

[www.webhi.com/register.php/?language=french](https://www.webhi.com/register.php/?language=french)[Afficher le panier \(https://www.webhi.com/cart.php/?language=french\)](https://www.webhi.com/cart.php/?language=french)<https://www.webhi.com/?language=french>

Vous avez une question? Écrivez-la ici et appuyer sur entrée

## Guide complet pour installer OpenStack manuellement sur Linux

📁 **Administration système Linux** (<https://www.webhi.com/how-to/fr/tutorial/administration-os-linux/>), **Virtualisation et informatique en nuage** (<https://www.webhi.com/how-to/fr/tutorial/virtualisation-informatique-en-nuage-cloud/>)



### Table des matières

1. Introduction
2. Prérequis
3. Configuration de l'environnement
4. Installer et configurer les composants OpenStack
  - Keystone (Service d'identité)
  - Glance (Service d'image)
  - Nova (Service de calcul)
  - Neutron (Service de réseau)
  - Cinder (Service de stockage en bloc)
  - Horizon (Tableau de bord)
5. Vérification et dépannage
6. Conclusion

## 1. Introduction

**OpenStack** est une plateforme open-source utilisée pour construire et gérer des clouds publics et privés. Ce guide vous guide à travers une installation manuelle d'OpenStack sur un serveur Linux.

## 2. Prérequis

Assurez-vous que votre système répond aux exigences suivantes :

- Une installation propre d'une distribution Linux prise en charge (par exemple, Ubuntu 20.04 LTS ou CentOS 8).
- Au moins 8 Go de RAM et 2 CPU.
- Accès root ou sudo.
- Paramètres réseau correctement configurés.

## 3. Configuration de l'environnement

Mettre à jour et mettre à niveau votre système

```
$ sudo apt update && sudo apt upgrade -y
```

Installer les paquets essentiels

```
$ sudo apt install -y python3-pip python3-dev libffi-dev gcc libssl-dev
```

## 4. Installer et configurer les composants OpenStack

### a. Keystone (Service d'identité)

#### 1. Installer Keystone

```
$ sudo apt install -y keystone
```

#### 2. Configurer Keystone

Modifier le fichier `/etc/keystone/keystone.conf` :

```
$ sudo nano /etc/keystone/keystone.conf
```

Définissez les valeurs suivantes :

```
[database]
connection = mysql+pymysql://keystone:KEYSTONE_DBPASS@controller/keystone
[token]
provider = fernet
```

#### 3. Peupler la base de données du service d'identité

```
$ sudo keystone-manage db_sync
```

#### 4. Initialiser les dépôts de clés Fernet

```
$ sudo keystone-manage fernet_setup --keystone-user keystone --keystone-group keystone
$ sudo keystone-manage credential_setup --keystone-user keystone --keystone-group keystone
```

#### 5. Démarrer le service d'identité

```
$ sudo keystone-manage bootstrap --bootstrap-password ADMIN_PASS \  
--bootstrap-admin-url http://controller:5000/v3/ \  
--bootstrap-internal-url http://controller:5000/v3/ \  
--bootstrap-public-url http://controller:5000/v3/ \  
--bootstrap-region-id RegionOne
```

## 6. Configurer le serveur Apache HTTP

Modifier le fichier `/etc/apache2/apache2.conf` :

```
$ sudo nano /etc/apache2/apache2.conf
```

Ajouter la ligne suivante :

```
ServerName controller
```

Activer les services Apache et WSGI :

```
$ sudo service apache2 restart
```

## b. Glance (Service d'image)

### 1. Installer Glance

```
$ sudo apt install -y glance
```

### 2. Configurer Glance

Modifier le fichier `/etc/glance/glance-api.conf` :

```
$ sudo nano /etc/glance/glance-api.conf
```

Définissez les valeurs suivantes :

```
[database]  
connection = mysql+pymysql://glance:GLANCE_DBPASS@controller/glance  
[keystone_authtoken]  
www_authenticate_uri = http://controller:5000  
auth_url = http://controller:5000  
memcached_servers = controller:11211  
auth_type = password  
project_domain_name = Default  
user_domain_name = Default  
project_name = service  
username = glance  
password = GLANCE_PASS  
[paste_deploy]  
flavor = keystone
```

### 3. Peupler la base de données du service d'image

```
$ sudo glance-manage db_sync
```

### 4. Redémarrer le service Glance

```
$ sudo service glance-api restart
```

## c. Nova (Service de calcul)

### 1. Installer Nova

```
$ sudo apt install -y nova-api nova-conductor nova-consoleauth nova-novncproxy nova-scheduler
```

## 2. Configurer Nova

Modifier le fichier `/etc/nova/nova.conf` :

```
$ sudo nano /etc/nova/nova.conf
```

Définissez les valeurs suivantes :

```
[database]
connection = mysql+pymysql://nova:NOVA_DBPASS@controller/nova
[api_database]
connection = mysql+pymysql://nova:NOVA_DBPASS@controller/nova_api
[keystone_authtoken]
www_authenticate_uri = http://controller:5000
auth_url = http://controller:5000
memcached_servers = controller:11211
auth_type = password
project_domain_name = Default
user_domain_name = Default
project_name = service
username = nova
password = NOVA_PASS
[DEFAULT]
my_ip = MANAGEMENT_INTERFACE_IP_ADDRESS
use_neutron = True
firewall_driver = nova.virt.firewall.NoopFirewallDriver
[vnc]
enabled = True
vncserver_listen = $my_ip
vncserver_proxyclient_address = $my_ip
[glance]
api_servers = http://controller:9292
```

## 3. Peupler les bases de données du calcul

```
$ sudo nova-manage api_db sync
$ sudo nova-manage cell_v2 map_cell0
$ sudo nova-manage cell_v2 create_cell --name=cell1
$ sudo nova-manage db sync
```

## 4. Redémarrer les services Nova

```
$ sudo service nova-api restart
$ sudo service nova-scheduler restart
$ sudo service nova-conductor restart
$ sudo service nova-novncproxy restart
```

## d. Neutron (Service de réseau)

### 1. Installer Neutron

```
$ sudo apt install -y neutron-server neutron-plugin-ml2 neutron-linuxbridge-agent neutron-dhcp-agent neutro
```

### 2. Configurer Neutron

Modifier le fichier `/etc/neutron/neutron.conf` :

```
$ sudo nano /etc/neutron/neutron.conf
```

Définissez les valeurs suivantes :

```
[database]
connection = mysql+pymysql://neutron:NEUTRON_DBPASS@controller/neutron
[keystone_authtoken]
www_authenticate_uri = http://controller:5000
auth_url = http://controller:5000
memcached_servers = controller:11211
auth_type = password
project_domain_name = Default
user_domain_name = Default
project_name = service
username = neutron
password = NEUTRON_PASS
[DEFAULT]
core_plugin = ml2
service_plugins = router
allow_overlapping_ips = True
```

Modifier le fichier `/etc/neutron/plugins/ml2/ml2_conf.ini` :


```
$ sudo nano /etc/neutron/plugins/ml2/ml2_conf.ini
```

Définissez les valeurs suivantes :

```
[ml2]
type_drivers = flat,vlan
tenant_network_types =
mechanism_drivers = linuxbridge
extension_drivers = port_security
[ml2_type_flat]
flat_networks = provider
[securitygroup]
enable_ipset = True
```

### 3. Peupler la base de données du réseau

```
$ sudo neutron-db-manage --config-file /etc/neutron/neutron.conf --config-file /etc/neutron/plugins/ml2/ml2_
```



### 4. Redémarrer les services Neutron

```
$ sudo service neutron-server restart
$ sudo service neutron-linuxbridge-agent restart
$ sudo service neutron-dhcp-agent restart
$ sudo service neutron-metadata-agent restart
```

## e. Cinder (Service de stockage en bloc)

### 1. Installer Cinder

```
$ sudo apt install -y cinder-api cinder-scheduler
```

### 2. Configurer Cinder

Modifier le fichier `/etc/cinder/cinder.conf` :

```
$ sudo nano /etc/cinder/cinder.conf
```

Définissez les valeurs suivantes :

```
[database]
connection = mysql+pymysql://cinder:CINDER_DBPASS@controller/cinder
[keystone_authtoken]
www_authenticate_uri = http://controller:5000
auth_url = http://controller:5000
memcached_servers = controller:11211
auth_type = password
project_domain_name = Default
user_domain_name = Default
project_name = service
username = cinder
password = CINDER_PASS
[DEFAULT]
my_ip = MANAGEMENT_INTERFACE_IP_ADDRESS
```

### 3. Remplir la base de données de stockage par blocs

```
$ sudo cinder-manage db sync
```

### 4. Redémarrer les services Cinder

```
$ sudo service cinder-api restart
$ sudo service cinder-scheduler restart
```

## f. Horizon (Tableau de bord)

### 1. Installer Horizon

```
$ sudo apt install -y openstack-dashboard
```

### 2. Configurer Horizon

Modifier le fichier `/etc/openstack-dashboard/local_settings.py` :

```
$ sudo nano /etc/openstack-dashboard/local_settings.py
```

Définissez les valeurs suivantes :

```

OPENSTACK_HOST = "controller"
ALLOWED_HOSTS = ['*']
SESSION_ENGINE = 'django.contrib
.sessions.backends.cache'
CACHES = {
    'default': {
        'BACKEND': 'django.core.cache.backends.memcached.MemcachedCache',
        'LOCATION': 'controller:11211',
    }
}
OPENSTACK_KEYSTONE_URL = "http://s:5000/v3" % OPENSTACK_HOST
OPENSTACK_KEYSTONE_MULTIDOMAIN_SUPPORT = True
OPENSTACK_API_VERSIONS = {
    "identity": 3,
    "image": 2,
    "volume": 3,
}
OPENSTACK_KEYSTONE_DEFAULT_DOMAIN = "Default"

```

### 3. Redémarrer le serveur Apache

```
$ sudo service apache2 restart
```

## 5. Vérification et dépannage

### 1. Vérifier l'état du service

Assurez-vous que tous les services OpenStack sont en cours d'exécution :

```
$ sudo systemctl status <nom-du-service>
```

### 2. Vérifier le service Keystone

Authentifiez-vous à l'aide de l'interface de ligne de commande OpenStack :

```

$ export OS_USERNAME=admin
$ export OS_PASSWORD=ADMIN_PASS
$ export OS_PROJECT_NAME=admin
$ export OS_USER_DOMAIN_NAME=Default
$ export OS_PROJECT_DOMAIN_NAME=Default
$ export OS_AUTH_URL=http://controller:5000/v3
$ export OS_IDENTITY_API_VERSION=3

```

Vérifier le service Keystone :

```
$ openstack token issue
```

### 3. Vérifier le service Glance

Télécharger une image exemple :

```

$ wget http://download.cirros-cloud.net/0.5.2/cirros-0.5.2-x86_64-disk.img
$ openstack image create "cirros" --file cirros-0.5.2-x86_64-disk.img --disk-format qcow2 --container-format raw

```

Lister les images :

```
$ openstack image list
```

### 4. Vérifier le service Nova

Créer un flavor :

```
$ openstack flavor create --id 0 --vcpus 1 --ram 64 --disk 1 m1.nano
```

Lister les flavors :

```
$ openstack flavor list
```

## 5. Vérifier le service Neutron

Créer un réseau :

```
$ openstack network create demo-net  
$ openstack subnet create --network demo-net --subnet-range 10.0.0.0/24 demo-subnet
```

Lister les réseaux :

```
$ openstack network list
```

## 6. Vérifier le service Cinder

Créer un volume :

```
$ openstack volume create --size 1 demo-volume
```

Lister les volumes :

```
$ openstack volume list
```

## 7. Vérifier le tableau de bord Horizon

Ouvrez un navigateur web et accédez à `http://<controller-ip>/dashboard` . Connectez-vous avec vos identifiants admin.

## 6. Conclusion

L'installation manuelle d'OpenStack peut être complexe en raison de ses nombreux composants et dépendances. Cependant, en suivant ce guide, vous devriez être en mesure de configurer un environnement OpenStack fonctionnel.

Pour une personnalisation et une mise à l'échelle supplémentaires, consultez la documentation officielle d'OpenStack et les ressources communautaires.

## Conseils de dépannage

- **Journaux** : Vérifiez toujours les journaux situés dans `/var/log/` pour obtenir des messages d'erreur détaillés.
- **Réseau** : Assurez-vous que votre configuration réseau est correcte et qu'il n'y a pas de conflits.
- **Services** : Utilisez `systemctl` pour démarrer, arrêter et vérifier l'état des services.
- **Bases de données** : Vérifiez les connexions aux bases de données et que les tables sont correctement remplies.

## Ressources supplémentaires

- Documentation officielle OpenStack (<https://docs.openstack.org/>)
- Communauté OpenStack (<https://www.openstack.org/community/>)

En suivant ces étapes, vous devriez disposer d'un environnement OpenStack fonctionnel adapté à vos besoins.

<< Comment installer AbanteCart : Un tutoriel complet (<https://www.webhi.com/how-to/fr/installer-abantecart-tutoriel-linux/>)  
Gestion d'Active Directory avec PowerShell : Un Guide Complet >> (<https://www.webhi.com/how-to/fr/gestion-active-directory-powershell-tutoriel/>)



# One Reply to “Guide complet pour installer OpenStack manuellement sur Linux”



**KEMTSA Bydias a dit:**

juillet 9, 2024 à 11:43 (<https://www.webhi.com/how-to/fr/tutoriel-complet-installer-configurer-openstack-manuellement-sur-linux-debian-ubuntu/#comment-2582>)

Très instructif merci

Répondre

## Laisser un commentaire

Commentaire\*

Votre nom\*

E-mail

☐

Enregistrer mon nom, mon e-mail et mon site dans le navigateur pour mon prochain commentaire.

Soumettre le commentaire

## Catégories

Administration système Linux (<https://www.webhi.com/how-to/fr/tutorial/administration-os-linux/>)

Administration système Windows (<https://www.webhi.com/how-to/fr/tutorial/administration-os-windows/>)

Bases de données (<https://www.webhi.com/how-to/fr/tutorial/base-de-donnee/>)

Certificats SSL (<https://www.webhi.com/how-to/fr/tutorial/certificat-ssl/>)

CMS & Développement Web (<https://www.webhi.com/how-to/fr/tutorial/cms-developpement-web/>)

Intelligence Artificielle (<https://www.webhi.com/how-to/fr/tutorial/intelligence-artificielle/>)

Logiciels et Outils (<https://www.webhi.com/how-to/fr/tutorial/logiciels-outils/>)

Panneaux d'hébergement web (<https://www.webhi.com/how-to/fr/tutorial/panneau-hebergement-web/>)

Sécurité (<https://www.webhi.com/how-to/fr/tutorial/cybersecurite/>)

Serveurs de messagerie (<https://www.webhi.com/how-to/fr/tutorial/serveurs-mail/>)

Serveurs web (<https://www.webhi.com/how-to/fr/tutorial/serveurs-web/>)

Virtualisation et informatique en nuage (<https://www.webhi.com/how-to/fr/tutorial/virtualisation-informatique-en-nuage-cloud/>)

## Tutoriels récents

Configurer un serveur SMTP sécurisé avec Exim (<https://www.webhi.com/how-to/fr/configurer-un-serveur-smtp-securise-avec-exim/>)

Comprendre et configurer la gestion des files d'attente Postfix (<https://www.webhi.com/how-to/fr/comprendre-et-configurer-la-gestion-des-files-dattente-postfix-2/>)

Comprendre et configurer la gestion des files d'attente Postfix (<https://www.webhi.com/how-to/fr/comprendre-et-configurer-la-gestion-des-files-dattente-postfix/>)


Comment installer et configurer Mailman pour les listes de diffusion (<https://www.webhi.com/how-to/fr/comment-installer-et-configurer-mailman-pour-les-listes-de-diffusion/>)


Introduction à la génération d'images par IA avec l'API DALL-E (<https://www.webhi.com/how-to/fr/introduction-a-la-generation-dimages-par-ia-avec-lapi-dall%c2%b7e/>)

---

## Changer la langue

 Français (/how-to/fr)

 English (/how-to)

 العربية (/how-to/ar)

## Nos produits

Hébergement web

(<https://www.webhi.com/hebergement-site-web>)

Serveur Dédié

(<https://www.webhi.com/serveur-dedie>)

Serveur Linux VPS

(<https://www.webhi.com/serveur-virtuel-vps-linux>)

Serveur VPS Windows

RDP

(<https://www.webhi.com/serveur-virtuel-vps-windows>)

Hostname DNS

dynamique

(<https://www.webhi.com/host-dns-dynamique>)

Création Pro des sites

web

(<https://www.webhi.com/creation-site-web>)

Création d'application

mobile

(<https://www.webhi.com/creation-application-mobile>)

Nom de domaine

(<https://www.webhi.com/enregistrer-nom-domaine>)

Certificat SSL

(<https://www.webhi.com/certificat-ssl-maroc>)

Licence cPanel

(<https://www.webhi.com/licence-cpanel-whm>)

Licence Plesk

(<https://www.webhi.com/achat-licence-plesk>)

## A propos

A propos de l'entreprise

(<https://www.webhi.com/a-propos>)

Tutoriels & Guides

(<https://www.webhi.com/how-to/fr>)

Conditions générales

de vente

(<https://www.webhi.com/terms-of-service>)

Contactez nous

(<https://www.webhi.com/contact.php>)

## Suivez-nous

 Webhi.Technology

(<https://www.facebook.com/webhi>)

 @WebHiTechnology

(<https://twitter.com/WebHiTechnic>)

WebHi Technology SARL Maroc - All rights reserved.