

Tutoriel : Installer Postgresql + Odoo

I - Avoir accès à internet

Comment faire pour avoir accès à internet :

En premier lieu grâce à une nouvel VM entrer la commande `ifconfig`

```
root@mysql:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 3a:66:66:65:66:36
          inet adr:192.168.11.20  Bcast:192.168.11.255  Masque:255.255.255.0
          adr inet6: fe80::3866:66ff:fe65:6636/64  Scope:Lien
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:2528 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1649 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 lg file transmission:1000
          RX bytes:2961577 (2.8 MiB)  TX bytes:117947 (115.1 KiB)

lo        Link encap:Boucle locale
          inet adr:127.0.0.1  Masque:255.0.0.0
          adr inet6: ::1/128  Scope:Hôte
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:497 errors:0 dropped:0 overruns:0 frame:0
          TX packets:497 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 lg file transmission:0
          RX bytes:156025 (152.3 KiB)  TX bytes:156025 (152.3 KiB)
```

Interfaces (`nano /etc/network/interfaces`) afin d'attribuer une adresse ip s'il n'y on a pas.

```
GNU nano 2.2.6      Fichier : /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug eth0
iface eth0 inet static
    address 192.168.11.20
    netmask 255.255.255.0
    network 192.168.11.0
    broadcast 192.168.11.255
    gateway 192.168.11.254
```

`Route -n` (savoir si y a une passerelle)

```
root@mysql:~# route -n
Table de routage IP du noyau
Destination      Passerelle      Genmask          Indic Metric Ref       Use Iface
0.0.0.0          192.168.11.254  0.0.0.0          UG    0      0        0 eth0
192.168.11.0    0.0.0.0         255.255.255.0   U     0      0        0 eth0
root@mysql:~# _
```

Ping gw (pingué la passerelle)

```
root@mysql:~# ping 192.168.11.254
PING 192.168.11.254 (192.168.11.254) 56(84) bytes of data.
64 bytes from 192.168.11.254: icmp_seq=1 ttl=64 time=0.703 ms
64 bytes from 192.168.11.254: icmp_seq=2 ttl=64 time=0.745 ms
64 bytes from 192.168.11.254: icmp_seq=3 ttl=64 time=0.628 ms
^C
--- 192.168.11.254 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 0.628/0.692/0.745/0.048 ms
```

Apt.conf (nano /etc/apt/apt.conf)

```
GNU nano 2.2.6      Fichier : /etc/apt/apt.conf
Acquire::http::Proxy "http://172.16.0.27:3128";
Acquire::https::Proxy "https://172.16.0.27:3128";
Acquire::ftp::Proxy "ftp://172.16.0.27:3128";
```

Aptitude update (mise à jour)

Refaire un Ping gw (pingué la passerelle)

II - Installation Postgresql

Installer postgresql avec la commande suivante :

Aptitude install postgresql-9.4

Après l'installation modifier les fichiers de configuration de Postgresql afin de pouvoir écouter à distance notre serveur Odoon

Modifier le fichier : /etc/postgresql/9.4/main/postgresql.conf

```
# - Connection Settings -
listen_addresses = '*'
-
```

Modifier la ligne `listen_addresses = 'localhost'` par celle ci-dessus par une * qui permettra de faire écouter sur toutes les adresses.

Puis aller dans le fichier `/etc/postgresql/9.4/main/pg_hba.conf` puis ajouter ses deux lignes au fichier de configuration.

```
host      all      all      all md5
host      all      all      0.0.0.0/0 trust
```

Pour permettre a tous les hosts de pouvoir se connecter sur le serveur.

III - Installation d'Odoon

Sur une nouvelle machine, où vous allez installer Apache2 a l'aide de la commande suivantes :

Aptitude install apache2

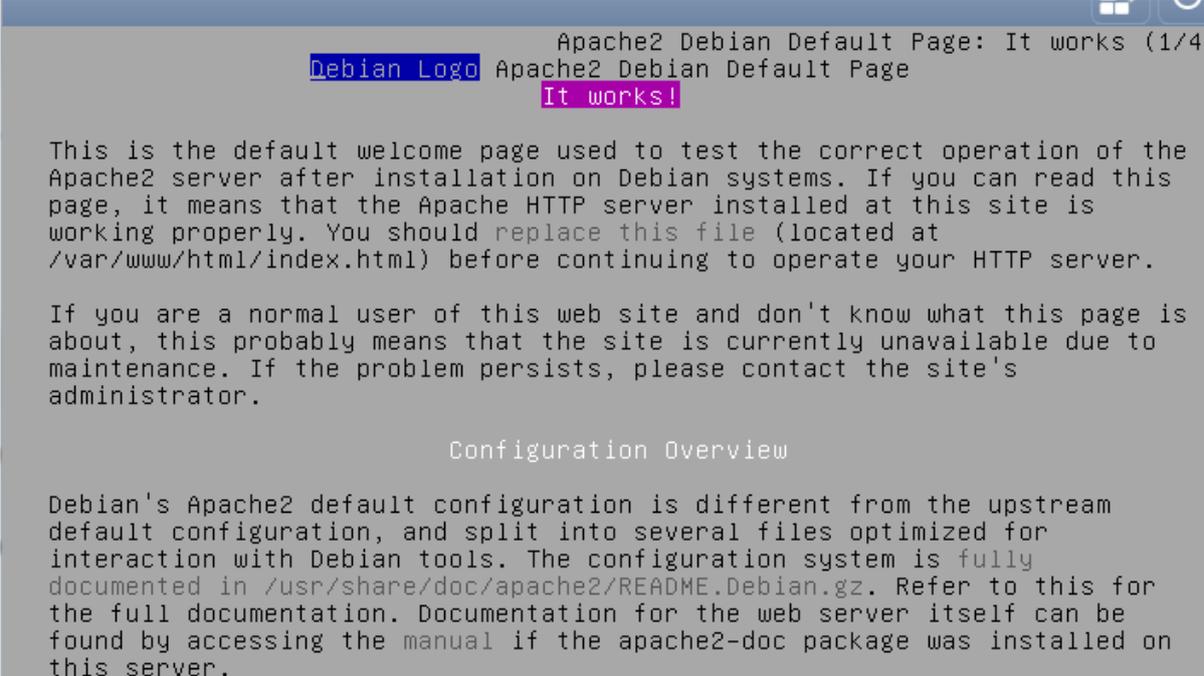
Et installer le paquet Elinks

Aptitude install Elinks

Testé le serveur apache2 taper la commande suivante :

Elinks http://localhost

Si votre service web est actif vous devriez afficher ceci :



```
Apache2 Debian Default Page: It works (1/4)
Debian Logo Apache2 Debian Default Page
It works!

This is the default welcome page used to test the correct operation of the
Apache2 server after installation on Debian systems. If you can read this
page, it means that the Apache HTTP server installed at this site is
working properly. You should replace this file (located at
/var/www/html/index.html) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is
about, this probably means that the site is currently unavailable due to
maintenance. If the problem persists, please contact the site's
administrator.

Configuration Overview

Debian's Apache2 default configuration is different from the upstream
default configuration, and split into several files optimized for
interaction with Debian tools. The configuration system is fully
documented in /usr/share/doc/apache2/README.Debian.gz. Refer to this for
the full documentation. Documentation for the web server itself can be
found by accessing the manual if the apache2-doc package was installed on
this server.
```

Maintenant nous allons utiliser l'utilitaire Wget (Aptitude intall Wget si vous ne le possédez pas)

Puis modifier le proxy du lycée (ici) dans le fichier suivant :

Nano /etc/wgetrc

```
# Set this to on to use timestamping by default:
#timestamping = off

# It is a good idea to make Wget send your email address in a `From:'
# header with your request (so that server administrators can contact
# you in case of errors). Wget does *not* send `From:' by default.
#header = From: Your Name <username@site.domain>

# You can set up other headers, like Accept-Language. Accept-Language
# is *not* sent by default.
#header = Accept-Language: en

# You can set the default proxies for Wget to use for http, https, and ftp.
# They will override the value in the environment.
https_proxy = http://proxy.vinci-melun.org:3128/
http_proxy = http://proxy.vinci-melun.org:3128/
ftp_proxy = http://proxy.vinci-melun.org:3128/

# If you do not want to use proxy at all, set this to off.
```

Une fois le proxy mis nous devant maintenant écrire cette série de commande :

```
# wget -O - https://nightly.odoo.com/odoo.key | apt-key add -
# echo "deb http://nightly.odoo.com/9.0/nightly/deb/ ." >> /etc/apt/sources.list
# apt-get update && apt-get install odoo
```

/!\ ATTENTION /!\ Si vous avez une erreur ici c'est souvent parce que une des phrases est mal orthographiée ou vous vous êtes trompez dans le –O

Une fois cela fait, nous devons aller dans le fichier de configuration d'Odoo :

Nano /etc/odoo/openerp-server.conf

```
[options]
; This is the password that allows database operations:
; admin_passwd = admin
db_host = 192.168.11.20
db_port = 5432
db_user = odoo
db_password = sio
addons_path = /usr/lib/python2.7/dist-packages/openerp/addons
```

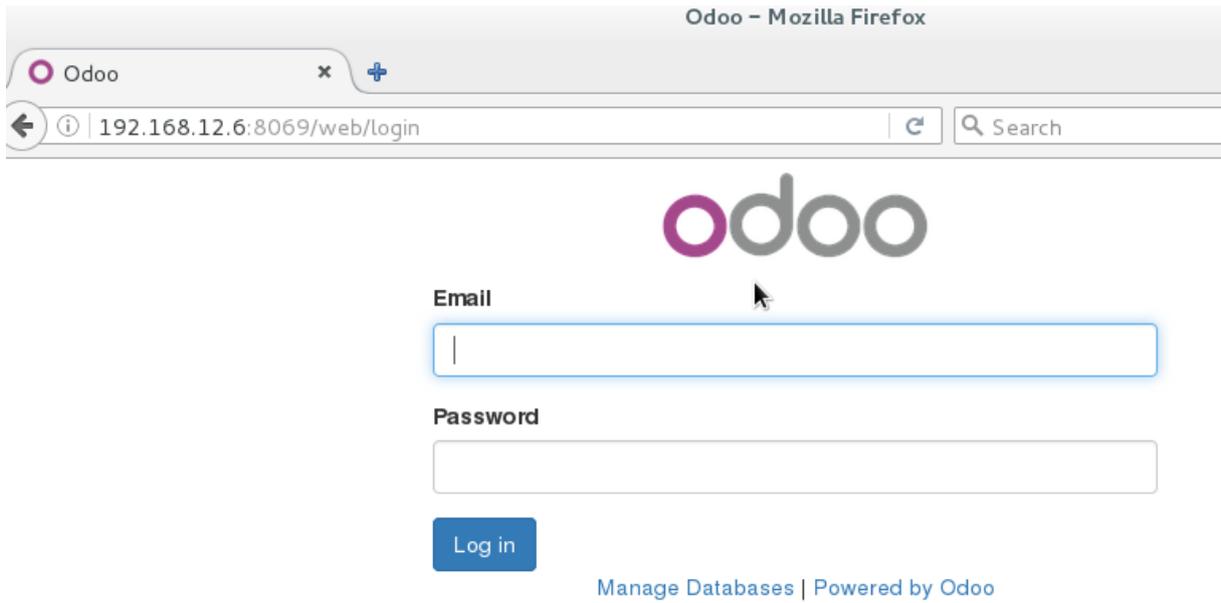
Dans db_host entrée l'ip de votre serveur postgresql

Db_port le port par défaut de postgresql qui est 5432

Faire la commande suivante : service odoo restart

Vous entrez alors vos identifiants de votre base de donné.

Maintenant ouvrez un navigateur internet, avec un interface graphique puis taper le nom de votre machine accompagné du port : 8069



Vous arrivez ici, normalement les ID sont : **admin admin**

Et voilà l'installation est termin , nous pouvons maintenant installer des modules.

