

# Déploiement d'une infrastructure virtuelle avec OpenStack [PPP]

# [OpenStack]

Cloud & IoT

Oussama Arifallah

## Table des matières

Introduction :	3
<b>Les composants de OpenStack :</b>	<b>4</b>
Nova :	4
Swift :	4
Cinder :	4
Neutron :	4
Horizon :	4
Keystone :	5
Glance :	5
Ceilometer :	5
Heat :	5 1-
Configuration :	6
OpenStack Dashboard :	6
Installation :	8
Configuration :	9
Connexion à OpenStack :	9 2-
Networks :	10
<b>Création d'un réseau :</b>	<b>10 Les</b>
réseaux :	11
Création des groupes de securité :	12
Création d'un routeur :	13
Ensemble des réseaux créés :	13 IP
flottantes :	14
Attacher l'instance à l'IP flottantes:	14
Topologie de réseau :	15 3-
User Authentification :	16
Création de l'Admin «arifallahoussama» :	16
Connexion de l'admin «arifallahoussama »:	16
<b>Création d'un utilisateur «abouchouarmohamed » :</b>	<b>17 Les</b>
deux utilisateurs sont bien créés :	17 4-
Compute :	18
<b>Création de l'image Cirros :</b>	<b>18 Le</b>
réseau privé :	Erreur ! Signet non défini.
Lancer une instance :	19
L'instance a été bien créé :	21
<b>Les informations de l'instance :</b>	<b>21 5-</b>
Volume :	22
Création d'un volume :	22
<b>Attacher le volume à l'instance :</b>	<b>22 6-</b>
Final :	23
L'instance est visible au public :	23

Lancement avec succès de l'instance : .....	23
Conclusion .....	24

# Introduction :

OpenStack permet aux utilisateurs de déployer des machines virtuelles et d'autres instances qui gèrent différentes tâches pour gérer un environnement cloud à la volée. Il facilite la mise à l'échelle horizontale, ce qui signifie que les tâches qui bénéficient de l'exécution simultanée peuvent facilement servir plus ou moins d'utilisateurs à la volée en faisant simplement tourner plus d'instances. Par exemple, une application mobile qui doit communiquer avec un serveur distant peut être en mesure de répartir le travail de communication avec chaque utilisateur sur plusieurs instances différentes, toutes communiquant entre elles mais évoluant rapidement et facilement à mesure que l'application gagne plus d'utilisateurs.

Et surtout, OpenStack est un logiciel open source, ce qui signifie que toute personne qui choisit d'accéder au code source, d'apporter les changements ou modifications dont elle a besoin, et de partager librement ces changements avec la communauté dans son ensemble. Cela signifie également qu'OpenStack a l'avantage de milliers de développeurs du monde entier travaillant en tandem pour développer le produit le plus solide, le plus robuste et le plus sécurisé possible.

## Les composants de OpenStack

### ■ Nova :

Est le principal moteur informatique derrière OpenStack. Il est utilisé pour déployer et gérer un grand nombre de machines virtuelles et d'autres instances pour gérer les tâches informatiques.

### Swift :

Est un système de stockage d'objets et de fichiers. Plutôt que l'idée traditionnelle de faire référence aux fichiers par leur emplacement sur un lecteur de disque, les développeurs peuvent plutôt se référer à un identifiant unique faisant référence au fichier ou à l'information et laisser OpenStack décider où stocker ces informations. Cela facilite la mise à l'échelle, car les développeurs n'ont pas à se soucier de la capacité d'un seul système derrière le

logiciel. Il permet également au système, plutôt qu'au développeur, de se soucier de la meilleure façon de s'assurer que les données sont sauvegardées en cas de défaillance d'une machine ou d'une connexion réseau.

### **Cinder :**

Est un composant de stockage en bloc, qui est plus analogue à la notion traditionnelle d'un ordinateur pouvant accéder à des emplacements spécifiques sur un lecteur de disque. Cette façon plus traditionnelle d'accéder aux fichiers peut être importante dans les scénarios où la vitesse d'accès aux données est la considération la plus importante.

### **Neutron :**

Fournit la capacité de mise en réseau pour OpenStack. Il permet de garantir que chacun des composants d'un déploiement OpenStack peut communiquer rapidement et efficacement les uns avec les autres.

### **Horizon :**

Est le tableau de bord derrière OpenStack. Il s'agit de la seule interface graphique d'OpenStack, donc pour les utilisateurs qui souhaitent essayer OpenStack, il s'agit peut-être du premier composant qu'ils «voient». Les développeurs peuvent accéder individuellement à tous les composants d'OpenStack via une interface de programmation d'application (API), mais le tableau de bord fournit aux administrateurs système un aperçu de ce qui se passe dans le cloud et le gère selon les besoins.

### **Keystone :**

Fournit des services d'identité pour OpenStack. Il s'agit essentiellement d'une liste centrale de tous les utilisateurs du cloud OpenStack, mise en correspondance avec tous les services fournis par le cloud, qu'ils sont autorisés à utiliser. Il fournit plusieurs moyens d'accès, ce qui signifie que les développeurs peuvent facilement mapper leurs méthodes d'accès utilisateur existantes contre Keystone.

### **Glance :**

Fournit des services d'imagerie à OpenStack. Dans ce cas, "images" fait référence aux images (ou copies virtuelles) des disques durs. Glance permet à ces images d'être utilisées comme modèles lors du déploiement de nouvelles

instances de machine virtuelle.

### Ceilometer :

Fournit des services de télémétrie, qui permettent au cloud de fournir des services de facturation aux utilisateurs individuels du cloud. Il conserve également un décompte vérifiable de l'utilisation du système par chaque utilisateur de chacun des différents composants d'un cloud OpenStack. Pensez à la mesure et aux rapports d'utilisation.

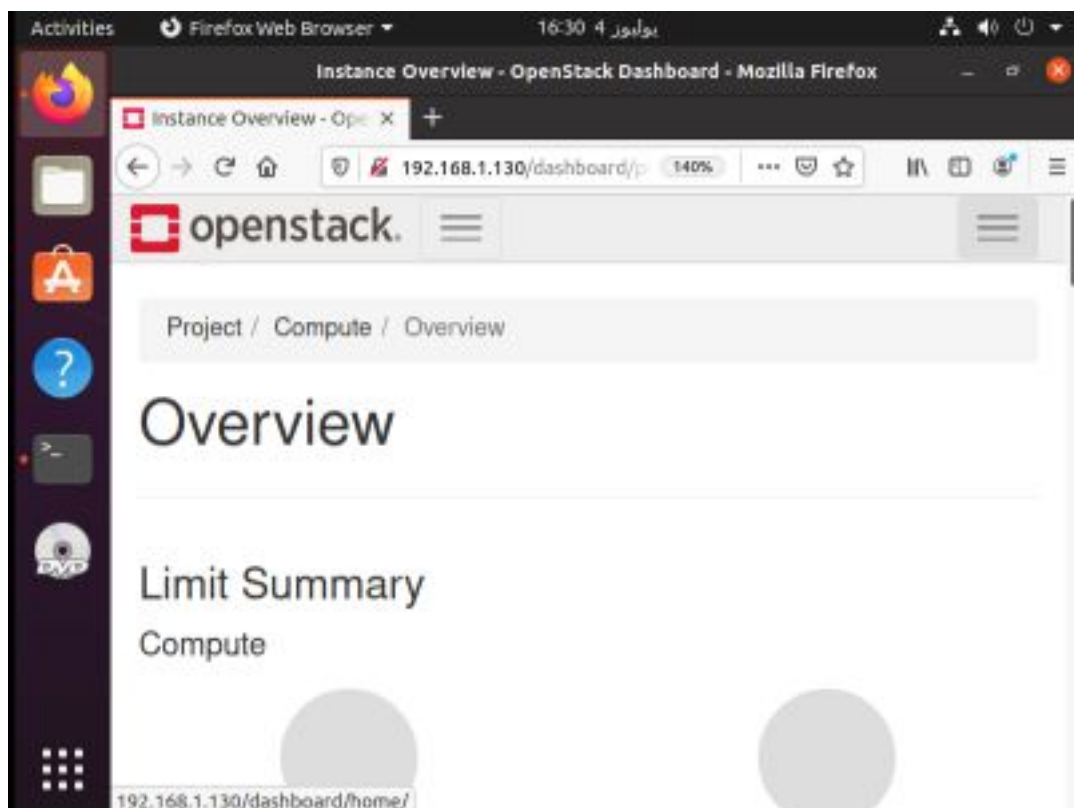
### Heat :

est le composant d'orchestration d'OpenStack, qui permet aux développeurs de stocker les exigences d'une application cloud dans un fichier qui définit les ressources nécessaires pour cette application. De cette façon, il aide à gérer l'infrastructure nécessaire à l'exécution d'un service cloud.

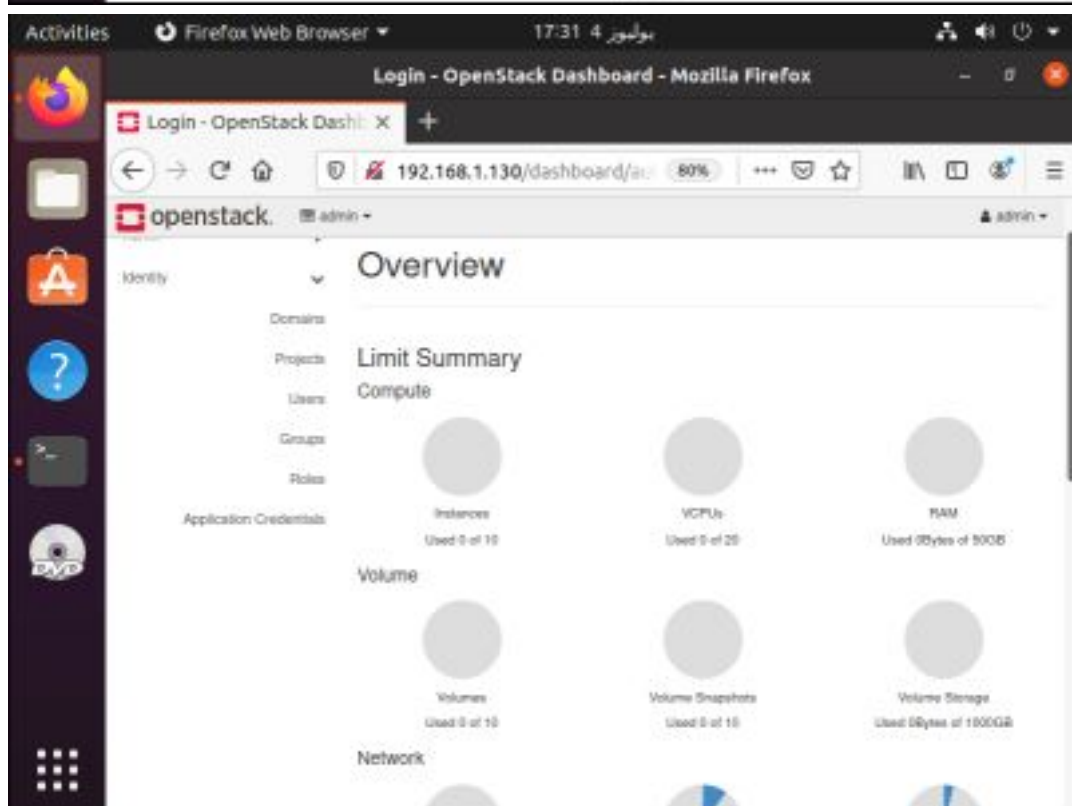
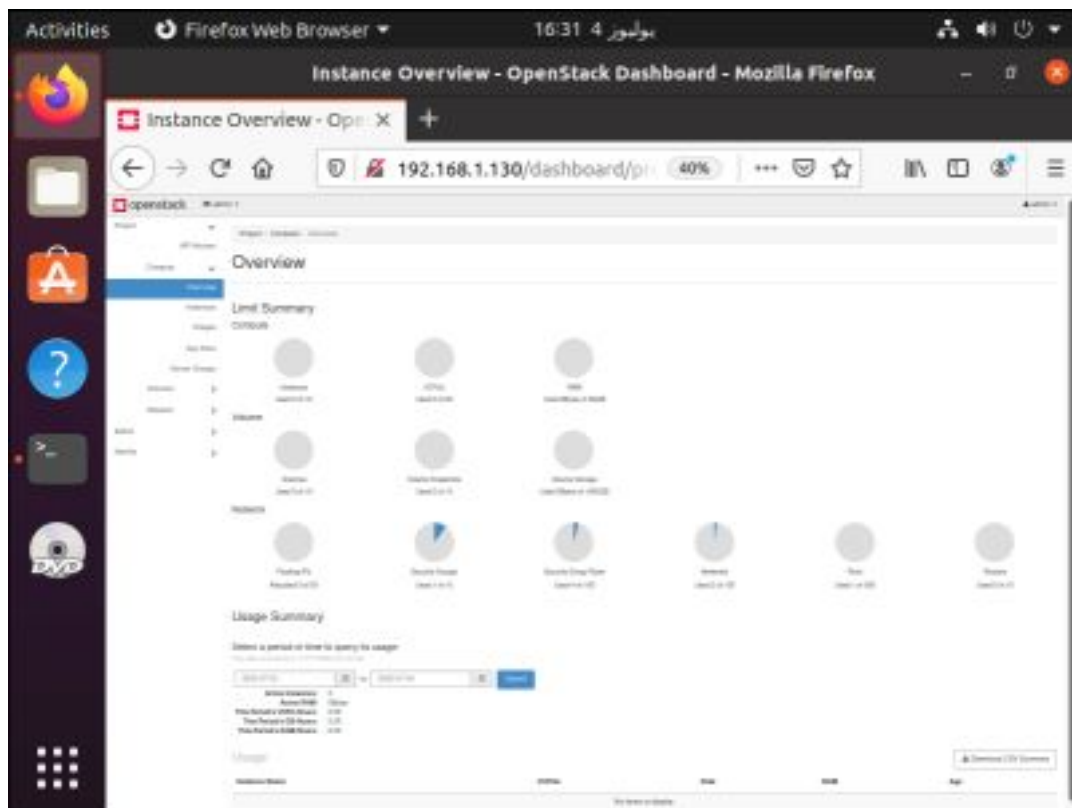
#### 1- Configuration :

##### OpenStack Dashboard :

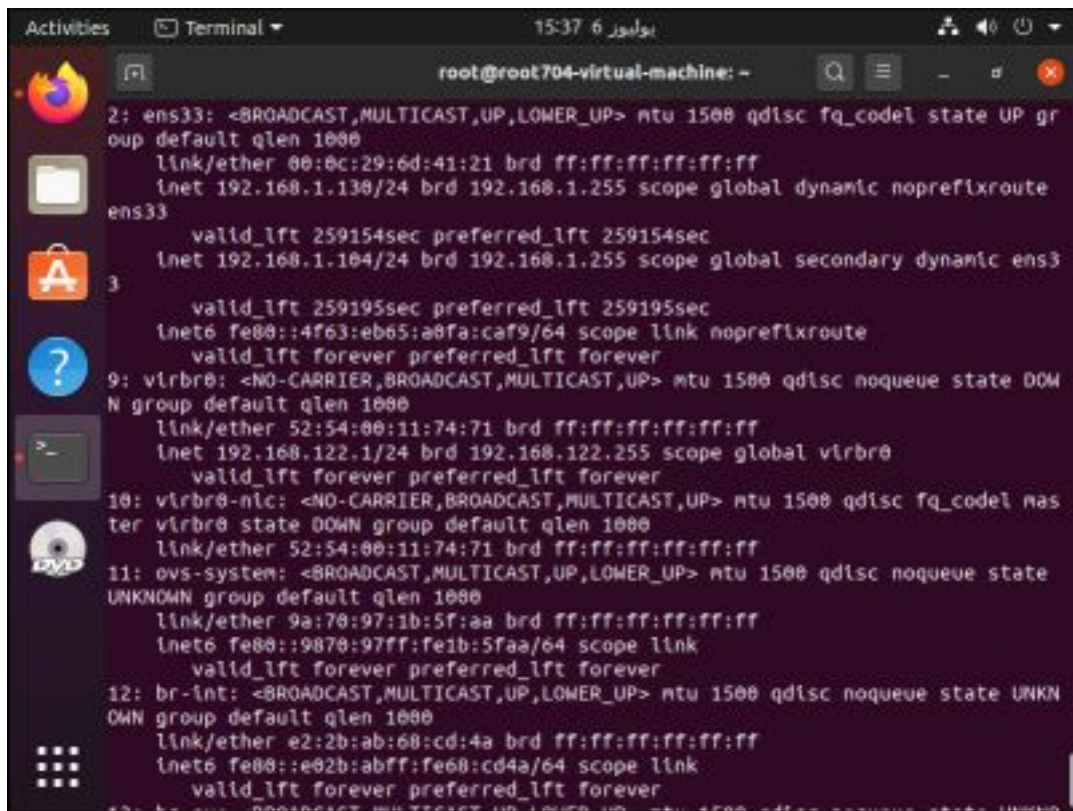
Après l'installation d'UBUNTU, on installe l'Openstack.



Afficher le dashboard.



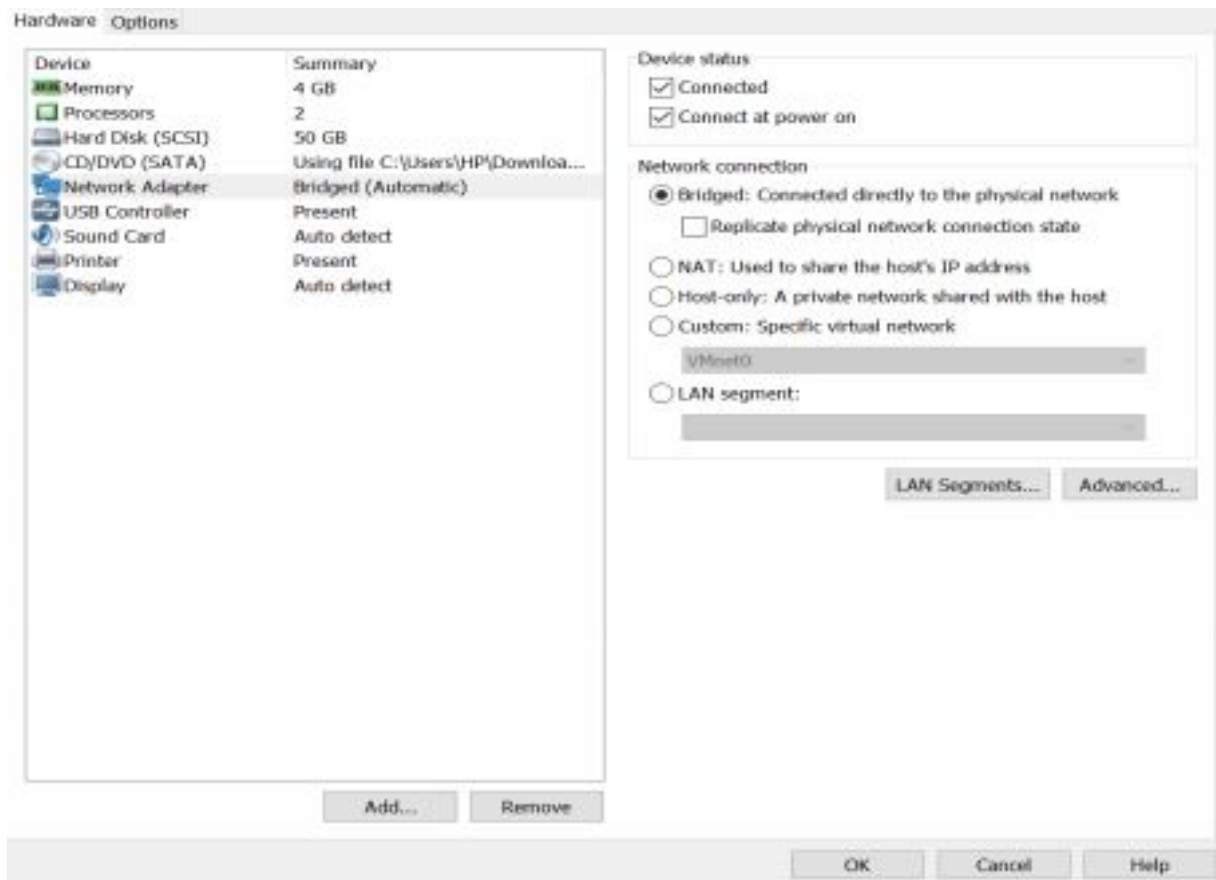
L'adresse ip du serveur :



```
root@root704-virtual-machine: ~
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:6d:41:21 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.130/24 brd 192.168.1.255 scope global dynamic noprefixroute ens33
        valid_lft 259154sec preferred_lft 259154sec
    inet 192.168.1.104/24 brd 192.168.1.255 scope global secondary dynamic ens3
        valid_lft 259195sec preferred_lft 259195sec
    inet6 fe80::4f63:eb65:a8fa:caf9/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
9: virbr0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default qlen 1000
    link/ether 52:54:00:11:74:71 brd ff:ff:ff:ff:ff:ff
    inet 192.168.122.1/24 brd 192.168.122.255 scope global virbr0
        valid_lft forever preferred_lft forever
10: virbr0-nic: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc fq_codel master virbr0 state DOWN group default qlen 1000
    link/ether 52:54:00:11:74:71 brd ff:ff:ff:ff:ff:ff
11: ovs-system: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UNKNOWN group default qlen 1000
    link/ether 9a:70:97:1b:5f:aa brd ff:ff:ff:ff:ff:ff
    inet6 fe80::9870:97ff:fe1b:5faa/64 scope link
        valid_lft forever preferred_lft forever
12: br-int: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UNKNOWN group default qlen 1000
    link/ether e2:2b:ab:60:cd:4a brd ff:ff:ff:ff:ff:ff
    inet6 fe80::e02b:abff:fe60:cd4a/64 scope link
        valid_lft forever preferred_lft forever
```

Configuration :

Pour notre cas on spécifie l'adaptateur réseau en Bridge afin de permettre l'accès au serveur à partir du windows.



## Connexion à OpenStack :

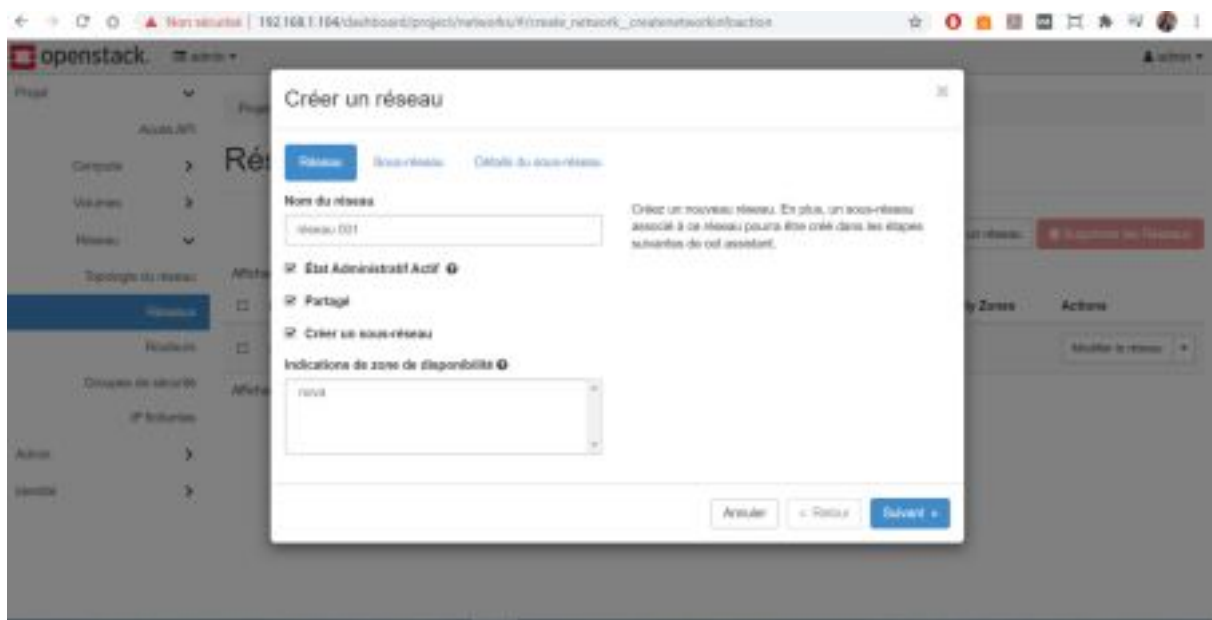
Connexion à l'Openstack utilisant l'adresse ip du serveur



## 2- Networks :

Création d'un réseau :





Création du sous-réseau.

**Créer un réseau**

[Réseau](#) **[Sous-réseau](#)** [Détails du sous-réseau](#)

**Nom du sous-réseau**

**Adresse réseau Source**

**Pool d'adresses\***

**Masque Réseau**

**Version IP**

**Adresse IP de la passerelle ?**

☐ Désactiver la passerelle

Crée un sous-réseau associé à un réseau. Vous devez entrer une "Adresse réseau" et une "Adresse IP de la passerelle" valide. Si vous n'entrez pas d'"Adresse IP de la passerelle", la première valeur (IP) de votre réseau sera assignée par défaut. Si vous ne souhaitez pas de passerelle, veuillez cocher "Désactiver la passerelle". Cliquez sur l'onglet "Détails Sous-réseaux" pour configurer des options avancées.

[Annuler](#) [Retour](#) **[Suivant >](#)**

Les réseaux :

# Réseaux

Nom

Filtrer

+ Créer un réseau

Supprimer les Réseaux

Affichage de 2 éléments

<input type="checkbox"/>	Name	Subnets Associated	Shared	External	Status	Admin State	Availability Zones	Actions
<input type="checkbox"/>	public	ipv6-public-subnet 2001:db8::54 public-subnet 172.24.4.0/24	non	Oui	Active	Actif	nova	Modifier le réseau
<input type="checkbox"/>	mon réseau privé 001	sous_reseau 801 10.0.0.0/26	Oui	non	Active	Actif	nova	Modifier le réseau

Affichage de 2 éléments

Project

Accès API

Compute

Volumes

Réseaux

Topologie du réseau

Routage

Groupes de sécurité

IP Subnet

Admin

Identité

Project / Réseau / Réseaux / public

public

Modifier le réseau

Vue d'ensemble

Sous-réseaux

Ports

Nom public

ID 6e996c518c0542ad6706e44505ae20e

ID du Project 18a7c0485b5d4262a3484aded5817d2

Statut Active

Etat Administrateur Actif

Partagé non

Réseau externe Oui

MTU 1500

Réseau fournisseur

Type de Réseau : flat

Réseau Physique : public

ID du segment --

Les sous-réseaux :

openstack

admin

Project

Accès API

Compute

Volumes

Réseaux

Topologie du réseau

Routage

Groupes de sécurité

IP Subnet

Admin

Identité

Project / Réseau / Réseaux / public

public

Modifier le réseau

Vue d'ensemble

Sous-réseaux

Ports

Sous-réseaux

Filtrer

+ Créer un sous-réseau

Supprimer les Sous-Réseaux

Affichage de 2 éléments

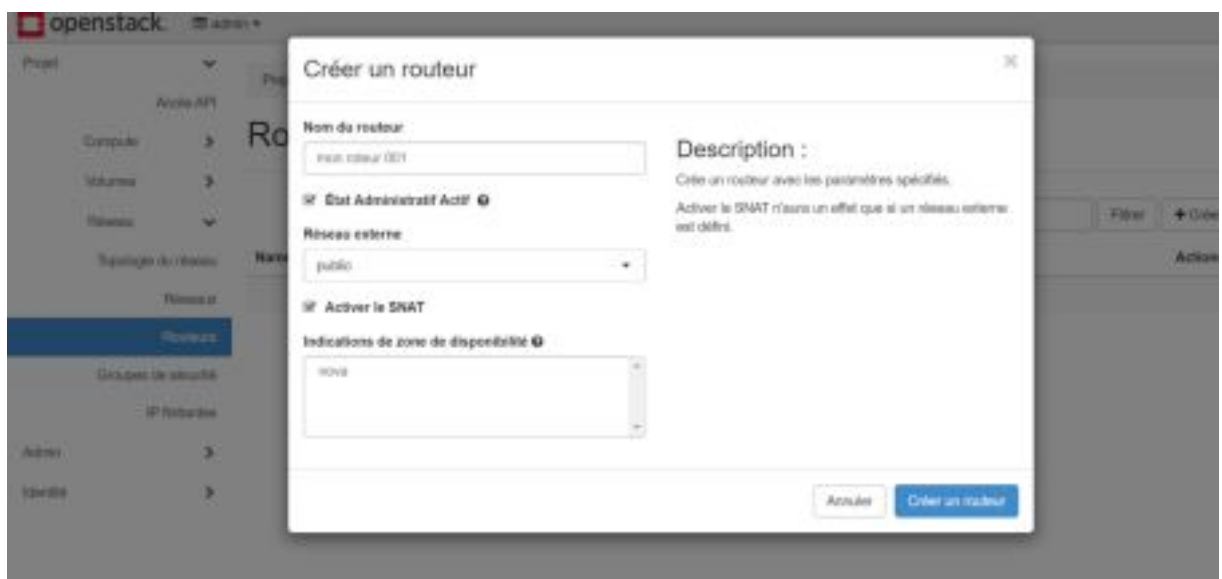
<input type="checkbox"/>	Name	Network Address	IP Version	Gateway IP	Actions
<input type="checkbox"/>	public-subnet	172.24.4.0/24	IPv4	172.24.4.1	Éditer le sous-réseau
<input type="checkbox"/>	ipv6-public-subnet	2001:db8::54	IPv6	2001:db8::2	Éditer le sous-réseau

Affichage de 2 éléments

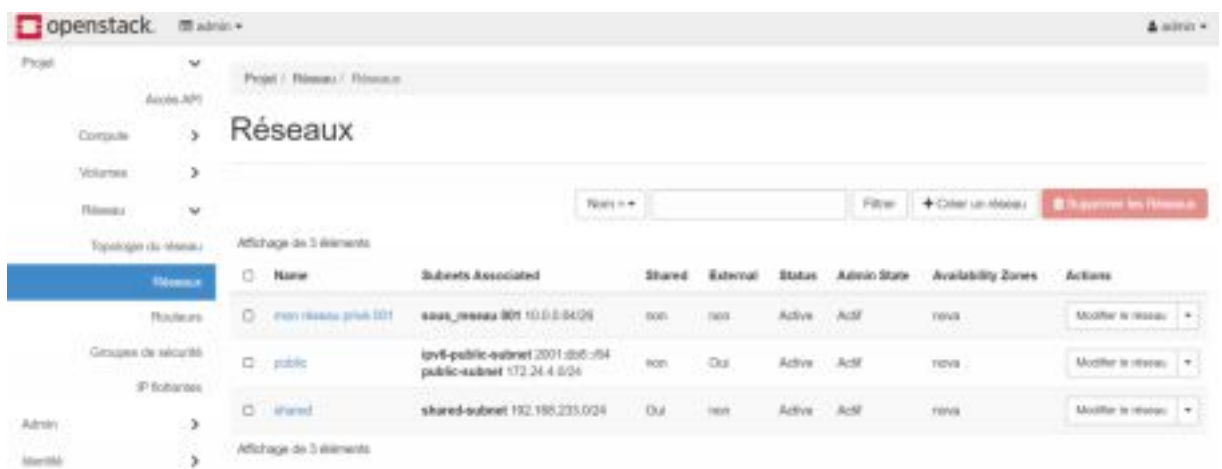
Création des groupes de sécurité pour contrôler les flux entrant sortant :



Création d'un routeur :



Ensemble des réseaux créés :



IP flottantes pour permettre l'accès à l'instance à l'aide de la connexion internet avec le compte arifallahoussama on va créer plus tard:



Attacher l'instance à l'IP flottantes:





Topologie de réseau créé:



3- User Authentication (Keystone) :

Création de l'Admin «arifallahoussama» :



Connexion de l'admin «arifallahoussama »:



Création d'un utilisateur «abouchouarmohamed » :



Les deux utilisateurs sont bien créés :



#### 4- Compute :

Création de l'image Cirros :



Lancer une instance :







L'instance a été bien créé :



Les informations de l'instance :



5- Volume avec le compte arifallahoussama :

Création d'un volume avec une taille d' 1Gib :





Attacher le volume à l'instance :



6- Lancement de l'instance :

L'instance est visible au public :



Lancement avec succès de l'instance :



## Conclusion

Le logiciel OpenStack contrôle de vastes pools de ressources de calcul, de stockage et de mise en réseau dans un centre de données, gérés via un tableau

de bord ou via l'API OpenStack. OpenStack fonctionne avec les technologies d'entreprise et open source populaires, ce qui le rend idéal pour les applications hétérogènes.