grâce à son caractère évolutif et sa modularité. De nouvelles fonctionnalités peuvent être ajoutées car cette technologie innovante est constamment étudiée par vmware .

# Bibliographie

- [1] 3S. « 3s. » (), adresse : <http://https://www.3s.com.tn/en/>.
- [2] —, (), adresse : <https://docs.vmware.com/fr/VMware-vSphere/index.html>.
- [3] DEVOPS. (), adresse : <https://fr.wikipedia.org/>.
- [4] COMPARAISON. [] (), adresse : <https://www.lemondeinformatique.fr/actualites/lire-quel-avenir-pour-la-virtualisation-de-serveurs%C2%A0-72313.html>.
- [5] N. AUTEUR. « titre de l'article, » Organisme. ().
- [6] STATISTIQUE. [] (), adresse : <https://www.programmez.com/actualites/hyper-v-une-reelle-concurrence-pour-vmware-3988>.
- [7] N. AUTEUR. « titre de l'article. » [], Organisme. (), adresse : [https://pathfinder.vmware.com/v3/activity/containers\\_k8s\\_101\\_hol/](https://pathfinder.vmware.com/v3/activity/containers_k8s_101_hol/).
- [8] —, « titre de l'article, » Organisme. ().
- [9] VSPHERE. [] (), adresse : [https://pathfinder.vmware.com/v3/activity/containers\\_k8s\\_101\\_hol](https://pathfinder.vmware.com/v3/activity/containers_k8s_101_hol).

## Résumé

Dans ce travail, nous avons étudié un ensemble d'outils et de technologies permettant l'utilisation de la solution de virtualisation "Vmware Tanzu". L'objectif est de mettre en place Vmware Tanzu dans les services de la société 3S, afin de fournir des environnements de travail modernisés, sécurisés et rapides aux utilisateurs. Tout d'abord, nous avons installé deux hyperviseurs de type 1 "Vmware ESXI" et un serveur Vcenter qui gère ces deux hyperviseurs. Ensuite, nous avons déployé TKG (Tanzu Kubernetes Grid). Enfin nous avons testé notre travail avec une application "Fitness" au sein d'un cluster de charge de travail et d'un cluster de gestion.

**Mots clés :** ESXI, Vcenter, TANZU ,TKG, HA ,DRS, HAproxy

## Abstract

In this work, we studied a set of tools and technologies that allow the use of the virtualization solution "Vmware Tanzu". The objective is to implement Vmware Tanzu in the services of the company 3S, in order to provide modernized, secure and fast working environments to users. First, we installed two type 1 hypervisors "Vmware ESXI" and a Vcenter server that manages these two hypervisors. Next, we deployed TKG (Tanzu Kubernetes Grid). Finally we tested our work with a "Fitness" application within a workload cluster and a management cluster.

**Keywords :** ESXI, Vcenter, TANZU ,TKG, HA ,DRS, HAproxy