RÉPUBLIQUE DU SÉNÉGAL

Un Peuple – Un But – Une Foi



MINISTERE DE L'ENSEIGNEMENT SUPERIEUR, DE LA RECHERCHE ET DE L'INNOVATION



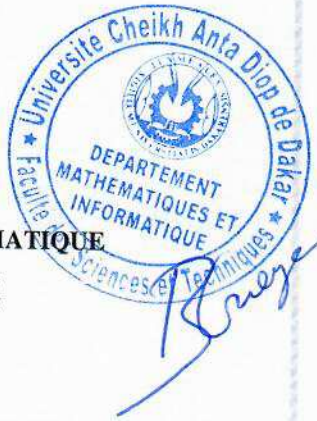
UNIVERSITE CHEIKH ANTA DIOP DE DAKAR

****

FACULTE DES SCIENCES ET TECHNIQUES

****

DEPARTEMENT DE MATHEMATIQUES ET INFORMATIQUE SECTION INFORMATIQUE



MÉMOIRE DE MASTER INFORMATIQUE

Spécialité : Réseaux et Télécommunications

Option : Réseaux, Systèmes et Services

Sujet : Étude et Mise en Place d'un Cloud Privé à L'ADIE

Présenté et Soutenu publiquement par :

M^me^ MAME FATOU NIANG

Encadré par Dr. Chérif DIALLO

Le : 29 / 05 / 2021

Jury :

Président :	Pr. Cheikh Ahmadou Bamba GUEYE	Professeur Titulaire, UCAD
Examinateur :	Dr. Abdourahmane RAIMY	Maître de Conférences, UCAD
Examinateur :	Dr. Mohamed Ould DEYE	Maître Assistant, UCAD
Examinateur :	Pr. Ibrahima NIANG	Professeur Titulaire, UCAD

Année universitaire 2019/2020

Remerciements

Après avoir rendu grâce à Dieu, le Tout Puissant et le Miséricordieux, je tiens à remercier vivement tous ceux qui m'ont apporté leur aide et qui ont contribué à l'élaboration de ce travail ainsi qu'à la réussite de ma formation universitaire. Qu'ils trouvent ici l'expression de ma profonde gratitude.

D'abord, je remercie Pr Cherif DIALLO, mon encadrant académique pour sa disponibilité, ses directives et sa rigueur qui m'ont permis de soigner et d'améliorer constamment la qualité de ce travail. Mes remerciements s'adressent également à Monsieur Alioune B. NIMAGA, mon encadrant professionnel, qui a toujours trouvé le temps de faire le suivi de mon travail.

Ensuite, Je remercie particulièrement M. Alassane NDOYE, Chef de Division Plateformes et Services (DPS) pour le temps qu'il m'a consacré et pour les précieuses informations qu'il m'a prodiguées avec intérêt et compréhension tout au long de la période du projet ainsi que M. Patrick E. BADJI Ingénieur Systèmes à la DPS.

Je remercie également M. Cheikh Sadibou THIAW chef de Division Etude et Architecture pour sa disponibilité et son sens d'écoute et d'échange et pour les conseils utiles qu'il n'a cessé de me communiquer.

J'exprime ma gratitude à tous les responsables et agents au sein de l'ADIE notamment M. Cheikh BAKHOUM le DG et M. Moustapha KANE le DRST qui m'ont donné l'opportunité de réaliser ce stage.

Je remercie chaleureusement Dr Fatoumata BALDE KASSE et Pr Bamba GUEYE ainsi que la famille FALL (oncle Djadji et tante Ngoné) depuis Thiès.

C'est avec une immense fierté que j'adresse mes remerciements les plus distingués à tous mes enseignants de la section informatique qui nous ont transmis leur savoir et nous ont assuré la meilleure des formations.

J'adresse aussi mes vifs remerciements aux membres du jury pour avoir bien voulu examiner et évaluer ce modeste travail sans oublier mon ami et partenaire de tous les temps M. Cheikh Madické NDIAYE et sa famille ainsi Thiara NDIAYE BA.

Dédicace

A mes Parents,

A mes frères et sœurs

A mes oncles et tantes

A toute ma famille,

A mes très chers amis, je ne pourrais tous les citer,

A mes enseignants tout au long de mon cursus scolaire et universitaire,

A toute personne qui a eu l'énorme générosité de m'accorder son temps,

Partager un rire, son amour, un souvenir impérissable,

Communiquer le gout et la force de vivre pleinement,

Table des matières

Remerciements 2
Dédicace 3
Table des matières 4
Table des Figures 7
Liste des Tableaux ..... **Erreur ! Signet non défini.**
Glossaire 10
Introduction Générale 12
1. Chapitre I : Cadre du projet 14
1.1 Introduction 14
1.2 Présentation de l'organisme d'accueil 14
1.2.1 Misions 14
1.2.2 Organisation 15
1.3 Présentation du projet 17
1.3.1 Contexte 17
1.3.2 Problématique 17
1.3.3 Objectifs 18
1.4 Etude de l'existant 19
1.4.1 Infrastructure Réseaux 19
1.4.2 Matériel et logiciels 20
1.4.3 Sécurité 20
1.5 Conclusion 20
2. Chapitre II: Généralités sur le Cloud 21
2.1 Introduction 21
2.2 Historique 21
2.3 Définition du Cloud 22
2.4 Caractéristiques du cloud computing 23
2.5 Modèles de déploiement 24
2.5.1 Cloud Privé (Private cloud) 25
2.5.2 Cloud Public (Public cloud) 26
2.5.3 Cloud Communautaire 28
2.5.4 Cloud Hybride 29
2.6 Types de services 31

2.6.1 Infrastructure as a Service (IaaS) 32
2.6.2 Platform as a Service (PaaS) 36
2.6.3 Software as a Service (SaaS) 40
2.7 Exigences du cloud ..... **Erreur ! Signet non défini.**
2.8 Cloud Écosystème 44
2.9 Les technologies pilotes du Cloud 44
2.9.1 SOA et Web services 45
2.9.2 Les technologies de stockage et de mémoire 47
2.9.3 Les technologies Réseau 47
2.9.4 La virtualisation 48
2.9.5 Types de virtualisation 53
2.10 Migration des machines virtuelles VMs 57
2.10.1 Techniques de migration 57
2.11 Les avantages et inconvénients du Cloud 62
2.11.1 Les avantages du Cloud 62
2.11.2 Les Inconvénients du Cloud 63
2.12 Conclusion 63
3. Chapitre III : Choix Technologique 65
3.1 Introduction 65
3.2 Solutions de cloud privés libres 65
3.2.1 OpenStack 65
3.2.2 OpenNebula 66
3.2.3 Apache CloudStack 67
3.2.4 Eucalyptus 69
3.3 Etude Comparative 70
3.4 Solution Choisie 73
3.4.1 Les composants d'OpenStack 74
4. Chapitre IV : Conception de la solution 80
4.1 Introduction 80
4.2 Partie Fonctionnelle 80
4.2.1 Identification de besoins 80
4.2.2 Identification des acteurs 82
4.2.3 Raffinements des Cas d'Utilisations 84
4.2.4 Diagrammes d'activité 90

4.3 Partie Technique 92
4.3.1 Gestion de l'infrastructure 92
4.3.2 Gestion des applications 100
4.4 Dimensionnement 103
4.4.1 Spécification matérielle 103
4.4.2 Quantités de ressources 104
5. Chapitre V : Mise en œuvre de la solution 107
Conclusion 114
Bibliographie 116
Webographie ..... **Erreur ! Signet non défini.**

Table des Figures

Figure 1:1: Infrastructure Réseaux de l'ADIE ..... **Erreur ! Signet non défini.**
Figure 2:1: Historiques et générations de l'informatique [1] 21
Figure 2:2: Aperçu du Cloud [3] 23
Figure 2:3: Modèles de déploiement du cloud computing [4] 25
Figure 2:4: Types de services du Cloud [6] 31
Figure 2:5:Responsabilités des utilisateurs et des fournisseurs pour chaque type de service : (a) IaaS, (b) PaaS, (c) SaaS [7] 32
Figure 2:6: Ecosystème du Cloud [8] 44
Figure 2:7: Technologies pilotes du cloud computing [9] 45
Figure 2:8: Participants et actions dans SOA [10] 46
Figure 2:9: Avant la virtualisation [11] Figure 2:10: Après la virtualisation [11] 48
Figure 2:11: Hyperviseur type 1 [12] Figure 2:12: Hyperviseur type 2 [12] 50
Figure 2:13: Virtualisation des processeurs [13] 51
Figure 2:14:Virtualisation de la mémoire [14] 52
Figure 2:15: Virtualisation du stockage [15] 52
Figure 2:16: Virtualisation du réseau [16] 53
Figure 2:17: Processus live migration pré-copie, source [17] 59
Figure 2:18: Processus de migration post-copie [18] 60
Figure 2:19: Processus de post-copie hybride, [19] 61
Figure 4:1: Architecture Openstack, source [22] 66
Figure 4:2: Architecture OpenNebula [23] 67
Figure 4:3:Composition architecturale de CloudStack, source [24] 68
Figure 4:4: Architecture Eucalyptus, source [25] 70
Figure 4:6: Architecture Conceptuelle d'OpenStack, source [26] 74
Figure 4:7: Architecture Fonctionnelle Openstack [27] 79
Figure 5:1: Diagramme de cas d'utilisation générale du Cloud 83
Figure 5:2: Cas d'utilisation « Consulter l'état du cloud» 84
Figure 5:3: Cas d'utilisation « Gérer la liste des instances » 85
Figure 5:4: Cas d'utilisation « Consulter la liste des services » 85
Figure 5:5: Cas d'utilisation « consulter la listes des flavors disponibles » 86
Figure 5:6: Cas d'utilisation « Gérer la liste des images du nuage » 87
Figure 5:7: Cas d'utilisation «Créer un Projet » 88
Figure 5:8: Cas d'utilisation « Gérer la liste des utilisateurs » 88
Figure 5:9: Diagramme des cas d'utilisation pour un membre d'un projet 89
Figure 5:10: Diagramme d'activité globale 91
Figure 5:11: Diagramme d'activité « créer une instance » 92
Figure 5:12: Types de réseaux de notre solution 96
Figure 5:14 : Première Architecture proposée pour notre solution 105
Figure 5:15: Deuxième architecture Proposée pour notre solution 106
Figure 5:1: Architecture de notre prototype 108
Figure 5:2: Interface d'accueil 108
Figure 5:3: Vue d'ensemble de notre cloud 109
Figure 5:4: Création Gabarit 1/2 110

Figure 6:5: Création Gabarit 2/2 110
Figure 5:6: liste des images d'OS 111
Figure 5:7: Topologie Réseau du cloud 111
Figure 5:8: Création de paire de clés 1/2 112
Figure 5:9: Création de paire de clés 2/2 112

Liste des Tableaux

Tableau 3:1: Comparaison des différentes solutions de Cloud Privé 71
Tableau 4:1: Comparaison entre GlusterFS et Ceph 98
Tableau 4:2: Les différents types de stockage dans Openstack 101
Tableau 5:1: Caractéristiques des machines de notre prototype 107

Glossaire

API : Application Program Interface

BGP : Border Gateway Protocol

CRM : Customer Relationship Management

CSV : Comma-Separated Values

DHCP : Dynamic Host Configuration Protocol

EC2 : Elastic Compute Cloud

ERP : Enterprise Resource Planning

FC : Fibre Channel

FOSS : Free and open-source software

GRE : Generic Routing Encapsulation

IDE : Integrated Development Environment

IP ECMP : IP Equal-Cost MultiPath

KVM : Kernel-Based Virtual Machine

LDAP : Lightweight Directory Access Protocol

LXD : Linux Container Daemon

ML2 : Modular Layer 2

MPLS : MultiProtocol Label Switching

NC : Node Controller

NIST : National Institute of Standards and Technology

QoS : Quality of Service

RDMA : Remote Direct Memory Access

S3 : Simple Storage Service d'Amazon

SDN : Software Defined Network

SGBD : Service de Gestion de Bases de Données

SOAP : Simple Object Access Protocol

UML : Unified Modeling Language

URL : Uniform Resource Locator

VLAN : Virtual Local Area Network

VMM : Virtual Machine Manager

VPN : Virtual Private Network

API Rest : Une API REST (également appelée API RESTful) est une interface de programmation d'application (API ou API web) qui respecte les contraintes du style d'architecture REST (Representational State Transfer) et permet d'interagir avec les services web RESTful.

F5 : est une entreprise informatique américaine fondée en 1996 établie à Seattle qui commercialise des équipements réseau.

Gestion des adresses IP (IPAM : IP Address Management) : Le processus d'automatisation de l'attribution des adresses IP, de la répartition des offres et de la gestion.

Libvirt : libvirt est une bibliothèque, une API, un daemon et des outils en logiciel libre de gestion de la virtualisation. Elle est notamment utilisée par KVM, Xen, VMware ESX, QEMU et d'autres solutions de virtualisation.

MainFrame : Le mainframe représente ainsi un ordinateur central de grande puissance chargé de gérer les sessions utilisateurs des différents terminaux qui lui étaient reliés.

Open vSwitch : est un commutateur virtuel multicouche de qualité de production sous licence Open source Apache 2.0. Il est conçu pour permettre l'automatisation massive du réseau grâce à l'extension programmatique, tout en soutenant les interfaces et protocoles de gestion standard (par exemple NetFlow, sFlow, SPAN, RSPAN, CLI, LACP, 802.1ag).

Pont Linux (Linux Bridge) : Logiciel qui permet à plusieurs VMs de partager une seule carte réseau physique (NIC) au sein de Compute.

RAID : Le RAID est un ensemble de techniques de virtualisation du stockage permettant de répartir des données sur plusieurs disques durs afin d'améliorer soit les performances, soit la sécurité ou la tolérance aux pannes de l'ensemble du ou des systèmes.

Introduction Générale

L'Agence De l'Informatique de l'Etat (ADIE) a pour mission principale d'accompagner la transition numérique de l'administration sénégalaise. Elle offre une panoplie de services à savoir de la messagerie, de l'hébergement web et du développement de sites et d'applications. Cependant, l'agence est confrontée à des difficultés dans sa fourniture de services car leur déploiement prend beaucoup de temps et la procédure à suivre par le demandeur est très longue.

Avec l'accroissement de la demande et l'expansion de la dématérialisation des procédures administratives, la durée de traitement des requêtes est plus longue aboutissant à une faible productivité et à une dégradation de la qualité de service. Sur ce, une problématique s'impose à l'évidence devant nous ; comment optimiser le déploiement des services et leur provisionnement aux clients ?

Par ailleurs, l'émergence d'un modèle informatique « à la demande » a bousculé les anciens schémas des technologies de l'information, le Cloud Computing. Ce modèle permet de provisionner des environnements évolutifs de développement et de déploiement, à la demande. Son efficacité au sein des entreprises n'est plus sujet à discussion vue qu'il a fait ses preuves durant ces dernières années, en réduisant les délais de fourniture de services d'où l'enthousiasme de l'ADIE à vouloir mettre en place un Cloud gouvernemental. Notre stage de fin de cycle de master s'inscrit dans ce contexte. Notre projet consiste à faire l'étude et la mise en place d'une solution open source de Cloud Computing de type IaaS. Le document est organisé en cinq chapitres.

Le premier chapitre sera consacré au cadre général du projet en présentant la structure d'accueil, puis le contexte et l'intérêt du projet en plus des objectifs fixés ainsi que l'étude de l'existant pour évaluer les ressources que dispose l'agence.

Quant'au deuxième chapitre, nous ferons l'état de l'art sur le Cloud Computing en décrivant ses caractéristiques, ses différents modèles de déploiement et de services et décrirons les technologies pilotes et les acteurs du Cloud Computing sans oublier leurs avantages et inconvénients.

Le troisième chapitre délimitera le choix technologique de la solution à mettre en place à la suite d'une étude comparative raffinée des différentes solutions libres les plus connues notamment Openstack, Eucalyptus, OpenNebula et CloudStack.

Nous entamerons ce quatrième chapitre dédié à la conception par la partie fonctionnelle en procédant à l' identification des besoins, à la présentation des utilisateurs du système et leurs fonctionnalités à travers des diagrammes de cas d'utilisation. Nous terminons ce chapitre par la partie technique. A ce niveau, nous parlerons de la gestion de l'infrastructure et des apllications mais également du dimensionnement qui sera faite en spécifiant les serveurs et la quantité de ressources.

Le cinquième et dernier chapitre détaillera les étapes de la réalisation et exposera l'environnement de travail ainsi que le déploiement de la solution proposée.

1. Chapitre I : Cadre du projet

1.1 Introduction

Dans ce chapitre, nous présentons le cadre général du projet. Nous introduisons l'entreprise d'accueil. Ensuite, nous mettons en évidence l'intérêt de notre projet, ce qui nous amènera à identifier sa problématique, ses objectifs et l'étude de l'existant.

1.2 Présentation de l'organisme d'accueil

L'Agence De l'Informatique de l'Etat (ADIE) est une structure autonome chargée de mettre en œuvre la politique d'informatisation de l'Etat du Sénégal. Elle a pour mission principale de doter l'Administration d'un dispositif cohérent de traitement et de diffusion de l'information, répondant aux normes internationales en matière de qualité, de sécurité, de performance et de disponibilité. A travers elle, le Gouvernement du Sénégal a adopté une démarche visant à améliorer le service public en s'appuyant sur les Technologies de l'Information et de la Communication (TIC). Dans cette optique, l'ADIE développe un ensemble de produits et services pour contribuer à la bonne gouvernance par la promotion d'une Administration numérique efficace et efficiente au service du citoyen.

1.2.1 Misions

De la Section Informatique du Bureau Organisation et Méthodes (BOM) à l'Agence De l'Informatique de l'Etat depuis 2004, l'ADIE s'est toujours tournée vers l'avenir en mettant tout en œuvre pour que l'Administration s'adapte aux différentes mutations technologiques. Ses missions dévolues sont :

- La modernisation de l'Administration Sénégalaise par la dématérialisation des procédures administratives ;
- La rationalisation des dépenses informatiques de l'Etat en mutualisant et en harmonisant les choix technologiques des services de l'Administration ;
- L'édification d'une infrastructure nationale de réseaux pour l'interconnexion des structures de l'Etat ;
- La mise à disposition d'un système d'information fiable pour un suivi efficace de l'action gouvernementale ;

- La coordination de la mise en place d'un cadre législatif et réglementaire propice au développement des technologies de l'information et de la communication.
- La réduction de la fracture numérique et l'exclusion sociale par la généralisation de l'accès aux tic.

1.2.2 Organisation

L'Agence De l'Informatique de l'Etat est constitué de la Direction Générale et ses services rattachés et des directions opérationnelles qui ont la charge de mettre en œuvre les projets et activités.

- **La Direction générale** : qui assure la coordination et fixé les objectifs. Elle comprend : le Directeur Général (DG), le Secrétaire Général (SG), les chargés des programmes et le coordinateur du bureau des projets (PMO) et s'appuie sur :
  - **L'Agence Comptable** : qui a pour mission de mobiliser les ressources de l'ADIE, de payer les dépenses, de conserver les valeurs et les fonds, de gérer la trésorerie, de tenir la comptabilité générale et d'établir les états financiers de synthèse.
  - **L'Audit et Contrôle de Gestion** : qui assure la performance de l'entreprise dans le respect des obligations légales et des procédures internes et participe au pilotage et à la prise de décision. Elle est chargée d'harmoniser les procédures comptables et financières et de superviser la clôture des comptes.
  - **La Cellule de passation des marchés** : est chargée de veiller à la qualité des dossiers de passation des marchés ainsi qu'au bon fonctionnement de la commission des marchés de l'ADIE.
  - **La Commission des marchés** : est chargée de l'exécution et du suivi des activités d'ouverture des plis, d'évaluation des offres et d'attribution provisoire des marchés de l'ADIE conformément aux dispositions du code des marchés publics.

Les directions opérationnelles sont au nombre de sept (07) sont :

- **La Direction des Réseaux, des Systèmes et des Télécommunications(DRST)** qui est la direction dans laquelle que nous avons

effectué notre stage. Elle est chargée de l'administration, de la maintenance des infrastructures systèmes, réseau et télécoms. Elle est également chargée de la gestion du parc informatique de l'Etat, du support aux utilisateurs, de la mise en production et de l'exploitation des systèmes d'information.

- **La Direction des Services de l'Ingénierie (DSI)** qui est chargée de la maitrise d'ouvrage, de la maitrise d'œuvre des projets et de la veille technologique. Elle assure l'accompagnement des structures administratives dans la mise en œuvre de leurs projets TIC et s'occupe du développement, de l'intégration et de la commercialisation des applications/produits/services.
- **La Direction des Relations Extérieures, du Marketing, de la Formation et de la Communication (DRMFC) :** qui a pour mission de promouvoir les produits, les services, les réalisations et l'image de l'ADIE. Elle est chargée de renforcer les relations de l'Agence avec les structures administratives. Elle contribue également à la mise en œuvre de la politique de renforcement des capacités des agents de l'Etat. Elle est aussi en charge de définir et d'adopter une démarche pour la rétribution de services.
- **La Direction Administrative et Financier (DAF)** : qui prend en charge la gestion comptable et financière, la gestion du personnel, l'intendance qui comprend notamment la gestion des locaux, des approvisionnements et des véhicules et la conduite des procédures administratives et juridiques.
- **La Direction de la Logistique et de la Maintenance (DLM)** : qui est chargée de concevoir, d'organiser et de définir les meilleures stratégies de gestion des processus, du suivi des acquisitions et de leurs entreposages, de leur distribution. Elle est chargée de la maintenance des équipements informatiques, de télécommunications et de production d'énergie (transformateurs, groupes électrogènes, onduleurs, régulateurs et climatiseurs…), ainsi que des matériels et outils de maintenance du réseau de l'intranet.
- **La Direction de la Solidarité Numérique (DSN)** : qui a pour mission générale de lutter contre la fracture numérique et l'exclusion sociale grâce à une intégration des TIC à l'école, à travers le recyclage utilitaire et durable des

équipements informatiques et la formation des enseignants et des couches vulnérables.

- **La Direction de la sécurité des Systèmes d'Information (DSSI)** : Créée au début de l'année 2015, cette création récente est la direction en charge de l'élaboration, de la mise en œuvre et du suivi de la politique de sécurité des systèmes d'informations de l'état. La sécurité des systèmes d'information repose d'abord sur des éléments organisationnels et fonctionnels, les éléments techniques viennent en compléments.

1.3 Présentation du projet

1.3.1 Contexte

Si en 2010, l'offre Cloud commençait à faire partie du paysage des solutions et des services liés aux systèmes d'information, le marché était en cours de structuration et l'offre ne répondait que partiellement aux besoins fonctionnels des DSI. « Faire différence, plus vite, moins cher, en optimisant les ressources », tel était alors le message des dirigeants. Ce qui impliquait pour l'entreprise performance, innovation, agilité et rapidité dans la conception, réalisation et distribution de nouveaux produits et services. Depuis lors, le Cloud Computing est progressivement devenu une opportunité incontournable et un vecteur de la transformation digitale. A cet effet, L'Agence de l'informatique de l'Etat exprime le besoin et a la volonté de disposer d'un cloud privé afin de résoudre les problèmes auxquels elle est confrontée dans sa fourniture de services.

1.3.2 Problématique

L'ADIE offre une multitude de services aux structures étatiques parmi lesquels l'hébergement web et le développement de sites et d'applications. Ces services prennent beaucoup de temps avant d'être livré au client. Cette lenteur est dû à la création manuelle des environnements sur lesquels se reposent les services fournis mais également à la longueur de la procédure à suivre par le demandeur. Celui-ci fait d'abord une expression de besoins à l'agence avec les spécificités de ressources voulues en calcul (CPU, RAM) et en stockage. Si la requête est favorable, une réunion de cadrage sera effectuée pour délimiter les ressources à allouer. Suite à cela, une template sera utilisée oubien une machine virtuelle sera créée suivi de l'installation des

prérequis et dépendances en fonction du besoin du client. Une fois l'environnement fonctionnel, les paramètres d'accès seront communiqués au client et des tests de vulnérabilités seront effectués avant publication du service. A cela s'ajoute, la dispersion des ressources. En effet, l'agence dispose de différentes plateformes de virtualisations avec des outils d'administration différents d'où la difficulté de monitoring et un manque de visibilité exacte sur les ressources consommées. Ces plateformes ont des limites par rapport aux nouveaux besoins de l'agence car elles ne prennent pas en charges les dernières versions de systèmes d'exploitation et ne peuvent pas être mis à jour sans interruption de service. Ces plateformes n'offrent pas des environnements élastiques car ne pouvant pas s'adapter aux variations de la demande (à la hausse ou à la baisse). Le cloisonnement d'environnements n'existe au sein de l'ADIE, que ça soit le développement, les tests comme la production tourne sur un seul environnement.

1.3.3 Objectifs

Le projet consiste à mettre en place un cloud privé à l'ADIE dans le cadre de notre mémoire de fin de cycle afin de faciliter le déploiement et le provisionnement des environnements. Son objectif consiste à :

- Automatiser la création et le déploiement des environnements pour réduire le temps d'exécution des services
- Disposer d'un portail self-service pour mieux répondre aux demandes et besoins des clients en provisionner rapidement des applications
- Mutualiser les ressources pour une gestion optimale et une supervision en temps réels des ressources consommées.
- Déployer des applications sans adhérence aux mises à jour de la plateforme.

Pour la mise en place de notre cloud privé, voici quelques recommandations permettant d'obtenir de bonnes performances : Serveurs Rackables, Double alimentation électrique, Système de ventilation redondant. 2 Sockets (processeurs), 6 cœurs de CPU par socket minimum 128 Go de RAM 2 Cartes réseaux 10 Gbps supportant le protocole RDMA 2 disques durs en RAID 1 pour supporter le système d'exploitation et l'hyperviseur.

1.4 Etude de l'existant

L'objectif de cette étude est d'évaluer les moyens techniques et matériels que détiennent l'agence.

1.4.1 Infrastructure Réseaux

L'ADIE a depuis longtemps adopté une architecture centralisée. Elle comporte deux datacenters fonctionnels. Le primaire se situe dans la banlieue dakaroise (Technopole) et le secondaire en centre-ville (Oranna). Le troisième datacenter n'est pas encore inauguré. L'agence dispose d'un intranet gouvernemental. En effet, elle dispose d'un réseau de données à l'échelle national comportant 4500 Km de fibre Optique, connecté à 1337 bâtiments administratifs réparti entre ses deux datacenters. Le Système de câblage F.O de l'agence supporte 10 Gb/s et admet un accès réseau redondants avec 2 liens de 10 Gb/s et une connectivité avec le Backbone MPLS.

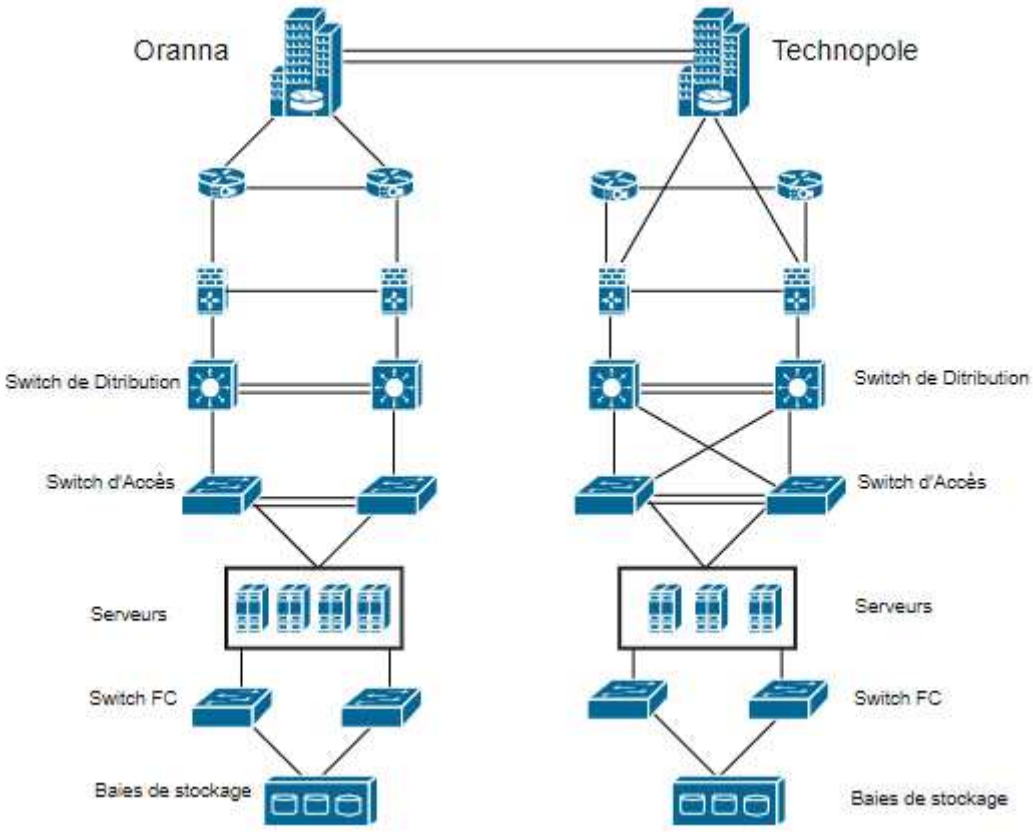


Figure 1:1: Infrastructure Réseaux de l'ADIE

1.4.2 Matériel et logiciels

L'ADIE a à sa disposition un parc de ressources dont des équipements réseaux majoritairement Huawei (des routeurs, des switchs FC et Ethernet, des firewalls, des serveurs de marques hétérogènes en rack et des baies de stockages.

- ❖ Equipements Réseaux :
  - Switchs d'accès
  - Routeurs
  - Switch Ethernet
- ❖ Systèmes d'exploitation :
  - Linux (Ubuntu, Debian, Kali)
  - Windows (Windows server 2012R2,2016)
- ❖ Outils de virtualisation
  - Proxmox
  - Fusion Sphère
  - VMware
- ❖ Stockage
  - 4 switchs FC
  - 4 SAN

1.4.3 Sécurité

Pour sécuriser son infrastructure, l'agence dispose de deux firewalls F5, deux DNS interne et externe, d'une solution VPN, d'un antivirus, d'un outil de gestion des certificats pour les sites web, une disponibilité de 2 sources d'énergie, des groupes électrogènes hautement sécurisées, une redondance électrique par baie, un système de climatisation et une salle de supervision 24/7.

1.5 Conclusion

Dans ce chapitre, nous avons parlé du cadre de notre projet en présentant d'abord la structure d'accueil avec ses missions et son organisation, puis le projet en tant que tel c'est-à-dire son contexte, la problématique et les objectifs visés et enfin l'étude de l'existant pour évaluer si les moyens que dispose que l'agence nous permettent de

mettre en place un cloud privé. Effectivement, l'existant permet de déployer un cloud privé. Nous allons passer maintenant à l'état de l'art sur le cloud computing.

2. Chapitre II: Généralités sur le Cloud

2.1 Introduction

Dans ce Chapitre, nous allons présenter les différents concepts du Cloud notamment son historique, ses différentes caractéristiques avant de venir aux différents modèles de déploiements et de services sans oublier leurs avantages et inconvénients mais également les technologies pilotes du cloud computing.

2.2 Historique

Les principes sous-jacents au *cloud computing* remontent aux années 1960 (longtemps avant que ne naisse l'expression « *cloud computing* »). À cette époque, les utilisateurs accédaient depuis leurs terminaux à des applications fonctionnant sur des systèmes centraux partagés (les *mainframes*), ancêtres des serveurs du *cloud*. La figure 2.1 montre l'évolution de l'information jusqu'à la naissance du cloud.

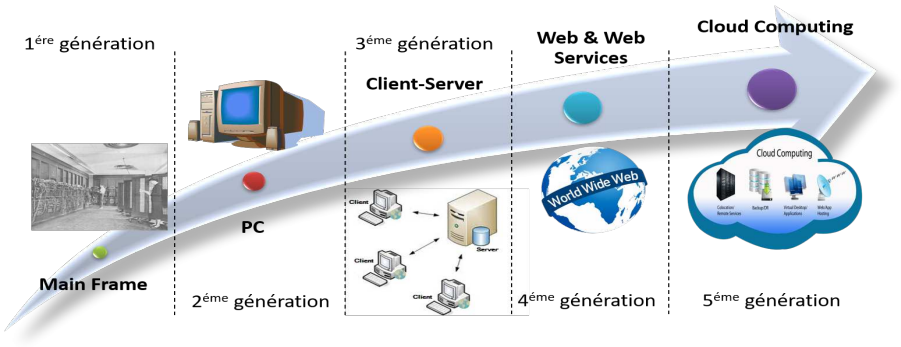


Figure 2:1: Historiques et générations de l'informatique [1]

— 1*re* génération (Mainframe) : Le terme se référait à l'origine aux grandes armoires qui abritaient l'unité centrale de traitement et la mémoire principale des premiers ordinateurs. Ces architectures de système ont été établies dans les années 1960, mais continuent d'évoluer. Les applications sont installées sur un ordinateur de grande puissance de traitement et qui sert d'unité centrale à un réseau de terminaux.

— 2*me* génération (PC) : Apparu en 1970s, l'utilisateur installe/accède ses applications directement sur son PC sans avoir besoin d'une tiers partie (réseau ou serveur).

— 3^{me} génération (Client-Serveur) : Dans ce type d'architecture, l'application est divisée en 2 entités : *(1)* le fournisseur de service appelé *Serveur* et le consommateur de service appelé *Client*. Les deux entités doivent être connectées par un réseau.

— 4^{me} génération (www) : Le web permet d'héberger des applications (dites applications web) sur des serveurs distants. Ces applications peuvent être accédées via le réseau internet en utilisant un navigateur.

— 5^{me} génération (cloud computing) : Apparu au début du 21^{me} siècle, des hébergeurs web sont capables d'héberger des applications dans leurs locaux informatiques. En particulier, Google et Amazon ont simultanément repris le terme 'cloud computing' en 2006. Dans ce contexte, l'ancêtre du SaaS correspondait au ASP (Application Service Provider). Les premières applications Web 2.0, qui ont été déployées en cloud computing, sont le courrier électronique, les outils collaboratifs, le CRM, les environnements de développement et de test (informatique).

2.3 Définition du Cloud

Le Cloud Computing, c'est avant tout la dématérialisation des ressources informatiques. Autrefois les entreprises n'avaient pas le choix. Lorsqu'elles avaient besoin d'utiliser un logiciel, elles achetaient une licence et l'installaient sur un ordinateur qui se situait dans leurs locaux, ou au mieux dans un Datacenter. Avec le Cloud désormais, cette même entreprise peut choisir, au lieu d'investir dans l'achat d'un logiciel, d'en louer l'usage auprès d'une société qui en assurera elle-même la maintenance, les mises à jour, et l'hébergement. Ceci a été rendu possible grâce à l'amélioration notable des débits internet d'une part, et l'évolution des navigateurs internet (Firefox, Chrome, IE, etc.) d'autre part.

Pour simplifier, le cloud computing est la fourniture de services informatiques (notamment des serveurs, du stockage, des bases de données, la gestion réseau, des logiciels, des outils d'analyse, l'intelligence artificielle) via Internet (le cloud) dans le but d'offrir une innovation plus rapide, des ressources flexibles et des économies d'échelle [2]. En règle générale, vous payez uniquement les services cloud que vous utilisez (réduisant ainsi vos coûts d'exploitation), gérez votre infrastructure plus efficacement et adaptez l'échelle des services en fonction des besoins de votre entreprise. Après avoir défini ce qu'est le cloud, Voyons donc ce qui caractérise un service cloud.

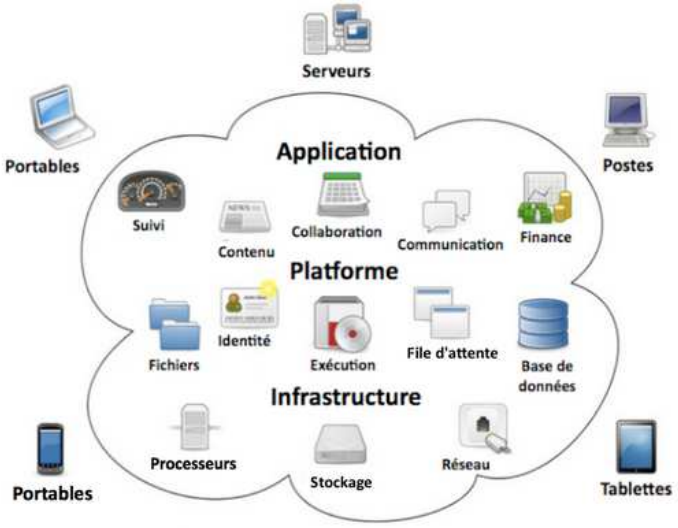


Figure 2:2: Aperçu du Cloud [3]

2.4 Caractéristiques du cloud computing

Bien que certains critères puissent être soumis à débat, il y a certaines notions indiscutables qui caractérisent un service en Cloud :

- **Service à la demande :**

Le service est fourni au client au moment de son besoin et de sa demande sans aucune interaction avec le fournisseur du service. Autrement dit, le service à la demande permet aux utilisateurs d'utiliser les ressources du Cloud selon leurs besoins, et un accès libre qui consiste au provisionnement des ressources sans interaction humaine entre l'utilisateur et le fournisseur de services.

- **Service mesurable :**

L'utilisation des ressources peut être surveillée, contrôlée et signalée, offrant une transparence pour le fournisseur et pour le consommateur du service utilisé. Le Cloud Computing permet au client de payer à l'usage, uniquement ce qui a été consommé selon le type de service (stockage, traitement, bande passante et comptes d'utilisateurs actifs, etc.) et le temps d'utilisation du service.

- **Mutualisation des ressources :**

Les mêmes ressources sont mises en commun et assignées aux différents clients de manière dynamique. Sur le Cloud une même application peut être utilisée par plusieurs

clients en même temps, en préservant la sécurité et les données privées de chaque client. Cela est possible en utilisant des outils de virtualisation qui permettent de partager un serveur entre plusieurs utilisateurs. Les différentes ressources physiques et virtuelles (par exemple, le stockage, le traitement, la mémoire et la bande passante du réseau) sont attribuées dynamiquement et réaffectées aux clients en fonction de la demande des consommateurs. Le client ne peut pas contrôler l'emplacement exact des ressources fournies. Néanmoins, il peut exiger l'emplacement souhaité pour ses ressources par exemple le pays où le continent, à travers l'intégration d'un niveau d'abstraction plus élevé.

- **Elasticité des ressources :**

Capacité d'allouer dynamiquement des ressources en fonction des besoins que ce soit à la hausse, à la baisse, de façon durable ou temporaire de manière très rapide

2.5 Modèles de déploiement

Le modèle de déploiement correspond aux moyens/méthodes utilisés pour mettre en place un environnement cloud. Ces modèles sont entièrement centrés sur l'utilisateur, c'est-à-dire qu'ils dépendent des exigences et de la commodité des utilisateurs. Autrement dit, il existe plusieurs utilisateurs du cloud et chaque utilisateur a ces propres besoins et un modèle de déploiement ne conviendra pas à tous ces utilisateurs. Selon le NIST, il existe 4 modèles de déploiements :

- Cloud privé (Private cloud) : est le modèle de déploiement le plus élémentaire pouvant être déployé par une seule organisation pour son usage personnel. Il n'est pas partagé avec d'autres organisations, et il n'est pas autorisé pour un usage public.
- Cloud public (Public cloud) : est le plus grand cloud. Il est ouvert au public et il est accessible depuis n'importe quel endroit dans le monde. Le fournisseur de services de cloud public facture les utilisateurs sur une base horaire.
- Cloud communautaire : Ce modèle représente une extension du cloud privé. Il est partagé par plusieurs organisations qui ont des causes communes.
- Cloud hybride : représente la combinaison de deux ou plusieurs modèles de déploiement.

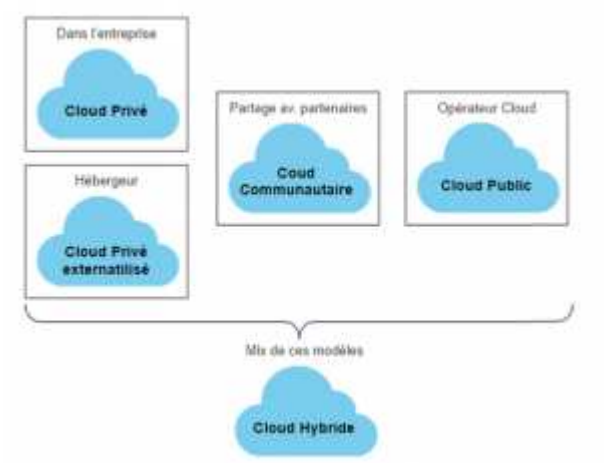


Figure 2:3: Modèles de déploiement du cloud computing [4]

La classification du cloud est basée sur plusieurs paramètres tels que la taille du cloud (nombre de ressources), le type de fournisseur de services, l'emplacement, le type d'utilisateurs, la sécurité et d'autres problèmes. Les quatre types de déploiements cloud sont discutés avec plus de détail dans les sections suivantes.

2.5.1 Cloud Privé (Private cloud)

Selon le NIST, le cloud privé est défini comme étant une infrastructure mise à disposition pour un usage exclusif par une organisation. Cette infrastructure peut exister sur ou hors site de l'organisation.

Ce type de cloud peut être détenu, géré et exploité par l'organisation, un tiers ou une combinaison des deux. En plus, il peut être déployé en utilisant des outils open-source (FOSS) tels que OpenStack et Eucalyptus.

Le cloud privé est de petite taille par rapport aux autres modèles de cloud. Dans ce cas, le cloud est déployé et maintenu par les organisations elles-mêmes.

2.5.1.1 Caractéristiques

Certaines caractéristiques du cloud privé sont les suivantes :

- Sécurité : le cloud privé est sécurisé car il est, généralement, déployé et géré par l'organisation. Par conséquent, il y a moins de risques de fuite de données hors du cloud. En plus, tous les utilisateurs appartiennent à la même organisation.

- Contrôle centralisé : Tant que l'organisation a le contrôle total sur le cloud, elle n'a pas besoin de quiconque pour gérer le cloud.
- Faible SLA : Les SLA formels peuvent ne pas exister dans un cloud privé. Mais s'ils existent, ils sont faibles puisqu'ils sont entre l'organisation et ses employés. En plus, la qualité du service n'est pas toujours optimale.

2.5.1.2 Avantages et inconvénients

Il y a plusieurs avantages et inconvénients d'un cloud privé.

Les avantages sont :

- Le cloud est de taille limité et facile à entretenir.
- Il offre un niveau élevé de sécurité et de confidentialité à l'utilisateur.
- Il est contrôlé par l'organisation

Les inconvénients sont :

- Les SLA ne sont pas respectés et le service peut ne pas être optimale.
- Contraintes budgétaire pour les petites entreprises.

2.5.2 Cloud Public (Public cloud)

Selon le NIST, le cloud public est l'infrastructure cloud à usage ouvert pour le grand public. Elle peut être détenue, géré et exploitée par une entreprise, un établissement universitaire ou un organisme gouvernementale [5]. Cette infrastructure est localisée dans les locaux du fournisseur du cloud (exemple : Google Datacenters).

Le cloud public peut servir tout type d'utilisateur. Un utilisateur peut simplement louer des ressources (Serveur, stockage, réseau ou software) sur une base horaire et utiliser ces ressources. Ces ressources sont disponibles dans les locaux du fournisseur de cloud. Généralement, les fournisseurs acceptent toutes les requêtes/demandes des utilisateurs (clients). Par conséquent, l'utilisateur aura

l'illusion que ces ressources sont infinies. Parmi les fournisseurs du cloud public on trouve *Amazon AWS* et *Microsoft Azure*.

2.5.2.1 Caractéristiques

On peut caractériser le cloud public par :

- Évolutive : le cloud public est hautement évolutive et l'approvisionnent des ressources passe à l'échelle rapidement (High Scalability). Le fournisseur du cloud doit s'assurer que toutes les demandes sont satisfaites.
- Abordable : les services du cloud public sont livrés aux public selon le model pay-asyou-go. Dans ce cas, l'utilisateur doit payer uniquement pour ce qu'il utilise (généralement sur une base horaire). Et cela n'implique aucun coût lié au déploiement.
- Moins sécurisé : le cloud public est moins sécurisé par rapport aux autres modèles. En effet, le cloud public est proposé par un tiers et il a un contrôle total sur le cloud. Bien que les SLA garantissent la confidentialité, il existe toujours un risque élevé de fuite de données.
- Hautement Disponible : le cloud public est hautement disponible, car n'importe qui dans n'importe où peut accéder avec des autorisations appropriées, ce qui n'est pas possible dans d'autres modèles car des restrictions d'accès géographiques ou autres peuvent exister.
- Strict SLAs : le SLA est très strict dans le cas du cloud public. La réputation commerciale du fournisseur de services dépend totalement de la qualité du service. Pour cela, il respecte strictement le SLA et évite tout type de violation.

2.5.2.2 Avantage et inconvénients

Il y a plusieurs avantages et inconvénients d'un cloud public :

Les Avantages sont :

- Il n'est pas nécessaire d'établir/acquérir une infrastructure pour configurer un cloud.
- Pas besoin d'assurer la maintenance du cloud.

- Moins chers par rapport aux autres modèles.
- Respect strict des SLAs.
- Pas de limite sur le nombre des utilisateurs.
- Passe à l'échelle rapidement (High Scalability).

Les Inconvénients sont :

- Problème de sécurité.
- La confidentialité et l'autonomie organisationnelle ne sont pas possibles.

2.5.3 Cloud Communautaire

Le cloud communautaire, selon le NIST, est l'infrastructure cloud utilisée par une communauté spécifique de consommateurs, ces consommateurs (organisations) partagent les mêmes intérêts. Cette infrastructure peut être gérée par un ou plusieurs organismes de la communauté. Le cloud communautaire est considéré comme une extension du cloud privé partagé entre plusieurs organismes.

L'avantage principale du cloud communautaire est le partage des ressources entre organisations en fonction des préoccupations bien déterminées. Ainsi, ici, les organisations sont en mesure d'extraire la puissance du cloud, qui est beaucoup plus grande que le cloud privé, et en même temps, elles peuvent l'utiliser à un coût généralement moins élevé. La communauté est formée sur la base d'une cause commune, mais finalement, tous les membres de la communauté en bénéficient.

Ce modèle est bénéfique dans le cas des organisations qui ne peuvent pas se permettre un cloud privé et ne peuvent pas non plus s'appuyer sur le cloud public.

2.5.3.1 Caractéristiques

Le cloud communautaire est caractérisé par :

- Maintenance collaborative et distributive : Le cloud communautaire est entièrement collaboratif, et généralement aucune partie n'a le contrôle total sur l'ensemble du cloud. La gestion de ce type de cloud est distributive et par conséquent, une bonne coopération donne de meilleurs résultats.

- Partiellement sécurisé : Partiellement sécurisé fait référence à la propriété du cloud communautaire où peu d'organisations partagent le cloud, il est donc possible que les données puissent être divulguées d'une organisation à une autre, bien qu'elles soient à l'abri du monde extérieur.
- Rentable : Le cloud communautaire est rentable car le cloud entier est partagé par plusieurs organisations ou une communauté. Habituellement, non seulement les coûts mais toutes les autres responsabilités partageables sont également partagées ou divisées entre les groupes.

2.5.3.2 Avantages et inconvénients

Ce qui suit décrit les nombreux avantages et inconvénients du cloud communautaire.

Les Avantages sont :

- Il permet d'établir un cloud privé avec un coût réduit.
- Il permet un travail collaboratif sur le cloud
- Il permet le partage des responsabilités entre l'organisation.
- Il est plus sécurisé que le cloud public.

Les Inconvénients sont :

- L'organisation n'est plus autonome.
- Les fonctionnalités de sécurité ne sont pas aussi efficaces que le cloud privé.
- Il ne convient pas s'il n'y a pas de collaboration.

2.5.4 Cloud Hybride

Le dernier modèle de déploiement est le cloud hybride. Selon le NIST, le cloud hybride est défini comme l'infrastructure cloud qui est une composition de deux ou plusieurs infrastructures cloud distinctes (privé, public ou communautaire) qui forment des entités uniques mais sont liées par une technologie standardisée ou propriétaire qui permet la portabilité des données et des applications.

Le cloud hybride est généralement une combinaison de clouds publics et privés. Ceci vise à combiner les avantages des clouds privés et publics. Ces avantages consistent à utiliser la puissance du cloud public tout en gardant les propriétés du cloud privé.

2.5.4.1 Caractéristiques

Le cloud hybride est caractérisé par :

- Évolutive (Scalable) : La principale raison d'avoir un cloud hybride est d'utiliser la propriété d'un cloud public avec un environnement de cloud privé. Le cloud public est utilisé en cas de besoin; par conséquent, comme le cloud public est évolutif, le cloud hybride avec l'aide de son homologue public est également évolutif.
- Partiellement sécurisé : Tant que le cloud hybride utilisant, généralement, le cloud public, le risque de violation de la sécurité est élevé. Ainsi, il ne peut pas être entièrement qualifié de sécurisé mais partiellement sécurisé.
- Strict SLA : Comme le cloud hybride implique une intervention du cloud public, les SLA sont stricts.
- Gestion complexe du cloud : La gestion du cloud est complexe et est une tâche difficile dans le cloud hybride car elle implique plusieurs modèles de déploiement et le nombre d'utilisateurs est également très élevé.

2.5.4.2 Avantages et inconvénients

Ce qui suit décrit les nombreux avantages et inconvénients du cloud communautaire.

Les Avantages sont :

- Il permet de réunir les avantages du cloud privé et public.
- Il est très évolutif

Les Inconvénients sont :

- ❖ Les fonctionnalités de sécurité ne sont pas aussi bonnes que le cloud public
- ❖ La gestion d'un cloud hybride est complexe.

2.6 Types de services

Le cloud computing est un modèle qui permet aux utilisateurs finaux d'accéder au pool partagé de ressources telles que le calcul, le réseau, le stockage, la base de données et les applications en tant que service à la demande sans avoir besoin de l'acheter ou de le posséder. Il existe trois principaux types de services de cloud computing, parfois appelés pile du cloud computing, car ils se superposent comme illustré ci-dessous :

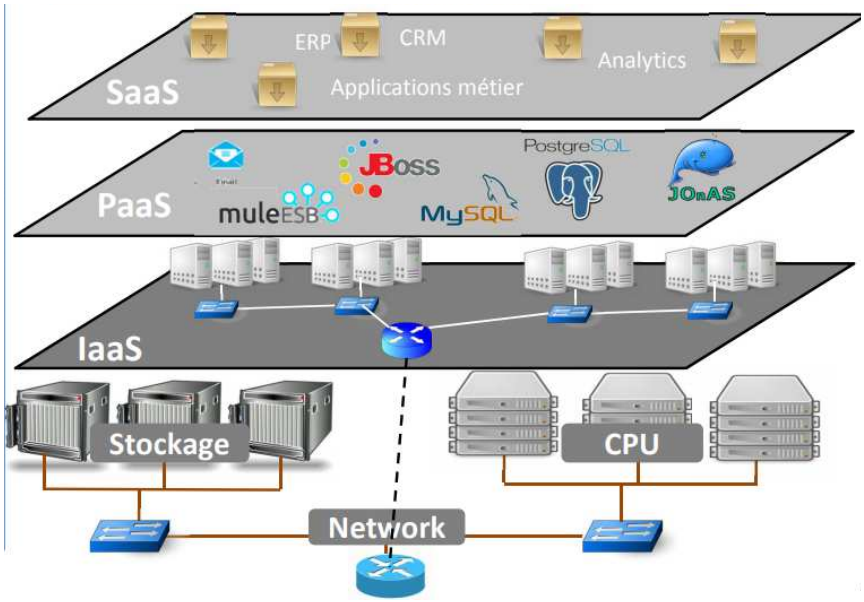


Figure 2:4: Types de services du Cloud [6]

Ces services sont fournis et gérés par le fournisseur de services, ce qui réduit l'effort de gestion du côté de l'utilisateur final comme indiqué sur la figure 1.4.

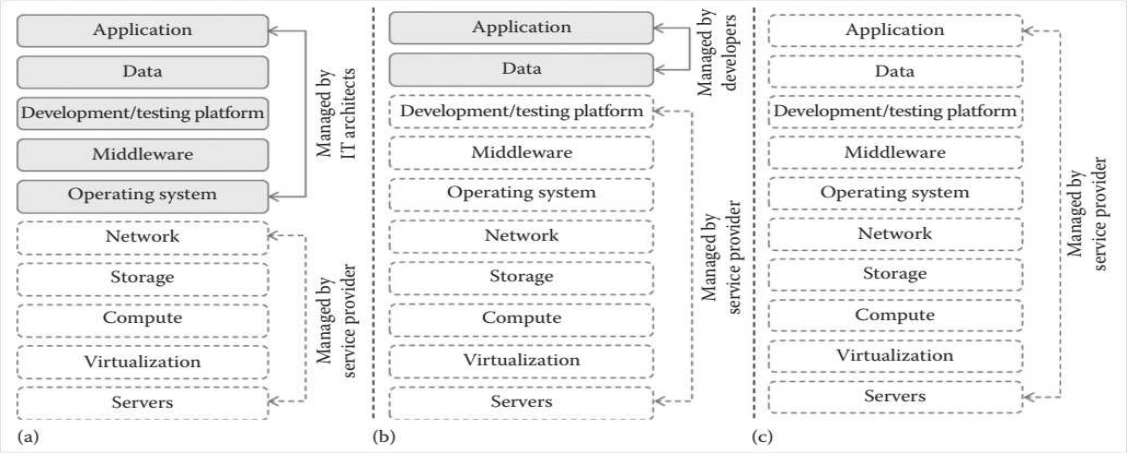


Figure 2:5:Responsabilités des utilisateurs et des fournisseurs pour chaque type de service : (a) IaaS, (b) PaaS, (c) SaaS [7]

Les différents types de services cloud ciblent différents publics. Par exemple, le modèle IaaS cible les architectes des technologies de l'information (IT), PaaS cible les développeurs et SaaS cible les utilisateurs finaux. En fonction des services souscrits, la responsabilité du public cible peut varier, comme le montre la figure 4. Les types de service peuvent être déployés en utilisant n'importe quel modèle de déploiement.

Cette section traite les caractéristiques, la pertinence et les avantages et les inconvénients des différents types de services cloud.

2.6.1 Infrastructure as a Service (IaaS)

Dans les centres de données traditionnels, la puissance de calcul est consommée par un accès physique à l'infrastructure. Actuellement, IaaS fournit (en location) des ressources (calcul, stockage et réseau) virtuelles en faisant abstraction des ressources physiques. La technologie de la *virtualisation* est utilisée pour fournir ces ressources qui sont regroupées en une seule entité qui s'appelle une machine virtuelle (VM). La VM est configurée par le fournisseur de service et livrée au consommateur finale (architecte IT par exemple).

L'utilisateur (architecte IT) contrôler ses ressources en via une interface utilisateur (web ou CLI[1]). Les fonctions typiques disponibles à partir de l'interface utilisateur incluent la création ou la suppression d'images du système d'exploitation, la mise à l'échelle des capacités (approvisionnement ou libération des ressources) requises ou

[1] . Command Line Interface

la définition de topologies de réseau. De plus, l'interface fournit les fonctionnalités requises pour les opérations, telles que le démarrage et l'arrêt des *vms*.

Dans le modèle de service IaaS, l'utilisateur est responsable de la maintenance de la plateforme de développement et de l'application s'exécutant au-dessus de l'infrastructure virtuelle. Les fournisseurs du service IaaS sont responsables de la maintenance du matériel sous-jacent (serveurs), comme illustré dans la figure 4(a). Un fournisseur IaaS peut fournir les services suivants :

- *Ressources de calcul* : Ce service comprend les processeurs (CPUs) et la mémoire principale pour les machines virtuelles.
- *Stockage* : Certains fournisseur fournissent des services de stockage des données distant comme *Amazon S3*.
- *Réseau* : Ce service inclue des composants de réseaux virtuels comme routeur virtuel, le commutateur et le pont pour les machines virtuelles
- *Équilibreur de charge (Load Balancer)* : Ce service fournit une capacité d'équilibrage de charge au niveau de la couche infrastructure.

1.1.1.1 Caractéristiques du IaaS

Les fournisseurs IaaS offrent aux consommateurs des ressources informatiques virtuelles sur une base de paiement à l'utilisation. En plus des caractéristiques du cloud, l'IaaS a ses propres caractéristiques.

- Accès aux ressources via le Web : le modèle IaaS permet aux utilisateurs d'accéder aux ressources d'infrastructure via Internet. Grâce à n'importe quel navigateur Web ou console de gestion, les utilisateurs peuvent accéder à l'infrastructure requise.
- Gestion centralisée : Les ressources réparties sur différentes parties peuvent être contrôlées à partir de n'importe quelle console de gestion. Cela garantit une gestion efficace des ressources et une utilisation efficace des ressources.
- Élasticité et mise à l'échelle dynamique : L'IaaS fournit des services élastiques où l'utilisation des ressources peut être augmentée ou diminuée en fonction des besoins. Ces derniers dépendent de la charge sur l'application.

- ❖ Infrastructure partagée : L'IaaS suit un modèle de livraison un-à-plusieurs et permet à plusieurs utilisateurs de partager la même infrastructure physique. Chaque utilisateur a ces propres VMs.
- ❖ Machines Virtuelles préconfigurées : Les fournisseurs IaaS proposent des VM préconfigurées avec des systèmes d'exploitation (OS), une configuration réseau, etc. Les utilisateurs informatiques peuvent sélectionner n'importe quel type de VM de leur choix. Ils peuvent directement commencer à utiliser les VM dès leur inscription aux services.
- ❖ Services mesurés : L'IaaS permet aux utilisateurs de louer les ressources au lieu de les acheter. Les services consommés par l'utilisateur seront mesurés et les utilisateurs seront facturés par les fournisseurs en fonction de la consommation.

2.6.1.1 Pertinences des IaaS

L'IaaS réduit le coût total de possession (TCO[2]) et augmente le retour sur investissement (ROI[3]) pour les start-ups qui ne peuvent pas investir davantage dans l'achat d'infrastructures.

L'IaaS peut être utilisé dans les cas suivants :

- ❖ Pics d'utilisation imprévisibles : Lorsque la demande est très volatile, nous ne pouvons pas prédire les pics et les creux en termes de demande de l'infrastructure. Dans cette situation, nous ne pouvons pas ajouter ou supprimer des ressources immédiatement en fonction de la demande dans une infrastructure traditionnelle. S'il existe une demande imprévisible d'infrastructures, il est recommandé d'utiliser les services IaaS.
- ❖ Contraintes d'investissements : Les start-ups ne peuvent pas investir davantage dans l'achat d'infrastructures pour leurs besoins commerciaux. Ainsi, en utilisant l'IaaS, les start-ups peuvent réduire l'investissement en capital sur le matériel. L'IaaS est l'option appropriée pour les start-ups qui investissent moins en capital sur le matériel.

[2] . Total Cost of Ownership.
[3] . Return On Investment : retour sur investissement.

- Infrastructure à la demande : Certaines organisations peuvent avoir besoin d'une grande infrastructure pendant une courte période (exemple : opérateur mobile pendant les périodes de l'Eid). À cette fin, une organisation ne peut pas se permettre d'acheter davantage de ressources. Au lieu de cela, ils peuvent louer l'infrastructure requise pour une période de temps spécifique.

L'IaaS aide les start-ups à limiter leurs dépenses d'investissement. Bien qu'il soit largement utilisé par les start-ups, il existe certaines situations où l'IaaS n'est peut-être pas la meilleure option. Dans les situations suivantes, les utilisateurs doivent éviter d'utiliser l'IaaS :

- Conformité réglementaire ne permet pas l'hébergement hors site de l'organisation : Pour certaines entreprises, sa réglementation peut ne pas permettre à l'application et aux données d'être hébergées sur une infrastructure tierce extérieurs.
- Utilisation minime : Lorsque l'infrastructure de l'organisation est suffisante et capable de satisfaire les besoins.
- Performance : Étant donné que les services IaaS sont accessibles via Internet, les performances peuvent parfois ne pas être celles attendues en raison de la latence du réseau.
- Besoin d'autonomie : Certaines organisations peuvent nécessiter un contrôle physique de l'infrastructure. Les services IaaS étant abstraits en tant que ressources virtuelles, il n'est pas possible d'avoir plus de contrôle sur l'infrastructure physique.

2.6.1.2 Avantages et inconvénients

Étant l'un des types de service les plus importants du cloud computing, l'IaaS offre de nombreux avantages aux utilisateurs informatiques. Voici ces avantages:

- Le modèle Pay-as-you-go : Les services IaaS sont livrés aux utilisateurs à la base du modèle pay-as-you-go. Cela garantit que les clients sont tenus de

payer pour ce qu'ils ont utilisé. Ce modèle élimine les dépenses inutiles pour l'achat de matériel.

- ❖ Ressources élastiques : L'IaaS fournit des ressources en fonction des besoins. Les utilisateurs peuvent augmenter ou réduire les ressources à tout moment.
- ❖ Réduire le TCO : Les utilisateurs informatiques peuvent louer l'infrastructure informatique plutôt que de l'acheter. L'IaaS réduit le besoin d'acheter des ressources matérielles et réduit ainsi le TCO.
- ❖ Écologique.

Même si l'IaaS offre des avantages liés aux coût, il ne permet pas de sécuriser les données. Voici les inconvénients du IaaS :

- ❖ Problème de sécurité : les fournisseurs du IaaS utilise les hyperviseurs pour fournir des machines virtuelles. Il existe de nombreuses attaques qui ciblent les hyperviseurs pour le compromettre. Si les hyperviseurs sont compromis, toutes les machines virtuelles peuvent être attaquées facilement.
- ❖ Problème d'interopérabilité : Il n'y a pas de normes standards suivies par les différents fournisseurs IaaS. Il est très difficile de migrer une machine virtuelle d'un fournisseur IaaS à l'autre.
- ❖ Problème de performance : La latence du réseau (Internet) joue un rôle important dans les performances du service fournit.

2.6.2 Platform as a Service (PaaS)

PaaS permet aux développeurs de développer leur application en ligne et leur permet également de la déployer immédiatement sur la même plateforme. Les consommateurs ou développeurs PaaS peuvent utiliser des environnements de programmation, des plateforme d'application, des bases de données, des outils de test et des outils de déploiement en tant que service sur Internet. Par conséquent, il réduit la complexité d'achat et de maintenance de différents outils pour développer une application. Dans PaaS, les utilisateurs sont responsables de la gestion de l'application qu'ils ont développée. L'infrastructure sous-jacente sera maintenue par le fournisseur d'infrastructure, comme le montre la figure 1.5(b). Certains fournisseurs PaaS peuvent fournir les services suivants :

- Langages de programmation : Les fournisseurs de PaaS fournissent une grande variété de langages de programmation pour les développeurs. Certains des langages de programmation les plus populaires fournis par les fournisseurs PaaS sont Java, Perl, PHP, Python, Ruby, Scala, Clojure et Go.
- Application frameworks : Les fournisseurs de PaaS fournissent des plateformes d'applications qui simplifient le développement d'applications. Par exemple : Node.js, Rails, Drupal, Joomla, WordPress, Django, EE6 et Spring.
- Base de données : Les bases de données sont livrées aussi par les fournisseurs de PaaS. Parmi les SGBD fournit on peut citer : ClearDB, PostgreSQL, cloudant, Membase, MongoDB et Redis.
- Autres outils : En plus, PaaS fournit les outils nécessaires pour développer, tester et déployer une application.

2.6.2.1 Caractéristiques du PaaS

Voici les caractéristiques essentielles qui rendent le PaaS unique par rapport aux plates-formes de développement traditionnelles :

- Tout en un : La plupart des fournisseurs PaaS offrent des services pour développer, tester, déployer, héberger et maintenir des applications dans le même IDE.
- Accès web : Grâce à l'interface utilisateur Web, tout développeur peut accéder à la plateforme de développement. L'interface utilisateur Web aide les développeurs à créer, modifier, tester et déployer différentes applications sur la même plate-forme.
- Accès hors ligne : Pour permettre le développement hors ligne, certains des fournisseurs PaaS permettent au développeur de synchroniser leur IDE local avec les services PaaS. Les développeurs peuvent développer une application localement et la déployer en ligne chaque fois qu'ils sont connectés à Internet.
- Évolutivité intégrée : Les services PaaS offrent une évolutivité intégrée à une application développée à l'aide de n'importe quel PaaS. Cela garantit que l'application est capable de s'adapter en fonction du besoin (passe à l'échelle).

- Plateforme collaborative : De nos jours, l'équipe de développement est composée de développeurs qui travaillent à différents endroits. La plupart des services PaaS prennent en charge le développement collaboratif.
- Variétés des outils Clients : Pour faciliter le développement, les fournisseurs PaaS fournissent une grande variété d'outils clients pour aider le développeur. Les outils clients incluent CLI, CLI Web, interface utilisateur Web, API REST et IDE.

2.6.2.2 Pertinence des PaaS

La plupart des start-ups de développement SaaS et des éditeurs de logiciels indépendants utilisent largement PaaS pour développer leurs applications. Le modèle PaaS attire également l'attention d'autres sociétés de développement de logiciels traditionnels. PaaS est une option appropriée pour les situations suivantes :

- Développement collaborative : Étant donné que les services PaaS fournissent un environnement de développement collaboratif, c'est une option appropriée pour les applications qui nécessitent une collaboration de plusieurs développeurs.
- Déploiement et test automatisé : L'équipe de développement doit se concentrer davantage sur le développement plutôt que sur les tests et le déploiement. La plupart des services PaaS offrent des capacités de test et de déploiement automatisées.
- Commercialisation rapide : Les services PaaS suivent les méthodologies de développement itératives et incrémentielles qui garantissent que l'application est sur le marché selon le délai imparti.

Même s'il est largement utilisé, il existe certaines situations où le PaaS peut ne pas être la meilleure option :

- Migration fréquente des applications : Puisqu'il n'y a pas de normes standards suivies par les fournisseurs PaaS, il est très difficile de migrer l'application d'un fournisseur PaaS vers un autre.

- Personnalisation au niveau de l'infrastructure : Certaines plates-formes de développement d'applications nécessitent une configuration ou une personnalisation de l'infrastructure sous-jacente. PaaS ne permet pas à l'utilisateur d'avoir un contrôle sur l'infrastructure.

2.6.2.3 Avantages et inconvénients

L'avantage principale du PaaS est de masquer la complexité de la plateforme et de l'infrastructure. Cela permet aux développeurs de se focaliser sur l'implémentation des fonctionnalités importantes de leurs applications. En plus de ça, le PaaS peut avoir les avantages suivants :

- Développement et déploiement rapide : La plupart des services PaaS automatisent le processus de test et de déploiement dès que le développeur termine le développement. Cela accélère le développement et le déploiement d'applications.
- Développement collaboratif : Les services PaaS aident les développeurs de différents endroits à travailler ensemble sur le même projet. Cela est possible grâce à la plateforme de développement commune en ligne fournie par les fournisseurs PaaS.
- Moins de frais de maintenance : Dans les Systèmes traditionnels, l'organisations a besoin de recruter des personnels qualifiés pour maintenir l'infrastructure et la pile de logiciels exécutés sur cette infrastructure.

PaaS fournit plusieurs avantages aux développeurs. Par contre, il contient des inconvénients, qui sont décrits ci-dessous :

- Vendor lock-in : La plupart des fournisseurs PaaS utilisent des technologies propriétaires qui ne sont pas compatibles avec les autres fournisseurs PaaS. Par conséquent, la migration des application d'un vendeur à un autre est impossible par faute de compatibilité.
- Problèmes de sécurité : Comme dans les autres services cloud, la sécurité est l'un des problèmes majeurs des services PaaS. Étant donné que les données

sont stockées sur des serveurs tiers hors site, de nombreux développeurs ont peur de recourir aux services PaaS.

- Besoin d'internet : Avec une connexion Internet lente, la convivialité et l'efficacité de la plate-forme PaaS ne satisfont pas aux exigences des développeurs.

2.6.3 Software as a Service (SaaS)

SaaS signifie les applications cloud qui s'adressent directement à l'utilisateur final. Ces applications (SaaS) sont livrées à la demande sur internet et il n'est pas nécessaire de les installer localement. L'utilisateur peut accéder aux services SaaS à tout moment et depuis n'importe quel appareil connecté tel que les laptops, tablettes, ou smart phones. En SaaS, l'utilisateur final est libre de maintenir l'infrastructure, la plate-forme de développement et l'application qu'il utilise. Toute la maintenance sera effectuée par les fournisseurs SaaS comme illustré Figure 1.5(c).

Le fournisseur SaaS peut fournir des services commerciaux, des réseaux sociaux, la gestion de documents et des services de messagerie :

- Business service : La plupart des fournisseurs de SaaS ont commencé à fournir une variété de services commerciaux qui attirent les start-ups. Parmi ces applications on trouve les ERP, CRM, logiciels de facturation et de vente ...etc.
- Réseaux sociaux : Étant donné que les sites de réseaux sociaux sont largement utilisés par le grand public, de nombreux fournisseurs de services de réseaux sociaux ont adopté le SaaS pour leur durabilité.
- Gestion des documents : la plupart des fournisseurs SaaS ont commencé à fournir des services qui sont utilisés pour créer, gérer et suivre les documents électroniques exemple Google docs.
- Messagerie électronique : Pour gérer le nombre imprévisible d'utilisateurs et la charge sur les services de messagerie, la plupart des fournisseurs de messagerie ont commencé à offrir leurs services en tant que services SaaS exemple Gmail.

2.6.3.1 Caractéristiques du SaaS

Les services SaaS sont différents et offrent plus d'avantages aux utilisateurs que les logiciels traditionnels. Voici les caractéristiques essentielles des services SaaS qui le rendent unique par rapport aux logiciels traditionnels :

- Multi-tenant (One-to-many) : Les services SaaS sont fournis comme un modèle un à-plusieurs où une seule instance de l'application peut être partagée par plusieurs locataires ou clients (tenants).
- Accès web : Les services SaaS fournissent un accès Web aux applications. Il permet à l'utilisateur d'accéder à l'application depuis n'importe quel emplacement si l'appareil est connecté à Internet.
- Prise en charge de plusieurs appareils : Les services SaaS sont accessibles à partir de tous les appareils des utilisateurs finaux tels que les ordinateurs de bureau, ordinateurs portables, tablettes, smartphones et clients légers (navigateurs web).
- Très évolutive : Étant donné que la plupart des services SaaS utilisent PaaS et IaaS pour leur développement et leur déploiement, ils garantissent une meilleure évolutivité que les logiciels traditionnels.
- Haute disponibilité : Les services SaaS garantissent la disponibilité à 99,99% des données utilisateur, car des mécanismes de sauvegarde et de restauration appropriés sont mis en œuvre en back-end.
- Interopérabilité : Les services SaaS ont la capacité de s'intégrer à d'autres logiciels ou services via des API standard.

2.6.3.2 Pertinences des SaaS

Le SaaS est populaire auprès des particuliers et des start-ups en raison des avantages qu'il offre. SaaS Services sont la meilleure solution dans les cas suivants :

- Application à la demande : Payer des licences pour des logiciels utilisés occasionnellement réduit le retour d'investissement et augmente les dépenses de l'entreprise. Pour cela, les utilisateurs recherchent un logiciel qu'ils peuvent utiliser au fur et à mesure de leurs besoins.

- Application pour les startups : Les startups qui ont des contraintes financières pour acquérir des infrastructures et des licence logiciel, SaaS est une solution très adéquate.
- Application compatible avec plusieurs appareils : Les services SaaS sont compatible avec la majorité des appareils existants.
- Charge imprévisible et variable : Nous ne pouvons pas prédire la charge des applications populaires telles que les sites de réseaux sociaux. L'utilisateur peut se connecter ou se déconnecter des applications à tout moment. Grâce aux capacités de mise à l'échelle dynamique, les applications SaaS peuvent gérer des charges variables d'une manière efficace.

D'autre part, les services SaaS peuvent ne pas être le meilleur choix pour tous les types de logiciels. Dans ce qui suit, nous mentionnons des situations où le SaaS ne convient pas :

- Application temps-réel : Étant donné que les applications SaaS dépendent de
- La connectivité Internet, elles peuvent ne pas fonctionner mieux avec une faible vitesse Internet.
- Application avec des données confidentielles : La sécurité des données, la gouvernance des données et la conformité des données sont toujours des problèmes avec les applications SaaS.
- Application locale satisfaisante : Certaines applications locale peuvent répondre à toutes les exigences de l'organisation. Dans de telles situations, la migration vers le modèle SaaS peut ne pas être bénéfique.

2.6.3.3 Avantages et inconvénients

Les applications SaaS sont utilisées par un large éventail de particuliers et de start-ups pour ses avantages liés aux coûts. Outre les avantages liés aux coûts, les services SaaS offrent les avantages suivants :

- Sans installation : Les utilisateurs peuvent accéder aux services directement sans aucune installation.

- Réduction du coût : Tant que, les services SaaS suivent le modèle pay-as-you-use, l'utilisateur ne payera que ce qu'il a utilisé.
- Moins de maintenance : L'utilisateur n'a pas besoin de s'inquiéter pour toutes les opérations de maintenance et mise à jour de l'application car le fournisseur SaaS prend en charge toutes ces opérations.
- Accès facile : Les services SaaS sont accessibles à partir de n'importe quel appareil s'il est connecté à Internet.
- Passage à l'échelle dynamique : Les services SaaS sont élastiques et ils peuvent gérer tout type de charges variables sans perturber le comportement normal de l'application.
- Reprise après sinistre : Avec des mécanismes de sauvegarde et de restauration appropriés, des répliques sont conservées pour tous les services SaaS.
- Mutualisation des ressources (Multitenancy) : la multitenancy permet à une instance d'application d'être partagée par plusieurs clients.

Bien que les services SaaS sont utilisés par plusieurs particuliers et startups, l'adoption par les grandes industries est très faible. Voici les principaux problèmes rencontrés avec les services Saas

- Sécurité : La sécurité est la principale préoccupation lors de la migration vers une application SaaS. Étant donné que l'application SaaS est partagée entre de nombreux utilisateurs, il existe une possibilité de fuite de données. L'utilisateur doit être prudent lors de la sélection du fournisseur SaaS pour éviter toute perte inutile de données
- Exigences de connectivité : Les applications SaaS nécessitent une connectivité Internet pour y accéder. Parfois, la connectivité Internet de l'utilisateur peut être très lente. Dans de telles situations, l'utilisateur ne peut pas accéder facilement aux services.
- Perte de contrôle sur les données : Étant donné que les données sont stockées dans un emplacement tiers et hors site, l'utilisateur final n'a aucun contrôle sur les données

2.7 Cloud Écosystème

L'écosystème Cloud (illustré par la figure 1.6) est un terme utilisé pour décrire un système d'un ensemble de composants/entités interdépendants qui travaillent ensemble pour réaliser, supporter ou consommer les services Cloud.

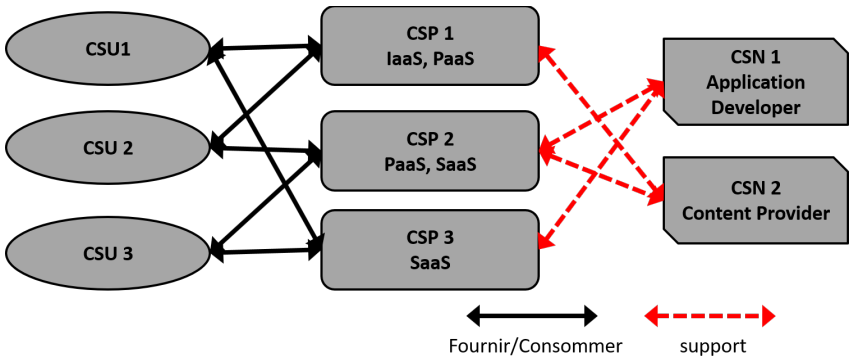


Figure 2:6: Ecosystème du Cloud [8]

- Cloud Service Users (CSUs) : consomme les services Cloud, exemple : personne normale, entreprise ou gouvernement.
- Cloud Service Providers (CSPs) : fournis/livre et maintient/gère les services Cloud.
- Cloud Service Partners (CSns) : Une personne ou une organisation (par exemple développeur d'applications, fournisseur de contenu, logiciel, matériel et / ou équipement, intégrateur de système et / ou auditeur) qui fournit un support à la création d'un service offert par un fournisseur CSP (par exemple, intégration de services).

2.8 Les technologies pilotes du Cloud

D'un point de vue technologique, le succès du cloud computing peut être associé aux avancés technologiques dans divers domaines tels que l'architecture orienté service (*SOA : Service Oriented Architecture*), les processeurs multi-cœurs, la technologie de stockage et de mémoire, la technologie de réseau, web 2.0 et web 3.0 et la virtualisation. Les progrès réalisés dans chacun de ces domaines technologiques ont considérablement contribué à l'adoption généralisée du cloud computing. Cette partie

présente d'une manière abstraite quelque technologie pilote du cloud et montre également comment les dernières avancées technologiques dans chacune de ces technologies habilitantes ont eu un impact sur le succès cloud computing.

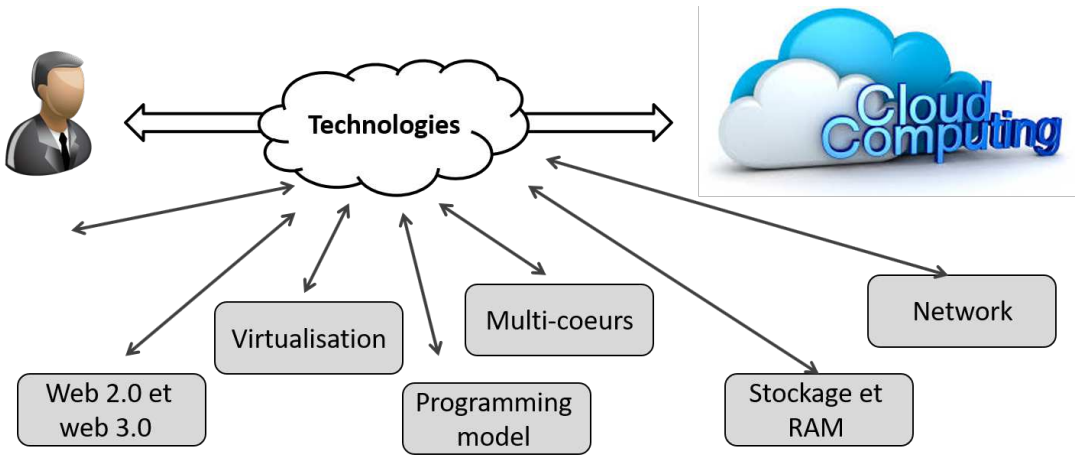


Figure 2:7: Technologies pilotes du cloud computing [9]

2.8.1 SOA et Web services

SOA est un style d'architecture logicielle, qui définit comment les services sont offerts et utilisés. L'architecture orientée services (AOS) est une réponse très efficace aux problématiques que rencontrent les entreprises en termes de réutilisabilité, d'interopérabilité et de réduction de couplage entre les différents systèmes qui implémentent leurs systèmes d'information.

Un service n'a pas besoin d'avoir une connaissance préalable de l'application appelante, et l'application n'a pas besoin d'avoir des connaissances sur la façon dont les tâches sont effectuées par un service.

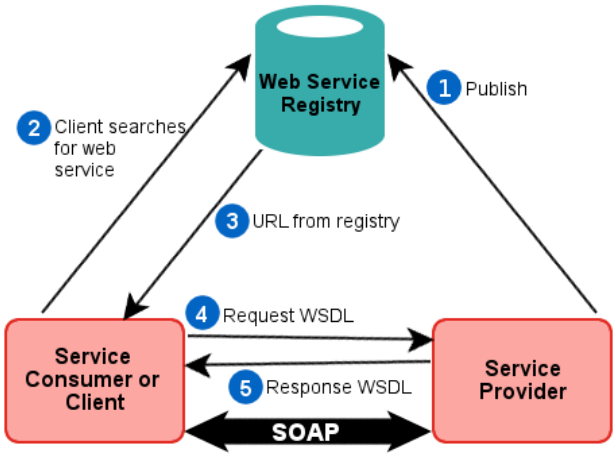


Figure 2:8: Participants et actions dans SOA [10]

2.8.1.1 Définition SOA

SOA est une architecture système qui représente des méthodes ou des applications différentes et incompatibles en tant que services réutilisables et accessibles de manière ouverte, permettant ainsi de les utiliser et les réutiliser de manière indépendante de la plate-forme ou de la langue.

2.8.1.2 Définition Web services

Une application logicielle identifiée par un URI, dont les interfaces et liaisons sont définies, décrites et découvertes par des artefacts XML. Il prend en charge les interactions directes avec d'autres applications logicielles en utilisant des messages XML via des protocoles Internet.

2.8.1.3 Avantages

- **Réutilisation** : Différents services peuvent être réutiliser par différentes applications, ce qui réduit les coûts de développement et de maintenance.
- **Agilité** : SOA peut apporter l'agilité architecturale dans une entreprise grâce à l'utilisation étendue des normes telles que les services web.
- **Monitoring** : Elle permet de surveiller les performances des différents services pour effectuer les changement requis.
- **Collaboration et extension** : Elle permet d'obtenir des services divers d'autres processus pour accomplir une tâche particulière.

2.8.2 Les technologies de stockage et de mémoire

La technologie ou les solutions de stockage utilisées dans l'environnement cloud doivent répondre aux exigences suivantes :

- Scalabilité : le système de stockage doit prendre en charge l'évolutivité des données de l'utilisateur.
- Haute disponibilité (High Availibilty) : le degré de disponibilité des solutions de stockage déployées dans le cloud devrait être très élevé.
- Bande passante élevée : Le système de stockage dans le cloud doit supporter le taux rapide du transfert de données.
- Performance constantes : aucun problème de performances ne doit être associé au système de stockage, et les performances doivent être constante s tout au long de la période du contrat.
- Équilibrage de charge (Load Balancing) : Afin d'obtenir une utilisation efficace des ressources, les systèmes de stockage déployés dans le cloud doivent être suffisamment intelligents pour prendre en charge l'équilibrage des charges automatique des données des utilisateurs.

2.8.3 Les technologies Réseau

Les technologies et les solutions réseaux utilisées dans le cloud doivent répondre à ces exigences :

- **Consolider les charges de travail et fournir l'infrastructure en tant que service (IaaS) à divers utilisateurs** : La technologie réseau doit permettre la consolidation de la charge de travail de l'entreprise, qui réduit les coûts de gestion et offre plus de flexibilité et d'évolutivité pour la gestion des machines virtuelles.
- **Connecter des machines virtuelles aux réseaux physiques et virtuels** : Le système de réseau cloud doit prendre en charge les fonctionnalités permettant de connecter les différentes machines virtuelles aux réseaux virtuels et au réseau physique. Il convient également de faire respecter l'application de la politique de sécurité, l'isolement et les niveaux de service.
- **Assurer la connectivité et gérer la bande passante du réseau** : Le système de réseau cloud doit disposer des fonctionnalités d'équilibrage de charge et de

basculement qui permettent l'agrégation de la bande passante du serveur et le basculement du trafic.

- **Accélérer les performances des applications et du serveur** : Diverses caractéristiques et technologies telles que les technologies à faible latence, le protocole de contrôle de transmission de centre de données pourraient être utilisés pour améliorer les performances de l'infrastructure de réseau en nuage.

2.8.4 La virtualisation

La virtualisation est la technologie la plus importante dans le cloud computing. Elle permet à une infrastructure physique de fonctionner comme plusieurs infrastructures ou ressources logiques. Elle permet également une utilisation efficace des ressources en exploitant la totalité des ressources disponible. La virtualisation n'est pas seulement limitée au matériel, elle peut prendre de nombreuses formes telles que la mémoire, le processeur, les E / S, le réseau, le système d'exploitation, les données et les applications. Les différentes formes de virtualisation seront discutées dans la section suivante.

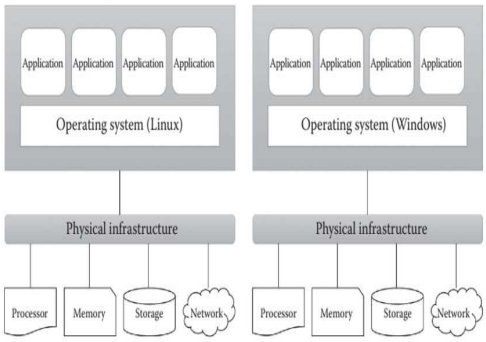


Figure 2:9: Avant la virtualisation [11]

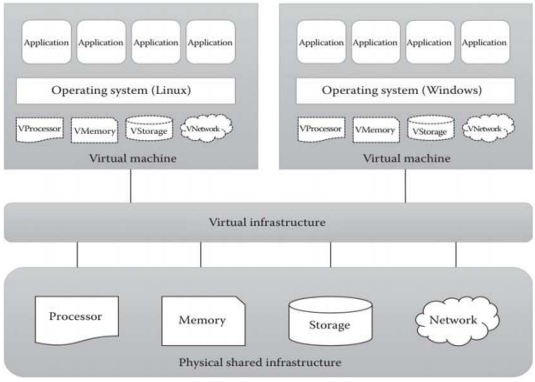


Figure 2:10: Après la virtualisation [11]

Avant la virtualisation (voir figure 2.9), une infrastructure physique (un serveur) a été utilisée pour exécuter un qu'seul système d'exploitation et ses applications, ce qui entraîne une sous-utilisation des ressources. La nature non partagée du matériel oblige les organisations à acheter un nouveau matériel pour répondre à leurs besoins de calcul supplémentaires. Par exemple, si une organisation souhaite expérimenter ou simuler sa nouvelle idée, elle doit utiliser des systèmes dédiés distincts pour différentes

expériences. Donc, pour mener à bien leurs travaux de recherche, ils ont tendance à acheter un nouveau matériel qui augmentera le CapEx[4].

Après l'introduction de la virtualisation (voir figure 2.10, différents systèmes d'exploitation et ces applications peuvent partager une seule infrastructure physique. La virtualisation réduit le montant énorme investi dans l'achat de ressources supplémentaires. La virtualisation devient un moteur clé dans l'industrie informatique, en particulier dans le cloud computing.

2.8.4.1 Hyperviseurs

Aujourd'hui, Les machines virtuelles (*VMs : Virtual Machines*) sont largement utilisées, au lieu des machines physiques, dans le domaine IT industriel. Ces VMs prennent en charge les solutions IT écologique. Étant donné que les machines virtuelles sont principalement utilisées, la technologie qui permet de créer ces environnements virtuels a aussi retenu l'attention du milieu industriel et académique.

L'environnement virtuel peut être créer avec l'aide d'un logiciel qui s'appelle Hyperviseur. L'hyperviseur est un méta-système d'exploitation minimal qui se situe entre les VMs et l'infrastructure physique. Son rôle consiste à fournir l'infrastructure virtuelle requise pour les VMs. Généralement, l'infrastructure virtuelle signifie des processeurs virtuels (*vCPUs*), mémoire virtuelle(*vRAMs*), carte réseau virtuelle (*vNICs*), stockage virtuel ...etc. Les hyperviseurs, aussi appelés VMMs (Virtual Machines Manager), sont considérés comme les principaux moteurs de la virtualisation dans les datacenters. Actuellement, il existe différents hyperviseurs qui sont utilisés dans l'industrie, parmi les VMMs les plus utilisés on trouve VMware ESXI, Xen, Microsoft Hyper-V et KVM. Ces VMMs sont classés en deux catégories.

[4] . CapEx : Capital Expenditures signifie investissement capitale.

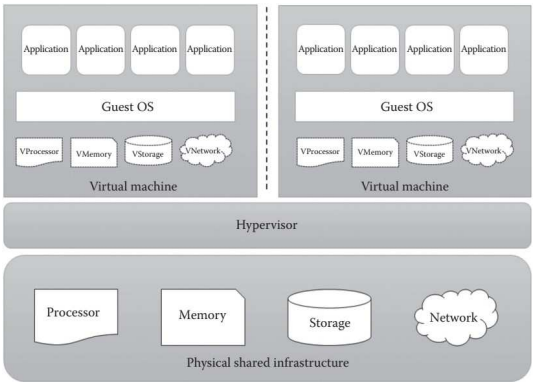


Figure 2:11: Hyperviseur type 1 [12]

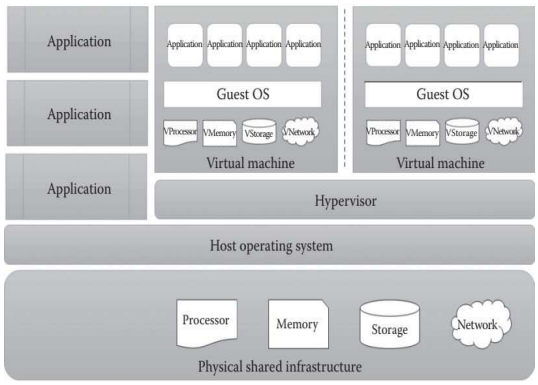


Figure 2:12: Hyperviseur type 2 [12]

Hyperviseurs type 2 (hyperviseurs intégrés). Ce type d'hyperviseur est installé, comme une application, au-dessus d'un système d'exploitation (voir figure 1.11). Il nécessite la communication avec l'OS hôte pour accéder aux ressources physiques. L'inconvénient majeur de cette approche est que l'échec du système d'exploitation (*OS : Operating System)* hôte implique l'échec des VMs. Donc, il est recommandé d'utiliser les VMMs type 2 uniquement sur les systèmes clients où l'efficacité est moins critique. Quelques exemples de VMM type 2 sont VMware Workstation, VMware Player et Oracle VirtualBox.

Hyperviseur type 1 (Bare metal) Il peut s'exécuter et accéder aux ressources physiques directement sans l'aide d'un OS hôte (voir figure 1.12). Comparé au hyperviseur type 2, la surcharge de la communication avec l'OS hôte est réduite qui offre une meilleure efficacité. Ce type d'hyperviseur est utilisé pour les serveurs qui gèrent des taches lourdes et nécessitent plus de sécurité. Parmi les VMM type 1 on trouve KVM, Microsoft Hyper-V, Citrix XenServer, VMware ESXI et Oracle VM.

2.8.4.2 Virtualisation des ressources

La virtualisation est le processus d'abstraction des ressources physiques dans le pool de ressources virtuelles qui peut être donné à n'importe quelle machine virtuelle (VM). Les différentes ressources comme la mémoire, les processeurs, le stockage et le réseau peuvent être virtualisées à l'aide des technologies de virtualisation appropriées. Dans cette section, nous allons discuter certaines des ressources qui peuvent être virtualisées.

2.9.4.2.1 Virtualisation du processeur

La virtualisation du processeur permet le partage du processeur physique en plusieurs processeurs logique (vCPU). Ce partage entre processeur se fait au niveau de l'hyperviseur (couche de la virtualisation voire section 2.8.4.1). Les CPUs sont ensuite utilisés/partagés par les machines virtuelles.

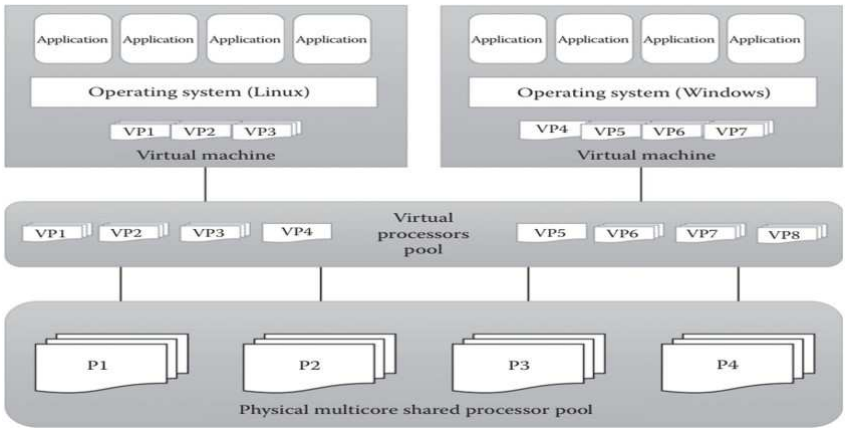


Figure 2:13: Virtualisation des processeurs [13]

2.9.4.2.2 Virtualisation de la mémoire

Une autre technique importante de la virtualisation des ressources est la virtualisation de la mémoire. Le processus de la virtualisation de la mémoire consiste à mapper la mémoire physique en un pool de mémoires virtuelles partagé entre les VMs. L'idée principale consiste à mapper les numéros de pages virtuels sur les numéros de pages physiques. Tous les processeurs x86 modernes prennent en charge la virtualisation de la mémoire principale (voir section(0)).

En plus, la virtualisation de la mémoire peut, également, être réalisée en utilisant l'hyperviseur. Normalement, dans les centres de données virtualisés, la mémoire principale inutilisée des différents serveurs se consolidera en un pool de mémoire principale virtuelle et pourra être donnée aux VM. Le concept de virtualisation de la mémoire principale est illustré à la figure 1.14.

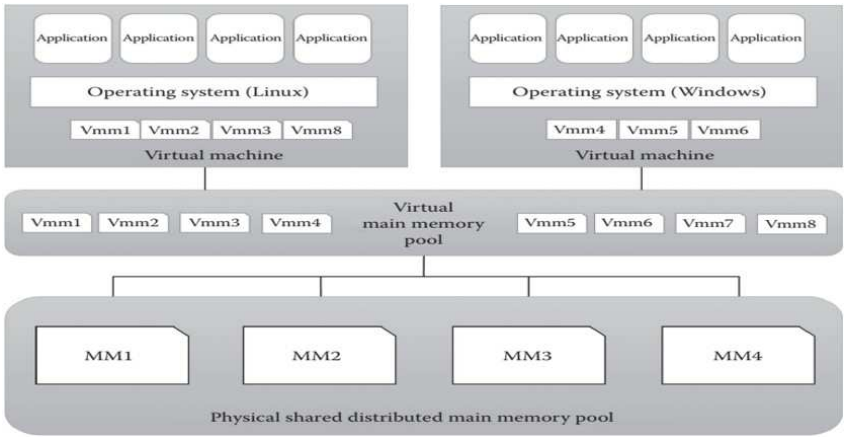


Figure 2:14:Virtualisation de la mémoire [14]

2.8.4.2.3 Virtualisation du stockage

La virtualisation du stockage est une forme de virtualisation de ressources où plusieurs disques de stockage physique sont virtualisés en tant que groupe de disques de stockage virtuel (logique) pour les VMs. Le processus de virtualisation du stockage est illustré dans la figure 1.15.

La virtualisation du stockage est principalement utilisée pour le maintien d'une sauvegarde ou une réplique des données qui sont stockées sur les machines virtuelles. De plus, il peut être étendu pour supporter la haute disponibilité des données. Parmi les techniques avancées de la virtualisation du stockage on peut citer Storage Area Network (SAN) et Network Attached Storage (NAS).

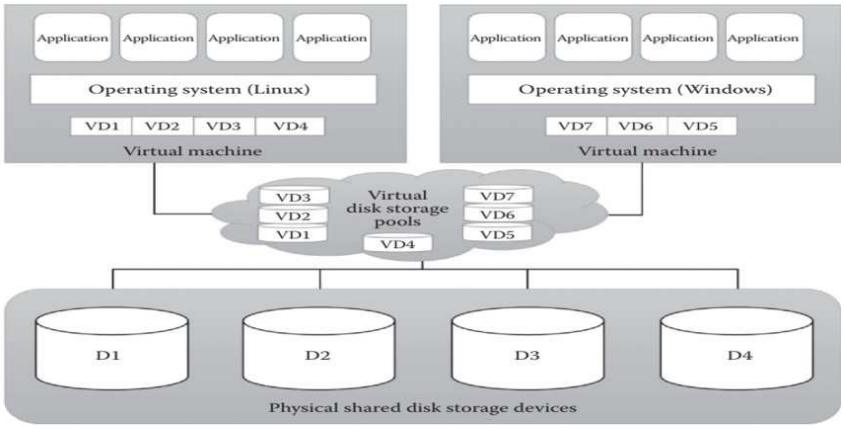


Figure 2:15: Virtualisation du stockage [15]

2.8.4.2.4 Virtualisation du réseau

La virtualisation du réseau (illustré dans la figure 2.16) correspond à la virtualisation de plusieurs foncions du réseau. Elle permet le partage d'une carte réseau par plusieurs machines virtuelles. Par conséquent, on peut avoir plusieurs réseaux virtuels au sein d'une seule machine physique avec une seule carte physique.

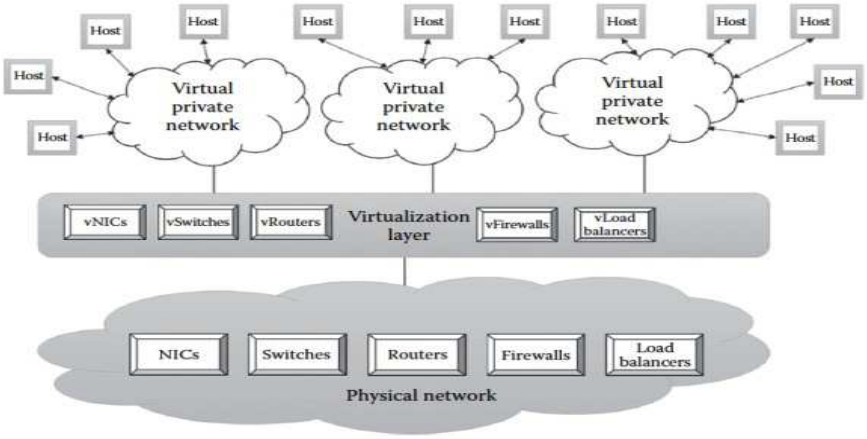


Figure 2:16: Virtualisation du réseau [16]

La virtualisation du réseau (VLAN) a les avantages suivants :

- Transparence : les périphériques distribués peuvent être regroupés dans un seul réseau logique. Les VLANs sont très utiles lors de la conception des IT pour des emplacements géographiquement disparates.
- Sécurité : Certains systèmes qui nécessite une certaine protection particulière peuvent être cachés dans un réseau virtuel séparé.

Par ailleurs, les VLAN impliquent plus de surcharge pour l'administration du réseau et pour la programmation des composants réseau actifs (switchs, etc.)

2.8.5 Types de virtualisation

Il existe 3 approches différentes de la virtualisation

2.8.5.1 *Virtualisation complète*

Dans la virtualisation complète, l'OS invité[5] est complètement abstrait de la couche physique car l'hyperviseur découple l'OS de l'infrastructure matériel. Dans ce cas l'OS invité ne sait pas qu'il est virtualisé et pense qu'il s'exécute sur du vrai matériel.

Dans cette approche l'hyperviseur fournit toutes les infrastructures virtuelles nécessaires aux machines virtuelles. Par conséquent, L'OS invité ne peut pas communiquer avec l'infrastructure physique directement, il nécessite l'aide de l'hyperviseur. Cette approche utilise les translations binaires (binary translation) pour transformer les instructions de l'OS invité non-virtualisées par de nouvelles séquences d'instructions qui ont le même effet prévu sur l'infrastructure virtuelle.

2.8.5.1.1 Avantages

- ❖ Cette approche fournit une meilleure isolation et sécurité pour les machines virtuelles.
- ❖ Elle permet d'exécuter plusieurs OS simultanément.
- ❖ L'OS invité peut facilement migrer pour s'exécuter sans virtualisation.
- ❖ Ne nécessite pas de modification sur le système d'exploitation invité.

2.8.5.1.2 Inconvénients

- Les translations binaires sont une surcharge supplémentaire qui réduit les performances globales du système.

2.8.5.2 *Paravirtualisation*

Cette approche est également connue sous le nom de *Partial virtualisation* ou bien *OS-assisted-virtualisation*. Elle fournit une simulation partielle de la couche physique. La différence majeure entre la virtualisation complète et la para-virtualisation est que L'OS invité sait qu'il s'exécute dans un environnement virtualisé dans la para-virtualisation. En plus, la para-virtualisation remplace la translation binaire des requêtes (non virtualisé) de l'OS invité par des hypercalls. Ces derniers sont similaires aux appels système et utilisés pour communiquer directement avec l'hyperviseur. Cette communication directe entre l'OS invité et l'hyperviseur améliore les

[5] . L'OS invité (OS Guest) est le système d'exploitation installé sur la machine virtuelle.

performance et l'efficacité de cette approche. Cependant, l'OS invité doit être modifié à cet effet.

2.8.5.2.1 Avantages

- La para-virtualisation élimine le surcoût supplémentaire de la translation binaire et améliore ainsi l'efficacité et les performances globales du système.
- Elle est plus facile à mettre en place que la virtualisation complète.

2.8.5.2.2 Inconvénients

- Il y'a une surcharge au niveau de L'OS invité.
- L'OS invité doit être modifié et il ne peut pas s'exécuté sans virtualisation.
- Les machines virtuelles ne peuvent pas être migrer vers d'autres serveurs (hôtes).

2.8.5.3 Virtualisation assisté par le matériel

Dans les deux approches précédentes, il y'a un surcout supplémentaire de la translation binaire ou la modification de l'OS invité pour réaliser la virtualisation. Mais dans cette approche, Les fournisseurs de matériel eux-mêmes offrent la prise en charge de la virtualisation, ce qui élimine les surcharges des approches précédentes. Ces vendeurs, comme *Intel* et *AMD*, ont ajouté une extension matérielle à leur processeur x86 pour prendre en charge la virtualisation.

Par exemple, *Intel* a lancé sa technologie de virtualisation Intel (*VT-x*) et *AMD* a lancé *AMD-v* pour simplifier la technique de virtualisation. Comme indiqué précédemment, la technique de virtualisation assistée par matériel supprime la traduction binaire et la paravirtualisation. Ici, les demandes du système d'exploitation communiquent directement avec l'hyperviseur sans aucune traduction.

2.8.5.3.1 Avantages

- Réduction de la surcharge de la translation binaire dans les virtualisations complètes.
- Ne modifie pas l'OS invité

2.8.5.3.2 Inconvénients

- Seuls les processeurs de nouvelle génération ont ces capacités.
- Un plus grand nombre d'interruptions de VM entraîne une surcharge élevée du processeur, une évolutivité limitée et une efficacité moindre dans la consolidation des serveurs.
- Les performances sont inférieures à la paravirtualisation.

2.8.5.4 Avantage de la virtualisation

Pour un fournisseur des services IT, l'utilisation des techniques de virtualisation a un nombre d'avantages :

- Consolidation des ressources : la possibilité de faire fonctionner plusieurs systèmes en parallèle sur un serveur, ce qui se traduit par la réduction du nombre de serveurs et la quantité de matériel informatique nécessaire dans le centre de données.
- Utilisation des ressources : la virtualisation permet d'accroître le taux d'utilisation des ressources (60 à 80%) contre 10 à 20% en non virtualisation.
- Gestion automatique : Les machines virtuelles peuvent être créées et configurées automatiquement selon les besoins.
- Consommation d'énergie : la consolidation des ressources diminue le nombre de composants physiques ce qui implique une réduction des dépenses énergétiques.

La virtualisation facilite l'installation, la sauvegarde, le déploiement des systèmes et des applications et simplifie la migration des applications sur de nouveau serveurs.

- Sécurisation et/ou isolation : La virtualisation Crée des environnements sécurisés et isolés.
- Comportement dynamique : En cas de goulots d'étranglement (surcharge), une machine virtuelle peut utiliser des ressources supplémentaires.
- Diminuer la surface au sol : Chaque espace carré de centre de données est rare et couteux. Avec la virtualisation, la même performance peut être obtenue sur une plus petite surface.

2.9 Migration des machines virtuelles VMs

La migration des machines virtuelles consiste à déplacer l'état d'une VM d'un serveur hôte physique à un autre. Elle inclut :

- La mémoire volatile et non-volatile (représente la majeure partie des données à transférer).
- L'état des CPUs virtuels (vCPUs).
- Les paramètres des interfaces réseaux et le détail des connexions actives.

On appelle les deux dernières parties les données annexes (états VCPUs + Paramètres VNICs). Les VMs peuvent être migrer/déplacer pour une des raisons suivantes :

- Équilibrer les charges des serveurs.
- Serveur hôte défectueux ou nécessite une maintenance.
- Améliorer les performances des applications.
- Pour des raison économie par exemple réduction de consommation énergétique.

2.9.1 Techniques de migration

On distingue deux types de migration des machines virtuelles

1. Migration à froid (STOP and COPY migration).
2. Migration à chaud (Livre migration).

2.9.1.1 Migration à froid (STOP and COPY migration)

La migration à froid des VMs est la stratégie la plus basique, elle consiste à *(i)* Mettre la VM hors tension afin de copie son état sur le serveur hôte cible. *(ii)* Une fois que la mémoire soit transférée entièrement la VM reprend son activité dans le même état lorsqu'elle s'est arrêtée. *(iii)* Le temps total de la migration est égal au temps d'arrêt de la machine virtuelle.

2.9.1.1.1 Avantages

- ❖ Facile à appliquer.
- ❖ Aucune faute lors de la migration de la mémoire.
- ❖ La machine une fois migrée pourra reprendre son activité dans le même état que lorsqu'elle s'est arrêtée.

2.9.1.1.2 Inconvénients

- ❖ Les services fonctionnant ne seront plus disponibles durant toute la durée de la migration.
- ❖ Ralentissement sur le réseau du cloud durant le transfert de la mémoire entièrement.

2.9.1.2 Migration à chaud (LIVE migration)

La migration à chaud est une stratégie utilisée pour pallier les problèmes de migration à froid. Elle permet à un hyperviseur *(i)* de transférer une machine virtuelle en cours d'exécution d'un hôte à un autre sans interruption de service. Elle assure *(ii)* une durée d'interruption minimale de la machine virtuelle.

Il existe deux principale approches de live migration :

1. Migration pré-copie.
2. Migration post-copie.

2.9.1.2.1 Migration pré-copie

L'algorithme de pré-copie est l'implémentation la plus répandue de la migration à chaud. Il consiste principalement à transmettre l'ensemble des pages mémoire de la VM vers l'hôte de destination tout en garantissant une faible durée d'interruption (voir figure 35). L'algorithme de pré-copie est un processus de transmission itératif. Il consiste à :

1. Envoyer l'ensemble des pages mémoire utilisées vers le serveur de destination.
2. Les pages mémoire modifiées durant chaque transfert sont envoyés lors de la prochaine itération.

3. Atteindre la phase arrêt et copie (nombre de pages restante inférieur à un certain seuil).
4. Mettre en pause la VM sur la machine source.
5. Transférer les dernières pages modifiées.
6. Activer la VM dans la machine destination.

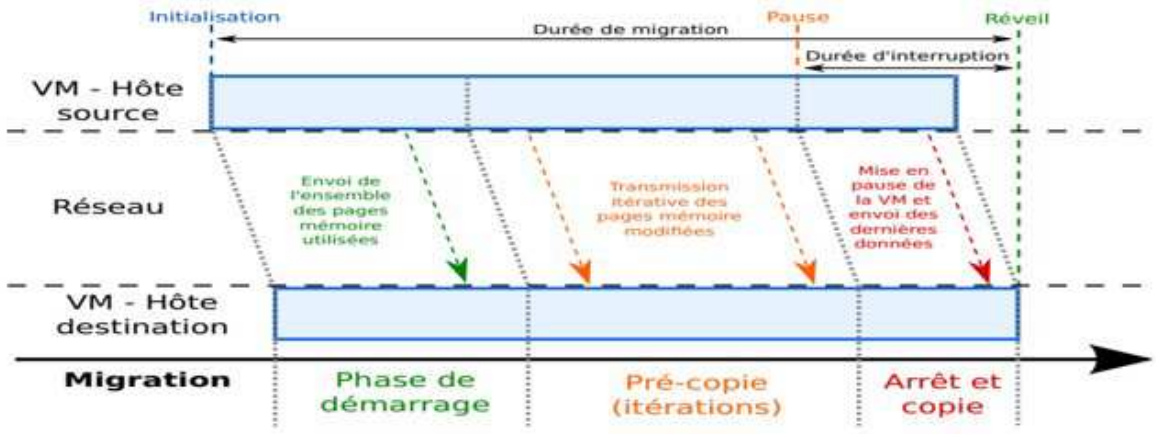


Figure 2:17: Processus live migration pré-copie, source [17]

2.9.1.2.1.1 Avantages

— Assure un temps d'interruption faible et borné ce qui implique temps de disponibilité fiable.

2.9.1.2.1.2 Inconvénients

— La durée d'interruption maximale est trop faible pour pouvoir envoyer les dernières pages modifiées au serveur destination ⇒ la migration peut durer indéfiniment.

2.9.1.3 Migration post-copie

L'algorithme de post-copie débute par l'arrêt immédiat de la machine virtuelle sur le serveur source puis transfère uniquement les données annexe avant d'activer laVM sur le serveur de destination (voir figure 2:18).

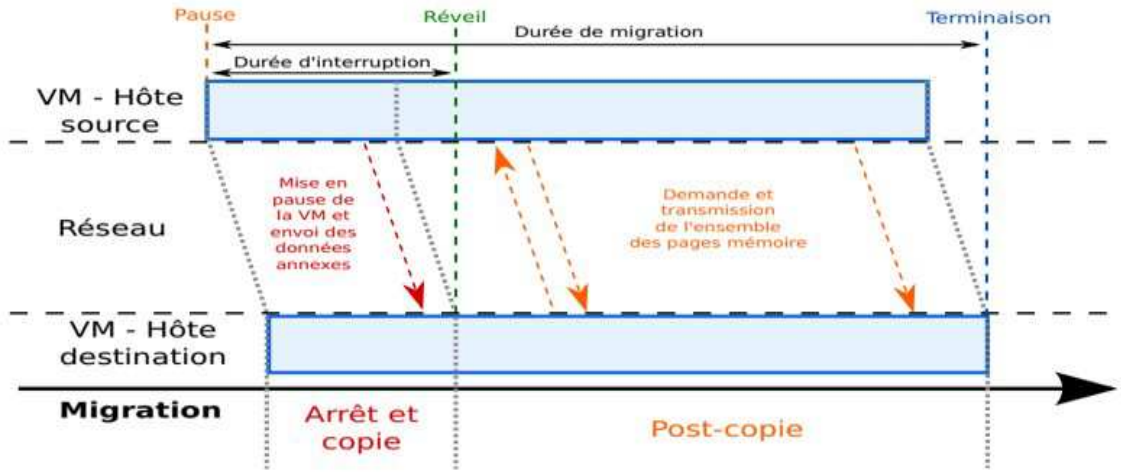


Figure 2:18: Processus de migration post-copie [18]

L'algorithme de migration post-copie consiste à:

1. L'arrêt de la VM sur l'hôte source.
2. Transfert des pages annexes et l'état d'exécution de la VM vers le serveur destination.
3. L'activation de la VM dans le serveur destination.
4. Générer les défauts de pages mémoires.
5. Renvoyer les pages mémoires modifiée.
6. Répéter le 4 et 5 jusqu'à l'arrivé complet de toutes les pages mémoires.

Les techniques utilisées dans la migration post-copie

1. La récupération à la demande (Demand Paging) : le serveur de destination demande la récupération des pages manquantes à l'hôte source lorsque la VM en a besoin.
2. La poussé active (Active-push) : cette méthode est utilisée pour transférer de façon continu les pages mémoire manquantes vers le serveur de destination. Elle envoie en priorité les pages mémoire ayant une dépendance résiduelle avec les pages déjà demandées.
3. La pré-pagination : Cette méthode permet d'améliorer la poussée active en prédisant les prochaines pages mémoire qui ont le plus de chances d'être accédées rapidement par la VM.

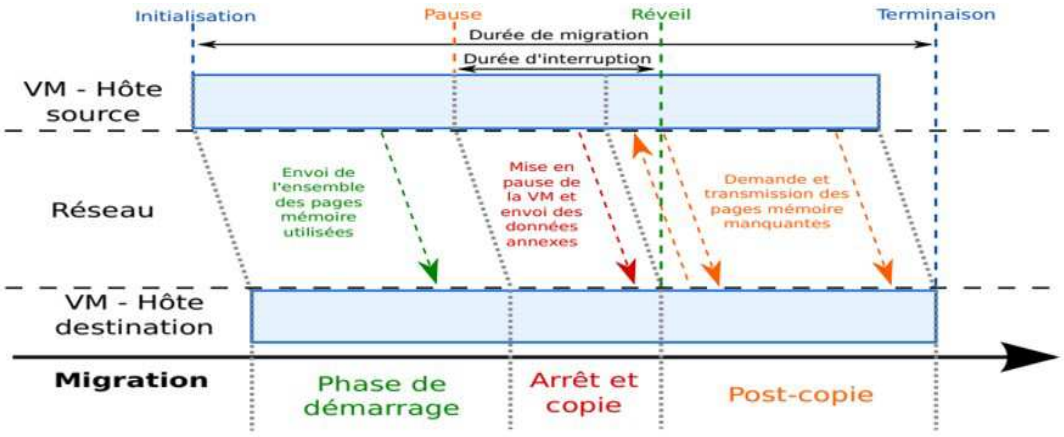


Figure 2:19: Processus de post-copie hybride, [19]

2.9.1.3.1 Avantages

— La post-copie permet de garantir une durée d'interruption fixe très faible quelle que soit l'activité mémoire de la VM à migrer.

2.9.1.3.2 Inconvénients

— La défaillance de l'un des deux hôtes, source ou destination, en cours de migration entraîne la perte inévitable de l'intégrité de l'état mémoire de la VM.

2.9.1.4 Migration post-copie hybride

L'algorithme effectue la première itération de pré-copie, qui revient à transférer l'ensemble des pages mémoire utilisées par la VM, avant de réveiller la VM sur l'hôte de destination. Ensuite, la phase de post-copie commence (voir figure 37).
L'algorithme de migration post-copie hybride consiste à :

1. Envoyer l'ensemble des pages mémoire utilisées vers le serveur de destination.
2. L'arrêt de la VM sur l'hôte source.
3. Transfert des pages annexes et l'état d'exécution de la VM vers le serveur destination.
4. L'activation de la VM dans le serveur destination.
5. Générer les défauts de pages mémoires.
6. Renvoyer les pages mémoires modifiées.
7. Répéter le 4 à 5 jusqu'à l'arrivé complet de toutes les pages mémoires.

2.9.1.4.1 Avantages

— La post-copie hybride permet de garantir une durée d'interruption fixe très faible quelle que soit l'activité mémoire de la VM à migrer.

— Améliorer les performances de l'algorithme post-copie.

2.10 Les avantages et inconvénients du Cloud

2.10.1 Les avantages du Cloud

Comme vous vous en doutez, si le Cloud Computing rencontre un tel succès, c'est que les avantages sont nombreux. Citons notamment :

- **L'accessibilité :** Les applications et services que vous utilisez dans le Cloud sont accessibles où que vous soyez à partir du moment où vous disposez d'un terminal et d'une connexion internet.
- **L'intérêt économique** : plus besoin d'investir dans l'achat de logiciels ou plateformes coûteuses, le coût d'acquisition est transformé en loyer ce qui permet à l'entreprise de convertir ses dépenses d'investissement (Capex) en dépenses d'exploitation (Opex). Et surtout, ce loyer s'adapte au fur et à mesure des besoins de l'entreprise, à la hausse comme à la baisse.
- **La facilité d'administration** : avec un service en cloud, la responsabilité de son fonctionnement repose sur l'hébergeur. L'entreprise n'a plus besoin de se soucier de l'hébergement du service, de sa maintenance logicielle, des mises à jour, de la sécurité du service, etc. L'entreprise peut ainsi bénéficier de conditions d'exploitation optimales (garanties de fonctionnement 24h/24, de sauvegardes, de support etc.) qu'elle n'aurait pas forcément les moyens d'assurer en interne.
- **La souplesse** : grâce à la mutualisation du service entre de nombreux clients, le fournisseur du service cloud autorise une très grande souplesse. Ce qui rend simple et sans conséquence les évènements tels que la nécessité de monter/baisser en charge très rapidement, ou tout simplement ne plus utiliser le service.
- **Le gain de productivité** : les entreprises peuvent davantage se concentrer sur l'utilisation du service plutôt que son administration. Cela permet de libérer des

ressources humaines et de les concentrer sur le cœur de métier plutôt que sur des tâches périphériques.

2.10.2 Les Inconvénients du Cloud

Bien sûr, comme tout concept, celui du Cloud Computing n'échappe pas à quelques contraintes et inconvénients dont il est préférable d'avoir conscience avant tout choix :

- **La perte de contrôle** : c'est la contrepartie inévitable au transfert de responsabilité. Comme l'hébergeur ou éditeur du service cloud se charge de tout, vous n'avez plus aucun contrôle sur le fonctionnement du service. Vous devez donc faire confiance à l'hébergeur, ce qui implique une vigilance toute particulière. Les garanties de service apportées par l'hébergeur (et sa réputation à tenir ses engagements) sont donc déterminantes.

- **Le coût** : Il peut bien être un atout comme un inconvénient. En effet, si le cloud permet généralement de réaliser des économies, cela ne reste vrai que si l'entreprise se dote de la volonté d'une réelle maîtrise des coûts. La facilité d'ajouter de nouvelles ressources peut conduire à un véritable gaspillage. Par exemple, un développeur pourrait être tenté d'ajouter de la puissance CPU en 1 clic plutôt que passer quelques heures à optimiser son code.

- **Les risques d'enfermement propriétaire** ("vendor lock-in") : de nombreux services cloud ont la particularité, volontairement ou non, d'utiliser des technologies ou environnements non standards, rendant ainsi complexe voire impossible la migration vers un hébergeur ou éditeur concurrent. Dans les cas extrêmes, il n'est même pas possible de récupérer ses propres données, ce qui oblige l'utilisateur du service à repartir de zéro s'il souhaite changer de fournisseur

2.11 Conclusion

Pour une compréhension claire du cloud computing, il y a certains concepts fondamentaux à connaître. Ce chapitre commence par l'historique puis définit de manière claire les raisons de son adoption en passant par ses caractéristiques. Il poursuit avec les modèles de déploiement et les types de services. Les modèles de déploiement sont basés sur plusieurs propriétés telles que la taille, l'emplacement et

la complexité. Il existe quatre types de modèles de déploiement, la description de chaque modèle avec ses caractéristiques et son adéquation à différents types de besoins est abordée dans ce chapitre. Les modèles de service décrivent le type de service livré par le Fournisseur du service (IaaS, PaaS et SaaS). Le chapitre a présenté les caractéristiques, l'adéquation, les avantages et les inconvénients de chaque type de service. Il termine avec les exigences du cloud, ses différents acteurs (écosystème), les technologies utilisées dans le cloud, la migration vers le cloud ainsi que les avantages et inconvénients de celui-ci. Nous allons passer maintenant à l'étude de la technologie de base à utiliser pour le déploiement de notre cloud.

3. Chapitre III : Choix Technologique

3.1 Introduction

Dans le domaine du Cloud Computing, on dispose de deux grandes catégories : des solutions propriétaires et d'autres open sources. Dans notre projet, on ne s'intéresse qu'à des solutions de cloud privé open sources que nous présenterons dans cette section.

3.2 Solutions de cloud privés libres

Il existe de multitude de solutions open sources sur le marché. Selon les besoins technologiques de notre entreprise, on s'intéresse à l'étude des solutions suivantes :

3.2.1 OpenStack

OpenStack est un ensemble de logiciels open source permettant de déployer des infrastructures de cloud computing (infrastructure en tant que service). La technologie possède une architecture modulaire composée de plusieurs projets corrélés (Nova, Swift, Glance...) qui permettent de contrôler les différentes ressources des machines virtuelles telles que la puissance de calcul, le stockage ou encore le réseau inhérent au centre de données sollicité. Le projet est porté par la Fondation OpenStack, une organisation non-commerciale qui a pour but de promouvoir le projet OpenStack ainsi que de protéger et d'aider les développeurs et toute la communauté OpenStack. De nombreuses entreprises (plus de 150) ont rejoint la fondation OpenStack. Parmi celles-ci on retrouve : Canonical, Red Hat, SUSE, eNovance, AT&T, Cisco, Dell, IBM, Yahoo!, Oracle, Orange, Cloudwatt, EMC, VMware, Intel, OVHcloud, NetApp. OpenStack est un logiciel libre distribué selon les termes de la licence Apache. En juillet 2010, Rackspace Hosting et la NASA ont lancé conjointement un nouveau projet open source dans le domaine du cloud computing sous le nom d'OpenStack. L'objectif du projet OpenStack est de permettre à toute organisation de créer et d'offrir des services de cloud computing en utilisant du matériel standard. La première version livrée par la communauté, dont le surnom est Austin, fut disponible dès octobre 2010.Il est prévu de livrer régulièrement des mises à jour logicielles à quelques mois d'intervalle. L'architecture d'OpenStack, représenté par la Figure 3.1 est modulaire fragmentée et distribuée. Elle comporte essentiellement trois projets en liaison : OpenStack Compute

(Nova), OpenStack Object Storage (Swift) et OpenStack Image Service (Glance). Ces trois composants permettent la gestion des différentes ressources des VMs : le calcul CPU, le stockage ou encore le réseau du datacenter associé. Ainsi que d'autres services : Identity, Dashboard, Block Storage, Network Service(Neutron) etc.

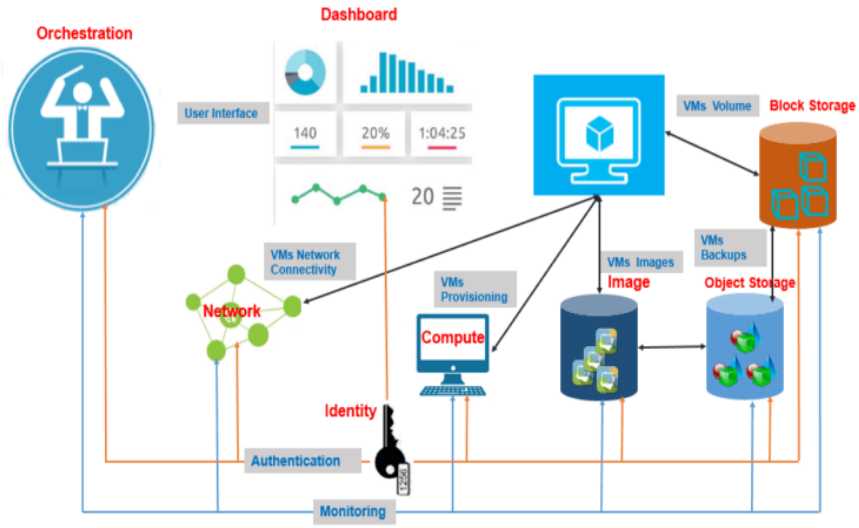


Figure 3:1: Architecture Openstack, source [22]

3.2.2 OpenNebula

OpenNebula voit le jour en 2005 à l'université Complutense de Madrid dans le cadre du projet européen open source RESERVOIR. Son objectif, dans le cadre de ce projet, est l'administration des IaaS virtualisés. Autrement dit, il fournit des services permettant de déployer et d'exécuter dans un environnement matériel virtualisé des VM. [12] Depuis sa version 5.6, OpenNebula peut gérer simultanément dans l'IaaS les hyperviseurs Xen, KVM et VMware.

OpenNebula présente l'IaaS sous forme de clusters (ensemble de machines physiques) et de VLAN (ensemble de VMs). Au moment de la création d'une VM, l'utilisateur choisit la machine physique et le réseau virtuel dans lequel il souhaite l'exécuter. L'architecture de OpenNebula est fortement centralisée autour d'une unique machine de l'IaaS nommée Frontend qui assure la coordination et la supervision de toutes les opérations d'administration.

La Figure 4.2 représente l'architecture classique de cette solution. Elle se compose de trois couches d'éléments qui sont : tools, core et drivers. La liaison entre les machines Node et Frontend se fait par le biais d'un réseau privé.

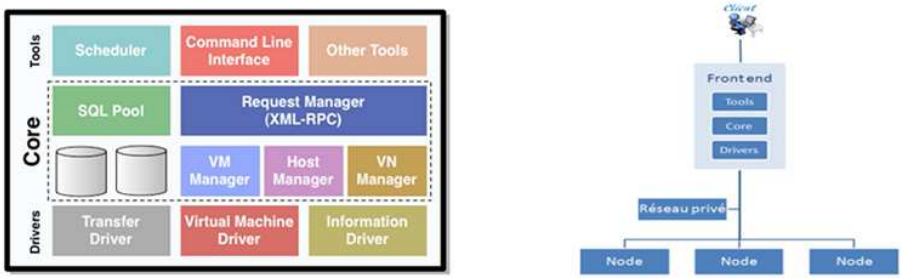


Figure 3:2: Architecture OpenNebula [23]

Détaillons chaque composant à part :

- **Interface de ligne de commande** : Une CLI pour les administrateurs et les utilisateurs d'infrastructure est fournie avec OpenNebula pour manipuler manuellement l'infrastructure virtuelle.
- **Scheduler (Planificateur)** : Le Scheduler est une entité indépendante dans l'architecture OpenNebula, de sorte qu'il peut être facilement adapté ou modifié car il est découplé du reste des composants.
- **Core (noyau)** : Le noyau consiste en un ensemble de composants pour contrôler et surveiller les machines virtuelles, les réseaux virtuels, le stockage et les hôtes.
- **Drivers (Pilotes)** : OpenNebula dispose d'un ensemble de modules enfichables pour interagir avec un middleware spécifique (par exemple, un hyperviseur de virtualisation, des services en nuage, des mécanismes de transfert de fichiers ou des services d'information), ces adaptateurs s'appellent Drivers.

3.2.3 Apache CloudStack

C'est une plateforme open source développée depuis 2008 par Cloud.com, racheté par Citrix en 2011 puis cédé en 2012 à Apache Software Foundation. Cette solution assure la création, la gestion et le déploiement des services Cloud d'infrastructures publiques, privées et hybrides (IaaS) dans les nuages. CloudStack fournit une panoplie d'outils nécessaire pour créer, déployer et gérer des applications Cloud multi-tenant à

travers des centres de données virtuels. CloudStack assume les hyperviseurs les plus courants : VMware, KVM, Xen Cloud, serveur Oracle VM et Microsoft Hyper-V. Il livre une API compatible avec AWS EC2 et S3 pour les clients désirant déployer des clouds hybrides.

L'architecture de CloudStack suit une structure hiérarchique, qui peut être déployé afin de gérer des milliers de serveurs physiques à partir d'une seule interface de gestion. La Figure 10 décrit les sept composants du CloudStack cités ci-dessous :

- Hosts : Serveurs fournissant des services ;
- Primary Storage : pour le stockage primaire de machines virtuelles ;
- Cluster : association d'un ensemble d'hôte et leur stockage ;
- Pod : Collection de clusters ;
- Network : Réseau logique associé aux offres de services ;
- Secondary Storage : Stockage secondaire d'ISO, de modèles et de captures instantanées;
- Zone : Collection de pods, offres de réseau et stockage secondaire ;
- Management Server Farm : Batterie de serveurs de gestion responsable de toutes les tâches de gestion et d'approvisionnement.

Le modèle de déploiement de CloudStack couvre les composants de base de CloudStack, à l'aide duquel CloudStack fournit toutes les fonctionnalités. Il couvre également la séparation logique des ressources pour aider à mieux les gérer.

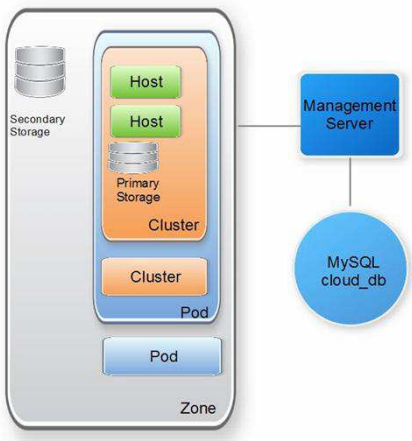


Figure 3:3:Composition architecturale de CloudStack, source [24]

3.2.4 Eucalyptus

Eucalyptus est l'acronyme de « Elastic Utility Computing Architecture for Linking Your Programs to Useful Systems » : Architecture pour lier vos programmes à des systèmes utiles. C'est une plateforme développée à partir de 2007.

En mars 2012, l'entreprise d'Eucalyptus Systems a déclaré une convenance explicite avec Amazon Web Services (AWS), aidant les administrateurs de la solution Eucalyptus d'affecter des instances entre le Cloud privé d'Eucalyptus et celui d'Amazon Elastic Compute Cloud (EC2) afin de construire un Cloud hybride. Ce couplage aide les admin Eucalyptus de collaborer avec les équipes d'Amazon dans le but de déployer des fonctionnalités AWS assimilables.

En tant que produit IaaS (Infrastructure as a Service), Eucalyptus permet aux utilisateurs de provisionner ses ressources de calcul et de stockage à la demande [2]. Ils peuvent augmenter ou diminuer de manière dynamique selon les charges de travail de l'application. Une configuration de Cloud fondée sur Eucalyptus intègre les cinq types de composants principaux suivant :

- **Contrôleur de Nœuds (NC)** : C'est le composant de base pour les nœuds. Il analyse le déroulement d'exécution des instances sur un nœud.
- **Cluster Controller (CC)** : Ce contrôleur couvre le déploiement et la gestion d'un ensemble de NCs. Il contribue aussi à la gestion de la mise en œuvre d'interfaçage réseau entre les VMs et les différents nœuds.
- **Contrôleur de stockage Walrus (WS3)** : Gère l'accès au service de stockage. Il constitue un service de stockage de fichiers constant et simple en utilisant des API SOAP compatible avec les API S3.
- **Contrôleur de stockage (SC) :** Ce composant fonctionne avec Warlus pour fournir un service de stockage persistant pour les instances. Il est similaire au service Elastic Block Storage (EBS) d'Amazon.
- **Contrôleur du Cloud (CLC)** : C'est un programme Java qui sert de Front-end l'infrastructure. Il offre, d'un côté une interface de gestion et de contrôle conforme aux services EC2/S3, une véritable boîte à outils complète, et de

l'autre côté, permet d'interagir avec les autres composants de l'infrastructure.

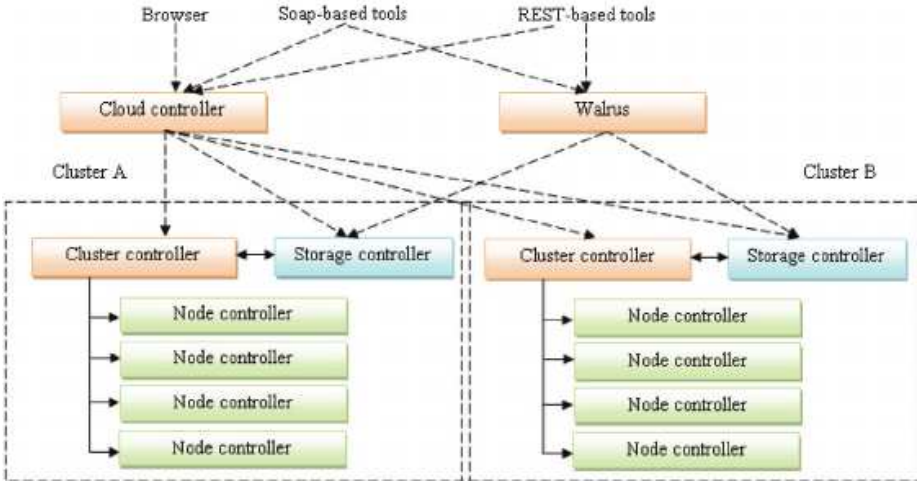


Figure 3:4: Architecture Eucalyptus, source [25]

3.3 Etude Comparative

Comme mentionné ci-dessus, il y a plusieurs projets et produits proposant le Cloud privé pour l'entreprise, tous ayant leurs points forts et leur discours commercial. Il est donc important de ne pas effectuer le mauvais choix. Comme pour toute décision professionnelle, l'étude comparative de différents produits est un élément clé pour décider de la solution de Cloud privé à adopter. La sélection de notre meilleure plate-forme de gestion cloud pour l'environnement souhaité dépend de nombreux éléments et de toutes les exigences de l'agence notamment :

- Open source
- Sécurisée
- Performante
- Modulaire et innovante
- Pouvant s'adapter à tous types d'infrastructures existantes
- Opérationnelle à toutes les tailles d'entreprises
- Extensible
- Large communauté

Dans le tableau ci-dessous, nous avons essayé de bien raffiner ces critères de comparaison en vue de mettre en avant les besoins de l'ADIE en termes de déploiement tous en comparant les quatre plates-formes cloud présentés précédemment en termes techniques et non techniques.

Tableau 3:1: Comparaison des différentes solutions de Cloud Privé

	OpenStack	**Eucalyptus**	**OpenNubula**	**CloudStack**
Code source	Open source, Apache v2.0	Open source GPL v3.0	Open source, Apache v2.0	Open source, Apache v2.0
Développeurs / Communauté	Rackspace, NASA, Dell, Citrix, Cisco, Canonical et plus de 150 autres organisations et 1800 contributeurs	- Apparu au début dans l'université Santa Barbara de l'université de Californie,	- L'union Européenne	- Fondation Apache
But	- Créer et offrir des fonctionnalités de Cloud Computing en utilisant un logiciel open source - Fonctionnant sur du matériel standard	- Une réponse open source pour le Cloud commerciale EC2	-Un Cloud privé	-Créer des clouds privés et Hybrides
Domaine d'utilisation	- Les sociétés, les fournisseurs de services, les chercheurs et les centres de données mondiaux qui cherchent à déployer à grande échelle leurs Cloud privés ou publiques	-Les entreprises	- Les chercheurs dans le domaine du Cloud Computing et de la virtualisation	-Les entreprises

Système d'exploitation supportés	- Linux et récemment Windows, - Processeur x64, x86	- Linux (Ubuntu, Fedora, CentOS, OpenSUSE et Debian)	- Linux (Ubuntu, RedHatEnterprise, Fedora et SUSE Linux Enterprise Server)	-Linux -Windows
Langage de programmation	Python	Java, C, Python	Java, C++, Ruby	-Java -Python
Hyperviseur	Xenserver, KVM, VMware ESXI, Hyper-V, Virtuzzo	Xen, KVM, VMware	Xen, KVM, VMware	Xenserver, KVM, VMware ESXI, Hyper-V
Orientation	- Cloud public, privé et hybride	- Cloud public et privé	- Cloud privé	-Cloud public, privé et hybride
Documentation	- Excellente, site bien fourni et facile d'accès avec à la fois un wiki contenant l'essentiel et une documentation officielle disponible et très détaillée.	- Correcte, complète mais pas toujours à jour.	- Complète, documentations, références de tous les fichiers de configuration, exemples	Bonne Documentation
Sécurité	Forte : Sécurité forte reposant sur des jetons et Groupes de sécurisation de calcul	Bien : chaque contrôleur est authentifié par clé SSH et fichiers de permission pour authentifier toutes les transactions.	Neutre :Firewall, Virtuel privé Prévu pour des utilisateurs de confiance ; exigence d'un compte root sans mot de passe, configuration de la CLI stockée en texte clair.	Neutre : Protection des VMs classique offrant des liens de sécurité de base avec vlan/firewall à travers le réseau établi
Réseau	Il s'intègre avec un déploiement à petit età grand échelle. VLAN, IP public Pare-feu, Équilibreur de charge, VPN; SDN	Non prévu pour un déploiement sur un réseau complexe. IP publique-IP privée; Serveur DHCP sur le	automatisé pour un réseau simple, manuel via contextualisation sur un réseau complexe.	Il s'intègre avec un déploiement à petit et à grand échelle. VLAN, IP public

		Contrôleur cluster		

3.4 Solution Choisie

Une fois choisie, une plate-forme de cloud privé ne peut pas être remplacée facilement par une autre. L'expertise de sa mise en place, son administration, les développements engendrés nécessitent un investissement conséquent. La pérennité de cet engagement est un critère de décision essentiel. Et OpenStack constitue, de ce point de vue, la solution libre offrant les meilleures garanties : la plupart des grands noms du secteur de l'informatique participent à ce projet. OpenNebula, Cloudstack, Eucalyptus sont d'autres solutions libres de cloud privé. Elles sont toutes éclipsées par OpenStack. Car, en quelques années, OpenStack s'est imposé comme un standard de fait. Et même si certaines de ces alternatives, comme Cloudstack, sont très abouties, il est peu probable qu'elles lui fassent un jour de l'ombre. Dans l'avenir, les interactions entre des plates-formes de cloud différentes vont avoir tendance à se multiplier. Le cloud computing est donc un domaine de l'informatique qui, à l'instar des réseaux, favorise l'émergence de technologies convergentes. En d'autres termes, OpenStack risque fort, dans quelques années, d'être au cloud computing ce qu'IP est au réseau Internet : incontournable. Dans ces conditions, les produits commerciaux comme VMware Vcloud ou Microsoft Azure ont un réel handicap, car ils tendent à emprisonner leurs clients dans l'utilisation de technologies (logiciels, protocoles, API) qui fonctionnent uniquement dans un cercle limité au bon vouloir du fournisseur. Et une fois à l'intérieur de ce cercle, difficile d'en sortir...

3.4.1 Les composants d'OpenStack

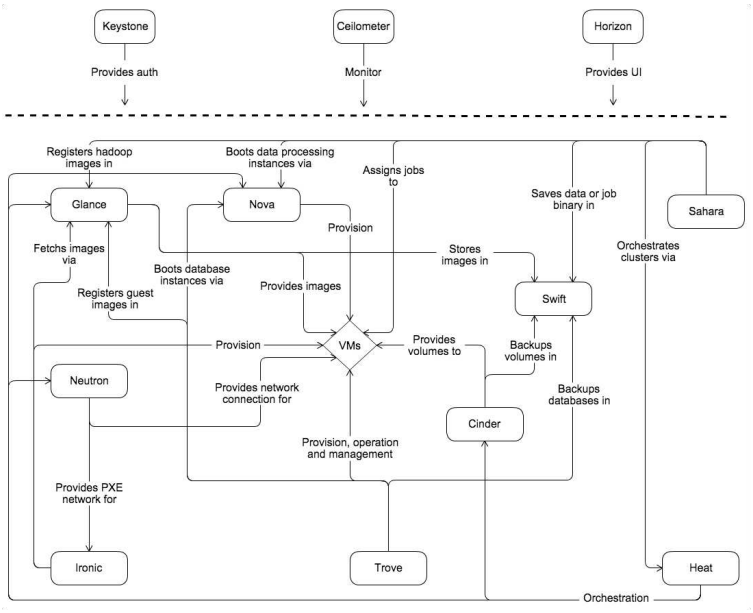


Figure 3:5: Architecture Conceptuelle d'OpenStack, source [26]

Regardons le rôle de chaque composant de l'architecture :

- **Calcul : Nova**

Nova est une des briques principales d'OpenStack. Son but est de gérer les ressources de calcul des infrastructures. Pour cela, Nova contrôle les hyperviseurs par l'intermédiaire de la **libvirt** ou directement par les API de certains hyperviseurs. Aujourd'hui l'hyperviseur le mieux supporté reste KVM, mais Nova fonctionne aussi avec Xen, ESX, et Hyper-V voire avec des gestionnaires de conteneur comme Docker et plus récemment l'hyperviseur LXD de Canonical.

L'architecture de la brique de Nova est conçue pour évoluer horizontalement en rajoutant du matériel. D'ailleurs Nova fonctionne avec du matériel non spécialisé ce qui permet de réutiliser des serveurs existants par exemple.

- **Stockage objet : Swift**

Le stockage objet d'OpenStack s'appelle Swift. C'est un système de stockage de données redondant et évolutif. Les fichiers sont écrits sur de multiples disques durs répartis sur plusieurs serveurs dans un Datacenter. Il s'assure de la réplication et de l'intégrité des données au sein du cluster. Le cluster Swift évolue horizontalement en

rajoutant simplement de nouveaux serveurs. Si un serveur ou un disque dur tombe en panne, Swift réplique son contenu depuis des nœuds actifs du cluster dans des emplacements nouveaux. Puisque toute la logique de Swift est applicative, elle permet l'utilisation de matériel peu coûteux et non spécialisé.

En août 2009, c'est Rackspace qui a commencé le développement de Swift, en remplacement de leur ancien produit nommé Cloud Files. Aujourd'hui c'est la société SwiftStack qui mène le développement de Swift avec la communauté.

- **Stockage bloc : Cinder**

Le service de stockage en mode bloc d'OpenStack s'appelle Cinder. Il fournit des périphériques persistants de type bloc aux instances OpenStack. Il gère les opérations de création, d'attachement et de détachement de ces périphériques sur les serveurs. En plus du stockage local sur le serveur, Cinder peut utiliser de multiples plateformes de stockage tel que Ceph, EMC (ScaleIO, VMAX et VNX), GlusterFS, Hitachi Data Systems, IBM Storage (Storwize family, SAN Volume Controller, XIV Storage System, et GPFS), NetApp, HP (StoreVirtual et 3PAR) et bien d'autres.

Le stockage en mode bloc est utilisé pour des scénarios performants comme celui du stockage de base de données, mais aussi pour fournir au serveur un accès bas niveau au périphérique de stockage. Cinder gère aussi la création d'instantanés (snapshots), très utile pour sauvegarder des données contenues dans les périphériques de type bloc. Les instantanés peuvent être restaurés ou utilisés pour créer de nouveaux volumes.

- **Le réseau : Neutron**

Le service Neutron d'Openstack (anciennement Quantum) permet de gérer et manipuler les réseaux et l'adressage IP au sein d'OpenStack. Grâce à Neutron, les utilisateurs peuvent créer leurs propres réseaux, contrôler le trafic à travers des groupes de sécurité (security groups) et connecter leurs instances à un ou plusieurs réseaux. Neutron gère aussi l'adressage IP des instances en leur assignant des adresses IP statiques ou par l'intermédiaire du service DHCP. Il fournit aussi un service d'adresse IP flottante que l'on peut assigner aux instances afin d'assurer une

connectivité depuis Internet. Ces adresses IP flottantes peuvent être réassignées à d'autres instances en cas de maintenance ou de défaillance de l'instance originelle.

Neutron fournit différents types de déploiement réseau en fonction de l'infrastructure cible. Les types de réseaux les plus déployés sont les réseaux plats (flat network), les réseaux à VLAN, VXLAN ou à tunnel GRE. Neutron gère ses déploiements grâce à des modules complémentaires qui lui permettent de communiquer avec des équipements ou logiciels de gestions réseau. Les plug-ins les plus utilisés sont OpenVswitch, ML2, LinuxBridge, mais aussi Cisco Nexus, Juniper OpenContrail et d'autres.

Dans son architecture, Neutron a été construit en suivant la philosophie des réseaux de nouvelle génération dite SDN. Bien qu'il ne le gère pas lui-même, certains modules tirent parti des fonctionnalités SDN des équipements qu'ils contrôlent. Lors de son utilisation avec OpenVswitch par exemple, Neutron utilise une combinaison de règles Iptables et OpenFlow pour gérer le trafic vers les instances.

- **Tableau de bord : Horizon**

OpenStack fournit un tableau de bord qui s'appelle Horizon. Il s'agit d'une application web qui permet aux utilisateurs et aux administrateurs de gérer leurs clouds à travers une interface graphique. Comme toutes les briques d'OpenStack cette application est libre et il n'est donc pas rare de voir des versions modifiées par les fournisseurs de cloud computing ou par d'autres sociétés commerciales ne serait-ce que pour y faire apparaître leur nom et logo, mais aussi pour y intégrer leurs systèmes de métrologie ou de facturation par exemple. Cette application est écrite en python et notamment grâce aux frameworks de développement web tels que Django et elle tire parti des API REST fournies par les autres composants d'OpenStack comme Nova, Cinder ou Neutron.

- **Service d'identité : Keystone**

Le service d'identité d'OpenStack s'appelle Keystone. Il fournit un annuaire central contenant la liste des services et la liste des utilisateurs d'Openstack ainsi que leurs rôles et autorisations. Au sein d'Openstack tous les services et tous les utilisateurs utilisent Keystone afin de s'authentifier les uns avec les autres. Keystone peut

s'interfacer avec d'autres services d'annuaire comme LDAP. Il supporte plusieurs formats d'authentification comme les mots de passe et autres.

- **Service d'image : Glance**

Le service d'image d'OpenStack s'appelle Glance. Il permet la découverte, l'envoi et la distribution d'image disque vers les instances. Les images stockées font office de modèle de disque. Le service glance permet aussi de stocker des sauvegardes de ces disques. Glance peut stocker ces images disques de plusieurs façons : dans un dossier sur serveur, mais aussi à travers le service de stockage objet d'OpenStack ou dans des stockages décentralisés comme Ceph. Glance ne stocke pas seulement des images, mais aussi des informations sur celles-ci, les métadonnées. Ces métadonnées sont par exemple le format du disque (comme QCOW2 ou RAW) ou les conteneurs de celles-ci (OVF par exemple).

- **Télémétrie : Ceilometer**

Le service de télémétrie d'OpenStack s'appelle Ceilometer. Il permet de collecter différentes métriques sur l'utilisation du cloud. Par exemple il permet de récolter le nombre d'instances lancé dans un projet et depuis combien de temps. Ces métriques peuvent être utilisées pour fournir des informations nécessaires à un système de facturation par exemple. Ces métriques sont aussi utilisées dans les applications ou par d'autres composants d'Openstack pour définir des actions en fonction de certains seuils comme avec le composant d'orchestration.

- **Orchestration : Heat**

Heat est le composant d'orchestration d'Openstack. Il permet de décrire une infrastructure sous forme de modèles. Dans Heat, ces modèles sont appelés des stack. Heat consomme ensuite ces modèles pour aller déployer l'infrastructure décrite sur Openstack. Il peut aussi utiliser les métriques fournies par Ceilometer pour décider de créer des instances supplémentaires en fonction de la charge d'une application par exemple.

- **Service de base de données : Trove**

Trove est le service qui permet d'installer et de gérer facilement des instances de base de données relationnelle et NoSQL au sein d'OpenStack. À ce jour les services de base de données supportés sont les suivants : MySQL, Redis, PostgreSQL, Mongodb, Cassandra, Couchbase et Percona.

- **Traitement des données : Sahara**

Sahara a pour but de fournir aux utilisateurs les moyens simples de provisionner des clusters de Hadoop en spécifiant plusieurs paramètres comme la version, la topologie du cluster ou d'autres. Après avoir rempli ces paramètres, Sahara déploie le cluster en quelques minutes. Sahara fournit aussi les moyens d'évolution du cluster en rajoutant des nœuds à la demande.

Il existe d'autres services comme :

- Ironic : service de Bare Metal provisioning
- Manila : service de gestion des systèmes de fichier partagés
- Zaqar : service de middleware à la demande
- Designate : service de gestion des DNS
- Barbican : service de gestion des clés et secrets
- Magnum : déploiement de clusters Kubernetes, Docker Swarm en tant que service
- Zun : service de gestion des conteneurs

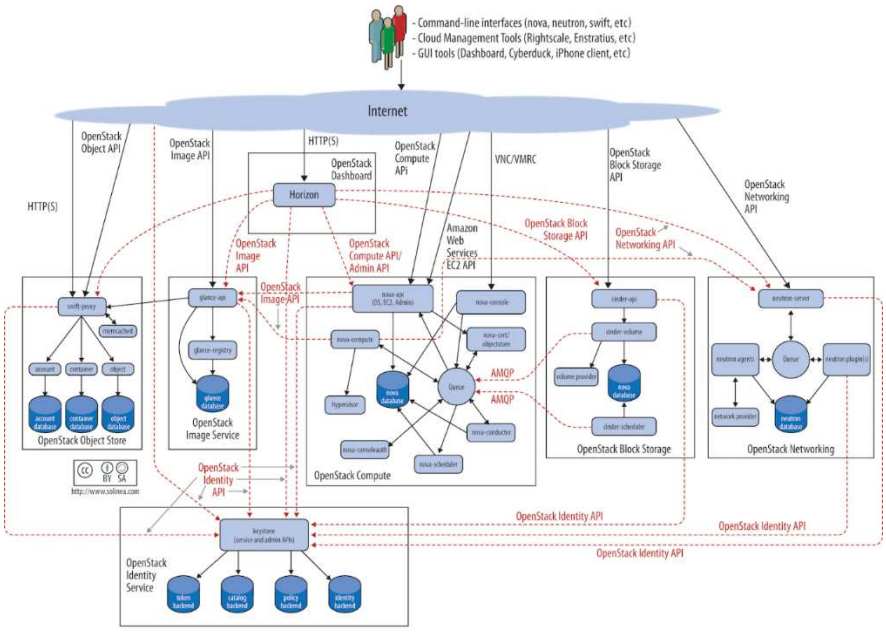


Figure 3:6: Architecture Fonctionnelle Openstack [27]

4. Chapitre IV : Conception de la solution

4.1 Introduction

Ce chapitre décrit la branche fonctionnelle et technique de notre solution. En effet, une première section du chapitre sera consacrée à la spécification et l'élaboration des différents cas d'utilisations. Donc, elle sera orientée à énoncer les besoins fonctionnels auxquels devrait répondre le cloud privé à réaliser, ainsi que les besoins non fonctionnels qu'il devrait respecter. La deuxième section de ce chapitre sera pour la partie technique. Dans laquelle le projet entame la phase technique qui permettra de décrire de manière détaillée l'architecture du cloud afin de faciliter sa réalisation.

4.2 Partie Fonctionnelle

4.2.1 Identification de besoins

Notre cloud privé s'adresse essentiellement à deux types d'utilisateurs : l'administrateur et les clients finaux du projet de l'ADIE c'est-à-dire tout utilisateur qui est susceptible de se connecter au cloud. Cette première partie, sera pour énoncer et analyser les différents besoins fonctionnels et non fonctionnels du cloud.

4.2.1.1 Besoins fonctionnels

Cette partie, est pour détailler l'ensemble des fonctionnalités que le nuage, à travers son portail, doit offrir aux utilisateurs. En effet, le système à réaliser doit répondre aux besoins fonctionnels suivants :

4.2.1.1.1 Gestion d'images

On parle d'images disques stockées par le service Glance. L'utilisateur pourrait consulter la liste des images autorisées pour les projets, les éditer. Il sera aussi possible de lancer de nouvelles instances de cette image, créer une nouvelle ou supprimer une existante.

4.2.1.1.2 Gestion d'instances

Une instance est une machine virtuelle en cours d'exécution ou dans un état connu comme « suspendue » qui peut être utilisé comme un serveur matériel. L'utilisateur pourrait consulter la liste d'instances des machines virtuelles actuelles plus quelques informations globales comme le projet auquel elles appartiennent, le serveur hôte,

l'adresse IP, la taille, le statut et les actions en cours. Il (l'utilisateur) aurait aussi les possibilités d'éditer, mettre fin, pause, redémarrer ou supprimer une instance. Aussi Il pourrait se connecter à la console VNC de l'instance ou créer une nouvelle.

4.2.1.1.3 Gestion des volumes

Le cloud permettrait à l'utilisateur de consulter la liste des volumes disques virtuels existants, la création d'un nouveau volume et la modification d'un ancien.

4.2.1.1.4 Gestion des flavors ou gabarits

Un Flavor est une configuration de matériel disponible dans un serveur. Chaque Flavor possède une combinaison unique d'espace disque et la capacité de mémoire. L'utilisateur pourrait consulter la liste des types d'instances disponibles, leurs spécifications en nombre de CPUs, mémoire, espace disque et créer des nouvelles définitions d'instance.

4.2.1.1.5 Gestion des projets

Un projet est un groupement logique des utilisateurs au sein de Nova, utilisé pour définir les limites des ressources pour ce projet et l'accès aux images des machines virtuelles. Il serait possible de consulter les projets existants et leur statut et de créer de nouveaux projets.

4.2.1.1.6 Gestion des utilisateurs

L'utilisateur aurait la possibilité de consulter la liste d'utilisateurs enregistrés, avec la possibilité d'ajouter ou d'éditer les détails mais pas d'ajouter l'utilisateur à plusieurs projets.

4.2.1.1.7 Gestion de la sécurité et de l'accès

L'utilisateur pourrait consulter les adresses IP disponibles pour connecter les instances au réseau public avec la possibilité de création de groupes de règles de pare-feu et leur interface d'édition et enfin la liste des clés SSH avec l'import ou la création de certificat.

4.2.1.2 Besoins non fonctionnels

4.2.1.2.1 Simplicité d'un service à la demande

Un utilisateur peut de manière unilatérale, immédiatement et généralement sans intervention humaine, avoir à sa disposition les ressources informatiques dont il a besoin (temps de calcul de serveurs, capacité de stockage, etc.).

4.2.1.2.2 Extrême flexibilité

Les ressources mises à disposition ont une capacité d'adaptation forte et rapide à une demande d'évolution, généralement de manière transparente pour l'utilisateur.

4.2.1.2.3 Accès « léger »

L'accès aux ressources ne nécessite pas d'équipement ou de logiciel propriétaire. Il se fait au travers d'applications facilement disponibles (parfois libres), généralement depuis un simple navigateur Internet.

4.2.1.2.4 Sûreté

Un évènement indésirable ne devrait pas se produire pendant l'accès d'une machine virtuelle aux ressources informatiques.

4.2.1.2.5 Vivacité

Une action souhaitée par une machine virtuelle arrivera nécessairement à être réalisée pour garantir la progression du programme.

4.2.2 Identification des acteurs

Cette phase a pour objectif de décrire le comportement attendu de l'application. Pour cela, l'utilisation du diagramme des cas d'utilisation qui représente un élément essentiel de la modélisation orientée objet, assure de bons résultats. Elle (cette phase) permet de modéliser les fonctionnalités de l'application du point de vue des besoins de l'utilisateur. Elle sert aussi à définir et à modéliser le produit à développer.

4.2.2.1 Les acteurs du système

Les acteurs qui manipuleront notre application sont :

- L'administrateur: IL possède les droits administratifs qui lui permettrait de contrôler tout le cloud et lui permettrait d'accéder à l'interface d'administration sur le portail « Dashboard » ainsi qu'à tous les autres projets.

- Client final « utilisateur »: C'est un collaborateur, membre de l'agence ou un client externe appartenant à l'Administration. Il peut commander une instance avec le service voulu et il n'aurait accès qu'à son (ses) projet(s).

Par la suite les principaux cas d'utilisations qui assurent toutes les tâches exécutées par le système seront mis en place. Avant cela, l'utilisateur (administrateur, client final) doit s'authentifier avec la saisie de son login, et son mot de passe.

Après vérification, si l'utilisateur est accepté, il aura accès au cloud et selon son rôle, des projets et des fonctionnalités s'activeront sinon on aura un message d'erreur.

4.2.2.2 Diagramme de cas d'utilisation Générale

La figure 4.1 présente le diagramme de cas d'utilisation générale du Cloud.

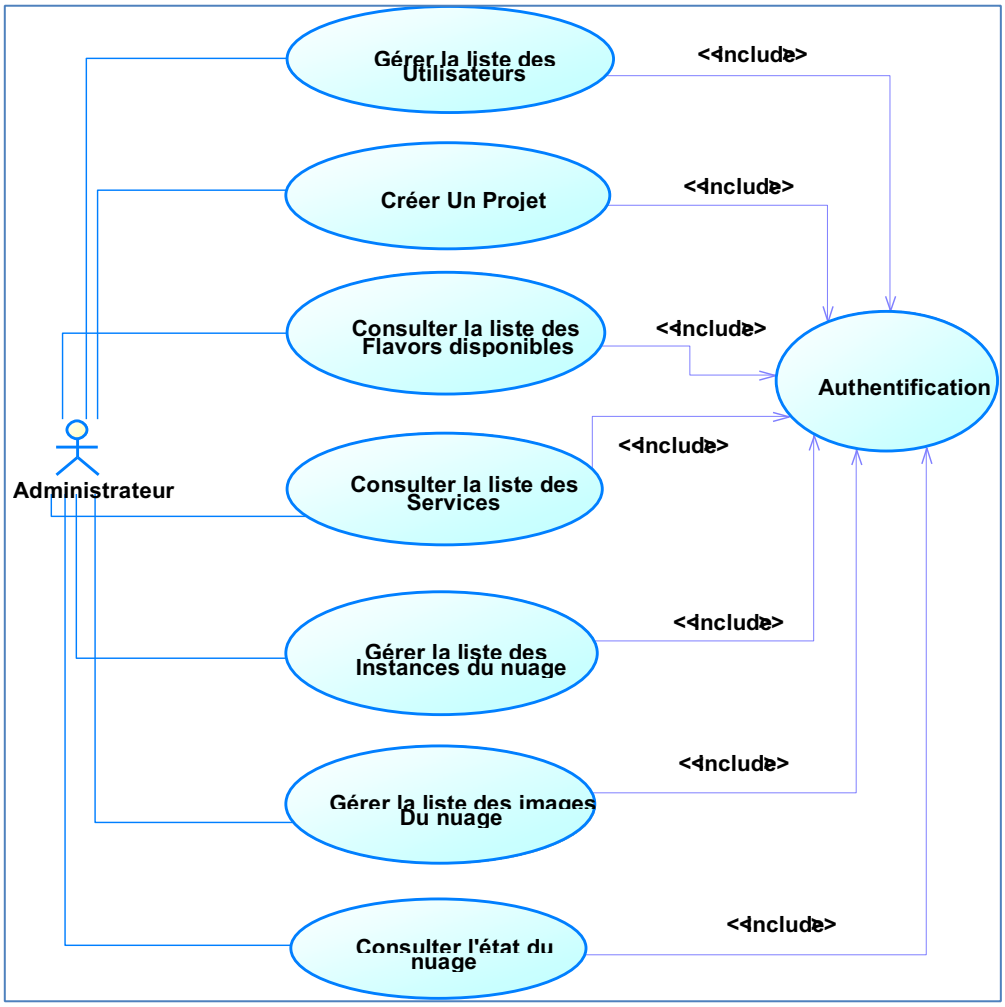


Figure 4:1: Diagramme de cas d'utilisation générale du Cloud

4.2.3 Raffinements des Cas d'Utilisations

4.2.3.1 Cas d'utilisation « Consulter l'état du nuage »

L'administrateur pourrait :

- Consulter l'usage des serveurs par projet, utilisation actuelle en nombre de CPU virtuels, RAM et Disques puis compteur en CPU et espace disque (GB) par heures.
- Générer un rapport sous format CSV.

La figure 4.2 présente le cas d'utilisation « Consulter l'état du nuage ».

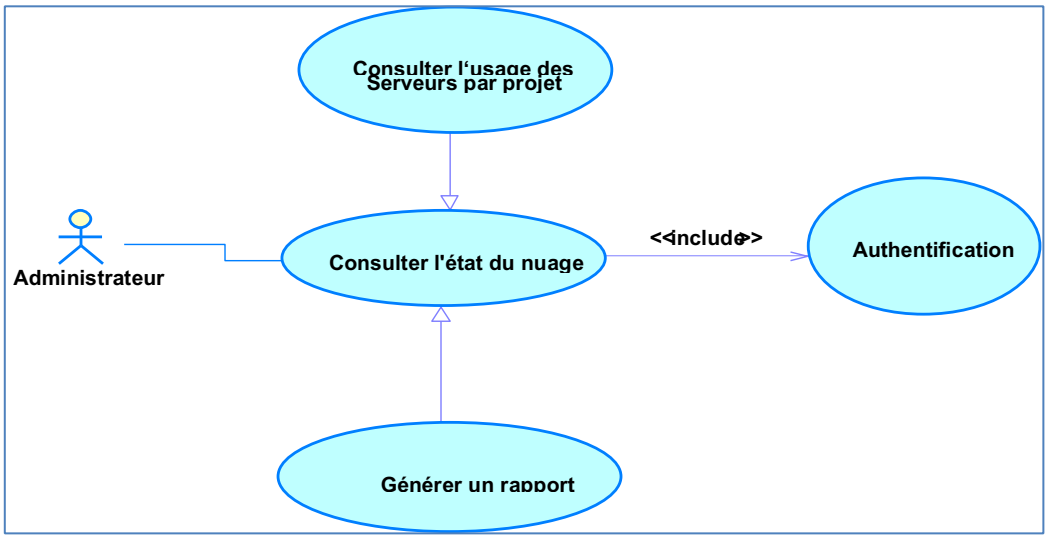


Figure 4:2: Cas d'utilisation « Consulter l'état du cloud»

4.2.3.1.1 Cas d'utilisation « Gérer la liste des instances »

L'administrateur pourrait gérer les instances existantes sur le nuage, il aurait la possibilité de :

- Consulter la liste des instances existantes et leurs détails.
- Editer les détails d'une instance.
- Mettre fin à une instance.
- La suspendre.
- La redémarrer.
- La supprimer.
- Connecter à sa console VNC.

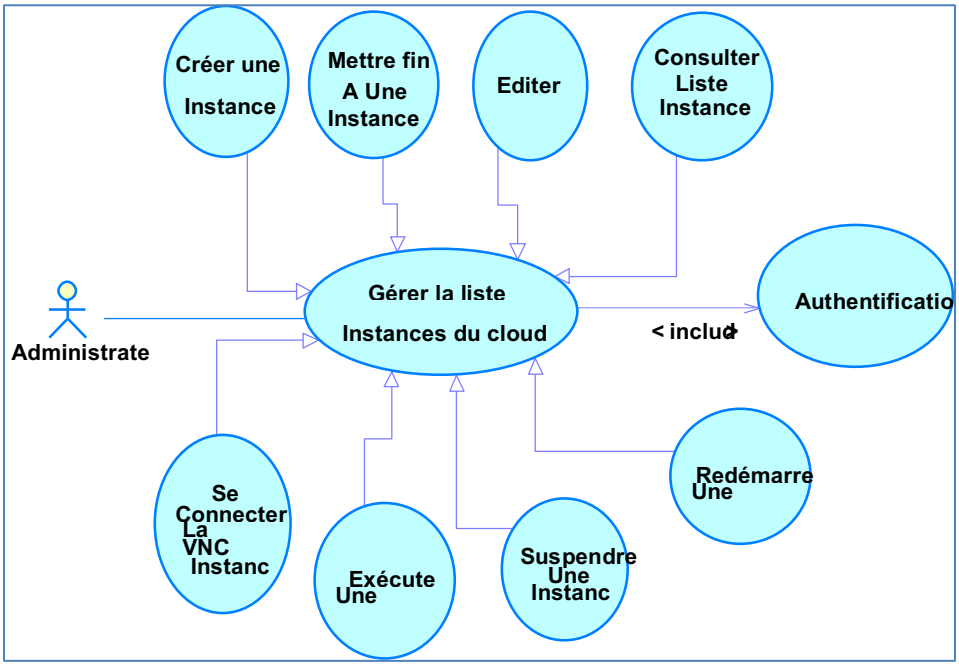


Figure 4:3: Cas d'utilisation « Gérer la liste des instances »

4.2.3.1.2 Cas d'utilisation « Consulter la liste des services »

L'administrateur pourrait :

- Consulter la liste des services (Volume, Glance, Nova, Keystone...) activés,
- Modifier l'état des services (activé / désactivé)
- Gérer les serveurs hôte et leurs statut (activé/désactivé).

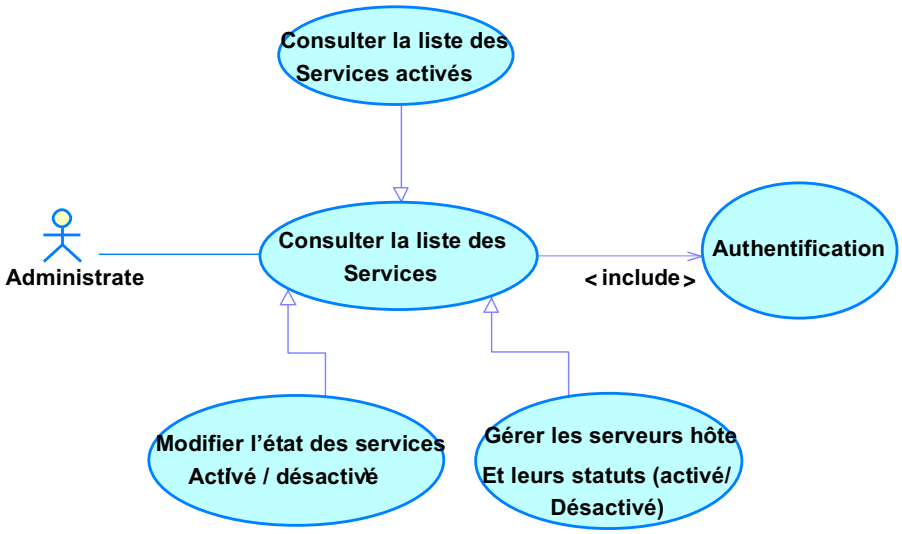


Figure 4:4: Cas d'utilisation « Consulter la liste des services »

4.2.3.1.3 Cas d'utilisation « consulter la liste des flavors disponibles »

L'administrateur pourrait:

- Consulter la liste de Flavors actuellement disponibles qui pourraient être utilisés pour lancer une instance.
- Créer des Flavors personnalisées.
- Editer des Flavors.
- Supprimer des Flavors existants.

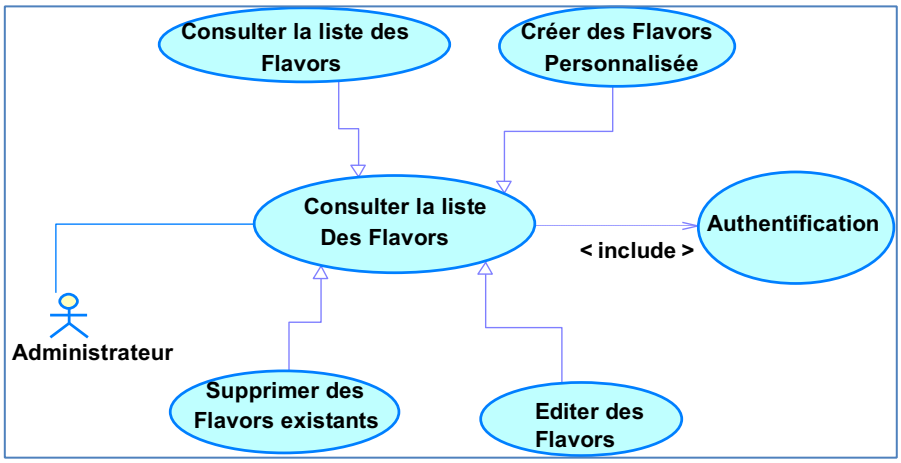


Figure 4:5: Cas d'utilisation « consulter la listes des flavors disponibles »

4.2.3.1.4 Cas d'utilisation « Gérer les images »

L'administrateur aurait la possibilité de:

- Consulter la liste des images disponibles.
- Editer les détails d'une image (nom, noyau ID, Ram, disk ID, architecture, format, public ou privé).
- Supprimer des images si elles ne sont plus nécessaires.

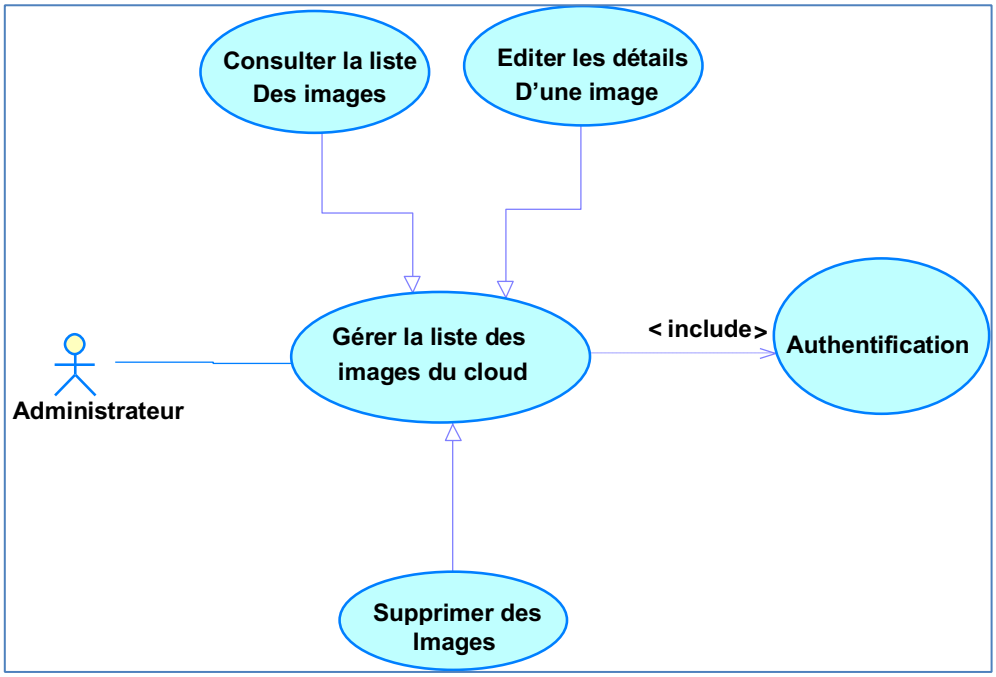


Figure 4:6: Cas d'utilisation « Gérer la liste des images du nuage »

4.2.3.1.5 Cas d'utilisation « Gérer les projets »

L'administrateur aurait la possibilité de gérer les projets existants sur le cloud, ainsi il pourrait:

- Consulter la liste des projets disponibles (locataires) qui ont été créés, leurs détails et leurs utilisations.
- Créer des nouveaux projets.
- Affecter des utilisateurs à un projet.
- Modifier les détails d'un projet.
- Supprimer un projet.

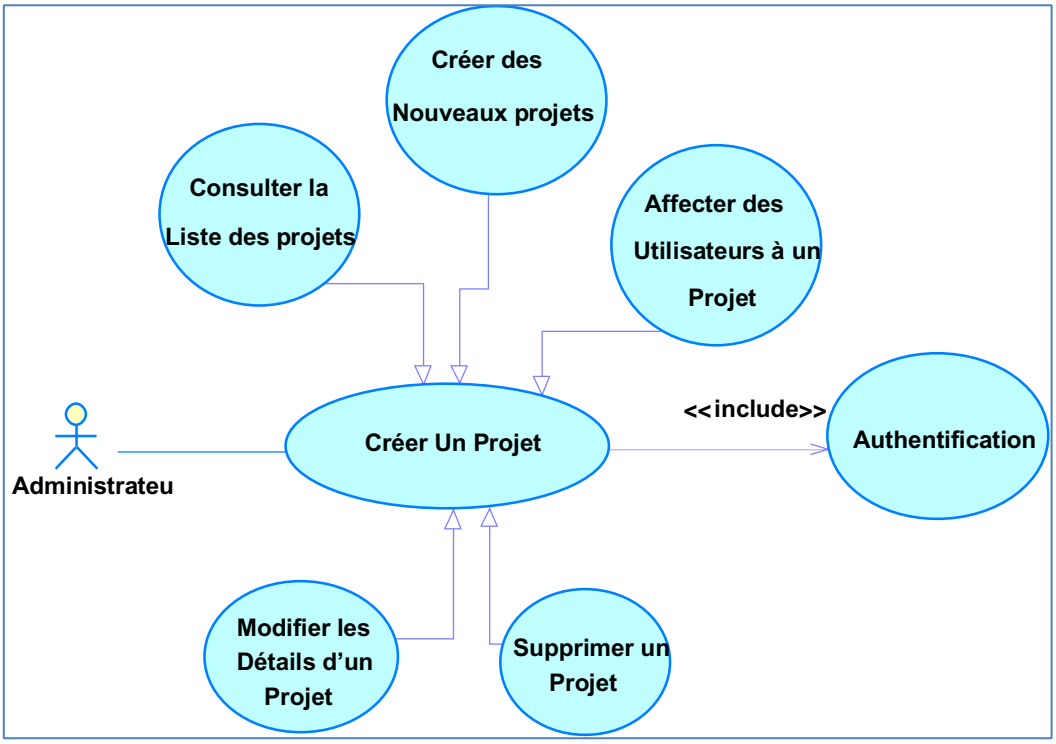


Figure 4:7: Cas d'utilisation «Créer un Projet »

4.2.3.1.6 Cas d'utilisation « Gérer les utilisateurs »

L'administrateur pourrait gérer les comptes utilisateurs existants sur le nuage, ainsi il pourrait:

- Consulter la liste des utilisateurs qui ont été créés et leurs détails.
- Créer de nouveaux utilisateurs.
- Désactiver / supprimer des utilisateurs existants.

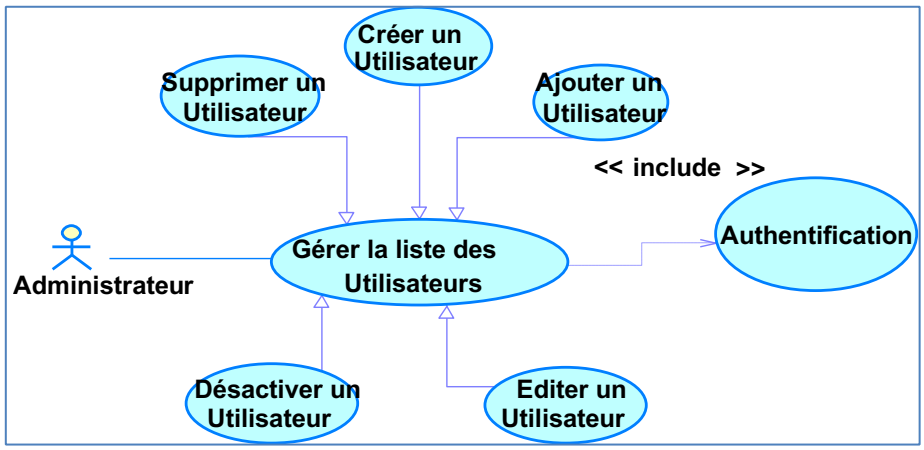


Figure 4:8: Cas d'utilisation « Gérer la liste des utilisateurs »

4.2.3.2 Diagramme de cas d'utilisation « membre d'un projet »

La figure 22 présente le diagramme de cas d'utilisation d'un membre d'un projet existant dans le cloud c'est-à-dire, le client final.

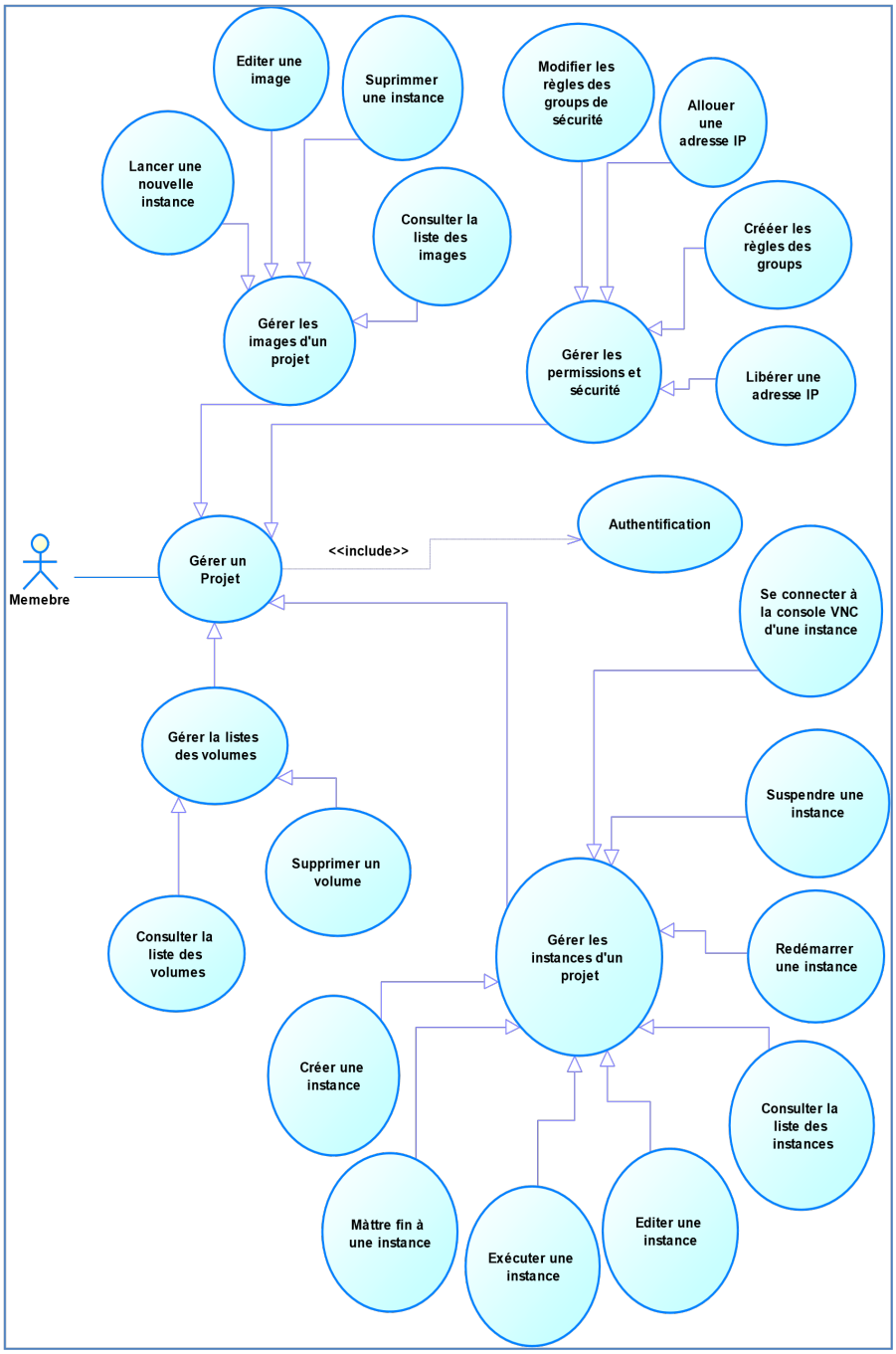


Figure 4:9: Diagramme des cas d'utilisation pour un membre d'un projet

Un membre ne pourrait consulter et gérer que les ressources des projets auxquels il appartient, ainsi il serait possible pour lui de:

4.2.3.2.1 Consulter l'état de ses projets

Un membre pourrait consulter l'usage des serveurs d'un projet sélectionné, utilisation actuelle en nombre de CPU virtuels, RAM et Disques puis compteur en CPU et espace disque (GB) par heures.

4.2.3.2.2 Gérer les instances et les volumes

Un membre pourrait consulter la liste des instances existantes et aurait la possibilité de les éditer, de créer ou de modifier des volumes disques virtuels.

4.2.3.2.3 Gérer les images et leurs instances

Un membre aurait la possibilité de consulter la liste des images autorisées pour le projet et lancer de nouvelles instances.

4.2.3.2.4 Gérer la sécurité et l'accès

Un membre pourrait consulter la liste des adresses IP disponibles pour connecter les instances au réseau public avec la possibilité de création des groupes de règles et de Pare-feu. Il aurait aussi la possibilité de consulter la liste des clés SSH et de créer de certificat.

4.2.4 Diagrammes d'activité

Les diagrammes d'activité système, permet de représenter le déclenchement d'événements en fonction des états du système et de modéliser des comportements parallèles (multithreads ou multiprocessus). Le diagramme d'activité est également utilisé pour décrire un flux de travail (workflow).

Ce paragraphe, sera consacré à la présentation de quelques diagrammes d'activités les plus significatifs.

4.2.4.1 Diagrammes d'activité globale

La figure 23 : montre le diagramme d'activité globale: les scénarios de quelques cas.

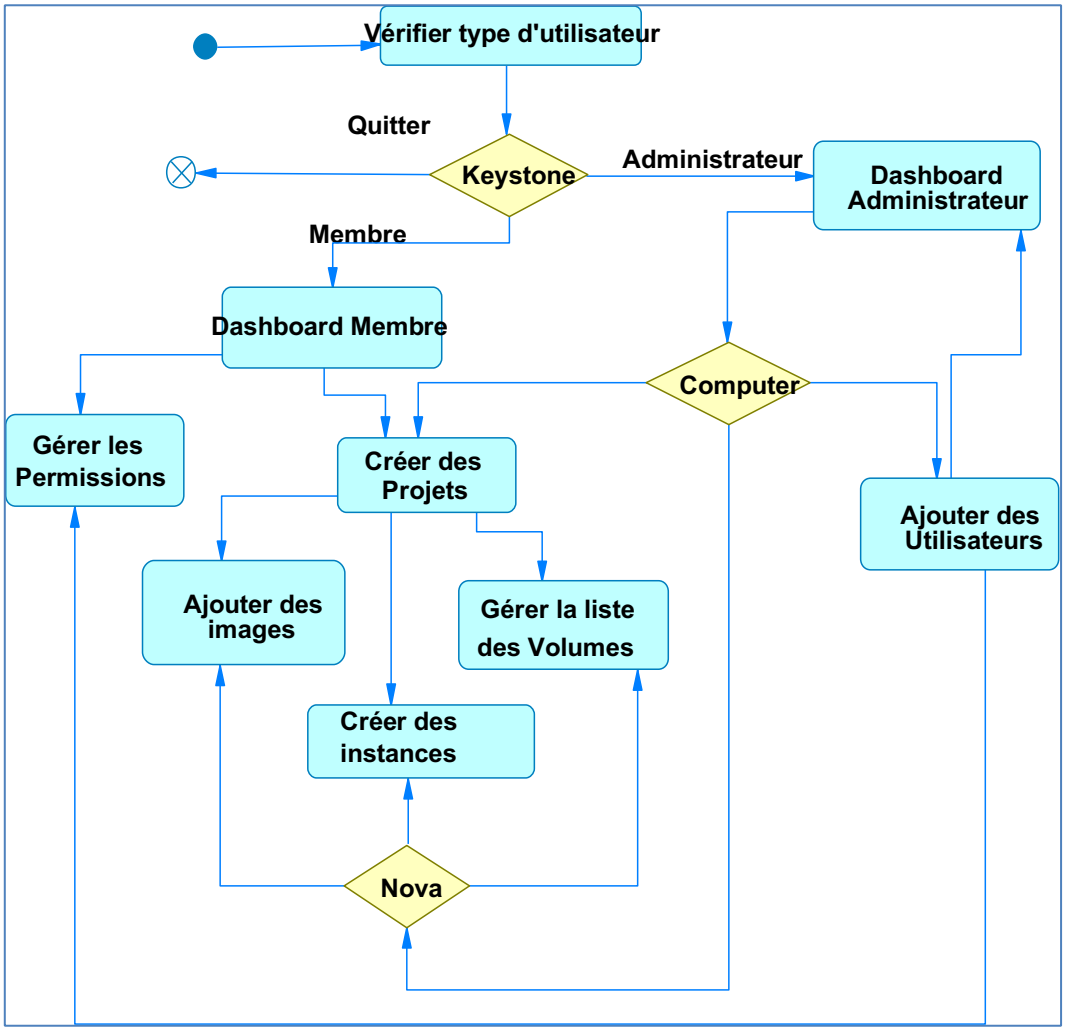


Figure 4:10: Diagramme d'activité globale

4.2.4.2 Diagramme d'activité « Créer une instance »

La figure 24 ci-dessous montre les étapes à suivre pour construire une instance « machine virtuelle »

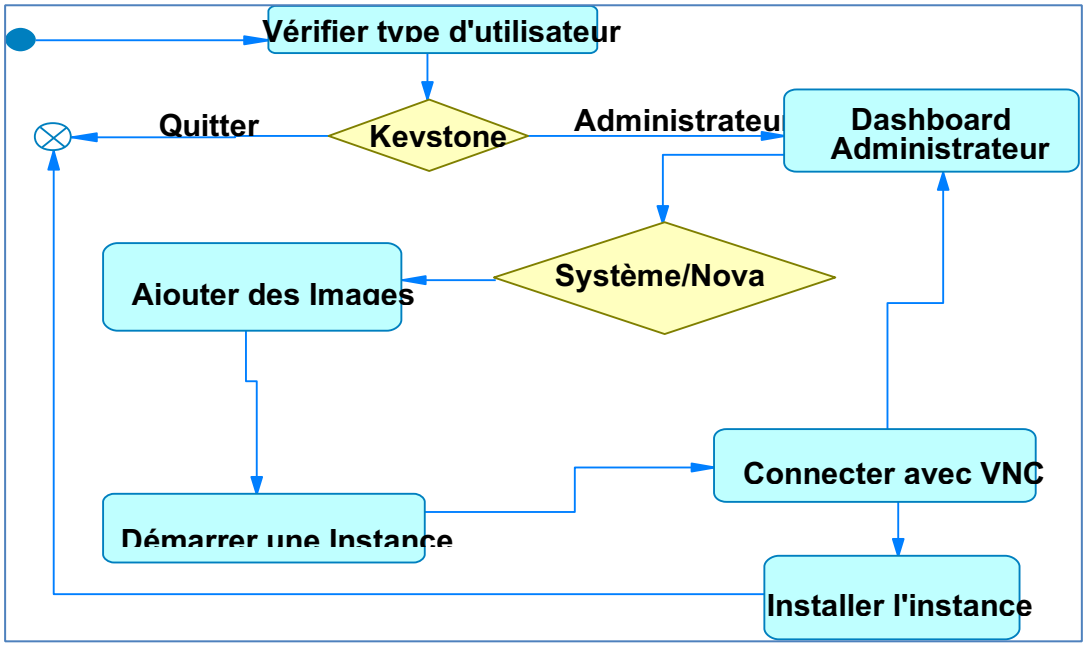


Figure 4:11: Diagramme d'activité « créer une instance »

A travers cette partie, l'exposition des besoins tant fonctionnels que non fonctionnels du projet est faite.

De plus, la présentation des diagrammes des cas d'utilisations et des diagrammes d'activités en définissant les acteurs et les détails des cas d'utilisations qui sont mis en place ainsi que quelques diagrammes de séquences détaillés seront présentés dans le chapitre suivant.

4.3 Partie Technique

4.3.1 Gestion de l'infrastructure

4.3.1.1 Ressources de calculs

Pour la gestion de notre infrastructure, dans le cadre du processus de planification, nous devons non seulement déterminer quel matériel utiliser, mais aussi si des ressources de calcul seront fournies dans un seul pool ou dans plusieurs pools ou zones de disponibilité. Nous devons également choisir l'hyperviseur à utiliser.

4.3.1.1.1 Serveurs

Le type de processeur dans un nœud de calcul est une décision très importante. Nous devons nous assurer que le Processeur prend en charge la virtualisation par le moyen de VT-x pour les puces Intel et AMD-v pour les puces AMD.

Le nombre de cœurs que le Processeur dispose, affecte également notre décision. Il est courant que les processeurs actuels aient jusqu'à 24 cœurs. En outre, si un processeur Intel prend en charge l'hyper-threading, ces 24 cœurs sont doublés à 48 cœurs. Si nous achetons un serveur qui prend en charge plusieurs processeurs, le nombre de cœurs est encore multiplié. Hyper-Threading est la mise en œuvre exclusive multithreading simultanée d'Intel utilisée pour améliorer la parallélisation de leurs processeurs. Nous envisageons d'activer Hyper-Threading pour améliorer les performances des applications multicœur. La capacité et le placement des charges de travail sont des considérations de conception clés pour les clouds. Un plan de capacité à long terme pour ces conceptions doit intégrer la croissance au fil du temps. En plus de l'hyper-threading, nous utilisons le **surabonnement** qui est une méthode pour émuler plus de capacité que ce qui peut être physiquement présent. Le taux de sursouscription de chaque hôte de calcul est de 16 :1 c'est-à-dire, chaque cœur physique du processeur peut supporter 16 cœurs virtuels. Pour la RAM, Le taux de sursouscription, c'est 1,5 :1, ce qui veut dire que la capacité de la RAM du serveur physique sera multipliée par 1,5 pour nous avoir la capacité virtuelle disponible au niveau de chaque serveur.

4.3.1.1.2 Hyperviseurs

Un hyperviseur fournit un logiciel pour gérer l'accès de la machine virtuelle au matériel sous-jacent. L'hyperviseur crée, gère et surveille les machines virtuelles. OpenStack Compute (nova) prend en charge de nombreux hyperviseurs à divers degrés, notamment:

- KVM - Machine virtuelle basée sur le noyau. Les formats de disque virtuel qu'il prend en charge sont hérités de QEMU car il utilise un programme QEMU modifié pour lancer la machine virtuelle. Les formats pris en charge incluent les images brutes, les formats qcow2 et VMware.

- LXC - Conteneurs Linux (via libvirt), utilisés pour exécuter des machines virtuelles basées sur Linux.
- QEMU - Quick EMUlator, généralement utilisé uniquement à des fins de développement.
- UML - Mode utilisateur Linux, généralement utilisé uniquement à des fins de développement.
- VMware vSphere 5.1.0 et plus récent, exécute des images Linux et Windows basées sur VMware via une connexion avec un serveur vCenter.
- Xen (utilisant libvirt) - Hyperviseur de projet Xen utilisant libvirt comme interface de gestion nova-compute pour exécuter des machines virtuelles Linux, Windows, FreeBSD et NetBSD.
- XenServer, Xen Cloud Platform (XCP) et d'autres variantes de Xen basées sur XAPI exécutent des machines virtuelles Linux ou Windows. Vous devez installer le nova-compute service dans une VM para-virtualisée.
- Hyper-V - Virtualisation de serveur avec Microsoft Hyper-V, à utiliser pour exécuter des machines virtuelles Windows, Linux et FreeBSD. S'exécute en nova-computemode natif sur la plate-forme de virtualisation Windows.
- Virtuozzo 7.0.0 et plus récent - Conteneurs OS et machines virtuelles basées sur le noyau pris en charge via libvirt virt_type = parallels. Les formats pris en charge incluent les images ploop et qcow2.

Selon le récent sondage des utilisateurs d'OpenStack, KVM est l'hyperviseur le plus largement adopté dans la communauté OpenStack. D 'autres hyperviseurs tels que LXC, VMware, Xen et Hyper-V sont soit moins utilisés, des hyperviseurs de niche, ou ont des fonctionnalités limitées par rapport aux hyperviseurs plus couramment utilisés. La comparaison des différents hyperviseurs est disponible [28]

Il est également possible d'exécuter plusieurs hyperviseurs dans un même déploiement à l'aide d'agrégats ou de cellules d'hôtes. Cependant, un nœud de calcul individuel ne peut exécuter qu'un seul hyperviseur à la fois.

4.3.1.1.3 Technique de Migration des instances

La migration en direct fait partie intégrante des opérations du cloud. Cette fonctionnalité offre la possibilité de déplacer en toute transparence les instances d'un hôte physique à l'autre, une nécessité pour effectuer des mises à niveau qui

nécessitent des redémarrages des hôtes de calcul, mais ne fonctionne bien qu'avec le stockage partagé. Pour cela, un système de fichiers distribués doit être configuré. Les options possibles incluent :

- NFS (par défaut pour Linux)
- Céph
- GlusterFS (GlusterFS)
- MooseFS (en)
- Lustre

Nous avons porté notre choix sur le premier (NFS) car étant une recommandation de la communauté.

4.3.1.2 Ressources Réseaux

4.3.1.2.1 Types de Réseaux

Le matériel serveur sélectionné doit avoir le nombre approprié de connexions réseau, ainsi que le bon type de connexions réseau, afin de prendre en charge l'architecture proposée. Nous devons assurer qu'au moins deux connexions de réseaux différentes entrent dans chaque rack.

Bien que chaque installation d'entreprise soit différente, les réseaux suivants avec leur bande passante proposée sont fortement recommandés pour l'installation de notre solution en production de base.

Réseau interne ou de gestion - Utilisé comme réseau de communication interne entre les nœuds contrôleur et de calcul. Peut également être utilisé comme un réseau pour la communication iSCSI entre le calcul et les nœuds de stockage iSCSI. Encore une fois, cela devrait être au minimum d'un NIC de 1 Go et devrait être un réseau non acheminé. Cette interface doit être redondante pour la haute disponibilité (HA).

Réseau de locataires(utilisateurs) - Un réseau privé qui permet la communication entre les instances de chaque locataire. Si vous utilisez des réseaux fixes et des réseaux de fournisseurs, ce réseau est facultatif. Ce réseau doit également être isolé de tous les autres réseaux pour la conformité à la sécurité. Une interface de 1 Go doit être suffisante et redondante pour HA.

Réseau de stockage - Un réseau privé qui pourrait être connecté au frontend Ceph ou à d'autres stockages partagés. Pour les fins HA cela devrait être une configuration redondante avec suggéré 10 Go NICs. Ce réseau isole le stockage pour les instances loin des autres réseaux. Sous charge, ce trafic de stockage pourrait submerger d'autres réseaux et provoquer des pannes sur d'autres services OpenStack.

Réseau externe ou public (facultatif) - Ce réseau est utilisé pour communiquer à l'extérieur des V vers l'espace du réseau public. Ces adresses sont généralement manipulées par l'agent neutronique sur les nœuds du contrôleur et peuvent également être manipulées par un SDN autre que le neutron. Toutefois, lors de l'utilisation du DVR (Distribued Virtual Routing) neutronique avec OVS, ce réseau doit être présent sur le nœud de calcul puisque le trafic nord et sud ne sera pas géré par les nœuds du contrôleur, mais par le nœud de calcul lui-même.

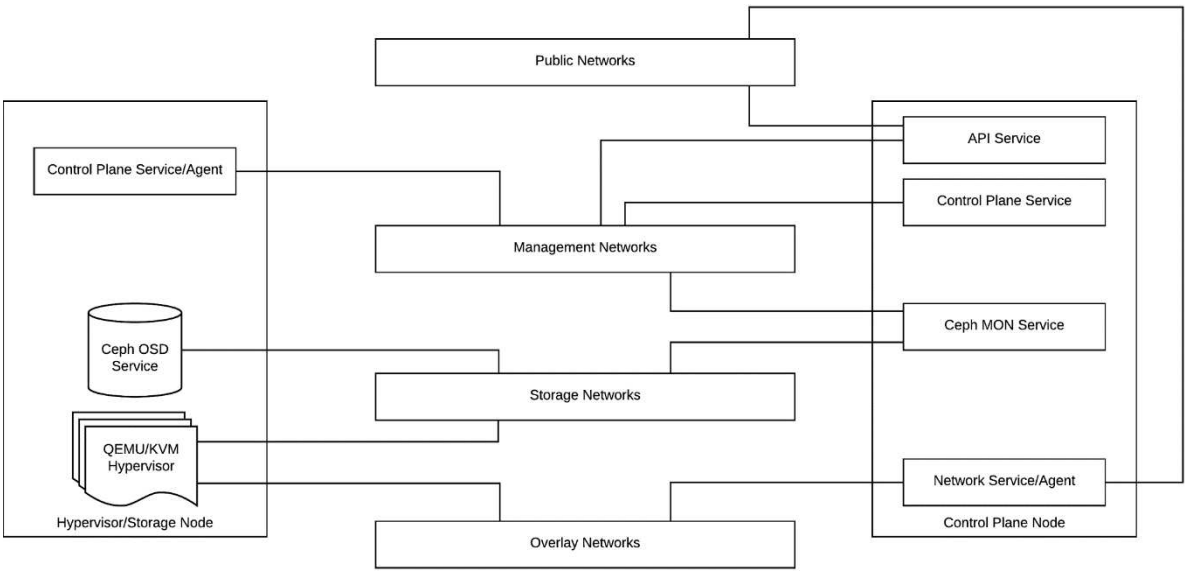


Figure 4:12: Types de réseaux de notre solution

4.3.1.2.2 Service Réseau (Neutron)

OpenStack Networking nous permet de créer et de gérer des objets réseau, tels que des réseaux, des sous-réseaux et des ports, que d'autres services OpenStack peuvent utiliser. Les plug-ins peuvent être implémentés pour accueillir différents équipements et logiciels de réseaux, offrant ainsi une flexibilité à l'architecture et au déploiement d'OpenStack.

Le service networking, nom de code neutron, fournit une API qui nous permet de définir la connectivité réseau et l'adressage dans le cloud. Le service Networking permet aux opérateurs de tirer parti des différentes technologies de réseautage pour alimenter leur réseau cloud. Le service networking fournit également une API pour configurer et gérer une variété de services réseau allant du l3 de transport et NAT à l'équilibrage de charge, pare-feu périphérique, et les réseaux privés virtuels. Il comprend les composants suivants :

Serveur API

L'API Réseau OpenStack inclut la prise en charge de la gestion des réseaux de couche 2 et des adresses IP (IPAM), ainsi qu'une extension pour une construction de routeur de couche 3 qui permet le routage entre les réseaux de la couche 2 et les passerelles vers les réseaux externes. OpenStack Networking comprend une liste croissante de plug-ins qui permettent l'interopérabilité avec diverses technologies de réseau commercial et open source, y compris les routeurs, les commutateurs, les commutateurs virtuels et les contrôleurs de réseau définis par logiciel (SDN).

Plug-in openstack networking et agents

Branche et débranche les ports, crée des réseaux ou des sous-réseaux et fournit une adresse IP. Le plug-in choisi et les agents diffèrent selon le fournisseur et les technologies utilisées dans le cloud particulier. Il est important de mentionner qu'un seul plug-in peut être utilisé à la fois.

File d'attente de messagerie

Accepte et a l'itinéraire des demandes RPC entre les agents pour compléter les opérations api. La file d'attente de messages est utilisée dans le plug-in ML2 pour RPC (Remote Protocol Communication) entre le serveur neutronique et les agents neutroniques qui fonctionnent sur chaque hyperviseur, dans les pilotes de mécanisme ML2 pour open vSwitch et Linux bridge.

4.3.1.3 Ressources de stockage

4.3.1.3.1 Technologie de stockage

Diverses technologies de stockage de produits de base sont disponibles. Selon les besoins de nos utilisateurs de cloud, nous pouvons implémenter une ou plusieurs de

ces technologies dans différentes combinaisons (Ceph, GlusterFS). Ceph est une solution de stockage évolutive qui reproduit les données sur les nœuds de stockage.

Ceph utilise un mécanisme de stockage d'objets pour le stockage de données et expose les données via différents types d'interfaces de stockage à l'utilisateur final, il prend en charge les interfaces pour: Stockage d'objets, Stockage en bloc et Interfaces système de fichiers

Ceph fournit une prise en charge de la même API de stockage d'objets que Swift et peut être utilisé comme back end pour le service block storage (Cinder) ainsi que le stockage back-end pour les images de Glance.

Ceph prend en charge l'approvisionnement mince (thin Provisionning) mis en œuvre à l'aide de copier-sur-écrire. Cela peut être utile lors du démarrage à partir du volume parce qu'un nouveau volume peut être provisionné très rapidement. Ceph prend également en charge l'authentification basée sur keystone (à partir de la version 0.56), de sorte qu'il peut être un échange sans faille pour la mise en œuvre rapide OpenStack par défaut.

Les avantages de Ceph incluent :

- L'administrateur a un contrôle plus fin sur les stratégies de distribution et de réplication des données.
- Consolidation du stockage des objets et stockage des blocs.
- Fourniture rapide d'instances de démarrage à partir de volume à l'aide d'un approvisionnement mince.
- Prise en charge de l'interface système de fichiers distribué CephFS.

Les inconvénients de Ceph :

Du fait du grand nombre de composantes prévues, un réseau complet est nécessaire pour pouvoir exploiter les atouts de Ceph. En outre, l'installation est relativement laborieuse et l'utilisateur ne comprend pas nécessairement où les données sont stockées physiquement [28].

Tableau 4:1: Comparaison entre GlusterFS et Ceph

GlusterFS	Ceph

Des atouts dans l'aspect système de fichiers	Des atouts dans l'aspect Object Store
Un algorithme de stockage plus rapide	Une meilleure performance sur un matériel simple
Aucun serveur de métadonnées central n'est nécessaire	Une intégration simple dans tous les systèmes, quel que soit le système d'exploitation utilisé
Une faible complexité	Block Device pour Linux
Une meilleure adéquation au stockage de gros fichiers (à partir de 4 Mo par fichier)	La simplicité des modifications spécifiques au client
Une meilleure adéquation pour les fichiers avec un accès séquentiel	La compatibilité RADOS intégration sous forme de mémoire orientée bloc grâce à KVM/QEMU

4.3.1.4 Supervision et Centralisation des logs

Les journaux sur les nœuds de calcul, ou tout serveur exécutant nova-calcul (par exemple dans une architecture hyper-convergée), sont les principaux points pour les problèmes de dépannage avec l'hyperviseur et les services de calcul. En outre, les journaux du système d'exploitation peuvent également fournir des informations utiles.

Au fur et à mesure que l'environnement cloud augmente, la quantité de données journal augmente de façon exponentielle. L'activation du débogage des services de notre cloud ou du système d'exploitation aggrave encore les problèmes de données. Le cloud produit beaucoup d'informations utiles sur l'enregistrement, mais pour que les informations soient utiles à des fins d'exploitation, nous avons envisager d'avoir un serveur de journalisation central pour envoyer des journaux, et un système d'analyse de journaux tel que Elastic Stack [anciennement connu sous le nom elk].

Elastic Stack se compose principalement de trois composants : Elasticsearch (recherche et analyse de journaux), Logstash (prise de journal, traitement et sortie) et Kibana (service de tableau de bord de journal).

4.3.2 Gestion des applications

4.3.2.1 Base de données

La plupart des composants de notre cloud nécessitent un accès aux services de base de données back-end pour stocker les informations d'état et de configuration. Nous avons choisi MySQL comme base de données principale appropriée qui répond aux exigences de disponibilité et de tolérance aux pannes des services OpenStack mais d'autres bases de données compatibles sont disponibles. Notons que la télémétrie utilise une base de données NoSQL : MongoDB.

La solution de base de données haute disponibilité choisie change en fonction de la base de données sélectionnée. MySQL, par exemple, propose plusieurs options. Utilisez une technologie de réplication telle que Galera pour le clustering actif-actif. Pour une utilisation active-passive, une forme de stockage partagée est utilisée. Chacune de ces solutions potentielles à un impact sur la conception:

Les solutions qui utilisent Galera / MariaDB nécessitent au moins trois nœuds MySQL.

MongoDB a ses propres considérations de conception pour la haute disponibilité.

4.3.2.2 Stockage des instances

Le stockage se trouve dans de nombreuses parties de l'environnement cloud OpenStack. Il est important de comprendre la distinction entre le stockage éphémère et le stockage persistant :

- Stockage éphémère - Si vous ne déployez que le service OpenStack Compute (nova), par défaut, vos utilisateurs n'ont accès à aucune forme de stockage persistant. Les disques associés aux machines virtuelles sont éphémères, ce qui signifie que du point de vue de l'utilisateur, ils disparaissent lorsqu'une machine virtuelle est terminée.
- Stockage persistant - Le stockage persistant signifie que la ressource de stockage survit à toute autre ressource et est toujours disponible, quel que soit l'état d'une instance en cours d'exécution.

Le cloud OpenStack supportent explicitement trois types de stockage persistant : le stockage d'objets, le stockage en bloc et le stockage basé sur les fichiers.

Ci-dessous, le tableau récapitulatif des différents types de stockage et leurs fonctions.

Tableau 4:2: Les différents types de stockage dans Openstack

	Stockage éphémère	Stockage en bloc	Stockage d'objets	Stockage partagé du système de fichiers
Application	Exécuter le système d'exploitation et l'espace de grattage	Ajouter un stockage persistant supplémentaire à une machine virtuelle (VM)	Stocker des données, y compris des images VM	Ajouter un stockage persistant supplémentaire à une machine virtuelle
Accessible par...	Un système de fichiers	Un périphérique de bloc qui peut être partitionné, formaté et monté (tel que, /dev/vdc)	L'API REST	Une part de service Systèmes de fichiers partagés (gérée par Manille ou externe enregistrée à Manille) qui peut être partitionnée, formatée et montée (comme /dev/vdc)
Accessible depuis...	Au sein d'un VM	Au sein d'un VM	quelque part	Au sein d'un VM
Géré par...	OpenStack Compute (nova)	OpenStack Block Storage (cinder)	OpenStack Object Storage (glance)	OpenStack Shared File System Storage (manilla)
Persiste jusqu'à ce que...	VM est résilié	Supprimé par l'utilisateur	Supprimé par l'utilisateur	Supprimé par l'utilisateur

Dimensionnement déterminé par...	Configuration des paramètres de taille par administrateur, connue sous le nom *de saveurs*	Spécification de l'utilisateur dans la demande initiale	Quantité de stockage physique disponible	• Spécification de l'utilisateur dans la demande initiale • Demandes de prolongation • Devis disponibles au niveau de l'utilisateur • Limitations appliquées parl'administrateur
Configuration de cryptage	Paramètre dans **nova.conf**	Admin établissant le type de volume crypté, puis l'utilisateur sélectionnant le volume crypté	Pas encore disponible	Le service Systèmes de fichiers partagés n'applique aucun cryptage supplémentaire au-dessus de ce que le stockage back-end de l'action fournit
Exemple d'utilisation typique...	10 Go premier disque, 30 Go deuxième disque	Disque de 1 To	10 s de TBs de stockage de jeu de données	Dépend entièrement de la taille du stockage back-end spécifié lors de la création d'une action. En cas d'approvision-nement mince, il peut s'agir d'une réservation partielle de l'espace

Il existe trois approches principales pour fournir un stockage temporaire :

- Hors du stockage des nœuds de calcul — système de fichiers partagé
- Sur le stockage des nœuds de calcul — système de fichiers partagé

- Sur le stockage des nœuds de calcul — système de fichiers non partagé

Nous avons porté notre choix sur le premier car l'exécution d'un système de fichiers partagé sur un système de stockage en dehors des nœuds de calcul est idéale pour les nuages où la fiabilité et l'évolutivité sont les facteurs les plus importants. L'exécution d'un système de fichiers partagé sur les nœuds de calcul eux-mêmes peut être le meilleur dans un scénario où nous devons déployer sur des serveurs préexistants pour lesquels vous avez peu ou pas de contrôle sur leurs spécifications ou ont des besoins spécifiques de performances de stockage, mais n'ont pas besoin de stockage persistant.

De plus, de nombreux opérateurs utilisent des hôtes de calcul et de stockage distincts au lieu d'une solution hyper convergée. Les services de calcul et de stockage ont des exigences différentes, et les hôtes de calcul nécessitent généralement plus de processeur et de RAM que les hôtes de stockage. Par conséquent, pour un budget fixe, il est logique d'avoir différentes configurations pour vos nœuds de calcul et vos nœuds de stockage. Les nœuds de calcul seront investis dans le Processeur et la RAM, et les nœuds de stockage seront investis dans le stockage en bloc.

4.4 Dimensionnement

4.4.1 Spécification matérielle

Le matériel serveur suivant est proposé pour notre architecture en production en tenant compte de la haute disponibilité, d'une bonne performance et de l'évolutivité de notre solution. Pour les nœuds de Calcul (Compute), la configuration suivante est proposée.

- Processeurs de base 2x12 E5-2680v3
- 256 Go de RAM
- 2x800 Go SSD Intel S3610
- 2x10 Go Intel X710 dual-port NICs

Pour les nœuds de stockage, la configuration suivante est proposée.

- 2x12 Core CPUs E5-2670v3

- RAM de 128 Go
- 2x480GB SSD Intel S3610
- Disque dur de 2x1 To
- 2x10 Go Intel X710 dual-port NICs

4.4.2 Quantités de ressources

4.4.2.1 Ressources de Calculs

Le nombre de nœuds Compute/Controller dans l'environnement : 6 nœuds

Le nombre de cœurs CPU mis à la disposition des hyperviseurs et des services de contrôle : 6*24=144 cœurs. Avec le taux de souscription 16 :1 du CPU, on aura une capacité de : 144*16= 2304 vCPU.

La quantité de RAM disponible pour les hyperviseurs et les services de contrôle : 256*6=1536 Go. Avec le taux de souscription 1,5 :1, on aura une capacité de : 1536*1,5=2304 vRAM.

NB : Un serveur standard sera utilisé pour la centralisation des logs avec Elastic Stack.

4.4.2.2 Ressources de stockage

- Le nombre de nœuds de stockage dans l'environnement : 3 nœuds
- Nombre de cœurs CPU disponibles pour les services de stockage :3*24= 72 cœurs
- La quantité de RAM disponible pour le cache de stockage: 3*128=384 Go
- La taille totale de l'espace disque brut disponible sur les nœuds de stockage : 3*2=6Tb.

Les serveurs sont installés dans 3 racks reliés par des commutateurs ToR à des commutateurs de colonne vertébrale.

4.4.2.3 Ressources réseaux

L'environnement utilise la topologie des commutateurs de feuilles (leaf switches) dans le réseau de sous-couche (underlay). Le protocole BGP est utilisé dans le réseau de sous-couche pour assurer MLAG (multipath links aggregation) l'agrégation des liens

multi chemins (IP ECMP) aux commutateurs de feuilles. Les commutateurs de feuilles ToR sont reliés aux épines avec des liaisons ascendantes de 40 GoE.

Les commutateurs de feuilles (leaf switches) utilisent des VXLAN pour fournir un réseau de superposition aux serveurs, et l'agrégation MLAG pour assurer la disponibilité et les performances sur les liaisons en aval. Les serveurs sont connectés aux commutateurs ToR avec des liaisons port-canal de 40 GbE (4x10 GbE avec agrégation MLAG).

- Commutateurs de colonne vertébrale (Spine switches): Arista 7508E (4x2900PS, 6xFabric-E modules, 1xSupervisorE module)
- Commutateurs ToR (ToR switches): Arista 7050X

Le diagramme suivant décrit le schéma réseau de l'environnement :

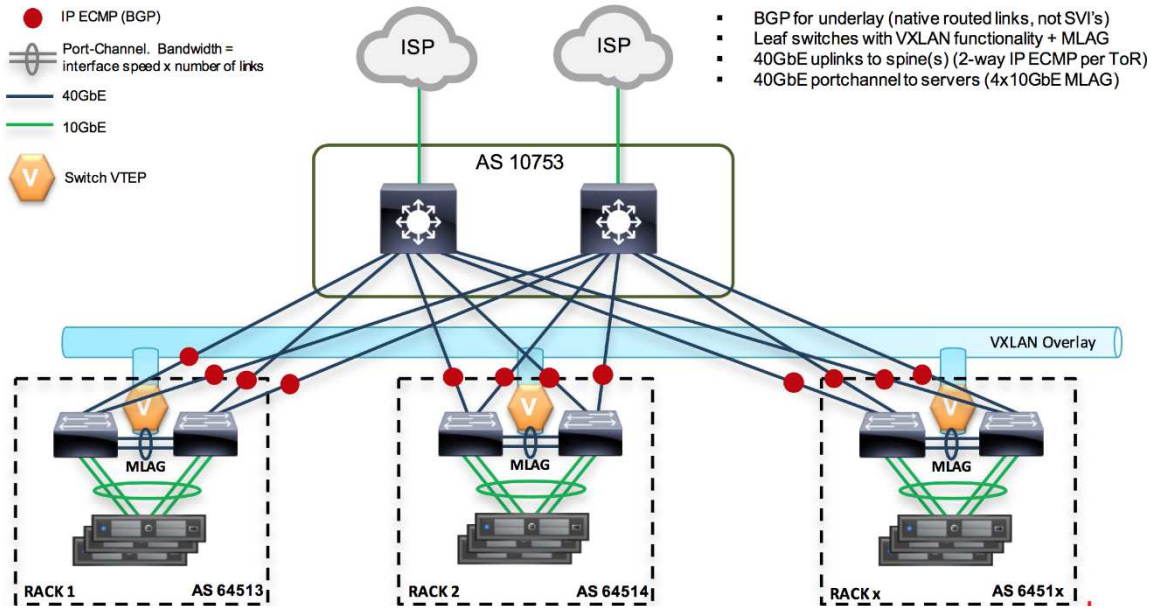


Figure 4:13 : Première Architecture proposée pour notre solution

Au cas où nous ne disposons pas de commutateurs ToR, nous pouvons utiliser un switch L3 et trois switchs L2.

Le switch L3 va desservir les switchs L2.

Nous aurons :

- Un switch pour le réseau de management
- Un switch pour le réseau de stockage

❖ Un switch pour le réseau privé des instances (overlay) et un autre pour leur accès depuis internet, c'est le réseau public

L'architecture proposée est la suivante :

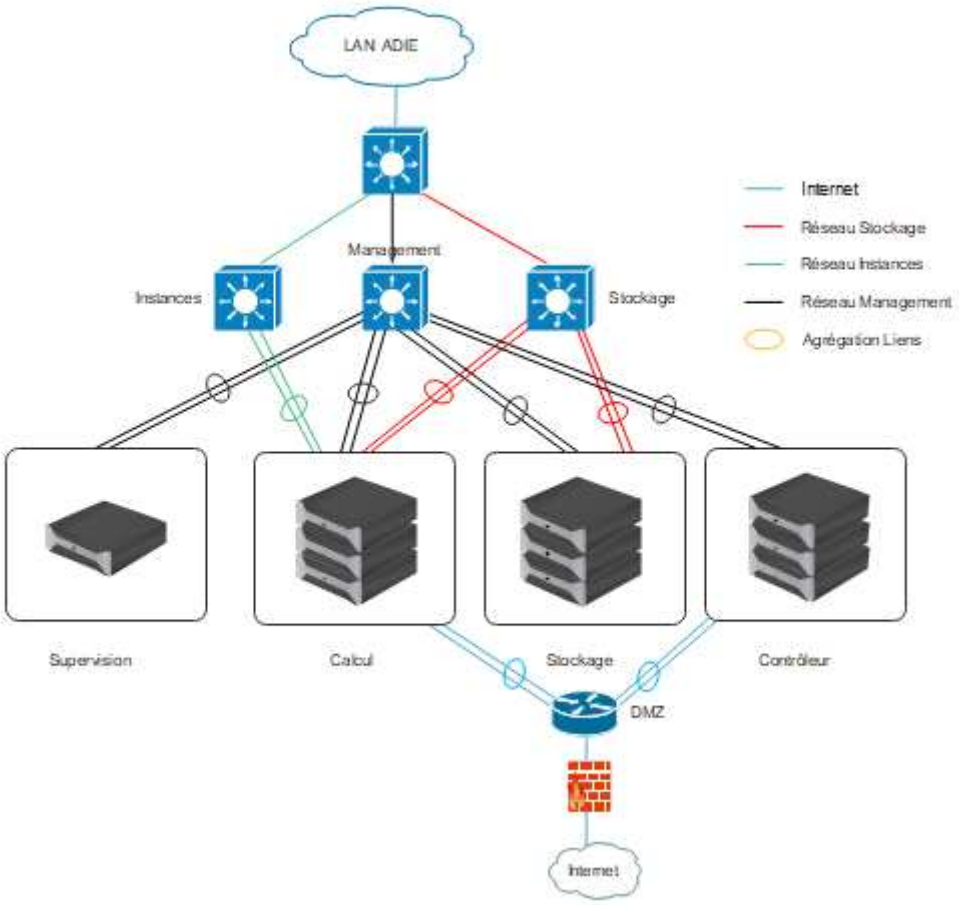


Figure 4:14: Deuxième architecture Proposée pour notre solution

5. Chapitre V : Mise en œuvre de la solution

Dans le chapitre précédant, nous avons fait le dimensionnement de l'architecture de production proposée. Pour notre prototype, nous avons travaillé sur une architecture de 6 nœuds dont 4 serveurs physiques et 2 VM qui tourne sur VmWare ESXI 6. Le système d'exploitation CentOS 8.2 a été installé sur l'ensemble de nos serveurs. Les modules installés sont les suivants : Keystone, Glance, Nova, Neutron, Horizon, Cinder, Swift et Heat. Nous allons présenter l'environnement de travail et montrer les résultats après installation et configuration.

Les caractéristiques des machines que nous avons utilisées sont les suivants :

Tableau 5:1: Caractéristiques des machines de notre prototype

	1 Nœud de Control	1 Nœud de Calcul	1Noeud de Réseau	1 Nœud de Stockage Bloc	2 Nœuds de Stocka-ge Objet
Modèle	Dell Inc. Power Edge R730	Dell Inc. Power Edge R730	HP ProLiant DL380 Gen9	Dell Inc. Power Edge R730	VMware, Inc.VMwae Virtual Platform
RAM en Go	256	256	256	256	8
Processeur	16xIntel(R)Xeon(R) CPU E5-2623 v3 @ 3.00GHz	16xIntel(R) Xeon(R) CPU E5-2623 v3 @ 3.00GHz	16xIntel(R) Xeon(R) CPU E5-2637 v4 @ 3.50GHz	16xIntel(R)Xeon(R) CPU E5-2623v3@ 3.00GHz	1xIntel(R) Xeon(R) CPU E5-2670 v3 @ 2.30GHz
Disque	1.64 Tb	1.64 Tb	0.838 Tb	1.8 Tb	50 Go
Réseau	eno1 : .83	eno1: .84 eno2 :sans IP	eno1: .85 eno2 :sans IP	eno1: .86	eno1:.87/.88

La figure suivante présente l'architecture que nous avons adoptée.

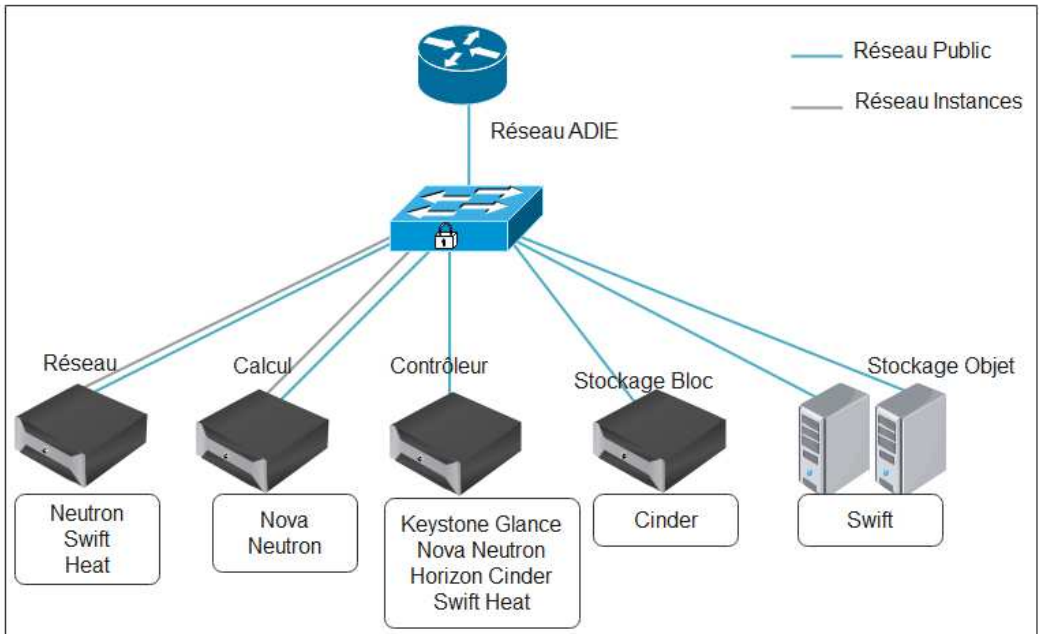


Figure 5:1: Architecture de notre prototype

Après installation et configuration des services, voici le tableau de bord auquel nous allons nous connecter.

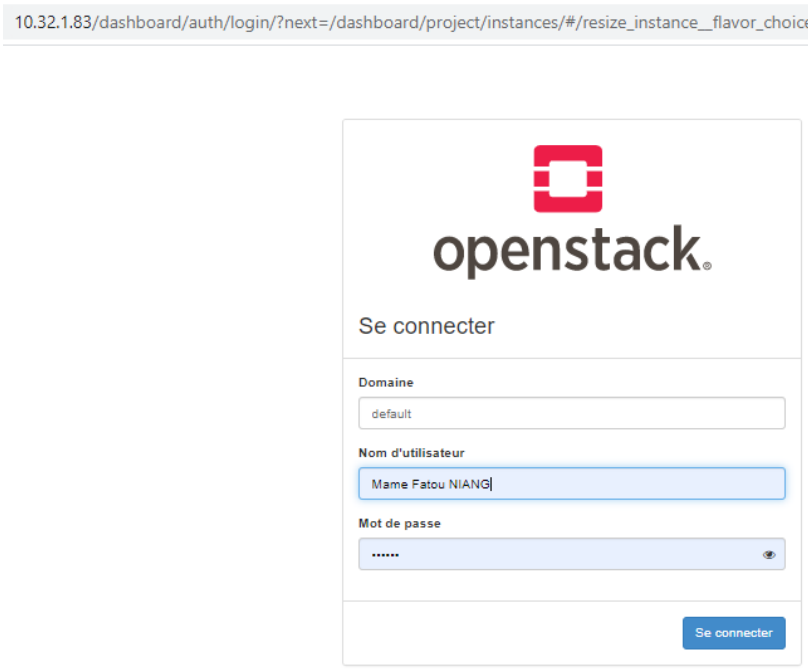


Figure 5:2: Interface d'accueil

Après connexion, nous aurons l'ensemble des ressources disponibles au niveau de notre cloud.

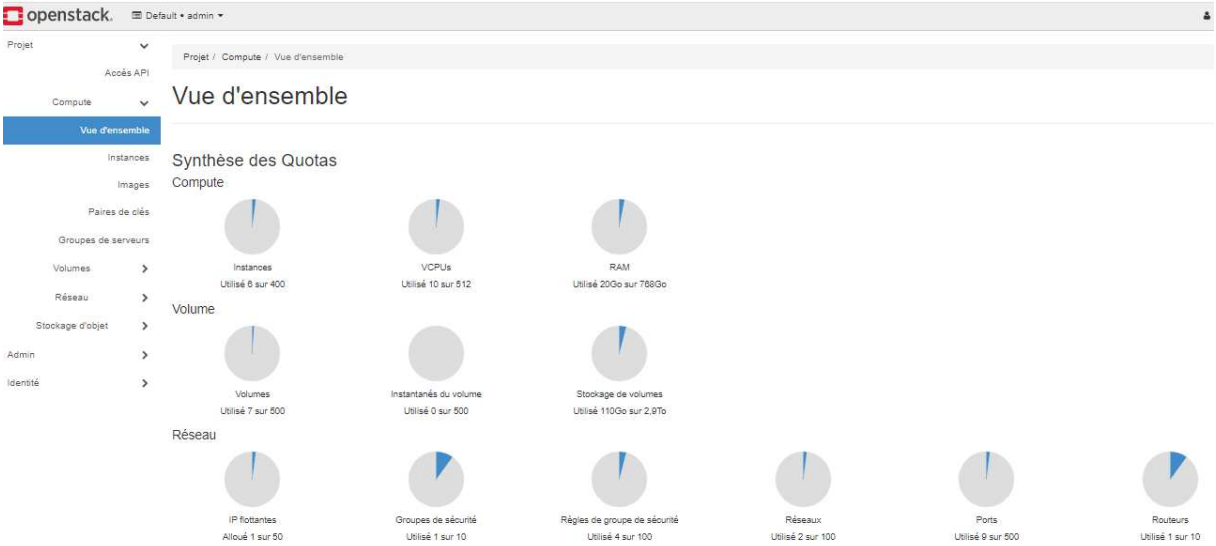


Figure 5:3: Vue d'ensemble de notre cloud

A ce niveau, nous pouvons commencer à utiliser notre cloud en créant des instances.

La création d'instances nécessite des prérequis :

- Création de Gabarits
- Création d'images
- Création de paires de clé pour l'authentification au niveau de notre instance
- Création d'un réseau pour la connectivité des instances
- Création d'un groupe de sécurité

Création de gabarits (flavors)

Les gabarits définissent la taille de la RAM et du disque, le nombre de cœurs, ainsi que d'autres ressources. Ils peuvent être choisis quand les utilisateurs déploient des instances. Pour la création d'un gabarit, l'administrateur va dans l'onglet Compute du projet Admin, puis sur l'onglet Gabarit et il clique sur Créer un gabarit comme illustré sur la figure suivante :

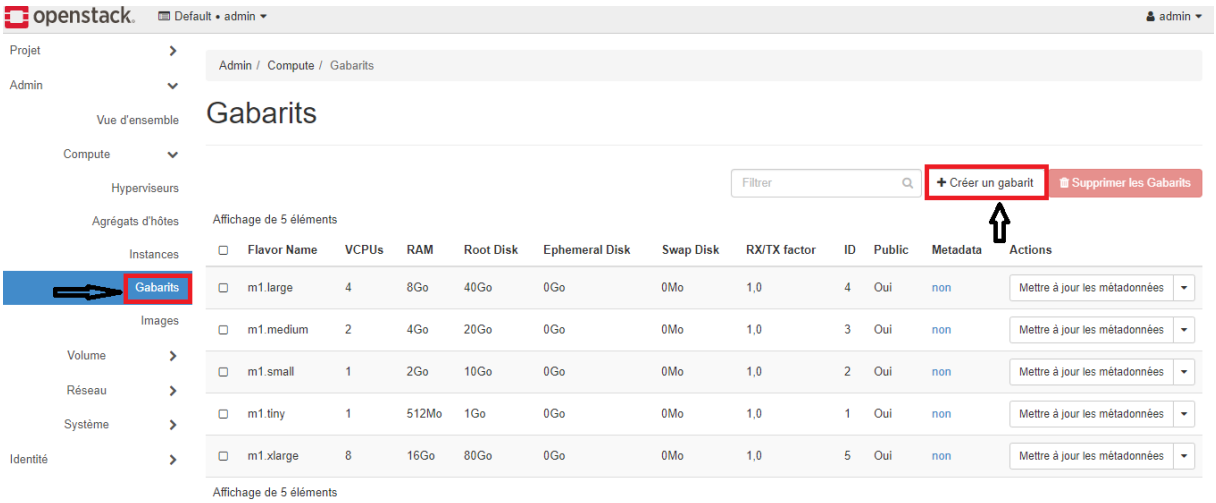


Figure 5:4: Création Gabarit 1/2

L'interface de création s'affiche et l'administrateur pourra renseigner la taille de CPU, RAM et disque voulue comme illustré ci-après :

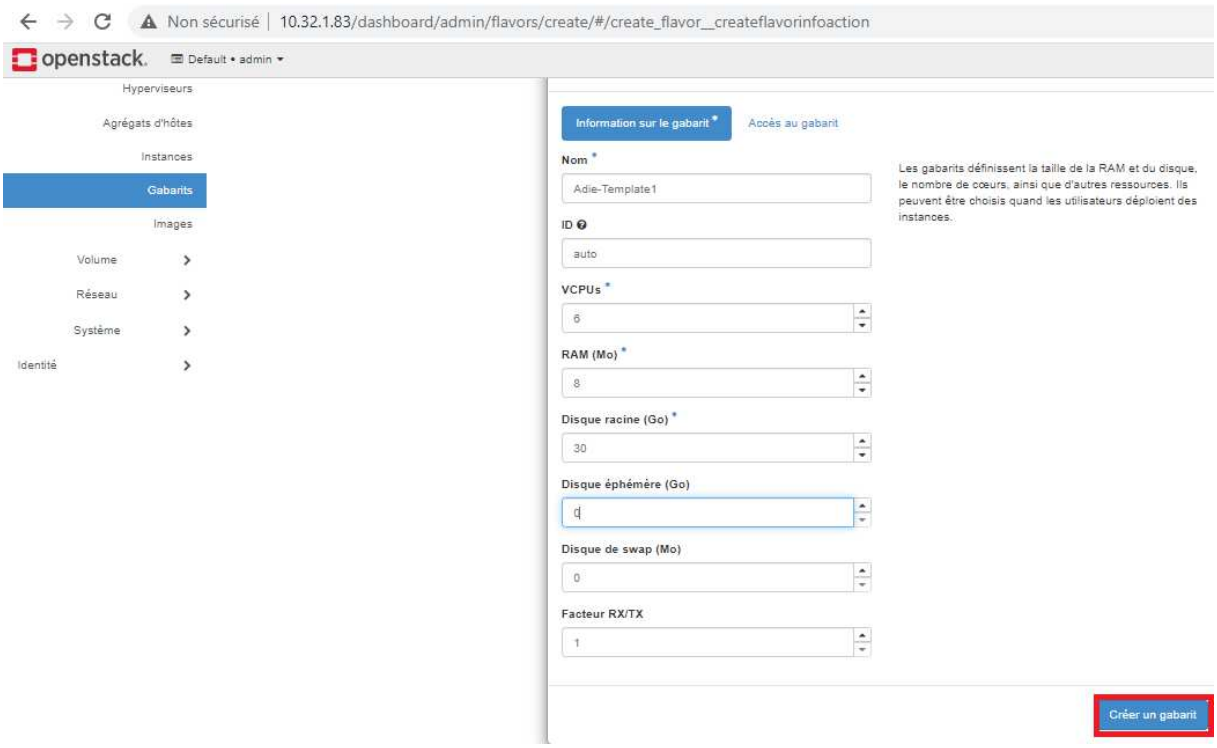


Figure 5:5: Création Gabarit 2/2

Gestion des images

Après avoir créé notre gabarit Adie-Template1, nous allons passer à la gestion des images sachant que les images sont créées au préalable avec tout le nécessaire, cloud-init, virt-builder et Packer. La figure ci-dessous représente les images systèmes disponibles sur notre cloud.

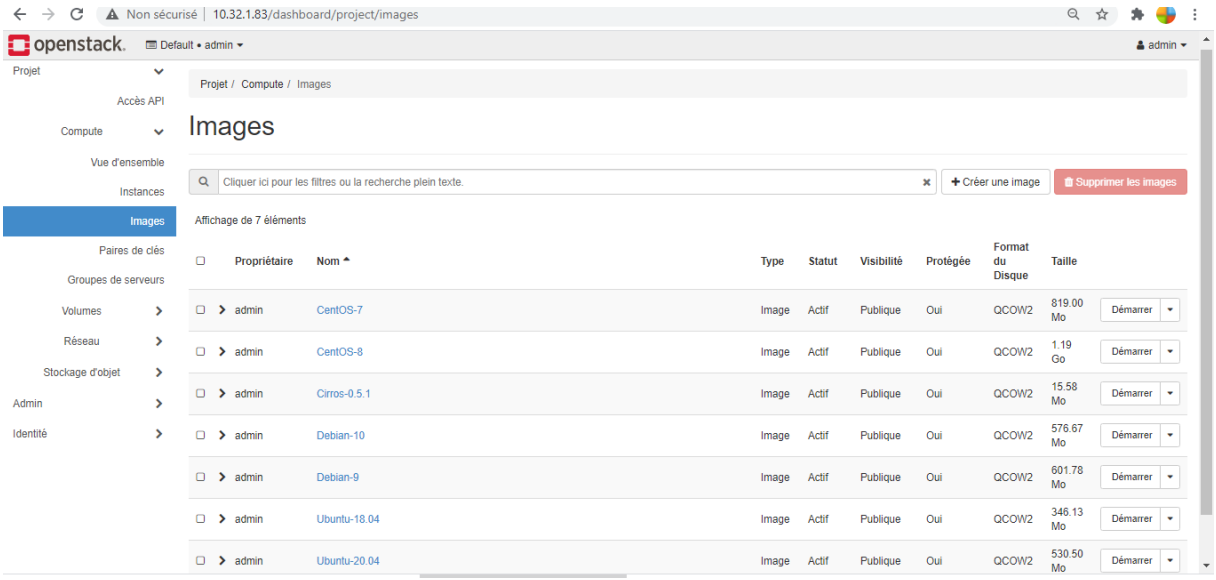


Figure 5:6: liste des images d'OS

Gestion du réseau

Les différents réseaux disponibles au niveau de notre cloud sont représentés dans la topologie qui suit :

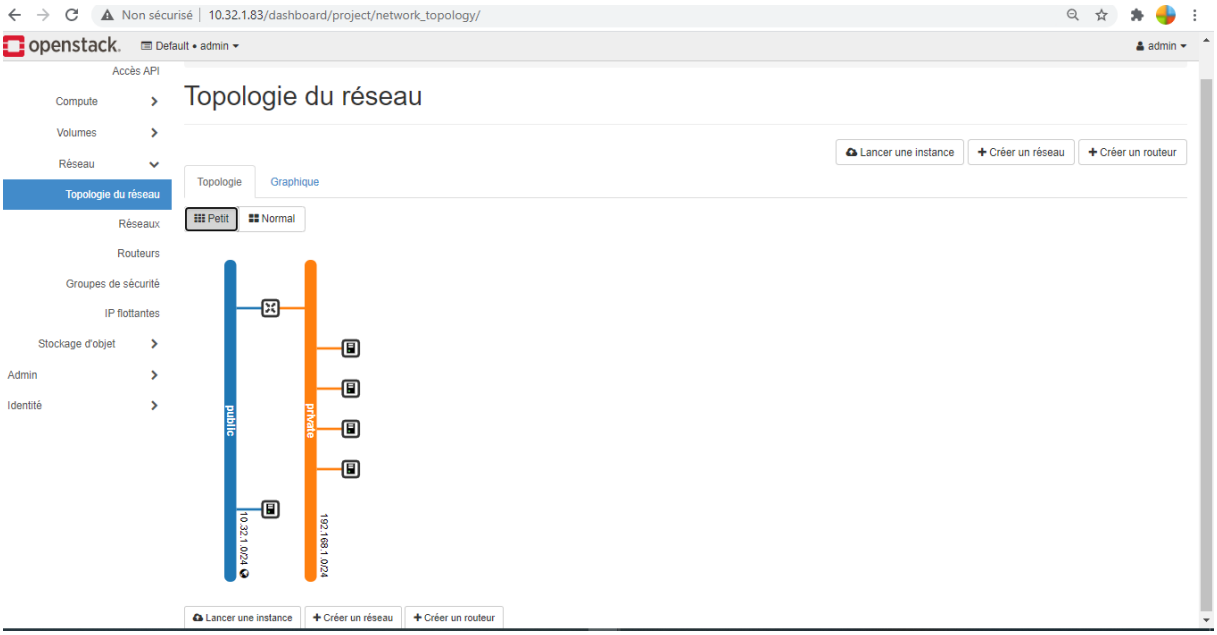


Figure 5:7: Topologie Réseau du cloud

En bleu, nous avons le réseau qui nous permet d'accéder à nos instances depuis internet et en orange, le réseau interne.

Création de la paire de clés

La paire de clés nous permet d'accéder à nos instances via ssh. Lors de la création d'une instance, une paire de clé est requis pour pouvoir nous connecter à l'instance.

La procédure de création est la suivante :

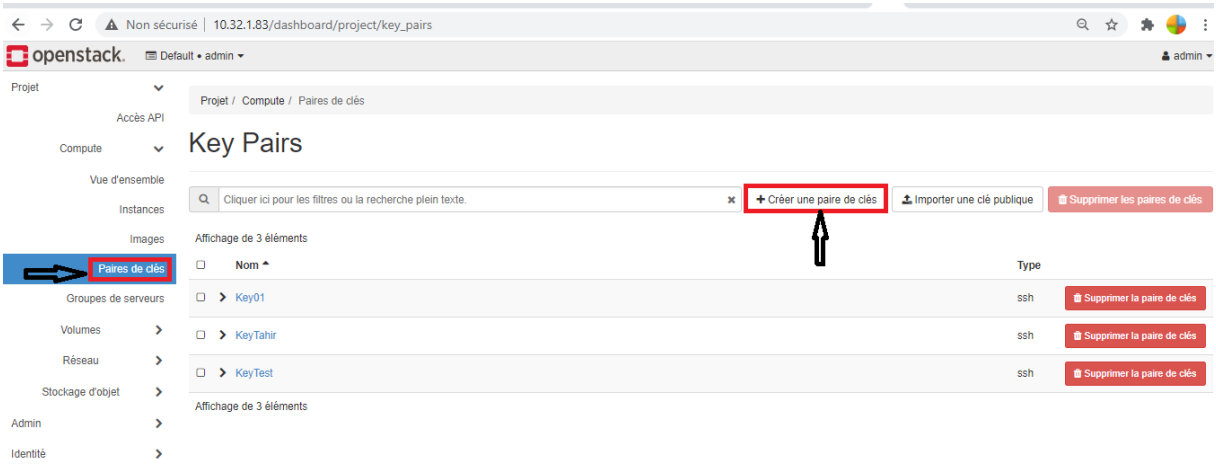


Figure 5:8: Création de paire de clés 1/2

Après avoir cliqué sur l'onglet « **créer une paire de clé »,** une autre interface s'affiche.

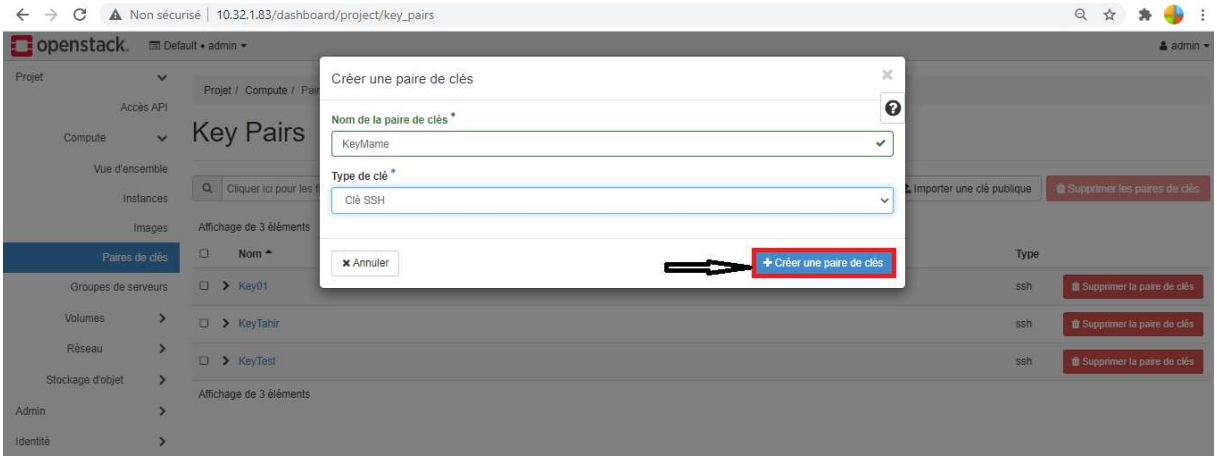


Figure 5:9: Création de paire de clés 2/2

A ce stade, il faut renseigner le nom de la clé, puis le type de la clé : **Clé SSH** ou **Certificat X509** et cliquer sur le bouton **Créer une paire de clé** et automatiquement la paire se crée et vous propose d'enregistrer le fichier KeyMame.pem dans le répertoire de votre choix.

Création d'instances

Maintenant, nous allons passer à la création des instances, toutes les étapes précédentes vont intervenir à ce niveau.

Conclusion

Dans ce chapitre, nous avons parlé de notre environnement de travail avec les caractéristiques des différents serveurs utilisés, les services disponibles et la capacité d'accueil de notre cloud et quelques exemples d'utilisations de notre plateforme.

Conclusion

L'objet de ce projet était de faire l'étude et la mise en place d'un cloud privé à l'ADIE dans l'optique de faciliter la création et le provisionnement des services aux clients. La motivation d'une telle solution a été nourrie par la lenteur de déploiement des services dû à la création manuelle des environnements et la longue procédure à suivre par le demandeur de services avant livraison mais également la gestion distribuée des différentes plateformes de virtualisation qui résulte d'un manque de visibilité des ressources consommées.

Bien que le cloud Computing ne soit une technologie nouvelle, son mode de fonctionnement diffère de l'informatique classique cause pour laquelle nous avons fait dans le premier chapitre l'état de l'art sur le Cloud computing en parlant de : son historique, ses caractéristiques, ses types de services (IaaS, PaaS, SaaS), ses modèles de déploiement (public, privé, communautaire et hybride) avec leurs avantages et inconvénients. Nous avons vu également les technologies pilotes, les différents acteurs (écosystème) du cloud, les techniques de migration vers le cloud ainsi que leurs avantages et leurs inconvénients.

Dans le deuxième chapitre, nous avons parlé du cadre du projet à savoir la présentation de la structure d'accueil et du projet ainsi que l'étude de l'existant.

Dans le troisième chapitre, nous avons présenté les différentes solutions de cloud privé ciblées (Openstack, OpenNebula, CloudStack et Eucalyptus) puis une étude comparative sur ces différentes solutions a été faite et notre choix fut porté sur Openstack. Cette solution est modulaire et innovante permet d'orchestrer et automatiser les tâches de gestion sur les ressources informatiques, fournit une communication centralisée entre les utilisateurs et une souplesse d'utilisation des différents services mises en place pour une exploitation immédiate.

Au niveau du quatrième chapitre, nous avons fait la conception de notre solution qui est scindée entre une partie fonctionnelle où nous avons identifié nos besoins (fonctionnels comme non fonctionnels), nos différents acteurs et les différents cas d'utilisations. Au niveau de la partie technique, nous avons parlé d'abord, de la manière de gérer notre infrastructure et des applications déployées déçues. Puis, nous avons dimensionné notre infrastructure cible avec deux possibilités de configuration.

Au cinquième et dernier chapitre, nous avons mis en œuvre notre solution en donnant d'abord les caractéristiques des différents serveurs utilisés puis l'architecture adoptée, la capacité d'accueil et quelques captures de cas d'utilisations.

Au bout de ce travail, nous avons déployer un portail self-service que les ministères par exemple peuvent utiliser en ayant un pool de ressources à leur disposition avec un quota définit au préalable par un administrateur. Tout utilisateur autorisé peut utiliser le service dont il a besoin par exemple un LAMP (Linux Apache PHP MySQL), un serveur web sans l'installer car étant préconfiguré et mis à sa disposition. Nous

avons également centralisé la gestion des ressources car nous pouvons visualiser en temps réel les ressources consommées des différents utilisateurs.

En définitif, nous avons atteint notre objectif qui était de faciliter la création et le provisionnement des services à travers un cloud privé. Comme perspectives, nous voulons migrer une bonne partie de l'infrastructure existante sur notre cloud privé et créer les nouvelles demandes de services dessus.

Bibliographie

[1] : Cloud Computing : services informatiques dynamiques basés sur le Web - Concepts et notions de base - REGUIEG Hicham 17 décembre 2020

[6] : Cours Master 2 Sécurité des systèmes informatiques Université Paris-Est Créteil Val-de-Marne (Paris12), Noureddine BOUYAHIAOU

[7] : Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy Katz,

Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, and Matei Zaharia. Above the clouds : A berkeley view of cloud computing. Technical report, University of California at Berkeley, February 2009

[8] : Cloud Computing : services informatiques dynamiques basés sur le Web - Concepts et notions de base - REGUIEG Hicham 17 décembre 2020

[9] : Mastering OpenStack Design, deploy, and manage a scalable OpenStack infrastructure Omar Khedher

[10] : Guerid Hachem. Cours :"Cloud computing". Master 2 RSID Informatique. Université des Sciences et de la Technologie d'Oran. 2019-2020

[11] : Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy Katz, Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, and Matei Zaharia. Above the clouds : A berkeley view of cloud computing. Technical report, University of California at Berkeley, February 2009

[12] : K. Chandrasekaran. *Essentials of Cloud Computing*. Chapman and Hall/CRC, 1st edition, 2014.

[13] : Christian Baun, Marcel Kunze, Jens Nimis, and Stefan Tai. *Cloud computing : Webbased dynamic IT services*. Springer, Berlin, 2011. 46.02.12; LK 01; Berlin [u.a.] : Springer, 2011.

[14] : Cloud Computing : services informatiques dynamiques basés sur le Web - Concepts et notions de base - REGUIEG Hicham 17 décembre 2020

[15] : Christian Baun, Marcel Kunze, Jens Nimis, and Stefan Tai. *Cloud computing : Webbased dynamic IT services*. Springer, Berlin, 2011. 46.02.12; LK 01; Berlin [u.a.] : Springer, 2011.

[16] : K. Chandrasekaran. *Essentials of Cloud Computing*. Chapman and Hall/CRC, 1st edition, 2014.

[17] : Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy Katz,

Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, and Matei Zaharia. Above the clouds : A berkeley view of cloud computing. Technical report, University of California at Berkeley, February 2009

[18] : Guerid Hachem. Cours :"Cloud computing". Master 2 RSID Informatique. Université des Sciences et de la Technologie d'Oran. 2019-2020

[19] : Mastering OpenStack Design, deploy, and manage a scalable OpenStack infrastructure Omar Khedher

[20] : Cloud Computing : services informatiques dynamiques basés sur le Web - Concepts et notions de base - REGUIEG Hicham 17 décembre 2020

[22] : http://www.techplayon.com/openstack-openstack-component-and-its-deployment-models/

5.1 Webographie

[2] : https://azure.microsoft.com/fr-fr/overview/what-is-cloud-computing/

[3] : https://formadoct.doctorat-bretagneloire.fr/c.php?g=491596&p=3362221

[4] : https://www.nuageo.fr/wpcontent/webpcpassthru.php?src=https://www.nuageo.fr/wp-content/uploads/2015/04/Cloud_MD.png&nocache=1

[5] : https://www.computerland.fr/difference-cloud-prive-public/

[21] : https://www.adie.sn/

[28] : https://www.ionos.fr/digitalguide/serveur/know-how /glusterfs-vs-ceph/