

OpenSupport Installation Manual

Table of Contents

[Prerequisites: 1](#_Toc36649028)

[Note: 1](#_Toc36649029)

[PHP 5.6+: 2](#_Toc36649030)

[MySQL 4.1+: 2](#_Toc36649031)

[Apache2.4+: 3](#_Toc36649032)

[PDO Extension: 4](#_Toc36649033)

[OpenSupport installation: 7](#_Toc36649034)

[Final result: 13](#_Toc36649035)

[Sources: 13](#_Toc36649036)

# Prerequisites:

* PHP 5.6+
* MySQL 4.1+
* PDO Extension
* Apache 2.4+

### Note:

We start with a fresh installation of Ubuntu 18.04 Server (64-bit), with internet connectivity during this installation tutorial.

Other Linux operating systems can be used, and steps will be similar.

Preform the following commands to retrieve the latest packages:  
*sudo apt-get update*

*sudo apt-get upgrade*

Press **Y** whenever you’re shown the following message during the installation of packages:

A picture containing object

Description automatically generated

### PHP 5.6+:

A clock that is on display

Description automatically generated

Preform the following command: *sudo apt-get install php*

This will install the latest version of PHP on our system.

### MySQL 4.1+:



Preform the following command: *sudo apt-get install mysql-server*

This will install the latest version of MySQL Server on our system.

We must also configure MySQL Server:



Start the configuration process by inputting the following command:

*sudo mysql\_secure\_installation*

You will be asked if you want to install the ‘Validate password plugin’.

You should go ahead and press **Y** when the OpenSupport application is internet facing.

A picture containing clock

Description automatically generated

You will then be asked which level of password validation you would like to apply.

For demonstration purposes I will choose ‘Medium’ (1).

Obviously this is dependent of your company security policy.

A screen shot of a clock

Description automatically generated

You will be asked to set a password for the root account.

Please choose this password carefully.

A screen shot of a person

Description automatically generated

After inputting a valid password, you will be given an estimate of the password strength and will be asked whether or not you wish to continue using this password.

If the password you have chosen suffices, press **Y**.

A close up of a screen

Description automatically generated

You will be asked if you want to remove the anonymous user, provided by default in MySQL.

For security purposes I recommend removing this user.

Next, you will be asked whether or not you wish to restrict logins to the root account.

A picture containing clock, monitor, object

Description automatically generated

Again this is up to you, and the company policy. But for security reasons I recommend disabling remote loggings to the Root account.

You will be asked whether or not you’d like to remove the test database and access to it.

A screen shot of a clock

Description automatically generated

Again, the choice is yours.

But for security purposes we recommend removing it by pressing **Y**.

Finally, you’ll be asked if you want to reload privilege tables.

A large clock mounted to the monitor

Description automatically generated

You should go ahead and press **Y**, this will ensure that all changes made so far will take effect.



For now, the MySQL Server installation is done!

### Apache2.4+:

We must also install Apache2, this will act as our webserver.

First we will install the Apache2 package using the following command:



*sudo apt-get install apache2*

Afterwards, we should make sure our UFW firewall allows the apache2 service to accept incoming and outgoing requests.

If you are unsure whether your company policy allows you to open these ports in your firewall, **please consult your security team beforehand**!

We can configure this using the following command:



*sudo ufw allow ‘Apache’*

Afterwards we will make sure the apache2 service is running and will automatically run when rebooting the server.

We will use the following two commands to achieve this:



1. *sudo systemctl enable apache2*



1. *sudo systemctl start apache2*

To verify our apache2 webserver is functional, we can browse to <http://your_server_ip> and you should be greeted with the following default page:

A screenshot of a cell phone

Description automatically generated

### PDO Extension:

The final prerequisite is to install the PDO extension. On a Linux server this is very simple using this command:



*sudo apt-get install php-mysql*

The next step is to enable the .pdo extensions in our php.ini file.

If you have followed this manual correctly, the active php.ini configuration should be located here: */etc/php/7.2/apache2/php.ini*

If you want to verify where your active php.ini file is located do the following:

Create a *phpinfo.php* file inside */var/www/html/* containing the following line of code:

<?php phpinfo(); ?>

You can then browse to <http://your_server_ip/phpinfo.php> and the following will appear:

A screenshot of a social media post

Description automatically generated

To enable the pdo extensions in your active php.ini file, browse to the given file and open it with the following command:  


*sudo nano /etc/php7.2/apache2/php.ini*

The file will be opened inside nano text editor:

A screenshot of a cell phone

Description automatically generated

Continue to hold the down arrow key untill you notice the following text:

A screenshot of a cell phone

Description automatically generated

Here we will add extension=pdo.so and extension=pdo\_mysql.so

A picture containing black, sitting, screen, table

Description automatically generated

Continue by pressing ctrl+o and ctrl+x to save and exit the file.

Afterwards we should restart the apache2 service to apply the changes we’ve made using the following command:



*sudo service apache2 restart*

# OpenSupport installation:

First make sure you are in the /var/www/html/ directory or any subdirectory of /var/www/html/<sub\_dir> of choice.



*cd /var/www/html/*

You can find the latest OpenSupport version here:

<http://www.opensupports.com/download/>

We will use the wget command to get the .zip file onto our machine.

Run the following command:  


sudo wget <https://github.com/opensupports/opensupports/releases/download/v4.6.1/opensupports_v4.6.1.zip>

Install the unzip package on your machine on order to be able to unzip the downloaded .zip file using this command:



*sudo apt-get install unzip*

We can now unzip the file with this command:



*sudo unzip opensupports\_v4.6.1.zip*

Upon typing the command ls -al we can see the files we had previously are still present in the directory.

A close up of a screen

Description automatically generated We should remove them for security purposes and to minimize confusion.

Do this using the sudo rm <filename> command.



We can now browse to <ip>/<directory>/index.php, if everything went successful you should be greeted with the following screen:

A screenshot of a cell phone

Description automatically generated

Choose the language of your choice, and press **next**.

A screenshot of a cell phone

Description automatically generated

As you can see, there are certain requirements we must fulfill before proceeding to the next step. We must give write permissions to the file /api/config.php and the files folder inside /api/

We can achieve this by using the following commands:

sudo chmod a+rwx files/

sudo chmod a+rwx config.php



If you then press the ‘Refresh’ button you can validate that the correct permissions have been set.

A screenshot of a cell phone

Description automatically generated

We can now press next.

A screenshot of a cell phone

Description automatically generated

In the third step we must input the MySQL Server credentials.

We can’t use the root account for security purposes, there for we must create a new MySQL account with privileges.

A screenshot of a cell phone

Description automatically generated

1. MySQL is running on the same server as OpenSupport, therefor we can use 127.0.0.1 (localhost)
2. We can use the default port (3306)
3. We must create a new database:
   1. CREATE DATABASE opensupport;
4. We must create a user in mysql we do this as followed:
5. sudo mysql -u root
6. CREATE USER 'opensupport'@'localhost' IDENTIFIED BY 'Testtest123!';
7. GRANT ALL PRIVILEGES ON opensupport.\* TO 'opensupport'@'localhost';
8. Input the MySQL username and password.

Press next

You will be greeted with the fourth step:

A screenshot of a cell phone

Description automatically generated

Press Next

In the next step we configure the SMPT server for OpenSupport.

A screenshot of a cell phone

Description automatically generated

For testing purposes I have chosen to use gmail.

A screenshot of a cell phone

Description automatically generated

Do not forget to enable access from less secure apps in your gmail account settings.

Configuring the Admin account:

A screenshot of a cell phone

Description automatically generated

Fill in the details to your liking as shown above. And press next.

If all went well, the installation is now complete and you can continue by logging in to the web panel:

A screenshot of a cell phone

Description automatically generated

# Final result:

A screenshot of a cell phone

Description automatically generated

# Sources:

<https://www.digitalocean.com/community/tutorials/how-to-install-mysql-on-ubuntu-18-04>  
<https://www.digitalocean.com/community/tutorials/how-to-install-the-apache-web-server-on-ubuntu-18-04-quickstart>

<https://www.ostraining.com/blog/coding/phpini-file/>

<https://stackoverflow.com/questions/13375061/installing-pdo-driver-on-mysql-linux-server>

<https://github.com/opensupports/opensupports/wiki/Installation>