

Below are the steps to install Orangescrum Self Hosted version of Cloud Edition in CentOS Server

Last Updated: *Oct 18, 2018*

STEP 1

Download the Orangescrum Self Hosted version of Cloud Edition Extract the archive file.

You will find the following list in files/folder

1. orangescrum - folder

You will find user license wise sql file(Ex. database.sql) inside orangescrum folder.

2. installationmanual.pdf

3. Perpetual license doc

Note: Strictly follow the below steps to install self hosted orangescrum.

***Note: If Gmail ID used make sure to enable the “less secure app” function. You can do this by logging into the gmail account and go to “My Accounts” then click on “Sign-in & Security” and the enable the “Allow less secure apps: ON”.**

***Note: Make sure to whitelist the application server domain/IP address from the private Mail Server.**

***Note: OrangeScrum will only works on MySQL 5.4-5.7, Apache 2.4 and PHP 7.x**

- *If SELinux is installed and enabled, please set it to permissive mode.
For temporary “setenforce 0”
For Permanent: edit the “/etc/selinux/config” and set selinux=permissive*
- *Added the below firewall rules in the server
firewall-cmd --permanent --zone=public --add-service=http
firewall-cmd --permanent --zone=public --add-service=mysql
firewall-cmd --permanent --zone=public --add-service=https
firewall-cmd --reload*

STEP 2

Required packages to install: PHP, MySQL, Apache

Requirement:

* Apache with `mod_rewrite`

* Enable curl in php.ini

* Change the 'post_max_size' and `upload_max_filesize` to 200Mb in php.ini

* PHP 7.x

Required PHP Extensions: -

Php7.2-gd

Php7.2-curl

Php7.2-common

Php7.2-fpm

Php7.2-cli

php57.2-imap
php7.2-intl
php7.2-ldap
php7.2-mysql
php7.2-snmp
php7.2-tidy
php7.2-mcrypt
php7.2-mbstring
php7.2-soap
php7.2-zip
php7.2-dba

* MySQL 5.5- 5.7

* If STRICT mode is On, turn it Off.

wkhtmltopdf installation Process

yum -y groupinstall "Development Tools"

yum -y install wget

yum install --y epel-release

wget https://github.com/wkhtmltopdf/wkhtmltopdf/releases/download/0.12.5/wkhtmltox-0.12.5-1.centos7.x86_64.rpm

rpm -Uvh wkhtmltox-0.12.5-1.centos7.x86_64.rpm

Check version - wkhtmltopdf -V

Define the path for wkhtmltopdf in Constants.php

For Ex. ***define('PDF_LIB_PATH', '/usr/local/bin/wkhtmltopdf');***

STEP 3

- Extract the archive file.
- Upload folder (orangescrum) to your working directory (/var/www/html).
- Provide proper write permission to orangescrum folder and their subfolders.
- Provide proper write permission to " app/tmp" folders and their sub folders
- Provide Proper ownership to orangescrum folder and their sub folders.

```
chmod -R 0755 /var/www/html/orangescrum  
chmod -R 0777 app/tmp  
chown -R apache:apache /var/www/html/orangescrum
```

STEP 4

Install the Apache Server:

```
yum -y install httpd
```

Apache start in system boot time:

```
systemctl enable httpd
```

Start Apache Service:

```
/etc/init.d/httpd start Or service httpd start
```

STEP 5

Install MySQL: (Login as root user)

```
yum -y localinstall https://dev.mysql.com/get/mysql57-community-release-el7-9.noarch.rpm  
yum -y install mysql-community-server  
systemctl enable mysqld  
service mysqld restart
```

Then we will setup the system startup links for MySQL (MySQL starts automatically whenever the system boots):`chkconfig --levels 235 mysqld on`

Start the MySQL server:

```
/etc/init.d/mysqld start
```

Setup the MySQL root password:

```
mysql_secure_installation
```

NOTE:

During password creation, MySQL will ask for root password

Run the below command to get the temporary MySQL root password

```
sudo grep "temporary password" /var/log/mysqld.log
```

After putting the temporary password, you may ask to change the MySQL root password.

Password must be 8 characters including 1 uppercase letter, number, and special characters

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MySQL SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MySQL to secure it, we'll need the current password for the root user. If you've just installed MySQL, and you haven't set the root password yet, the password will be blank, so you should just press enter here

Enter current password for root (enter for none): OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the MySQL root user without the proper authorisation.

Set root password? [Y/n] ← **ENTER**

New password: ← **yourrootsqlpassword**>

Re-enter new password: ← **yourrootsqlpassword**>

Password updated successfully!

Reloading privilege tables..

... Success!

By default, a MySQL installation has an anonymous user, allowing anyone to log into MySQL without having to have a user account created for them. This is intended only for testing, and to make the installation go a bit smoother. You should remove them before moving into a production environment.

Remove anonymous users? [Y/n] ← **ENTER**

... Success!

Normally, root should only be allowed to connect from 'localhost'. This ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] ← ENTER

... Success!

By default, MySQL comes with a database named 'test' that anyone can access. This is also intended only for testing, and should be removed before moving into a production environment.

Remove test database and access to it? [Y/n] ← ENTER

- Dropping test database...

... Success!

- Removing privileges on test database...

... Success!

Reloading the privilege tables will ensure that all changes made so far will take effect immediately.

Reload privilege tables now? [Y/n] ← ENTER

... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MySQL installation should now be secure.

Thanks for using MySQL!

STEP 6

Login To MySQL & Create Database named "orangescrum" (you can rename to any name)

login to mysql:

```
[root@server ~]# mysql -u root -p
```

Enter password:

create the database:

```
mysql> create database orangescrum; (database name)
```

verify that it's there:

```
mysql> show databases;
```

create the user:

```
mysql> create user orangescrum; (database user name)
```

Grant all privileges while assigning the password:

```
mysql> grant all on orangescrum.* to 'orangescrum'@'localhost' identified by 'your_password';
```

Exit from the database:

```
mysql> exit
```

Import database sql file:

Navigate to /var/www/html/orangescrum directory by typing

```
cd /var/www/html/ orangescrum
```

```
[root@server ~]# mysql -u orangescrum -p orangescrum < database.sql
```

Enter password:

Login to the database and check whether your tables are created or not:

```
[root@server ~]# mysql -u orangescrum -p
```

Enter password:

```
mysql> show databases
```

if your database exist then trigger the below command:

```
mysql> use orangescrum;
```

```
mysql> show tables;
```

```
mysql> exit
```

```
mysql> exit
```

[Manage MySQL Databases \(Optional\)](#)

Install phpMyAdmin(To access database Graphically)

```
yum install phpmyadmin
```

To access phpmyadmin in browser

Open the file

```
vim /etc/httpd/conf.d/phpMyAdmin.conf
```

And add

Require all granted

in the below patterns.

```
<IfModule mod_authz_core.c>
```

```
  # Apache 2.4
```

```
  <RequireAny
```

```
  >
```

```
    Require ip
```

```
    127.0.0.1 Require
```

```
    ip ::1
```

```
    Require all granted
```

```
  </RequireAny>
```

```
</IfModule>
```

```
<IfModule !mod_authz_core.c>
```

After that restart apache server.

```
service httpd restart
```


Open Browser and access the phpmyadmin

<http://IPADDRESS/phpmyadmin>

STEP 7

Install the PHP Packages: (Recommended Version PHP 72)

```
yum -y install http://rpms.remirepo.net/enterprise/remi-release-7.rpm
```

```
yum install yum-utils.noarch remi-release.noarch
```

```
yum-config-manager --enable remi-php72
```

```
yum -y install php php-cli.x86_64 php-common.x86_64 php-bcmath.x86_64 php-dba.x86_64  
php-embedded.x86_64 php-enchant.x86_64 php-fpm.x86_64 php-gd.x86_64 php-imap.x86_64  
php-intl.x86_64 php-ldap.x86_64 php-mbstring.x86_64 php-mcrypt.x86_64 php-mysql.x86_64  
php-pdo.x86_64 php-pecl-zip.x86_64 php-pecl-memcache.x86_64 php-pecl-imagick.x86_64  
php-soap.x86_64 php-tidy.x86_64 php-xml.x86_64 php-opcache.x86_64
```

Restart the Apache service:

```
/etc/init.d/httpd restart
```

Or

```
service httpd restart
```

STEP 8

Setup the database information in app/config/database.php

Update the database connection details. (host, login, password and database name).

STEP 9

Virtual Host setup in Apache conf file.

```
vi /etc/httpd/conf/httpd.conf
```

Add this below details to last line in httpd.conf file

```
<VirtualHost *:80>
ServerName localhost # eg: demo.orangescrum.com
DocumentRoot /var/www/html/orangescrum /
<Directory /var/www/html/orangescrum />
Options Indexes FollowSymLinks MultiViews
AllowOverride All
Order allow,deny
allow from all
</Directory>
</VirtualHost>
```

Save the file and Restart the Apache service

```
/etc/init.d/httpd restart / service httpd restart
```

STEP 10

General Configuration management:

MySQL:

- If STRICT mode is On, turn it Off.

Disable Strict mode on mysql for Centos/Fedora :-

```
vim /etc/my.conf
```

```
sql_mode=NO_ENGINE_SUBSTITUTION,STRICT_TRANS_TABLES
```

to

```
sql_mode=""
```

```
sudo service mysql restart
```

Make sure that, the .htaccess is working in your server.

SMTP (IMPORTANT)

Provide your valid Gmail ID and Password for SMTP email sending Or, you can use Sendgrid or Mandrill in app/Config/Constants.php file

FROM_EMAIL - this will be the default from email through out Orangescrum.

FROM_EMAIL_EC All the task created/updated notification email will be sent from this Email ID.

SUPPORT_EMAIL All Other Emails and support related Emails will be sent from this Email ID.

Please follow below steps to configure your SMTP:

Step 1:

Go to the path in your application: app/Config/constants.php

Step 2:

Set the following in Gmail SMTP:

```
define("SMTP_HOST", "ssl://smtp.gmail.com");
```

```
define("SMTP_PORT", "465");
```

```
define("SMTP_UNAME", "youremail@gmail.com");
```

```
define("SMTP_PASSWORD", "*****");
```

Please provide your smtp username and password to configure your SMTP.

If you are finding problem after setting the username and password, then please check the below:

Go to your php.ini file and enable the "php_openssl" in you PHP setting.

You need to enable extension=php_openssl.dll on php.ini file. If you are still facing the problem, then the port 465 is not available. Try with 587 port.

Otherwise, please contact with your server administrator.

Please run the following URL to check email is working or not:

http://YOURDOMAINNAME/cron/test_email/?to=emailId

Example:

http://localhost/orangescrum/cron/test_email/?to=emailId (if you are using localhost)

http://127.0.0.1/orangescrum/cron/test_email/?to=emailId (if you are using IP)

http://myprojects.orangescrum.com/cron/test_email/?to=emailId (if you are using any valid domain)

If the email is not working, then please follow the installation guide again.

Check Gmail security issues here. Please see below links, you have to change some configuration at email end.

<https://support.google.com/accounts/answer/6010255?hl=en>

<https://www.google.com/settings/security/lesssecureapps>

Also check app/tmp/logs/os-email.log file for any error.

Check SMTP Port:

If ports (465,587) are not enabled, then open it.

Still mail functionality is not working then try with changing the ports to (465,587, 25)

Add the hostname according to the mail server imap setting and Imap port. If ssl is enable then add no validate-cert parameter to the host name.

Then Add HOST Name, User Name & Password

To Use Own Email Server:

For own Email server: Add HOST_NAME, PORT, SMTP_UNAME, SMTP_PASSWORD in constant.php instead of gmailconfiguration.

Incorrect authentication data error:

Please double check the username and password.

If both are correct then check the server configuration. It's somehow blocking the server to connect to the mail server. We cannot do anything to fix this.

Also make sure you're not using 2 step verification and less secure app setting is ON.

FROM_EMAIL_EC All the task created/updated notification email will be sent from this Email ID.

SUPPORT_EMAIL All Other Emails and support related Emails will be sent from this Email ID.

STEP 12

Email Reply - Nohup Cron Job setup (Linux Server)

Make sure to do the following changes on the Email server connection details in the app/webroot/EmailReply.php file.

\$username- This will be the FROM_EMAIL_EC Email set on your app/Config/constants.php

All the task created/updated notification email will be sent from FROM_EMAIL_EC . When somebody will reply on that task created/updated notification email, the FROM_EMAIL_EC will get that Email in the inbox.

EmailReply.phppage is going to read the emails from FROM_EMAIL_EC and It will post to them as a reply to the respective tasks in Orangescrum.

\$password- Password of FROM_EMAIL_EC

client- Change it, if you are not using Gmail

After this setup, you can reply to a task created/updated notification email and that email reply will be posted to Orangescrum under that Task. This will help you to respond to a task while on the go from your Mobile.

(Assuming your Application is in "/var/www/html/orangescrum/ ")

Enable extension=php_imap in your php.ini file

Create a orangescrum.sh file in your server

vi orangescrum.sh (or, open that file to write the below code)

```
#!/bin/bash while(true)
```

```
do
cd /var/www/html/orangescrum/app/webroot
php q EmailReply.php 1>&2
sleep 1?
done
```

Give the execute permission for orangescrum.sh: `chmod +x orangescrum.sh`
Start the Nohup using the command: `nohup sh orangescrum.sh > customout.log &`

STEP 13

Browse the Orangescrum website <http://YourIpAddress> or server IP address or domain name.
Ex. <http://localhost/orangescrum>

STEP 14

You will be asked to provide your Company Name, Email address and a Password to login and start using Orangescrum.

STEP 15

Advanced Setup

Google Signup, Google Login, Google Contact

To setup the Google Signup, Google Login, Google Contact, define the following details in the ***app/Config/constants.php file***

```
##### Google Keys (Login, Drive, Contacts)
#####
define("CLIENT_ID", "XXXXXXXXXXXX.apps.googleusercontent.com");
define("CLIENT_ID_NUM", "XXXXXXXXXXXX"); define("CLIENT_SECRET",
"xXxXxXxxx_xXxXxXxxx"); define("API_KEY",
"xXxXxXxxxXXXXXXXXXXXXXxXxXxxx"); define("REDIRECT_URI",
HTTP_ROOT . "users/googleConnect");
define("USE_GOOGLE", 0); //Set this parameter to 1, to use Google Login, Drive and Contacts
```

Dropbox Setup

```
##### Dropbox Key #####
define("DROPBOX_KEY", "xXxXxXxX");
```

```
define("USE_DROPBOX", 0); //Set this parameter to 1, to use Dropbox file sharing
```

AWS S3 Bucket Setup

```
define('USE_LOCAL', 1);  
##### AWS S3 Bucket #####  
define('USE_S3', 0); //Set this parameter to 1 to use AWS S3 Bucket ("0" for local storage)  
define('BUCKET_NAME', 'Bucket Name');  
define('DOWNLOAD_BUCKET_NAME', 'Download Bucket Name');  
define('awsAccessKey', 'XXXXXXXXXXXXXXXX');  
define('awsSecretKey', 'XXXX/XXXXXXXXXXXXXXXX/+XXXXXXXXXXXXXXXX');
```

STEP 16

For Installing Node.js

Added the node repository:-

```
curl -sL https://rpm.nodesource.com/setup_8.x | sudo bash -
```

Optional:

```
curl -o-
```

```
https://raw.githubusercontent.com/creationix/nvm/v0.33.11/install.sh |  
bash
```

- Install nodejs:-
yum install nodejs
- Check the node and npm version:-
node -v
npm -v
- Install socket.io:-
npm install socket.io
- Install pm2 tools to run the notification.js file:-
npm install pm2 -g
- Go to the directory where the notification.js file is present and run the command:-
pm2 start notification.js
- Run the command to see the notification:-
pm2 list
- If you want to use http, then no need to change in the notification.js file.
- If you want to use https, then enable the ssl part and added the ssl details like below:-

```
var options = {  
  key: fs.readFileSync('/etc/ssl/certs/grantham.key'),
```



```
cert: fs.readFileSync('/etc/ssl/certs/grantham.crt'),
ca: fs.readFileSync('/etc/ssl/certs/grantham.ca-bundle')
};
var app = require('https').createServer(options, handler);
```

And disable the http part like below:

```
//var app = require('http').createServer(handler); // For non-ssl server
```

- After that stop the pm2 tools with pid 0
pm2 stop 0
- At last start the pm2 tools with pid 0
pm2 start 0

Now node server is ready. You can test by checking the below URL

<http://your-domain:3002>

Ex. <http://localhost:3002>

<http://127.0.0.1:3002>

<http://myorange.com:3002>

If output is "Welcome to socket.io." then node.js is working fine and you can use that url in the constant.php as "NODE_HOST"

Note: you can find notification.js, node-js-installation-linux.pdf, node-js-installation-windows.pdf file in Orangescrum folder after install the In-App chat.

STEP 17

Profile Setting

- Go to Setting, click on My profile to set up your profile photo.
- Select Time zone for your account
- Click on Update to save the changes

Notifications Setting

- Go to Settings
- Click on Notifications under Profile Setting

- Select “Yes” to get notifications in your email inbox

- Click on Email Reports from the Personal setting
- Select Yes to get Email Reports

Company Setting

- Go to Setting
- Select My Company from the Company Settings
- Upload the company logo and click on Update to save changes

For Any Queries, Contact us: support@orangescrum.com