Before You Begin[Permalink](https://linode.com/docs/websites/cms/install-odoo-10-on-ubuntu-16-04/#before-you-begin)

1. Complete the [Getting Started](https://linode.com/docs/getting-started) guide.
2. Follow the [Securing Your Server](https://linode.com/docs/security/securing-your-server/) guide to create a standard user account, harden SSH access and remove unnecessary network services. **Do not** follow the *Configuring a Firewall*section in the Securing Your Server Guide–we will configure the firewall for an Odoo production server in the next section.

This guide will use sudo wherever possible.

1. Update your packages:
2. sudo apt update && sudo apt upgrade

Install PostgreSQL Database and Server Dependencies[Permalink](https://linode.com/docs/websites/cms/install-odoo-10-on-ubuntu-16-04/#install-postgresql-database-and-server-dependencies)

Install the PostgreSQL database, Python, and other necessary server libraries:

sudo apt install git python-pip postgresql postgresql-server-dev-9.5 python-all-dev python-dev python-setuptools libxml2-dev libxslt1-dev libevent-dev libsasl2-dev libldap2-dev pkg-config libtiff5-dev libjpeg8-dev libjpeg-dev zlib1g-dev libfreetype6-dev liblcms2-dev liblcms2-utils libwebp-dev tcl8.6-dev tk8.6-dev python-tk libyaml-dev fontconfig

Create a PostgreSQL User[Permalink](https://linode.com/docs/websites/cms/install-odoo-10-on-ubuntu-16-04/#create-a-postgresql-user)

1. Switch to the postgres user:
2. sudo su - postgres
3. Set a strong password for the database user and record it in a secure location, you’ll need it in the following sections:
4. createuser odoo -U postgres -dRSP
5. Press **CTRL+D** to exit the postgres user session.

**Note**

If you want to run multiple Odoo instances on the same Linode remember to check your PostgreSQL client configuration file (as of the date this guide is published, located at /etc/postgresql/9.5/main/pg\_hba.conf) and modify it according your needs.

Create an Odoo User[Permalink](https://linode.com/docs/websites/cms/install-odoo-10-on-ubuntu-16-04/#create-an-odoo-user)

In order to separate Odoo from other services, create a new Odoo system user to run its processes:

sudo adduser --system --home=/opt/odoo --group odoo

**Note**

If you’re running multiple Odoo versions on the same Linode, you may want to use different users and directories for each instance.

Configure Logs[Permalink](https://linode.com/docs/websites/cms/install-odoo-10-on-ubuntu-16-04/#configure-logs)

For logging, Ubuntu 16.04 uses systemd and journald by default. With that in mind, you can set up Odoo 10 logs in several ways. We’ll consider two scenarios in this guide. Your choice of option will depend on your particular needs:

1. To use journals **and** a separate Odoo log file at the same time, create the corresponding directory:
2. sudo mkdir /var/log/odoo
3. To only use journald for logging, you don’t need to create any directory.

Install Odoo[Permalink](https://linode.com/docs/websites/cms/install-odoo-10-on-ubuntu-16-04/#install-odoo)

Clone the Odoo files onto your server:

sudo git clone <https://github.com/OCA/OCB.git> --depth 1 --branch 10.0 --single-branch /opt/odoo

**Note**

Using git offers great flexibility. When a new upgrade is available, pull the new branch. You can even install a different version alongside the production one, just change the destination directory and the --branch X.x flag. Before upgrading, remember to make a full backup of your database and custom files.

Install Dependencies for Odoo Applications[Permalink](https://linode.com/docs/websites/cms/install-odoo-10-on-ubuntu-16-04/#install-dependencies-for-odoo-applications)

Before your Odoo applications are ready to use, you’ll need to install some dependencies. We’ll use the Python package manager, pip, instead of apt to guarantee that you install the correct versions. We’ll also refrain from using Ubuntu’s packaged versions of [Wkhtmltopdf](http://wkhtmltopdf.org/) and [node-Less](http://lesscss.org/).

Be sure to follow the steps in this section as your limited, non-root user (not the odoo user).

Install Python Dependencies[Permalink](https://linode.com/docs/websites/cms/install-odoo-10-on-ubuntu-16-04/#install-python-dependencies)

Install the required Python libraries:

sudo pip install -r /opt/odoo/doc/requirements.txt

sudo pip install -r /opt/odoo/requirements.txt

These commands use the requirements.txt files provided with your Odoo installation to ensure you’re getting the correct versions of the packages your applications depend on.

Install Less CSS via Node.js and npm[Permalink](https://linode.com/docs/websites/cms/install-odoo-10-on-ubuntu-16-04/#install-less-css-via-node-js-and-npm)

1. Download and run the nodejs installation script from nodesource:
2. sudo curl -sL https://deb.nodesource.com/setup\_4.x | sudo -E bash -
3. Now that our repository list is updated install nodejs using apt.
4. sudo apt install -y nodejs
5. Install a newer version of Less via npm, the Node.js package manager.
6. sudo npm install -g less less-plugin-clean-css

Install Stable Wkhtmltopdf Version[Permalink](https://linode.com/docs/websites/cms/install-odoo-10-on-ubuntu-16-04/#install-stable-wkhtmltopdf-version)

1. Navigate to a temporary directory:
2. cd /tmp
3. Download the recommended version of wkhtmltopdf for Odoo server. In this example, we use **0.12.1**. See the [Odoo repository](https://github.com/odoo/odoo/wiki/Wkhtmltopdf) for an up-to-date list of compatible versions:
4. sudo wget https://downloads.wkhtmltopdf.org/0.12/0.12.1/wkhtmltox-0.12.1\_linux-trusty-amd64.deb
5. Install the package using dpkg:
6. sudo dpkg -i wkhtmltox-0.12.1\_linux-trusty-amd64.deb
7. To ensure wkhtmltopdf functions properly, copy the binaries to a location in your executable path:
8. sudo cp /usr/local/bin/wkhtmltopdf /usr/bin
9. sudo cp /usr/local/bin/wkhtmltoimage /usr/bin

**Note**

While wkhtmltopdf version 0.12.2.4 is available in the official Ubuntu 16.04 repository, we don’t advise installing it from there due to the large number of dependencies including: xserver, gstreamer, libcups, wayland, qt5 and many more. There isn’t an official Xenial package from the project page yet, but the Trusty package from Ubuntu 14.04 is compatible as of this publication.

Odoo Server Configuration[Permalink](https://linode.com/docs/websites/cms/install-odoo-10-on-ubuntu-16-04/#odoo-server-configuration)

1. Copy the included configuration file to a more convenient location, changing its name to odoo-server.conf
2. sudo cp /opt/odoo/debian/odoo.conf /etc/odoo-server.conf
3. Next, modify the configuration file. The complete file should look similar to this, depending on your deployment needs:

**/etc/odoo-server.conf**

|  |  |
| --- | --- |
| 1 2 3 4 5 6 7 8 910 | [options]  admin\_passwd = admin  db\_host = False  db\_port = False  db\_user = odoo  db\_password = FALSE  addons\_path = /opt/odoo/addons  ;Uncomment the following line to enable a custom log  ;logfile = /var/log/odoo/odoo-server.log  xmlrpc\_port = 8069 |

* admin\_passwd = admin - This is the password that allows database operations. Be sure to change admin to something more secure.
* db\_host = False - Unless you plan to connect to a different database server address, leave this line untouched.
* db\_port = False - Odoo uses PostgreSQL default port 5432, change this only if you’re using custom PostgreSQL settings.
* db\_user = odoo - Name of the PostgreSQL database user. In this case we used the default name, but if you used a different name when creating your user, substitute that here.
* db\_password = FALSE - Change FALSE to the PostgreSQL password you created previously.
* addons\_path = - Modify this line to read: addons\_path = /opt/odoo/addons. Add </path/to/custom/modules> if you’re using custom modules, substituting your own path.
* Include the path to log files, and add a new line: logfile = /var/log/odoo/odoo-server.log. You can skip this line if you plan to only use journald for logging.
* Optionally, we could include a new line specifying the Odoo Frontend port used for connection: xmlrpc\_port = 8069. This only makes sense if you’re planning to run multiple Odoo instances (or versions) on the same server. For normal installation, you can skip this line and this instance of Odoo will connect by default to port 8069.

**Note**

As explained in the [Configure Logs](https://linode.com/docs/websites/cms/install-odoo-10-on-ubuntu-16-04/#configure-logs) section, you have many options for Odoo logging in Ubuntu 16.04. This configuration file assumes you’ll use Ubuntu system journals in addition to a custom log path.

Create an Odoo Service[Permalink](https://linode.com/docs/websites/cms/install-odoo-10-on-ubuntu-16-04/#create-an-odoo-service)

Create a systemd unit called odoo-server to allow your application to behave as a service. Create a new file at /lib/systemd/system/odoo-server.service and add the following contents:

**/lib/systemd/system/odoo-server.service**

|  |  |
| --- | --- |
| 1 2 3 4 5 6 7 8 91011121314151617 | [Unit]  Description=Odoo Open Source ERP and CRM  Requires=postgresql.service  After=network.target postgresql.service  [Service]  Type=simple  PermissionsStartOnly=true  SyslogIdentifier=odoo-server  User=odoo  Group=odoo  ExecStart=/opt/odoo/odoo-bin --config=/etc/odoo-server.conf --addons-path=/opt/odoo/addons/  WorkingDirectory=/opt/odoo/  StandardOutput=journal+console  [Install]  WantedBy=multi-user.target |

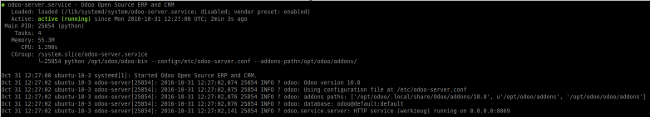
The most relevant line in this file is StandardOutput=journal+console. As configured in the example above, Odoo logs will be completely managed by the system journal (Option 2 in the [Configure Logs](https://linode.com/docs/websites/cms/install-odoo-10-on-ubuntu-16-04/#configure-logs) section). If you want a separate log file, omit that line and configure odoo-server.confaccordingly, specifying the location of your log file. Remember that journald will always capture main Odoo service activity (service start, stop, reboot, errors), using a separate log file will only exclude journal “info” messages like webserver messages, rendering engine, etc.

Change File Ownership and Permissions[Permalink](https://linode.com/docs/websites/cms/install-odoo-10-on-ubuntu-16-04/#change-file-ownership-and-permissions)

1. Change the odoo-server service permissions and ownership so only root can write to it, while the odoo user will only be able to read and execute it.
2. sudo chmod 755 /lib/systemd/system/odoo-server.service
3. sudo chown root: /lib/systemd/system/odoo-server.service
4. Since the odoo user will run the application, change its ownership accordingly.
5. sudo chown -R odoo: /opt/odoo/
6. If you chose to use a custom log, set the odoo user as the owner of log directory as well (this applies only if you decided to use a separate log file):
7. sudo chown odoo:root /var/log/odoo
8. Finally, protect the server configuration file. Change its ownership and permissions so no other non-root user can access it:
9. sudo chown odoo: /etc/odoo-server.conf
10. sudo chmod 640 /etc/odoo-server.conf

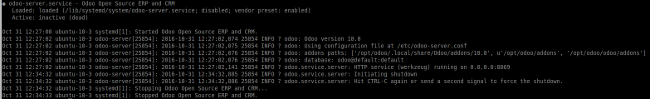
Test the Server[Permalink](https://linode.com/docs/websites/cms/install-odoo-10-on-ubuntu-16-04/#test-the-server)

1. Confirm that everything is working as expected. First, start the Odoo server:
2. sudo systemctl start odoo-server
3. Check the service status to make sure it’s running. This will also include journal logs. You can see an example output in the picture bellow:
4. sudo systemctl status odoo-server

[](https://linode.com/docs/assets/odoo_servicerunning.png)

1. Check the database journal to make sure there are no errors:
2. sudo journalctl -u postgresql
3. Verify that the server is able to stop properly:
4. sudo systemctl stop odoo-server
5. Run a service status check again to make sure there were no errors:
6. sudo systemctl status odoo-server

Your output should look similar to this:

[](https://linode.com/docs/assets/odoo_servicestopped.png)

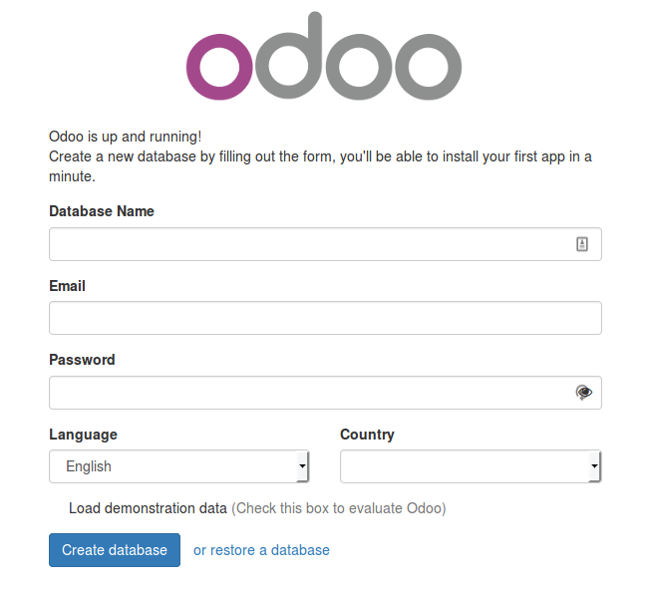
Enable the Odoo Service[Permalink](https://linode.com/docs/websites/cms/install-odoo-10-on-ubuntu-16-04/#enable-the-odoo-service)

1. If your system journal doesn’t indicate any problems, enable the odoo-server unit to start and stop with the server:
2. sudo systemctl enable odoo-server
3. Log into the Linode Manager and reboot your Linode to see if everything is working as expected.
4. Once restarted, log in via SSH and verify your journal messages:
5. sudo journalctl -u odoo-server

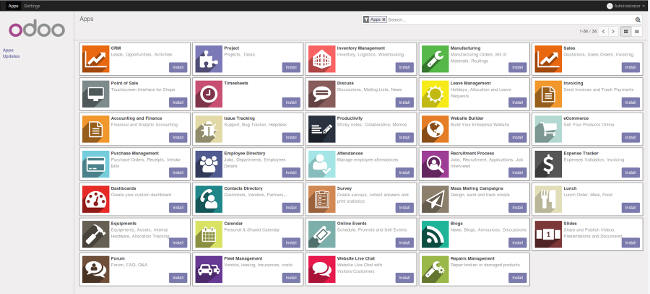
The output should include a message indicating that Odoo has started successfully.

Test Odoo[Permalink](https://linode.com/docs/websites/cms/install-odoo-10-on-ubuntu-16-04/#test-odoo)

1. Open a new browser window and visit the address:
2. http://<your\_domain\_or\_IP\_address>:8069
3. If everything is working properly, a screen similar to this should appear:

[](https://linode.com/docs/assets/odoo_10_db_creation.png)

1. Congratulations, now you can create your first database and start using Odoo 10 applications!

[](https://linode.com/docs/assets/odoo_10_applications.png)