

[Skip to content](#)

[Search](#)



disamTECHbench

[Linux/Ubuntu](#)

Installing and Configuring Multiple Tomcat Server Instances In Ubuntu Server.

July 17, 2014 Madura Dissanayake

Hello all, today I 'm taking you through the steps to configure multiple server instances of Apache Tomcat server in a single machine. In this case I have taken Ubuntu server 12.04 version to be demonstrated. So first of all let's look at the what are the steps for the server.

1. Availability of JDK and JRE.
2. Installing the tomcat server.
3. Create multiple instances.

In here let's check whether that your server has the JDK and JRE.

```
$ sudo java -version
```

In here let's check whether that your server has the JDK and JRE.

```
$sudo java -version
```

```
java version "1.6.0_31"  
OpenJDK Runtime Environment (IcedTea6 1.13.3) (6b31-1.13.3-1ubuntu1~0.12.04.2)  
OpenJDK 64-Bit Server VM (build 23.25-b01, mixed mode)
```

If u r getting the response like this you have already installed the JDK and JRE otherwise you have to install JDK and JRE. following commands will help you to install the latest version of JDK and JRE.

```
$ sudo apt-get update
$ sudo apt-get install default-jre
$ sudo apt-get install default-jdk
```

After installation is completed, u can check the JDK and JRE has been installed or not using java version command.

```
$ sudo java -version
```

Now, we have to set the JAVA_HOME variable path to the relevant directory, do the followings.

```
$ sudo nano ~/.bashrc
```

Then add the following line and reboot the system.

```
export JAVA_HOME=/usr/lib/jvm/default-java
```

After verified the JDK and JRE is installed correctly, let's move to the next step. In here let's install the apache tomcat with tomcat admin.

```
$ sudo apt-get install tomcat7 tomcat7-admin
```

Now you can test the tomcat is working using <http://myserver.com:8080> ,

It works !

If you're seeing this page via a web browser, it means you've setup Tomcat successfully. Congratulations!

This is the default Tomcat home page. It can be found on the local filesystem at:

```
/var/lib/tomcat7/webapps/ROOT/index.html
```

Tomcat7 veterans might be pleased to learn that this system instance of Tomcat is installed with CATALINA_HOME in /usr/share/tomcat7 and CATALINA_BASE in /var/lib/tomcat7, following the rules from /usr/share/doc/tomcat7-common/RUNNING.txt.gz.

You might consider installing the following packages, if you haven't already done so:

tomcat7-docs: This package installs a web application that allows to browse the Tomcat 7 documentation locally. Once installed, you can access it by clicking [here](#).

tomcat7-examples: This package installs a web application that allows to access the Tomcat 7 Servlet and JSP examples. Once installed, you can access it by clicking [here](#).

tomcat7-admin: This package installs two web applications that can help managing this Tomcat instance. Once installed, you can access the [manager webapp](#) and the [host-manager webapp](#).

NOTE: For security reasons, using the manager webapp is restricted to users with role "manager". The host-manager webapp is restricted to users with role "admin". Users are defined in /etc/tomcat7/tomcat-users.xml.

Image - 01

Now let's create multiple instances on Tomcat server called tomcat7-a, tomcat7-b, tomcat7-c. follow the commndes given below.

```

$ cd /var/lib/
$ sudo mkdir tomcat7-a
$ sudo mkdir tomcat7-b
$ sudo mkdir tomcat7-c

$ cd /var/lib/tomcat7-a
$ sudo mkdir conf temp webapps work logs
$ cd /var/lib/tomcat7-b
$ sudo mkdir conf temp webapps work logs
$ cd /var/lib/tomcat7-c

$ sudo mkdir conf temp webapps work logs
$ cd ..

$ sudo chown -R tomcat7:tomcat7 tomcat7-a tomcat7-b tomcat7-c

```

Now we have created the directories of new instances, in next step we need link main configurations files which are available in main tomcat7 server instance.

```

$ sudo ln- s /var/lib/tomcat7/conf/catalina.properties /var/lib/tomcat7-a/conf
$ sudo ln- s /var/lib/tomcat7/conf/catalina.properties /var/lib/tomcat7-b/conf
$ sudo ln- s /var/lib/tomcat7/conf/catalina.properties /var/lib/tomcat7-c/conf

$ sudo ln- s /var/lib/tomcat7/conf/web.xml /var/lib/tomcat7-a/conf
$ sudo ln- s /var/lib/tomcat7/conf/ web.xml /var/lib/tomcat7-b/conf
$ sudo ln- s /var/lib/tomcat7/conf/ web.xml /var/lib/tomcat7-c/conf

$ sudo ln- s /var/lib/tomcat7/conf/policy.d/ /var/lib/tomcat7-a/conf
$ sudo ln- s /var/lib/tomcat7/conf/policy.d/ /var/lib/tomcat7-b/conf
$ sudo ln- s /var/lib/tomcat7/conf/policy.d/ /var/lib/tomcat7-c/conf

```

Now the necessary files have been copied. So in next step let's configure the server port, shutdown port and the connector port on server.xml file.

```

$ sudo cp /var/lib/tomcat7/conf/server.xml /var/lib/tomcat7-a/conf/
$ sudo cp /var/lib/tomcat7/conf/server.xml /var/lib/tomcat7-b/conf/
$ sudo cp /var/lib/tomcat7/conf/server.xml /var/lib/tomcat7-c/conf/

```

Open the server.xml files separatly and change the above mentioned port numbers.

```
<Server port="8005" shutdown="SHUTDOWN">
```

```

.
.
.

```

```

<Connector port="8080" protocol="HTTP/1.1"
    connectionTimeout="20000"
    URIEncoding="UTF-8"
    redirectPort="8443" />

```

```

.
.
.

```

```
<Connector port="8009" protocol="AJP/1.3" redirectPort="8443" />
```

```
.  
.
```

After setting up the port number on server.xml file , you have to change the configuration on the service file which is in "/etc/init.d/" . In here copy the main tomcat7 service file and make three copies and change the appropriate values.

```
$ cd /etc/init.d/  
$ sudo cp tomcat7 tomcat7-a  
$ sudo cp tomcat7 tomcat7-b  
$ sudo cp tomcat7 tomcat7-c
```

Then open those files and change the necessary configurations sample shown given below.

```
DESC="Tomcat servlet engine"  
CATALINA_BASE=/var/lib/$NAME  
CATALINA_PID="/var/run/$NAME.pid"
```

After configurations are completed save the files and start the tomcat instances.

```
$ sudo service tomcat7-a start  
$ sudo service tomcat7-b start  
$ sudo service tomcat7-c start
```

If services are not started properly, please make sure the configuration file changes are correct. The set those main tomcat instance services to the startup services.

```
$ sudo update -rc.d tomcat7-a defaults  
$ sudo update -rc.d tomcat7-b defaults  
$ sudo update -rc.d tomcat7-c defaults
```

Now try the instances using the appropriate URLs with port numbers. If you have set the port numbers like 8181, 8282, 8383, use the following URLs.

<http://localhost:8181> ,<http://localhost:8282>, <http://localhost:8383>

Thank You.

Advertisement

Advertisements

Report this ad

Report this ad

[Linux/Ubuntu](#)

Odoo 8 | Installing And Configuring in Ubuntu Server.

July 23, 2014 Madura Dissanayake

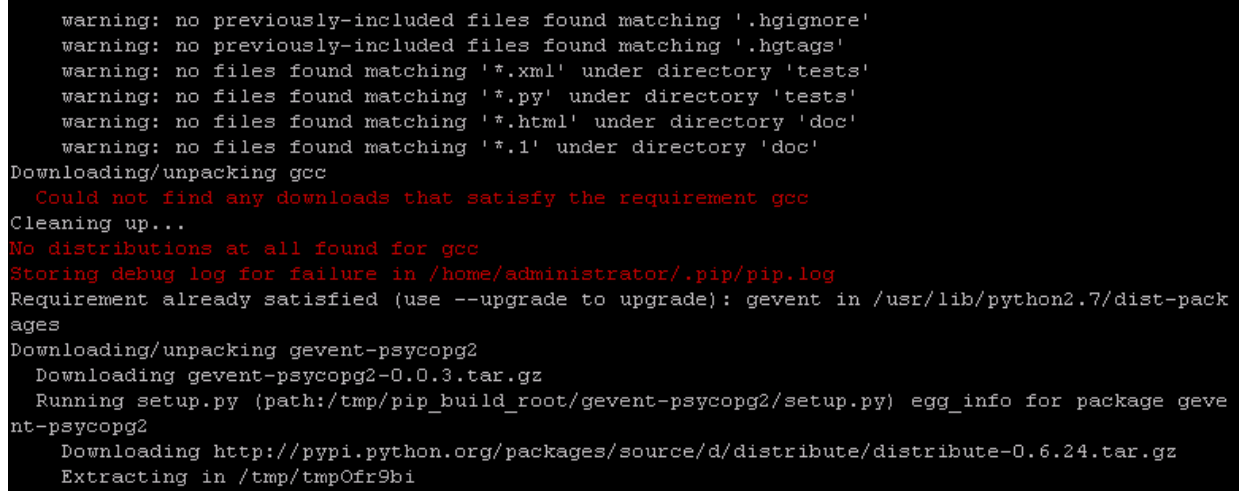
Hello All, Here, I'm going to show you how to install new version of "Odoo" ERP system on Ubuntu server. New version is "Odoo v8", which contains more reliable features. Now it has been very easy to install and configure. Before installing the Odoo v8 let's keep a clean installation on Ubuntu latest server version. The Ubuntu latest server version is Ubuntu Server 14.04. In here I have a clean Installation of Ubuntu Server 14.04.

Then let's start the steps to download, install and configure Odoo v8 in the new Ubuntu server.

Follow these steps.

```
$ wget https://raw.githubusercontent.com/lukebranch/openerp-install-  
scripts/master/odoo-saas4/ubuntu-14-04/odoo_install.sh  
$ sudo sh odoo_install.sh
```

While the script is running if you are getting errors like in the image-01.



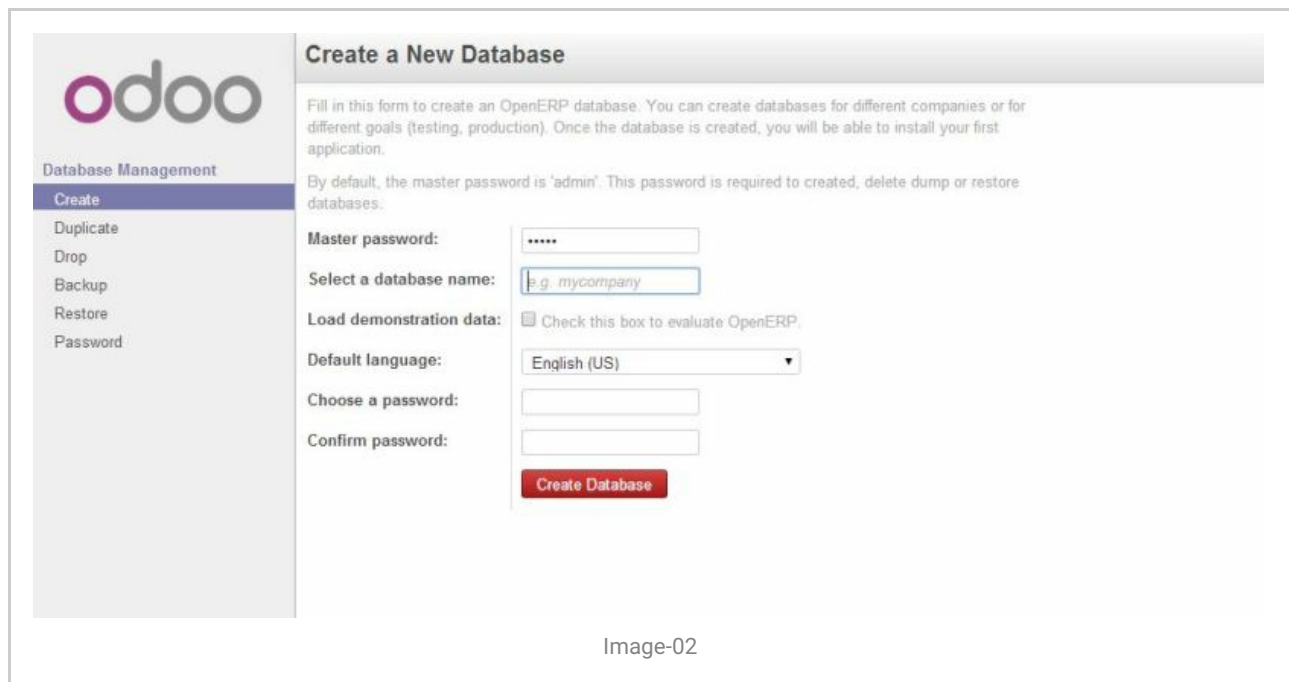
```
warning: no previously-included files found matching '.hgignore'  
warning: no previously-included files found matching '.hgtags'  
warning: no files found matching '*.xml' under directory 'tests'  
warning: no files found matching '*.py' under directory 'tests'  
warning: no files found matching '*.html' under directory 'doc'  
warning: no files found matching '*.1' under directory 'doc'  
Downloading/unpacking gcc  
  Could not find any downloads that satisfy the requirement gcc  
Cleaning up...  
No distributions at all found for gcc  
Storing debug log for failure in /home/administrator/.pip/pip.log  
Requirement already satisfied (use --upgrade to upgrade): gevent in /usr/lib/python2.7/dist-pack  
ages  
Downloading/unpacking gevent-psycpg2  
  Downloading gevent-psycpg2-0.0.3.tar.gz  
  Running setup.py (path:/tmp/pip_build_root/gevent-psycpg2/setup.py) egg_info for package geve  
nt-psycpg2  
    Downloading http://pypi.python.org/packages/source/d/distribute/distribute-0.6.24.tar.gz  
    Extracting in /tmp/tmpOfr9b1
```

Image-01

You can manually download and install the required per-requisites. In here I have shown how to install gcc and python with modules.

```
$ sudo apt-get install gcc  
$ sudo apt-get install python-dateutil python-docutils python-feedparser python-gdata  
python-jinja2 python-ldap python-libxslt1 python-lxml python-mako python-mock python-  
openid python-psycpg2 python-psutil python-pybabel python-pychart python-pydot python-  
pyparsing python-reportlab python-simplejson python-tz python-unittest2 python-  
vatnumber python-vobject python-webdav python-werkzeug python-xlwt python-yaml python-  
zsi
```

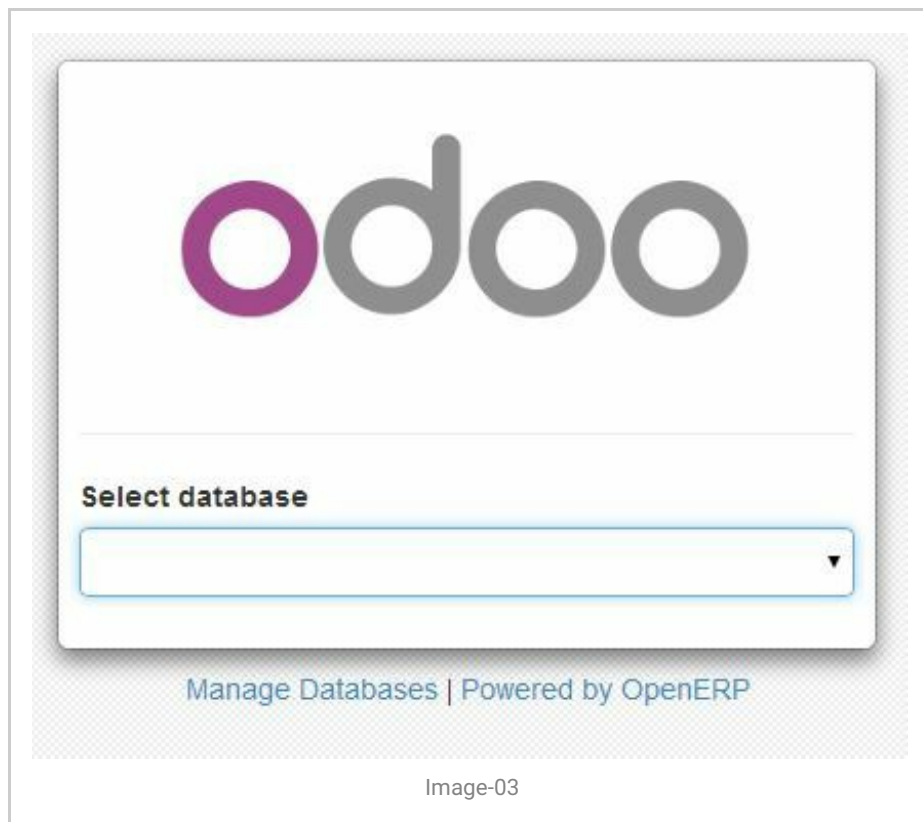
In here all the per-requisites are downloaded and installed and you will be asked to reboot the machine. After rebooting the machine, try to load the the Odoo home page using <http://youripaddress.com:8072>, then you will be appeared page like in image-02.



The screenshot shows the 'Create a New Database' form in the Odoo interface. On the left is a sidebar with the Odoo logo and a 'Database Management' menu containing options like 'Create', 'Duplicate', 'Drop', 'Backup', 'Restore', and 'Password'. The main area has a title 'Create a New Database' and instructions: 'Fill in this form to create an OpenERP database. You can create databases for different companies or for different goals (testing, production). Once the database is created, you will be able to install your first application.' Below this, it states: 'By default, the master password is 'admin'. This password is required to created, delete dump or restore databases.' The form fields include: 'Master password:' (a text box with four asterisks), 'Select a database name:' (a text box with 'e.g. mycompany' as a placeholder), 'Load demonstration data:' (a checkbox labeled 'Check this box to evaluate OpenERP.'), 'Default language:' (a dropdown menu showing 'English (US)'), 'Choose a password:' (a text box), and 'Confirm password:' (a text box). A red 'Create Database' button is at the bottom right of the form.

Image-02

Then fill the appropriate fields and create whatever the databases as you wish. After that you can logging to the account that you have already created.



The screenshot shows a dialog box with the Odoo logo at the top. Below the logo is a section titled 'Select database' with a dropdown menu. At the bottom of the dialog box, it says 'Manage Databases | Powered by OpenERP'.

Image-03

As in the image-02 you can select the database and then enter the logging details into relevant fields as shown in the Image-04.



Finally you will see the home page of the OpenERP.

Advertisements

Post navigation

[Previous Post](#)

[Hosting Virtual Hosts with Nginx Server on Ubuntu](#)

[Next Post](#)

[Odoo 8 | Installing And Configuring in Ubuntu Server.](#)

Archives

M	T	W	T	F	S	S
	1	2	3	4	5	6
7	8	9	10	11	12	13

M	T	W	T	F	S	S
14	15	16	<u>17</u>	18	19	20
21	22	<u>23</u>	24	25	26	27
28	29	30	31			

July 2014

Categories

New Posts

Recent Posts: disamTECHbench

Advertisements

Report this ad